Wizualization: A "Hard Magic" Visualization System for Immersive and Ubiquitous Analytics

Andrea Batch, Peter W. S. Butcher, Panagiotis D. Ritsos, Niklas Elmqvist

Fig. 1: **Becoming an Optomancer.** Our motivating scenario tells the story of Paul, Ana, and Carlos, three CDC experts analyzing health survey data across space and time to understand the impact of COVID-19 on U.S. diets. Along the way, they will use their voices and hand gestures to cast magic spells that weave together chart blocks, datasets, and interaction into immersive 3D visualizations in web-based eXtended Reality. All three of these spell casters use our Wizualization system on their Microsoft HoloLens 2 HMDs and smartphones to track down elusive answers to what the global pandemic has done to our very lives...

Abstract—What if magic could be used as an effective metaphor to perform data visualization and analysis using speech and gestures while mobile and on-the-go? In this paper, we introduce WIZUALIZATION, a visual analytics system for eXtended Reality (XR) that enables an analyst to author and interact with visualizations using such a magic system through gestures, speech commands, and touch interaction. Wizualization is a rendering system for current XR headsets that comprises several components: a cross-device (or ARCANE FOCUSES) infrastructure for signalling and view control (WEAVE), a code notebook (SPELLBOOK), and a grammar of graphics for XR (OPTOMANCY). The system offers users three modes of input: gestures, spoken commands, and materials. We demonstrate Wizualization and its components using a motivating scenario on collaborative data analysis of pandemic data across time and space.

Index Terms—Immersive analytics, situated analytics, ubiquitous analytics, gestural interaction, voice interaction.

1 INTRODUCTION

Arthur C. Clarke famously stated [15] that *"any sufficiently advanced technology is indistinguishable from magic."* This is certainly true for how many researchers envision people using immersive technologies such as virtual, augmented and mixed realities (VR/AR/MR), collectively refereed to as eXtended Reality (XR) hereafter, to analyze data in the future [60]; using vocal commands, gestures, and dataset "reagents"—all classic components of magic—to conjure data visualizations and analytics interfaces out of thin air, anytime and anywhere. Furthermore, recent work has proposed the idea of superpowers—arguably a specialized form of magic—as inspiration for data visualization [78].

- *Andrea Batch is with the U.S. Bureau of Economic Analysis in Washington, D.C., United States. E-mail: andrea.c.batch@gmail.com.*
- *Peter W. S. Butcher and Panagiotis Ritsos are with Bangor University, Bangor, United Kingdom. E-mail: {p.butcher, p.ritsos}@bangor.ac.uk.*
- *Niklas Elmqvist is with Aarhus University in Aarhus, Denmark; the work was conducted at University of Maryland, College Park, MD, United States. E-mail: elm@cs.au.dk.*

Manuscript received xx xxx. 201x; accepted xx xxx. 201x. Date of Publication xx xxx. 201x; date of current version xx xxx. 201x. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org. Digital Object Identifier: xx.xxxx/TVCG.201x.xxxxxxx

We propose an approach to visualization authoring based on fictional magic systems. We draw on prolific fantasy author Brandon Sanderson's concept of "hard magic" as magic systems that have strict rules explicitly described by the author [62]. We suggest that a hard magic system can be an engaging and useful mental model for users to author visualizations for immersive [48], situated [72], and ubiquitous analytics [20] (IA/SA/UA) experiences. These ideas are also compatible with recent movements within immersive and situated visualization [75] toward modeless grammars of graphics for immersive analytics [17], such as virtual hand and raycaster interactions for manipulating data [74].

We present WIZUALIZATION, a prototype implementation of a visualization authoring framework for XR based on this hard magic metaphor. Wizualization is a rendering system running on current-generation XR hardware (such as the Microsoft HoloLens 2, or even VR headsets that support hand-tracking) with the following components:

- **Arcane Focuses**: Verbal control (creation and interaction) using the Web Speech API (e.g, smartphone), gesture control for devices with gesture recognition (e.g, Hololens 2), and passive viewing capabilities for all devices (spectators).
- **Weave**: A signal propagation mechanism connecting external devices with the HMD and gesture controls.
- **Spellbook**: A code notebook and dictionary of mid-air gestures and spoken commands (spells); and
- **Optomancy**: A grammar of graphics (GoG) for immersive visualization and interactions for creating visualizations.

These components together form a "hard magic" system because they operationalize seemingly "magic" functions such as voice, gestures, and touch interaction as *spells* to generate visualization grammar specifications for XR. We have made all of the subsystems that constitute our contribution available to eponymous repositories at https://github.com/wizualization.

Beyond the engineering contributions of the Wizualization system, we propose the following as the key intellectual contributions of our work: (i) a comprehensive magic metaphor for using speech, gestures, and touch interaction to author and interact with immersive visualizations; (ii) an immersive and ubiquitous analytics system for collaborative data analysis in XR using the magic metaphor; and (iii) a WebXR and React implementation—Wizualization—of the above ideas.

2 Related Work

Our work is influenced by research in several domains, including direct manipulation and sketching, post-WIMP interaction for visualization, and the emergence of declarative grammars in visualization.

2.1 Direct Manipulation in Visualization

Direct manipulation [67] is designed to reduce the distance between physical input and virtual objects displayed on a computer's screen. Beyond games, one of the original applications of direct manipulation was visualization [67]. As a case in point, the seminal dynamic queries method minimizes indirection and reduces barriers between users and visual representations [1]. Building on the direct manipulation idea, Elmqvist et al. [22] proposed the notion of *fluid interaction*, arguing that "interaction in visualization is the catalyst for the user's dialogue with the data, and, ultimately, the user's actual understanding and insight into these data." They define a fluid interface as being one that: (1) promotes flow (a mental state of complete immersion in an activity) [18], (2) supports direct manipulation [67], and (3) minimizes Norman's gulfs of interaction [55] (i.e., the difference between a user's intended action and actions afforded by the system).

Sketch-based systems are an example of fluidity in interaction. In SketchStory, Lee et al. [41] enable users to give ad-hoc data presentations by authoring visualizations on the fly using sketch-based pen strokes. ScribbleQuery [53] applies touch-based sketching to brushing and selection in parallel coordinate plots. Schroeder and Keefe [65] discuss a system in Visualization-by-Sketching that augments the digital work of artists and designers with real data.

The advent of consumer-ready immersive displays reduces the barrier further still, with the most common modes of interaction in immersive analytical environments being virtual hands and virtual ray pointers [74]. *Scientific sketching* [36] allows for free-form 3D sketching to support data analytics in an immersive environment; the approach has since been adapted to multiple applications, including 3D fluid flow, collaborative analysis, and paleontology [56]. With that said, the use of sketching in immersive systems is not new; in fact, work in the area of direct manipulation was already exploring this mode of input for creating 3D scenes in 1996 with the SKETCH system [81]. However, this and similar work of that era focused on simulated objects or free-form design [33] rather than on data visualization.

2.2 Interaction in Immersive Analytics

Interaction has a central role in visualization [79], and this also holds true for immersive and situated visualization [49]. Nevertheless, enabled by technological advances in immersive technologies, recent efforts have explored novel interaction techniques and device synergies in an SA context [10]. For example, Bach et al. [2] assessed the effectiveness of direct tangible interaction with 3D holograms. They compared the use of a Microsoft HoloLens with fiducial markers for AR visualization to a handheld and a desktop-based setup.

Analyzing user interactions in MR spaces has also received attention. MRAT [52] visualizes usage data of interaction techniques in MR, providing mechanisms for interaction tracking, task definition and evaluation, and visual inspection with in-situ visualizations. Likewise, Büschel et al. [11] present MIRIA, a toolkit for in-situ visual analysis of spatial temporal interaction data in MR and multi-display environments. MIRIA provides mechanisms to analyze movement of users and tracked devices and to identify issues such as tracking problems or obstructions from physical objects. Flex-ER [44] is a web-based environment that enables users to design, run, and share investigations in MR, supporting different platforms and interfaces via a JSON specification of interactions and tasks. ReLive [31] provides similar cross-device MR functionality by combining in-situ and ex-situ (desktop) views.

Cross-device synergies have also been used for enhancing the analytical process within SA environments. Butscher et al. [13] investigate synergies between tabletops and AR-enabled HMDs to visualize and manipulate 3D parallel coordinate plots. Hubenschmid et al. [32] explore similar synergies with tablets, and interactions via touch and voice commands. Reipschläger et al. [59] does this for AR HMDs and touchscreens. Finally, Langner et al. present MARVIS [38], a framework enabling the combination of mobile devices and AR HMDs in analytical setting. MARVIS allows the depiction of 3D visualizations above and between devices through the HMD. However, these are all one-off designs for specific displays and devices. Fröhler et al [26] provide a more detailed survey of cross-device synergies in IA and SA environments, referring to the domain as "cross-virtuality analytics."

Of particular interest to our work are the interaction affordances of toolkits designed for building immersive experiences. DXR [68], a Unity-based IA/SA toolkit, supports multi-visualization workspaces, with interactions (toggles, filters etc.), specified in a JSON specification. Interactive elements include tooltips, view manipulation, and configuration controls, and its grammar can be extended to work with other modalities such as tangible, direct manipulation, gesture, and speech input. IATK [16] defines a high-level interaction model that provides filtering, brushing, linking and details on demand functionalities, harnessing GPU power to optimise performance. Building on IATK, RagRug [25] uses data streams from Internet-of-Things devices in SA. RagRug portrays the potential of cross-device connectivity with a visualization pipeline that combines IoT devices, data mediation via MQTT, Node-RED for filtering, and IATK for visual encoding and rendering in MR. Finally, VRIA [12], although predominately designed for VR, also works in AR settings [6], largely thanks to the ongoing development of the open WebXR specification. Our work here follows the same open web technology approach as VRIA.

Interestingly, Besançon et al. [8] point out that new interaction techniques for exploring, filtering, selecting, or manipulating 3D data are often published in non-visualization venues, and thus remain unnoticed by visualization researchers (e.g. [16, 17, 21]). They also note that leveraging sensing technologies and adapting 3D interaction techniques from other contexts has significant potential for positive impact on 3D visualization. Nevertheless, structured mechanisms for creating and defining interaction affordances in SA environments, especially for voice and gesture input (such as this paper), are much less explored.

2.3 Gestures and Speech for Visualization

Combining speech and gestures has long been a vision for the future of computing [28]. For example, Bolt [9] proposed "Put-That-There" in 1980, allowing a user to combine spoken commands with pointing to populate a room-sized space with 3D objects. Likewise, the use of speech and gestures for interacting with visualizations have also been advocated as potentially more natural compared to WIMP interfaces [40,60], yet remain largely unexplored. Recent work has explored multimodal interfaces, for example combining direct manipulation and natural language input [69]. However, we feel that in 3D, immersive space, gestures are a more natural fit, with most current and emerging HMDs (e.g., the Apple Vision Pro) supporting gestural interfaces.

The visualization *of* gestures [23, 34, 35] is more prevalent than the use of gestures for visualization. One exception is Proxemic Lenses [3], where collaborators use explicit gestures and implicit body language to interact with large data displays. Another is DA-TU [29], which features a tablet-based multi-finger gestural vocabulary for interacting with database objects. However, the use of mid-air gestures for immersive analytics has remained uncommon, with one notable exception in Filho et al.'s evaluation of an immersive space-time cube [24].

Considerably more work has been conducted in the area of speech as an input mode for visualization for traditional displays [19, 80], large screens [3], and immersive environments [4]. The Natural Language for Data Visualization (NL4DV) system [51] integrates contemporary visualization tools with multimodal user input and popular analytical tools and workflows. Even the subject of combining speech and touch—not gesture, but touchscreen interaction—has been addressed in visualization literature [61], with the results confirming that multimodal input is preferred over single modes of input for either speech or touch. However, we argue that *speech and mid-air gestures* have not been used in combination for immersive analytics in the existing literature. We note that gestures (and even speech) has been used for data-driven storytelling, such as the aforementioned SketchStory [41] as well as the more recent RealityTalk [43] and Hall et al.'s remote visualization presentation tool based on gestures [27]. Nevertheless, in this paper we show how this can be a powerful modality in mobile AR settings.

2.4 Visualization Grammars and Beyond

Visualization grammars, first introduced by Leland Wilkinson as the eponymous Grammar of Graphics [77] in 1999, provide combinatorial building blocks for specifying visual representations using a concise declarative language. This approach is radically different from the chart template galleries used in tools such as Microsoft Excel.

Wilkinson's grammar was quickly adopted by the visualization and statistics communities. Hadley Wickham's `ggplot2` [76] operationalized the grammar in the R language. Vega [64] is a low-level explanatory specification expressed as JSON and rendered on the web using SVG or Canvas; upon it is built the higher-level Vega-Lite [63], which facilitates rapidly building interactive visualization without exposing the full complexity of the Vega backend. Several special-purpose grammars have since evolved, including ATOM [57] for unit visualizations, Cicero [37] for responsive web-based visualizations, and PGoG [58], a probabilistic extension to Wilkinson's Grammar of Graphics.

Visualization grammars can also serve as low-level specification backends for higher-level visualization environments. This allows point-and-click interaction rather than textual specification. For example, Tableau (originally Polaris [70]) is built on the VisQL grammar, and Lyra 2 [82] generates Vega or Vega-Lite specifications as output.

Finally, there exists some visualization grammars designed specifically for immersive, situated, and ubiquitous analytics. ImAxes [17] is one such system, enabling a VR user to freely combine axes to author various multidimensional visualizations in 3D using direct manipulation. DXR [68] uses a JSON specification language similar to Vega-Lite to author 3D visualizations for immersive analytics in the Unity engine, even providing an interface for modifying the representations while in the immersive environment. VRIA [12] supports a similar Vega-like JSON specification, but is entirely implemented using open web technologies such as A-Frame[1], React[2], and D3.js[3] rather than Unity.

In contrast, we propose a visualization environment for ubiquitous and immersive analytics based on mid-air gesture and speech interaction. Similar to VRIA, our approach is built on open web technologies rather than a proprietary graphics engine. To the best of our knowledge, our work is the first to allow users to author visualizations specifications in 3D MR using such a direct manipulation method.

3 Overview

Wizualization is a magic-inspired visualization authoring environment for XR. It uses the metaphor of a "hard magic" system—one whose rules are explicitly described to the point where the audience can reason about how to use it themselves [62]—because magic is a familiar and useful mental model for many users. By mapping grammar specification to speech, touch, and gesture-based interactions, Wizualization provides a "hard magic"—i.e., operationalized and deterministic—approach to authoring and interacting with visualizations in XR. Here we will give an overview of Wizualization and its use before delving into the system architecture in the following section.

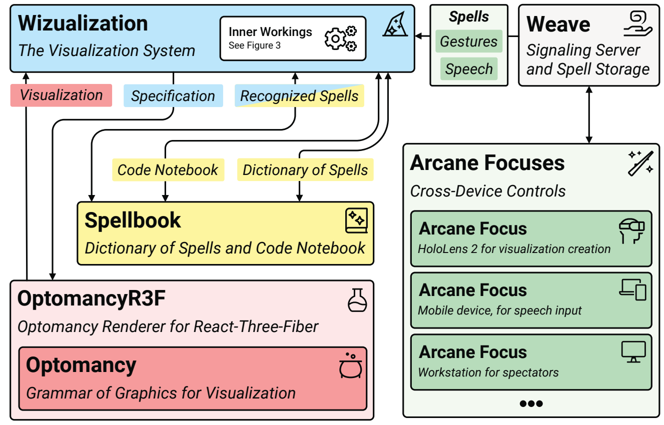


Fig. 2: **Wizualization System Overview.** The main system components: i) Wizualization comprises support for a set of devices, Arcane Focuses, which allow the user system control, ii) a signaling server, Weave, responsible for synchronizing the state of all connected clients (Arcane Focuses) and storing the state of cast spells in each room, iii) a dictionary of spells and code notebook, Spellbook, which stores the system's set of spell primitives and mappings to blocks of code for Optomancy, and iv) the grammar of graphics we use to produce demonstration visualizations.

At its most basic level, Wizualization (Fig. 2) is a web-based framework for IA, SA, and UA in XR based on the W3C WebXR Device API [73]. We developed the system for the Microsoft HoloLens 2, but it should be compatible with any WebXR-capable device (such as the Meta Quest Pro and any future HMD that supports the WebXR device API). In fact, the system will also work with XR-capable handheld devices, such as a smartphone or tablet; this enables a user to access their Wizualization workspace even when no HMD is available. However, the system is primarily designed for HMDs because it relies on hand tracking, voice recognition, and overlaid computer imagery, which are typically only partially available on handheld devices.

Wizualization user sessions are captured in *virtual 3D rooms* that represent a unique configuration of datasets, visualizations, and a *spellbook* (a dictionary of spells; Sec. 4.3). Initializing a Wizualization session typically means creating a new workspace and adding new datasets or using existing datasets already uploaded into the system. Workspaces are persistent and can be shared by multiple users, providing provenance and persistence.

The inputs to Wizualization are spoken words, finger gestures, or QR tags that have been associated with a set of chart and data transformations. A sequence of these inputs is what we refer to as a *spell*. Spells are *crafted* by the user by first recording the input that will be used to invoke the spell, and then specifying the chart and data transformations that the spell will perform. *Casting* a spell, consequently, involves merely performing the stored input sequence, causing the chart and data transformations to be executed akin to running a recorded macro.

The chart and data transformations that make up spells are represented as blocks of code (called *spell blocks*) that form chunks of input to the Optomancy declarative visualization grammar (Sec. 4.4). Spell blocks are rendered in the scene as sequentially-ordered linked nodes with snippets of the Optomancy specification that are germane to the spell each block represents. A selection of spells that we consider "primitives" are part of a shared Spellbook made available across all of the user's (or team's or organization's) rooms; others are created specifically for a particular room and dataset. A user can also create an Optomancy specification on the fly to immediately instantiate a visualization for a specific dataset rather than creating a spell for later use. In this sense, spells are visualization templates.

Finally, a Wizualization session is also characterized by the physical configuration of devices and displays used during the session. The system keeps track of these devices—each called an Arcane Focus—and

[1] https://aframe.io
[2] https://reactjs.org
[3] https://d3js.org

manages input routing between them using the Weave signaling service (Sec. 4.2). Since all rooms and configurations are server-side with operational consistency and persistence, Wizualization also trivially supports all forms of collaboration (Sec. 5.3): both asynchronous and synchronous, remote as well as co-located.

Throughout the technical description that follows, we are using a scenario involving a health informatician, Paul, and his colleagues working on a large-scale health survey spanning space and time. The scenario is inspired by real projects drawn from the first author's current workplace at the U.S. Bureau of Economic Analysis. We interweave vignettes of Paul and his team using the Wizualization system to perform a variety of analytical tasks. Some vignettes are illustrated in Fig. 1.

Paul — Introduction

PAUL is a postdoc at the National Center for Health Statistics in Hyattsville, MD, USA, a part of the U.S. Centers for Disease Control and Prevention. He has been asked to explore potential impacts of the COVID-19 pandemic on U.S. diets using the NHANES (National Health and Nutrition Examination Survey) dataset, which catalogs the health and nutritional status of adults and children in the United States. Since his research team is widely distributed across time and geography, he needs a data science environment that can support multiple forms of collaboration. Furthermore, the data is complex, spread across multiple years and ancillary databases, and often has an in-situ component when survey statisticians go into the field to collect local data in support of a research question.

4 THE WIZUALIZATION SYSTEM

Wizualization is the rendering and user input management system, and also connects four components: **Arcane Focuses** that facilitate verbal control for supported devices with a microphone (e.g., smartphone), gesture control for devices with gesture recognition (e.g., HoloLens 2), and passive viewing for all devices (spectators); **Weave**, a signal propagation framework that connects input and output devices in the tool; **Spellbook**, which serves as a notebook and dictionary of mid-air gestures and spoken commands (spells); and **Optomancy**, a grammar of graphics for immersive visualization and interactions.

4.1 System Summary

Wizualization is the client application layer for the implementation summarized in Fig. 2 and detailed in Fig. 3. It is written in TypeScript using ReactJS and depends on `react-three-fiber`[4] (R3F), a React wrapper for `three.js`.[5] It also depends on the WebXR API[6] for hand pose, the Web Speech API[7] for speech recognition, and the AR.js library for QR code recognition.[8] Wizualization creates the specifications for our grammar, Optomancy, which returns components from its R3F renderer, OptomancyR3F, to be rendered as visualization objects in the XR scene by Wizualization. Wizualization also imports our code notebook, Spellbook, and passes it a sequential list of code blocks designating each spell in the order it was cast; Spellbook then returns R3F components to be rendered as linked spell blocks representing the current workflow of the room. Wizualization events are communicated across devices using Weave, our signaling server.

In the prototype version of Wizualization (see project repository), as well as in the figures presented in this paper, we have included examples using a selection of sample datasets, chart types, and Optomancy grammar features. For the purposes of demonstration, we have randomized variable selection for Wizualization user sessions. Wizualization applies sensible default properties to charts; for example, if a user wishes to craft a bar chart, the first axis they cast will be assumed to be an `X` axis and will have a randomly selected nominal field from the dataset applied to it. The second axis will be of a quantitative data type and will be assumed to be a `Y` axis.

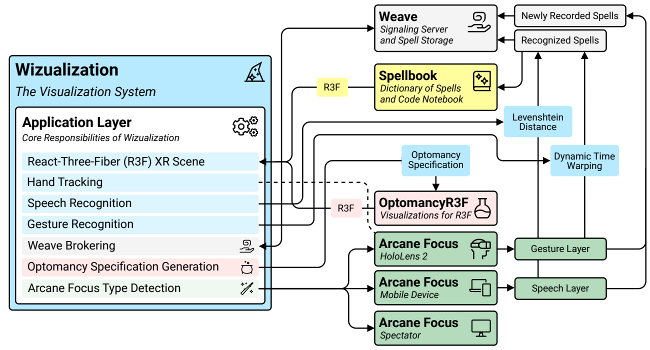


Fig. 3: **The inner workings of Wizualization.** The core responsibilities of Wizualization include: i) rendering the `react-three-fiber` WebXR scene, ii) enabling hand tracking on XR headsets, iii) speech and gesture recognition for casting and creating new spells, iv) brokering socket connections between Arcane Focuses via Weave, v) generating Optomancy specifications from spell input, and vi) presenting the appropriate user interface to each type of connected Arcane Focus.

Setting Up the Wizualization Session (Fig 1.I)

When PAUL arrives at work in the morning, he puts on his HoloLens 2 and fires up Wizualization and navigates to a room managed by Weave. He takes out his smartphone and navigates to the same room, and then does the same for his PC workstation. Weave restores his workspace based on the room he navigated to and uses his current whereabouts (his office) to organize the view. He uploads the limited-access up-to-date NHANES dataset into the workspace. Then he quickly surveys the visualizations presently occupying his workspace, deciding which ones to keep and which new ones to create. Given the scale and complexity of the NHANES dataset collection, Paul is particularly thankful for the virtually unlimited display space that the immersive workspace is providing him for his analysis.

4.2 Cross-Virtuality: Arcane Focuses & Weave

We discuss some examples of cross-device synergies in SA—or cross-virtuality analytics [26]—in Sec. 2.2. Our Node.js signaling server, Weave, acts as a relay linking all devices—or Arcane Focuses—inhabiting a given room via `socket.io`[9] connections. The signaling server can be layered over a database of existing spells and Optomancy specifications. Its two primary roles are:

1. **Device linking:** Wizualization runs on client devices and passes spoken commands and gestures, as well as recognized spell IDs, to Weave, which are used to connect all devices in the same room (i.e., connected via URLs with room IDs); and
2. **Room management and spell storage:** Weave is also the mechanism by which project spaces are organized in the form of rooms. Stored lists of spells—each assigned a unique ID by Weave—and Optomancy specification files are linked to a given room, which determines the initial layout of the visualization scene. The view is then updated via device signaling.

In other words, Weave is the glue connecting all devices, and acts as a persistence layer for changes to each room. It is a relatively lightweight, but essential, element of the overarching system.

The system is inherently collaborative, supporting asynchronous actions (spells/gestures). Users joining a room will see existing state from previous spell-casting sessions and, if they are joining with a device that supports gestures and/or speech, can continue adding or modifying the scene. This is similar to a user observing edits (as spectator) on a Google Doc[10], made by others, and choosing to edit sections themselves (even asynchronously).

[4] https://docs.pmnd.rs/react-three-fiber
[5] https://threejs.org
[6] https://immersive-web.github.io/webxr
[7] https://wicg.github.io/speech-api
[8] https://ar-js-org.github.io/AR.js-Docs

[9] https://socket.io
[10] https://docs.google.com

Configuring the Workspace

PAUL uses his hands to delete views that are no longer needed using gestures based on American Sign Language. Then he arranges the remaining views floating in mid-air around his office into a comfortable arrangement by directly grabbing and moving them. Finally, he pulls up the Diet Behavior and Nutrition (DBQ) and the COVID-19 (COQ) questionnaire components, the latter new for the 2021-2022 NHANES cycle, into his workspace as database glyphs floating in front of him.

4.3 Spellbook: An eXtended Reality Code Notebook

Spellbook is a learning support tool for Optomancy as well as a control mechanism comparable to a code notebook within Wizualization (Fig. 4(a)). With Spellbook, an analyst can use a collection of predefined gestures or sequences of spoken words to create visualizations of their data. They may also choose to craft their own spells—i.e., define their own methods, as one might record a macro—using gestures or spoken words. We use the visual metaphor of—naturally—a book as the interface for Spellbook, which serves three purposes:

Linked spell blocks representing each step of the spell sequence: While traditional code notebooks, such as Jupyter Notebook, typically present blocks of code in a scrollable, top-down page, Spellbook presents the user with floating windows, each of which are visually linked via a curve with the block that precedes it (Fig. 4(c), (d) and (e)). Spell blocks represent the user's workflow. The blocks are a collapsible element at the bottom of the right-hand page of the book (Fig. 4(b)). Once expanded, they can be splayed out in a horizontal sequence in front of the user and moved by pinching and dragging (Fig. 4(c)).

Variable selection: For the sake of precision, we provide a direct manipulation interface for selecting the variable to act as the target for new object creation spells (e.g., adding an axis). A menu of datasets (as tabs) and the variables contained in each dataset (as a list of buttons) is located at the top of the right-hand page (Fig. 4(a)). Selecting a variable button sets it as the target for the next object creation spell.

Compendium of primitives: Wizualization and Optomancy allow the user to record and translate gestures, speech, and materials into specification files. We have pre-recorded and loaded essential gestural and spoken primitives into Spellbook. A glossary of these primitives is represented in the book on the left-hand page (Fig. 4(a)).

Each spell block represents either a gesture or spoken command, or an element of the Optomancy specification generated by the user's interactions. Once expanded, spell blocks can be arranged anywhere in the space surrounding the user (Fig. 4(d) and (e)). The dashed curve linking each Spellbook block (Fig. 4(c), (d) and (e)) is animated such that line segments flow from each earlier step of the user's workflow, into the subsequent step. In this way, Spellbook represents the room's analysis history in the order in which it was performed. A two-handed gesture tapping the thumb of one hand with the little finger of the other reverts the workflow to the step before the tapped block. While considerable prior work has been conducted on the structuring and visualization of user workflows—for example, work by Maguire et al. [46], which implements and evaluates the Java tool *AutoMacron* designed to do exactly that—we limit the scope of Spellbook by restricting it to being a lightweight component library with linear representations of interactions expressed in the Optomancy grammar.

For our gestural primitives, we use conceptually-related words from American Sign Language (ASL), and for the corresponding spoken primitives, we use the English corollaries to the ASL word. An overview of the Spellbook primitives are listed in Tab. 1. Each primitive is stored in the data format described in Fig. 4(f). We have opted to use ASL due to its widespread adoption among sign language speakers and the consequent ready availability of learning tools and dictionaries for the language (e.g., `https://www.handspeak.com`). A glossary of the primitives (cf. Tab. 1) is given on the left-hand page of Spellbook.

Gestural Programming with Spell Blocks

PAUL wants to evaluate the steps he took throughout his earlier DBQ and COQ analysis. This mainly involves directly pinching and dragging the linked code blocks—or Spell Blocks—generated by Spellbook to retrace his steps and find the surprising correlation between the two tables that he found during his analysis. It looks like the COVID-19 pandemic has significantly harmed American diets, particularly in rural regions, but Paul is unsure why. He checks his analysis through a combination of hand gestures and spoken commands, the actions long since having become familiar to him. A visual representation of the current command sequence helps him keep track of what he is doing and where in the spell sequence he is editing. Gestures enable him to navigate back and forth in the sequence.

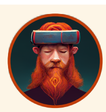
4.4 Optomancy: The Grammar of Wizualization

Wizualization's visualization output is powered by Optomancy, a grammar of graphics for the web, and OptomancyR3F, its `react-three-fiber` renderer for immersive and ubiquitous analytics applications. Like many of the visualization grammars mentioned in Sec. 2.4, Optomancy accepts a declarative JSON format (Fig. 5).

Optomancy takes this declarative JSON specification and compiles it into a full specification inferring sensible default values for missing properties. OptomancyR3F acts as an interpreter between Wizualization clients and Optomancy, taking the full specification from Optomancy and constructing visualizations as R3F components.

Visualizations in Wizualization are constructed in stages, with each spell block representing a chunk of the resulting Optomancy specification. When partial specifications are passed into Optomancy, it will generate some visual response for each recognized spell, as visual feedback at every step of the crafting process is vital for user understanding and discussion in solo or collaborative sessions. For example, if the user casts an axis into an empty view, it will display a 1-dimensional chart, even if the user or users intend to create multi-dimensional charts.

Furthermore, Optomancy allows for the visualization of multiple datasets within the same scene, meaning new data sources can be loaded in at any time without requiring a completely separate scene to be initialized. This separation is possible via the grammar's *workspaces* construct. Optomancy configurations can contain multiple workspaces (each with a distinct dataset), multiple views, and multiple layers allowing for a broad range of view compositions. At present, Optomancy supports the creation of simple Cartesian plots, in a similar way to other IA/SA toolkits and frameworks (e.g., [12,68]), with more types planned for the future. Nevertheless, as indicated in Sec. 4.3, Wizualization through Spellbook supports the recording of new spells that may yield new visualization types. The requirement for the user is to define appropriate gestures and voice commands that intuitively map to each visualization type.



Manual Magic with the Optomancy Specification

PAUL, who has learned the rules of Optomancy in part via Spellbook, is expecting to connect with his co-worker Ana in the field to track down real-world data supporting his analysis. Since Ana will be mobile, Paul decides to manually construct an Optomancy specification to streamline her data entry and curation. He installs the full Wizualization system stack on his local PC workstation from the public repository, creates an example specification file, and modifies the Wizualization application to use the example specification as a read-only presentation. Then he invites his co-workers to join the workspace.

5 Using Wizualization

We have already discussed the system architecture and the individual components. In this section, we present not just the interactive and dynamic usage of the system, but also our model for ubiquitous, immersive, and situated analytics for the web.

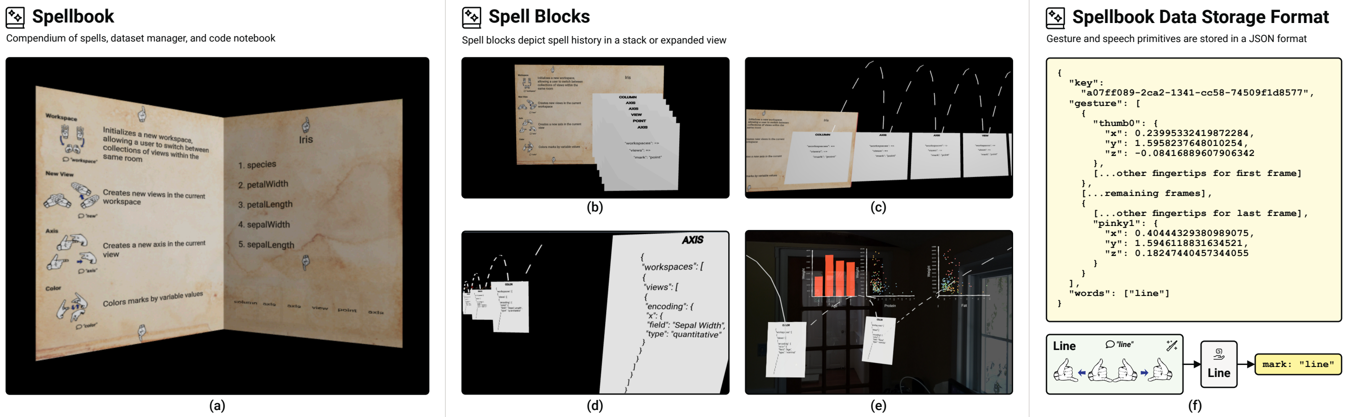


Fig. 4: **Spellbook, spell blocks, and the Spellbook data storage format.** Spellbook has multiple functions within Wizualization: (a) It acts as a learning tool for available spells within Spellbook (left page), a variable selection tool for subsequent spells (top of right page), and a code notebook comprising spell blocks depicting the history of cast spells in the session (bottom of right page). (b) Tapping the spell history at the bottom of the right page reveals a stack of spell blocks. (c) Tapping the stack of spell blocks further expands them in a line linked by curves. All spell blocks are linked in a temporal sequence based on the actions performed by the user during analysis. (d) + (e) Casting further spells adds spell blocks to the list that can then be dragged and rearranged within the scene by the analyst. (f) When a spell is recorded, a sequence of finger positions is stored alongside a list of associated spoken words that can be used in their place. Spells such as the one in this figure representing the ASL for "line" are encoded in a JSON format and stored in Weave for recall, with a selection of what we consider essential primitives stored in Spellbook in this format as well.

Overall, Wizualization was designed to support multimodal input, allowing for different input modalities to accomplish the same thing. While gesture and voice interaction are the main input modalities with HMDs, speech interactions are also exposed via the smartphone speech interface.

5.1 Indirect User Input

Wizualization detects both the words spoken by the user and the user's hand position and rotation, applying algorithms to identify the closest match to the spell that the user is casting. The currently selected Optomancy specification (Fig. 5) is updated dynamically as the user interacts with the system; such updates are propagated via Weave.

5.1.1 Somatic Components (Gestures)

Mid-air gestures in Wizualization are represented as unique objects assigned by Weave; the sequence of recognized fingertip positions is stored as an array of objects every tenth frame, which we found to be sufficient for our purposes. In conjunction with the verbal component, a spell sent to and returned from Weave takes the structure exemplified in Fig. 4(f), with fingertip positions between the first position of the first fingertip (the left hand thumb) at the start and the last position of the last fingertip (the right hand pinky) at the end omitted for brevity.

To interpret the user's gestures, Wizualization uses an implementation of the Dynamic Time Warping (DTW) algorithm (Fig. 3) [50]. Specifically, we use time series of forward vectors from the user's headset to all recognized fingertips as inputs to the algorithm, and selects the gesture with the minimum distance value output by the algorithm.

We use DTW for two reasons: First, it is appropriate for the problem—i.e., matching finger pose time series to existing recorded series that may be of differing length but exhibit the same general pattern. It is what the algorithm was designed for [7], and it is well-vetted, having been oft-used for nearly three decades up to the present-day for gesture matching applications exactly like our own [42,45,71]. Second, performance: It efficiently selects a best match among a number of options with reasonable accuracy and very little overhead [14]. Fig. 6 describes the gestures to initialize spell casting and recording.

5.1.2 Verbal Components (Spoken Commands)

Wizualization also recognizes the user's spoken commands. For the purpose of recognizing which spell the user is casting, we use the Levenshtein edit distance to enable "fuzzy" matching of existing spells (Fig. 3). This done by calculating the number of transformations—character insertion, substitution, or deletion—needed to turn the string of characters representing that spell's spoken words into each of the words among the stored spells' keywords, and then selecting the spell with the fewest transformations (i.e., the closest match). We opted to use the Levenshtein distance mainly for transparency (it is an easily explainable technique) as well as accuracy.

Field Data Collection and Curation (Fig 1.II)

ANA is a survey statistician with the CDC. Last night, she traveled to Fargo, ND to collect supporting data for Paul's project on the impact of COVID-19 on diets in rural regions. She is now braving the cold and dark fall morning to visit small groceries in rural communities surrounding Fargo. XR on her HoloLens 2 allows her to perform data collection and curation to supplement the DBQ component even in the field and while walking from her car. Her workspace is mobile and centered around her rather than fixed to the world. Since she is wearing gloves, she uses spoken commands to control her workspace and navigate the interactive map. She can use the visualization spells Paul has already prepared for her in the workspace, making it easy to see geographic areas where more data collection is needed.

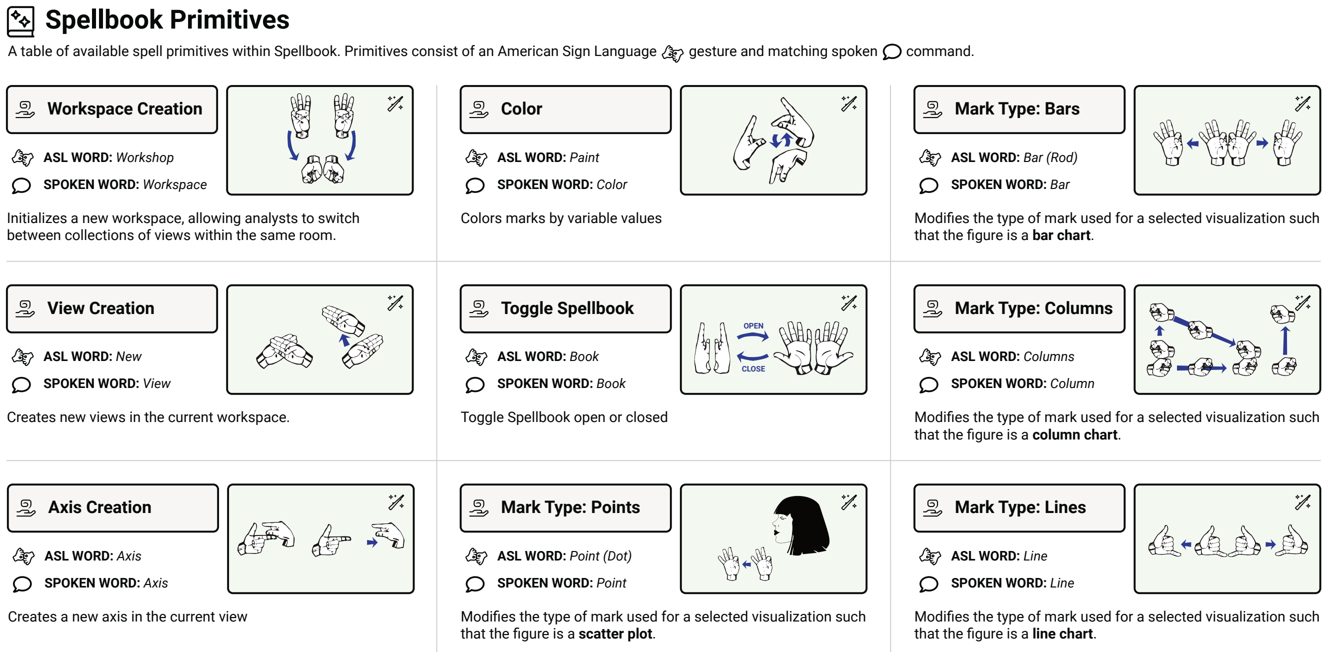
Because the Microsoft HoloLens 2 did not support the use of the Web Speech API in the Edge browser at the time of our system design—and the Edge browser is the only fully WebXR compatible browser that can be used on the HoloLens 2—we opted to offload verbal input to the user's handheld Arcane Focus (e.g., a smartphone). The Arcane Focus transmits spells spoken by the user to Weave, which then relays it to all other devices connected to the room.

5.1.3 Material Components ("Enchanted Items")

Optomancy supports the use of material components by treating virtual and real objects as triggers for spells with a focus on Optomancy specification updates that support situating visualizations in the real space around the user. Typically this is done using a barcode or fiducial marker on the object to uniquely identify such enchanted items. This identifier is sent to Weave as a parameter that is treated differently from the verbal and somatic components, since it has a natural spatial mapping that can be used to situate virtual objects in the scene.

Material components can thus trigger events similar to gestures or verbal commands, and can also be used to position and orient the results

Table 1: **Spellbook primitives**. Overview of the spoken and gestural primitives included in Spellbook.



Spellbook Primitives

A table of available spell primitives within Spellbook. Primitives consist of an American Sign Language gesture and matching spoken command.

Primitive	ASL WORD	SPOKEN WORD	Description
Workspace Creation	*Workshop*	*Workspace*	Initializes a new workspace, allowing analysts to switch between collections of views within the same room.
Color	*Paint*	*Color*	Colors marks by variable values
Mark Type: Bars	*Bar (Rod)*	*Bar*	Modifies the type of mark used for a selected visualization such that the figure is a **bar chart**.
View Creation	*New*	*View*	Creates new views in the current workspace.
Toggle Spellbook	*Book*	*Book*	Toggle Spellbook open or closed
Mark Type: Columns	*Columns*	*Column*	Modifies the type of mark used for a selected visualization such that the figure is a **column chart**.
Axis Creation	*Axis*	*Axis*	Creates a new axis in the current view
Mark Type: Points	*Point (Dot)*	*Point*	Modifies the type of mark used for a selected visualization such that the figure is a **scatter plot**.
Mark Type: Lines	*Line*	*Line*	Modifies the type of mark used for a selected visualization such that the figure is a **line chart**.

of the method they trigger via direct manipulation. We opted to keep the usage of material components relatively simple and limited for the time being, with an eye toward future extension both by users and in later iterations of the Wizualization system.

Data-Driven Storytelling (Fig. 1.III)

Meanwhile, CARLOS, who is Paul and Ana's boss, is sitting in an auditorium listening to a talk at the CDC's headquarters in Atlanta, GA while preparing a statement to journalists about the new study. His HMD allows him to keep an eye on the speaker while skimming through views that Paul and Ana are constructing in their common workspace. Since he is unable to both speak and gesture in the auditorium, he uses his smartphone that has been woven into Wizualization to interact with the 3D workspace suspended in mid-air in front of him. Carlos, a trained economist, is troubled by the poor nutrition of groceries in the rural supermarkets Ana is surveying, and his mind immediately goes to supply chains affected by the pandemic.

5.2 Interactions and Spell Chaining

One design consideration for Wizualization has been iterative simplification of the user's workflows. Suppose that a user finds themselves repeating the same sequence of tasks to create a visualization:

1. Set the mark type as point;
2. Select two scalar variables and join them into a 2D scatterplot;
3. Set the color variable for the 3D scatterplot; and
4. Add a third variable and create a 3D scatterplot.

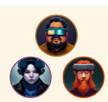
This is the workflow depicted in Fig. 5. What if the user could reduce this to a single gesture or spoken word? In our implementation, this is handled via Weave (Fig. 3); the iterative layering nature of Optomancy supports the reduction of analytical task components into abstracted user inputs. Once a spell has been crafted by the user in Wizualization then stored and propagated via Weave, it cannot only be cast at will, but can also be folded into future user-crafted spells.

Thus, a sequence like the above can be reduced to a simple phrase or wave of the hand. Even without this iterative simplification, IA systems have been praised by users as accelerating the creation of graphical representations of data compared to traditional 2D environments [5].

5.3 Collaboration in Cross-Virtuality Infrastructure

The necessity of synchronizing multiple devices for a single user has an implicit beneficial byproduct: It becomes trivial to support synchronous collaboration, both remotely and in co-located settings. Wizualization does not distinguish between individual users; rather, it presents all users with a shared room with a layout determined by the latest version of the Optomancy specification, which is updated through signals propagated across Arcane Focuses by Weave via `socket.io` messages. Thus, all users on all devices are privy to a continuously updated shared view—each room plays the role of an XR visualization equivalent to a Google Doc in terms of synchronization frequency.

Fig. 7 describes a typical collaborative user scenario. In a collaborative scenario, all users' spells are appended to the same spell sequence, which updates the view globally for all users in the room. Users interact freely with elements of the scene at their individual discretion; the active workspace or view is local to each user, but the grammar specification is global for the room. Each user can cast spells independently into the sequence, and then the recognized spell returns new spell blocks from Spellbook and visualizations from OptomancyR3F for all users. Any number of globally disparate spell casters can create a shared view of the datasets loaded in their room and discuss them in real time using any additional collaborative conferencing software of their choice as shown in the Fig. 7 example, or they can work asynchronously.



Synchronous & Remote Collaboration (Fig 1.IV & V)

CARLOS realizes that he lacks vital information for his upcoming press conference, so he leaves the talk to call an impromptu meeting with PAUL in Maryland and ANA in North Dakota. Wizualization supports synchronous remote collaboration, enabling all three to view Paul's workspace as he walks them through his data analysis. Paul agrees with Carlos' suspicion about supply chain disruptions caused by the pandemic, and is quickly able to pull up data from the U.S. Bureau of Economic Analysis (BEA) limited-access data on national logistics to confirm the suspicion. Then Ana presents the most recent findings from her ongoing supermarket survey. Her data shows that rural supermarkets are still lacking fresh produce and dairy, but that the supply chain situation is slowly improving. Satisfied, Carlos signs off and heads out to meet the press.

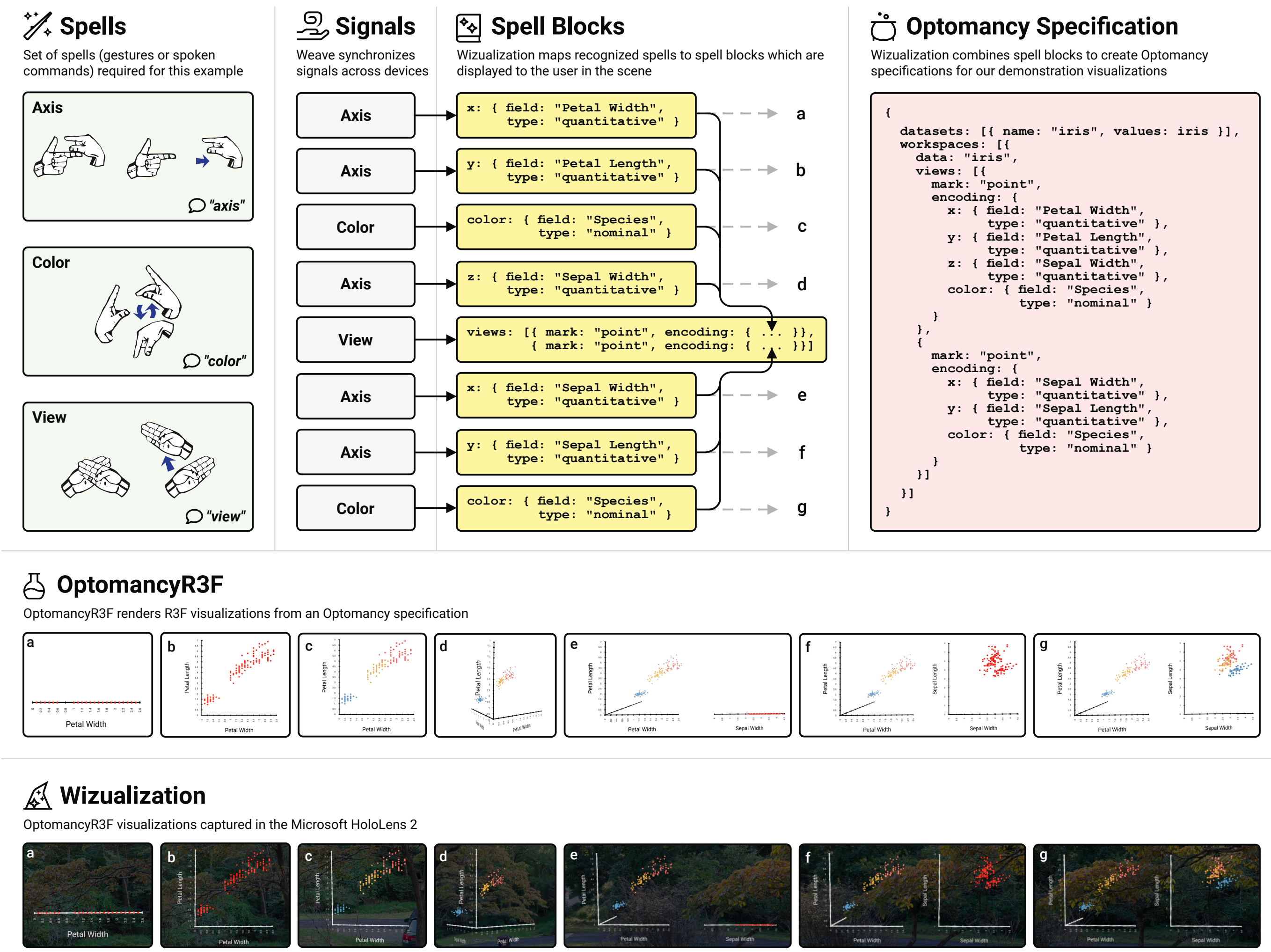


Fig. 5: **Example user workflow.** Once the user has loaded their dataset(s) into Wizualization, they may then begin crafting. The user can choose to cast spells via spoken commands or gestures (or both). Each recognized spell will return components from Spellbook to display the appropriate code block to the user in a set of connected spell blocks in the XR scene. The contents of each spell block are added to the room's Optomancy specification and the resulting visualization is produced in front of all connected users after each iteration.

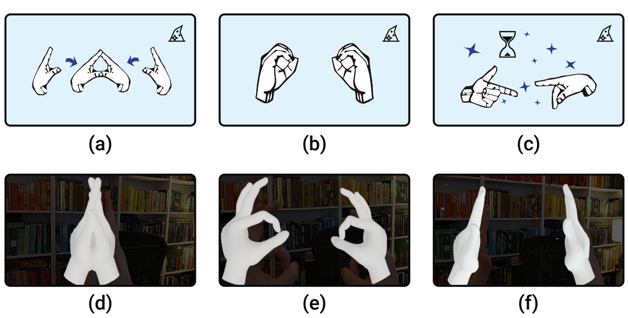


Fig. 6: **Gesture recognition.** The clapping gesture (a) and (d)—or any gesture that involves touching all opposite-hand fingertips recognized in the view—initializes a spell-casting period, while the double-pinch gesture (b) and (e) initializes a gesture recording period that will allow for macro creation (c). When the gesture recording period begins, after a 1-second delay, a sparkling aura appears around the user's hands. The white gloved hands shown in (d), (e), and (f), while removed in the final version of Wizualization, are left to better illustrate the use of hand pose tracking data as a central element of user interaction with the system.

6 Discussion

We base the magic approach in this paper on the fact that metaphors have power, both in terms of knowledge transfer and familiarity, but also for engaging and motivating the user. As a case in point, the desktop metaphor for personal computing—with icons, windows, files, folders, and trashcans—has persisted and empowered users since its introduction. And this despite significant early (and also more recent) criticism against the use of such metaphors. In addition, the supernatural and the fantastical—that which is beyond the visible and observable—are extraordinarily fascinating to humans, and serve as perfect catalysts for immersive analytics. This notion even has support in visualization literature given the recent Best Paper at IEEE InfoVis 2021, where Willet et al. [78] introduce the idea of superpowers as inspiration and motivation for specific data visualization features.

It is also worth asking what limitations may be introduced by basing our work on a magic metaphor. Are there specific features, transformations, or visualizations that are impractical in such a system? One such potential limitation is that 3D visualizations in general are plagued by several challenges, such as occlusion, perspective foreshortening, legibility, reach, and the need for 3D navigation. However, this challenge is not unique to our system, and, besides, many of these problems can be mitigated [47]. Another, more relevant, limitation is the discoverability and usability of both voice commands as well as gestures. As noted by Norman [54], "*a pure gestural system makes it difficult to discover*

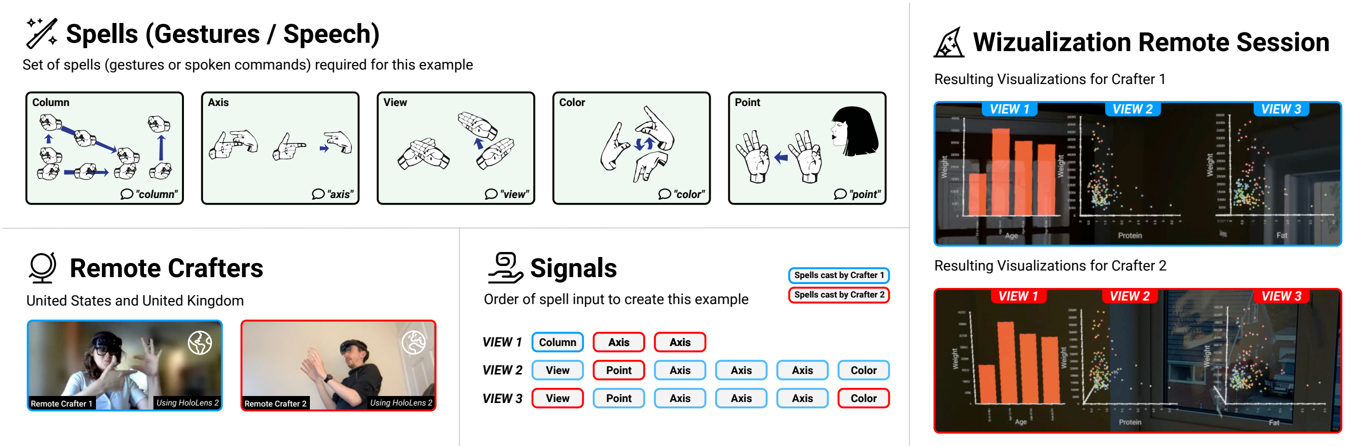
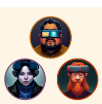


Fig. 7: **Collaborative user flow.** In this scenario, a user from the United States is analyzing a dataset in collaboration with a user from the United Kingdom over a video call while sharing a Wizualization room. Remote Crafter 1 (highlighted in blue) casts a spell specifying a column. Remote Crafter 2 (highlighted in red) casts an axis spell to add an axis to a view; then, he casts a second. Crafter 1 creates a new view, then Crafter 2 specifies that it is to be a scatter plot using the point gesture, and so on and so forth as illustrated in the "Signals" pane until they ultimately create the selection of views represented in the "Wizualization Remote Collaboration Session" pane.

the set of possibilities" that the system provides. The same is true of spoken commands. We attempt to address this in the paper by both providing visual guides, as well as basing our gestures on American Sign Language and as common spoken words, respectively.[11] However, addressing discoverability in "natural interfaces" is not our focus and we do not claim that our mitigation strategies are optimal.

We should also acknowledge that gestures, while consistent with the magic metaphor and well-suited to the AR environment, are also associated with interaction fatigue [30]. Fortunately, Wizualization supports multiple interaction modalities, including both speech and touch interaction, which reduces the impact of such interaction fatigue.

Of course, the present paper is primarily an engineering contribution and presents no empirical evidence supporting any of our claims about the utility of XR visualization authoring. Rather, our value proposition is qualitative: That visualization authoring in the field and on-the-go is only consistent with simple touchscreen interaction, mid-air gestures, and verbal commands in XR, and that magic is as effective a metaphor as any (and more engaging than most) to guide users in this task. Fastidiously typing visualization specifications in JSON, or even selecting visualization templates or building blocks from a touchscreen menu, is not. While we agree that it would be productive to conduct empirical user studies to guide specific design choices or validate the utility of specific features, we firmly believe that the overall utility of our Wizualization system is sound.



Paul, Ana & Carlos — Meeting the Press

The time has come for CARLOS to meet the press. While PAUL and ANA have not completed their data collection and analysis, the American public is clamoring for news about the food deserts in rural America whose expansion have only been accelerated by the onset of the COVID-19 pandemic. Carlos wears his Apple Vision Pro to the press conference because it allows him to use his hands to pull up Paul's and Ana's workspaces and present the different visualizations to the cameras. The audience can see the workspace from his point of view. Paul and Ana remain in voice communications with him and part of the Wizualization session so that they can aid his presentation, both by feeding him information in real-time in response to questions from reporters as well as by constructing new visualizations on the fly. In the end, Carlos is able to reassure an anxious public that while rural diets have indeed been disrupted by disease, social distancing, and poor logistics, the supply chains are being reestablished. After all, winter is coming. THE END.

[11] Happily, no *Expecto Patronum!* for us.

7 Conclusion and Future Work

We have presented WIZUALIZATION, an XR system for visualization authoring based on mid-air gestures and speech dressed up in a hard magic metaphor. Using this system, a user can create data visualizations in 3D space using a combination of hand gestures, verbal commands, and data bindings using an XR HMD such as a Microsoft HoloLens 2 for seamless ubiquitous analytics anywhere and anytime. We continue the magic metaphor to the components that make up the overall system: (i) *Optomancy* is our grammar of graphics for visualization; (ii) *Spellbook* is an XR analytical notebook and collection of gestures and verbal commands for casting various actions (spells); (iii) *Arcane Focus*es integrate "magical artifacts" (smartphones) into the system; and (iv) *Weave* propagates signals between the physical components. We have described each of these different components in depth and explained their roles in the overall system. We have also demonstrated Wizualization in a motivating example showcasing how it can be used to author immersive, situated, and ubiquitous analytics experiences.

We see significant potential for future work in this area. As discussed earlier, we anticipate conducting laboratory and field studies to validate the utility of the Wizualization system. Toolkit and system evaluation is often difficult because of ecological validity demands, lack of comparison baselines, training requirements, etc. Therefore, we anticipate turning to work such as that of Ledo et al. [39] for inspiration. Beyond evaluation, there is opportunity for new visual representations, new mid-air interaction techniques, and new analytics methods that are specialized for mobile analytics. In addition, many questions remain. What is the equivalent of a dashboard in a 3D workspace? How do you help users discover and remember gestures and commands? To what degree do current and future SA systems support the analytical process when it is conducted in situ [66]? And how do you overcome challenges intrinsic to 3D such as occlusion, distortion, and legibility? We look forward to building on the Wizualization system as we continue to answer such questions in our future research.

Acknowledgments

This work was supported partly by grant IIS-1908605 from the U.S. National Science Foundation, and partly by the DSP Centre funded by the ERDF through the Welsh Government. Any opinions, findings, and conclusions expressed in this material are those of the authors and do not necessarily reflect the views of the funding agency.

References

[1] C. Ahlberg, C. Williamson, and B. Shneiderman. Dynamic queries for information exploration: An implementation and evaluation. In *Proc. ACM CHI*, pp. 619–626. ACM, New York, 1992. doi: 10.1145/142750.143054 2

[2] B. Bach, R. Sicat, J. Beyer, M. Cordeil, and H. Pfister. The Hologram in My Hand: How Effective is Interactive Exploration of 3D Visualizations in Immersive Tangible Augmented Reality? *IEEE Trans Vis Comput Graph*, 24(1):457–467, 2018. doi: 10.1109/TVCG.2017.2745941 2

[3] S. K. Badam, F. Amini, N. Elmqvist, and P. Irani. Supporting visual exploration for multiple users in large display environments. In *Proc. IEEE VAST*. IEEE Computer Society, Los Alamitos, 2016. doi: 10.1109/VAST.2016.7883506 2, 3

[4] S. K. Badam, A. Srinivasan, N. Elmqvist, and J. Stasko. Affordances of input modalities for visual data exploration in immersive environments. In *Proc. IEEE VIS Workshop on Immersive Analytics*, 2017. 3

[5] A. Batch, A. Cunningham, M. Cordeil, N. Elmqvist, T. Dwyer, B. H. Thomas, and K. Marriott. There is no spoon: Evaluating performance, space use, and presence with expert domain users in immersive analytics. *IEEE Trans Vis Comput Graph*, 26(1):536–546, 2020. doi: 10.1109/TVCG.2019.2934803 7

[6] A. Batch, S. Shin, J. Liu, P. W. S. Butcher, P. D. Ritsos, and N. Elmqvist. Evaluating View Management for Situated Visualization in Web-based Handheld AR. *Comput Graph Forum*, 42(3):349–360, 2023. doi: 10.1111/cgf.14835 2

[7] D. J. Berndt and J. Clifford. Using dynamic time warping to find patterns in time series. In *Proc. ACM KDD Workshop on Knowledge Discovery in Databases*, pp. 359–370, 1994. 6

[8] L. Besançon, A. Ynnerman, D. F. Keefe, L. Yu, and T. Isenberg. The state of the art of spatial interfaces for 3D visualization. *Comput Graph Forum*, 40(1):293–326, 2021. doi: 10.1111/cgf.14189 2

[9] R. A. Bolt. "Put-That-There": Voice and gesture at the graphics interface. In *Proc. SIGGRAPH*, pp. 262–270. ACM, New York, 1980. doi: 10.1145/800250.807503 2

[10] W. Büschel, J. Chen, R. Dachselt, S. Drucker, T. Dwyer, C. Görg, T. Isenberg, A. Kerren, C. North, and W. Stuerzlinger. *Interaction for Immersive Analytics*, pp. 95–138. Springer International Publishing, Cham, 2018. doi: 10.1007/978-3-030-01388-2_4 2

[11] W. Büschel, A. Lehmann, and R. Dachselt. MIRIA: A mixed reality toolkit for the in-situ visualization and analysis of spatio-temporal interaction data. In *Proc. ACM CHI*. ACM, New York, 2021. doi: 10.1145/3411764.3445651 2

[12] P. W. S. Butcher, N. W. John, and P. D. Ritsos. VRIA: A web-based framework for creating immersive analytics experiences. *IEEE Trans Vis Comput Graph*, 27(7):3213–3225, 2021. doi: 10.1109/TVCG.2020.2965109 2, 3, 5

[13] S. Butscher, S. Hubenschmid, J. Müller, J. Fuchs, and H. Reiterer. Clusters, trends, and outliers: How immersive technologies can facilitate the collaborative analysis of multidimensional data. In *Proc. ACM CHI*, pp. 1–12. ACM, New York, 2018. doi: 10.1145/3173574.3173664 2

[14] F. Chin-Shyurng, S.-E. Lee, and M.-L. Wu. Real-time musical conducting gesture recognition based on a dynamic time warping classifier using a single-depth camera. *Applied Sciences*, 9(3), 2019. doi: 10.3390/app9030528 6

[15] A. C. Clarke. *Profiles of the Future: An Inquiry Into the Limits of the Possible*. Harper & Row, New York, 1972. 1

[16] M. Cordeil, A. Cunningham, B. Bach, C. Hurter, B. H. Thomas, K. Marriott, and T. Dwyer. IATK: an immersive analytics toolkit. In *Proc. IEEE VR*, pp. 200–209. IEEE Computer Society, Los Alamitos, 2019. doi: 10.1109/VR.2019.8797978 2

[17] M. Cordeil, A. Cunningham, T. Dwyer, B. H. Thomas, and K. Marriott. ImAxes: Immersive axes as embodied affordances for interactive multivariate data visualisation. In *Proc. ACM UIST*, pp. 71–83. ACM, New York, 2017. doi: 10.1145/3126594.3126613 1, 2, 3

[18] M. Csikszentmihalyi. *Flow: The Psychology of Optimal Experience*. Harper Collins, New York, 1991. 2

[19] W. Cui, X. Zhang, Y. Wang, H. Huang, B. Chen, L. Fang, H. Zhang, J.-G. Lou, and D. Zhang. Text-to-Viz: Automatic generation of infographics from proportion-related natural language statements. *IEEE Trans Vis Comput Graph*, 26(1):906–916, 2020. doi: 10.1109/TVCG.2019.2934785 3

[20] N. Elmqvist and P. Irani. Ubiquitous Analytics: Interacting with Big Data Anywhere, Anytime. *Computer*, 46(4):86–89, 2013. doi: 10.1109/MC.2013.147 1

[21] N. Elmqvist, M. E. Tudoreanu, and P. Tsigas. Evaluating motion constraints for 3D wayfinding in immersive and desktop virtual environments. In *Proc. ACM CHI*, pp. 1769–1778. ACM, New York, 2008. doi: 10.1145/1357054.1357330 2

[22] N. Elmqvist, A. Vande Moere, H.-C. Jetter, D. Cernea, H. Reiterer, and T. Jankun-Kelly. Fluid interaction for information visualization. *Information Visualization*, 10(4):327–340, Oct. 2011. doi: 10.1177/1473871611413180 2

[23] K. Fennedy, J. Hartmann, Q. Roy, S. T. Perrault, and D. Vogel. OctoPocus in VR: Using a dynamic guide for 3D mid-air gestures in virtual reality. *IEEE Trans Vis Comput Graph*, 27(12):4425–4438, 2021. doi: 10.1109/TVCG.2021.3101854 2

[24] J. A. W. Filho, W. Stuerzlinger, and L. Nedel. Evaluating an immersive space-time cube geovisualization for intuitive trajectory data exploration. *IEEE Trans Vis Comput Graph*, 26(1):514–524, 2020. doi: 10.1109/TVCG.2019.2934415 2

[25] P. Fleck, A. Sousa Calepso, S. Hubenschmid, M. Sedlmair, and D. Schmalstieg. RagRug: A toolkit for situated analytics. *IEEE Trans Vis Comput Graph*, 29(7):3281–3297, 2023. doi: 10.1109/TVCG.2022.3157058 2

[26] B. Fröhler, C. Anthes, F. Pointecker, J. Friedl, D. Schwajda, A. Riegler, S. Tripathi, C. Holzmann, M. Brunner, H. Jodlbauer, H.-C. Jetter, and C. Heinzl. A survey on cross-virtuality analytics. *Comput Graph Forum*, 41(1):465–494, 2022. doi: 10.1111/cgf.14447 2, 4

[27] B. D. Hall, L. Bartram, and M. Brehmer. Augmented chironomia for presenting data to remote audiences. In *Proc. ACM UIST*, pp. 18:1–18:14. ACM, New York, 2022. doi: 10.1145/3526113.3545614 3

[28] A. G. Hauptmann and P. McAvinney. Gestures with speech for graphic manipulation. *International Journal for Man-Machine Studies*, 38(2):231–249, 1993. doi: 10.1006/imms.1993.1011 2

[29] S. He, M. L. Huang, and L. Zhu. Natural user interface design in DA-TU: An interactive clustered data visualization system. In *Proc. IV*, pp. 83–88. IEEE Computer Society, Los Alamitos, 2015. doi: 10.1109/iV.2015.25 2

[30] J. D. Hincapié-Ramos, X. Guo, P. Moghadasian, and P. Irani. Consumed endurance: a metric to quantify arm fatigue of mid-air interactions. In *Proc. ACM CHI*, pp. 1063–1072. ACM, New York, 2014. doi: 10.1145/2556288.2557130 9

[31] S. Hubenschmid, J. Wieland, D. I. Fink, A. Batch, J. Zagermann, N. Elmqvist, and H. Reiterer. ReLive: Bridging in-situ and ex-situ visual analytics for analyzing mixed reality user studies. In *Proc. ACM CHI*, pp. 24:1–24:20. ACM, New York, 2022. doi: 10.1145/3491102.3517550 2

[32] S. Hubenschmid, J. Zagermann, S. Butscher, and H. Reiterer. STREAM: Exploring the combination of spatially-aware tablets with augmented reality head-mounted displays for immersive analytics. In *Proc. ACM CHI*. ACM, New York, 2021. doi: 10.1145/3411764.3445298 2

[33] T. Igarashi, S. Matsuoka, and H. Tanaka. Teddy: A sketching interface for 3D freeform design. In *Proc. SIGGRAPH*, pp. 409–416. ACM, New York, 1999. doi: 10.1145/311535.311602 2

[34] S. Jang, N. Elmqvist, and K. Ramani. MotionFlow: Visual abstraction and aggregation of sequential patterns in human motion tracking data. *IEEE Trans Vis Comput Graph*, 22(1):21–30, 2016. doi: 10.1109/TVCG.2015.2468292 2

[35] S. Kallio, J. Kela, J. Mäntyjärvi, and J. Plomp. Visualization of hand gestures for pervasive computing environments. In *Proc. ACM AVI*, pp. 480–483. ACM, New York, 2006. doi: 10.1145/1133265.1133363 2

[36] D. F. Keefe, D. A. Feliz, J. Miles, F. Drury, S. Swartz, and D. H. Laidlaw. Scientific sketching for collaborative VR visualization design. *IEEE Trans Vis Comput Graph*, 14(4):835–847, 2008. doi: 10.1109/TVCG.2008.31 2

[37] C. Kim, P. Chiu, and Y. Tjahjadi. A web-based remote assistance system with gravity-aware 3D hand gesture visualization. In *Proc. ACM ISS*, pp. 319–322. ACM, New York, 2019. doi: 10.1145/3343055.3360742 3

[38] R. Langner, M. Satkowski, W. Büschel, and R. Dachselt. MARVIS: Combining mobile devices and augmented reality for visual data analysis. In *Proc. ACM CHI*. ACM, New York, 5 2021. doi: 10.1145/3411764.3445593 2

[39] D. Ledo, S. Houben, J. Vermeulen, N. Marquardt, L. Oehlberg, and S. Greenberg. Evaluation strategies for HCI toolkit research. In *Proc. ACM CHI*, pp. 36:1–36:17. ACM, New York, 2018. doi: 10.1145/3173574.3173610 9

[40] B. Lee, P. Isenberg, N. H. Riche, and S. Carpendale. Beyond mouse and keyboard: Expanding design considerations for information visualization interactions. *IEEE Trans Vis Comput Graph*, 18(12):2689–2698, 2012.

doi: 10.1109/TVCG.2012.204 2

[41] B. Lee, R. H. Kazi, and G. Smith. SketchStory: Telling more engaging stories with data through freeform sketching. *IEEE Trans Vis Comput Graph*, 19(12):2416–2425, 2013. doi: 10.1109/TVCG.2013.191 2, 3

[42] H. Li, J. Liu, Z. Yang, R. W. Liu, K. Wu, and Y. Wan. Adaptively constrained dynamic time warping for time series classification and clustering. *Inf. Sci.*, 534:97–116, 2020. doi: 10.1016/j.ins.2020.04.009 6

[43] J. Liao, A. Karim, S. S. Jadon, R. H. Kazi, and R. Suzuki. RealityTalk: Real-time speech-driven augmented presentation for AR live storytelling. In *Proc. ACM UIST*, pp. 17:1–17:12. ACM, New York, 2022. doi: 10.1145/3526113.3545702 3

[44] M. J. Lobo, C. Hurter, and P. Irani. Flex-ER: A platform to evaluate interaction techniques for immersive visualizations. *PACM*, 4(ISS):1–20, 2020. doi: 10.1145/3427323 2

[45] Y. Luo, J. Liu, and S. Shimamoto. Wearable air-writing recognition system employing dynamic time warping. In *Proc. IEEE Consumer Communications & Networking Conference*, pp. 1–6, 2021. doi: 10.1109/CCNC49032.2021.9369458 6

[46] E. Maguire, P. Rocca-Serra, S.-A. Sansone, J. Davies, and M. Chen. Visual compression of workflow visualizations with automated detection of macro motifs. *IEEE Trans Vis Comput Graph*, 19(12):2576–2585, 2013. doi: 10.1109/TVCG.2013.225 5

[47] K. Marriott, J. Chen, M. Hlawatsch, T. Itoh, M. A. Nacenta, G. Reina, and W. Stuerzlinger. Immersive Analytics: Time to Reconsider the Value of 3D for Information Visualisation. In *Immersive Analytics*, vol. 11190 of *Lecture Notes in Computer Science*, pp. 25–55. Springer, Berlin, Germany, 2018. doi: 10.1007/978-3-030-01388-2_2 8

[48] K. Marriott, F. Schreiber, T. Dwyer, K. Klein, N. H. Riche, T. Itoh, W. Stuerzlinger, and B. H. Thomas, eds. *Immersive Analytics*, vol. 11190 of *Lecture Notes in Computer Science*. Springer International Publishing, New York, 2018. doi: 10.1007/978-3-030-01388-2 1

[49] J. McCormack, B. Roberts, Jonathan C.and Bach, C. D. S. Freitas, T. Itoh, C. Hurter, and K. Marriott. *Multisensory Immersive Analytics*, pp. 57–94. Springer International Publishing, Cham, 2018. doi: 10.1007/978-3-030-01388-2_3 2

[50] M. Müller. Dynamic time warping. In *Information Retrieval for Music and Motion*, pp. 69–84. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007. doi: 10.1007/978-3-540-74048-3_4 6

[51] A. Narechania, A. Srinivasan, and J. Stasko. NL4DV: A toolkit for generating analytic specifications for data visualization from natural language queries. *IEEE Trans Vis Comput Graph*, 27(2):369–379, 2021. doi: 10.1109/TVCG.2020.3030378 3

[52] M. Nebeling, M. Speicher, X. Wang, S. Rajaram, B. D. Hall, Z. Xie, A. R. E. Raistrick, M. Aebersold, E. G. Happ, J. Wang, Y. Sun, L. Zhang, L. E. Ramsier, and R. Kulkarni. MRAT: The mixed reality analytics toolkit. In *Proc. ACM CHI*, pp. 1–12. ACM, New York, 2020. doi: 10.1145/3313831.3376330 2

[53] M. Nielsen, N. Elmqvist, and K. Grønbæk. Scribble query: fluid touch brushing for multivariate data visualization. In *Proc. OzCHI*, pp. 381–390. ACM, New York, 2016. doi: 10.1145/3010915.3010951 2

[54] D. A. Norman. Natural user interfaces are not natural. *Interactions*, 17(3):6–10, 2010. doi: 10.1145/1744161.1744163 8

[55] D. A. Norman and S. W. Draper, eds. *User Centered System Design: New Perspectives on Human-Computer Interaction*. Lawrence Erlbaum Associates, Hillsdale, NJ, USA, 1986. 2

[56] J. Novotny, J. Tveite, M. L. Turner, S. Gatesy, F. Drury, P. Falkingham, and D. H. Laidlaw. Developing virtual reality visualizations for unsteady flow analysis of dinosaur track formation using scientific sketching. *IEEE Trans Vis Comput Graph*, 25(5):2145–2154, 2019. doi: 10.1109/TVCG.2019.2898796 2

[57] D. Park, S. M. Drucker, R. Fernandez, and N. Elmqvist. Atom: A grammar for unit visualizations. *IEEE Trans Vis Comput Graph*, 24(12):3032–3043, 2018. doi: 10.1109/TVCG.2017.2785807 3

[58] X. Pu and M. Kay. A probabilistic grammar of graphics. In *Proc. ACM CHI*, pp. 1–13. ACM, New York, 2020. doi: 10.1145/3313831.3376466 3

[59] P. Reipschläger, S. Engert, and R. Dachselt. Augmented displays: Seamlessly extending interactive surfaces with head-mounted augmented reality. In *Ext. Abs. ACM CHI*, pp. 1–4. ACM, New York, 2020. doi: 10.1145/3334480.3383138 2

[60] J. C. Roberts, P. D. Ritsos, S. K. Badam, D. Brodbeck, J. Kennedy, and N. Elmqvist. Visualization beyond the desktop – the next big thing. *IEEE Comput Graph Appl*, 34(6):26–34, Nov. 2014. doi: 10.1109/MCG.2014.82 1, 2

[61] A. Saktheeswaran, A. Srinivasan, and J. Stasko. Touch? Speech? or touch and speech? Investigating multimodal interaction for visual network exploration and analysis. *IEEE Trans Vis Comput Graph*, 26(6):2168–2179, 2020. doi: 10.1109/TVCG.2020.2970512 3

[62] B. Sanderson. Sanderson's first law of magics. `https://www.brandonsanderson.com/sandersons-first-law/`, 2007. 1, 3

[63] A. Satyanarayan, D. Moritz, K. Wongsuphasawat, and J. Heer. Vega-Lite: A grammar of interactive graphics. *IEEE Trans Vis Comput Graph*, 23(1):341–350, 2017. doi: 10.1109/TVCG.2016.2599030 3

[64] A. Satyanarayan, R. Russell, J. Hoffswell, and J. Heer. Reactive Vega: A streaming dataflow architecture for declarative interactive visualization. *IEEE Trans Vis Comput Graph*, 22(1):659–668, 2016. doi: 10.1109/TVCG.2015.2467091 3

[65] D. Schroeder and D. F. Keefe. Visualization-by-Sketching: An artist's interface for creating multivariate time-varying data visualizations. *IEEE Trans Vis Comput Graph*, 22(1):877–885, 2016. doi: 10.1109/TVCG.2015.2467153 2

[66] S. Shin, A. Batch, P. W. S. Butcher, P. D. Ritsos, and N. Elmqvist. The Reality of the Situation: A Survey of Situated Analytics. *IEEE Trans Vis Comput Graph*, 2023. To appear. doi: 10.1109/TVCG.2023.3285546 9

[67] B. Shneiderman. Direct manipulation: A step beyond programming languages. *Computer*, 16(8):57–69, 1983. doi: 10.1109/MC.1983.1654471 2

[68] R. Sicat, J. Li, J. Choi, M. Cordeil, W.-K. Jeong, B. Bach, and H. Pfister. DXR: a toolkit for building immersive data visualizations. *IEEE Trans Vis Comput Graph*, 25(1):715–725, 2019. doi: 10.1109/TVCG.2018.2865152 2, 3, 5

[69] A. Srinivasan, B. Lee, and J. Stasko. Interweaving multimodal interaction with flexible unit visualizations for data exploration. *IEEE Trans Vis Comput Graph*, 27(8):3519–3533, 2021. doi: 10.1109/TVCG.2020.2978050 2

[70] C. Stolte, D. Tang, and P. Hanrahan. Polaris: A system for query, analysis, and visualization of multidimensional relational databases. *IEEE Trans Vis Comput Graph*, 8(1):52–65, 2002. doi: 10.1109/2945.981851 3

[71] J. Tang, H. Cheng, Y. Zhao, and H. Guo. Structured dynamic time warping for continuous hand trajectory gesture recognition. *Pattern Recognition*, 80:21–31, 2018. doi: 10.1016/j.patcog.2018.02.011 6

[72] B. H. Thomas, G. F. Welch, P. Dragicevic, N. Elmqvist, P. Irani, Y. Jansen, D. Schmalstieg, A. Tabard, N. A. M. ElSayed, R. T. Smith, and W. Willett. Situated analytics. In *Immersive Analytics*, vol. 11190 of *Lecture Notes in Computer Science*, pp. 185–220. Springer, New York, 2018. doi: 10.1007/978-3-030-01388-2_7 1

[73] W3C. *WebXR*. [Online]. Available `https://www.w3.org/TR/webxr/`, 2017. Accessed: 07/11/22. 3

[74] J. Wagner, W. Stuerzlinger, and L. Nedel. Comparing and combining virtual hand and virtual ray pointer interactions for data manipulation in immersive analytics. *IEEE Trans Vis Comput Graph*, 27(5):2513–2523, 2021. doi: 10.1109/TVCG.2021.3067759 1, 2

[75] S. White and S. Feiner. SiteLens: Situated visualization techniques for urban site visits. In *Proc. ACM CHI*, pp. 1117–1120. ACM, New York, 2009. doi: 10.1145/1518701.1518871 1

[76] H. Wickham. A layered grammar of graphics. *Journal of Computational and Graphical Statistics*, 19(1):3–28, 2010. doi: 10.1198/jcgs.2009.07098 3

[77] L. Wilkinson. *The Grammar of Graphics*. Springer-Verlag, Berlin, Germany, 1999. 3

[78] W. Willett, B. A. Aseniero, S. Carpendale, P. Dragicevic, Y. Jansen, L. Oehlberg, and P. Isenberg. Perception! Immersion! Empowerment! Superpowers as inspiration for visualization. *IEEE Trans Vis Comput Graph*, 28(1):22–32, 2022. doi: 10.1109/TVCG.2021.3114844 1, 8

[79] J. S. Yi, Y. a. Kang, J. Stasko, and J. Jacko. Toward a deeper understanding of the role of interaction in information visualization. *IEEE Trans Vis Comput Graph*, 13(6):1224–1231, 2007. doi: 10.1109/TVCG.2007.70515 2

[80] B. Yu and C. T. Silva. FlowSense: A natural language interface for visual data exploration within a dataflow system. *IEEE Trans Vis Comput Graph*, 26(1):1–11, 2020. doi: 10.1109/TVCG.2019.2934668 3

[81] R. C. Zeleznik, K. P. Herndon, and J. F. Hughes. SKETCH: An interface for sketching 3D scenes. In *Proc. SIGGRAPH*, pp. 163–170. ACM, New York, 1996. doi: 10.1145/237170.237238 2

[82] J. Zong, D. Barnwal, R. Neogy, and A. Satyanarayan. Lyra 2: Designing interactive visualizations by demonstration. *IEEE Trans Vis Comput Graph*, 27(2):304–314, 2021. doi: 10.1109/TVCG.2020.3030367 3