End-to-End Example-based Motion Graphics Video Authoring for Novice Motion Designers

by

Amir Hosein Jahanlou

M.Ed., University of British Columbia, 2018
M.A., Middlesex University, 2014
B.A. (Hons.), Middlesex University, 2008

Thesis Submitted in Partial Fulfillment of the
Requirements for the Degree of
Doctor of Philosophy

in the
School of Interactive Arts and Technology
Faculty of Communication, Art and Technology

© Amir Hosein Jahanlou 2023
SIMON FRASER UNIVERSITY
Fall 2023

Declaration of Committee

Name: **Amir Hosein Jahanlou**

Degree: **Doctor of Philosophy**

Thesis title: **End-to-End Example-based Motion Graphics Video Authoring for Novice Motion Designers**

Committee: **Chair:**   Chris Shaw
Professor, Interactive Arts and Technology

Parmit Chilana
Supervisor
Associate Professor, Computing Science

William Odom
Committee Member
Associate Professor, Interactive Arts and Technology

Sheelagh Carpendale
Committee Member
Professor, Computing Science

Halil Erhan
Examiner
Associate Professor, Interactive Arts and Technology

Pourang Irani
External Examiner
Professor, Computer Science, Mathematics, Physics and Statistics
University of British Columbia

Ethics Statement

The author, whose name appears on the title page of this work, has obtained, for the research described in this work, either:

      a.   human research ethics approval from the Simon Fraser University Office of Research Ethics

or

      b.   advance approval of the animal care protocol from the University Animal Care Committee of Simon Fraser University

or has conducted the research

      c.   as a co-investigator, collaborator, or research assistant in a research project approved in advance.

A copy of the approval letter has been filed with the Theses Office of the University Library at the time of submission of this thesis or project.

The original application for approval and letter of approval are filed with the relevant offices. Inquiries may be directed to those authorities.

Simon Fraser University Library
Burnaby, British Columbia, Canada

Update Spring 2016

iii

Abstract

Motion graphics videos offer a powerful means of communicating complex concepts through engaging visuals and animation. While becoming increasingly popular, authoring these videos can be overwhelming. This dissertation investigates the challenges faced by both casual and professional users in motion graphics and offers Katika, a novel user-centric tool that supports end-to-end authoring of motion graphics and lowers the barrier to end-user animation.

To inform the design and implementation of Katika, we carried out multiple studies. We initially interviewed 19 motion designers of varying skill levels and did a follow-up survey with 207 respondents to understand current practices, processes, and challenges of motion graphics authoring. The results underscored the need for example-based learning, particularly for amateur users facing a significant learning curve in envisioning the different stages of animation. Moreover, the results revealed that most users were using and switching between multiple feature-rich applications to create even basic motion graphics videos.

In our next study, we investigated the phenomenon of application switching in more depth, broadening our investigation to knowledge workers using multiple software tools to complete a single task. We carried out interviews with 15 knowledge workers to understand the drivers of tool switching and any related productivity impediments linked with this behavior. Our insights culminated in a taxonomy of reasons for application switching that emphasized the role of collaboration and external factors as being key drivers. These results provide insights for designing tools that eliminate or minimize task-centric application switching across different domains, including motion graphics.

Based on the results of our formative studies, we designed and developed Katika, an end-to-end authoring system to mitigate complexities such as the need for multiple software tools, the lack of in-built content, the difficulties of animation and communication observed in creating motion graphics. This system, in particular, empowers amateurs to create motion graphics without facing a steep learning curve or switching several applications. Our observational lab study demonstrated Katika's initial usability and showed that even novice users could create a motion graphics video within an hour. To gain further ecological validity, we next deployed Katika within professional and amateur users' contexts to gain insights into

real-life user practices, expectations, and perceptions of end-to-end authoring. Among our key findings, we found that although casual motion designers wanted the design process to be further simplified, the professionals were concerned about their creative input and wanted more control over the design choices.

Overall, this dissertation charts a comprehensive journey from identifying the processes and challenges of users in creating motion graphics videos to developing, validating, and refining an interactive system in response to such challenges. At the core of this dissertation is the following thesis:

In the rapidly evolving domain of motion graphics software, there is a new class of users that are casual motion designers; this population is not equipped with a thorough understanding of various aspects of authoring such as content creation, animation or video editing. An end-to-end and example-based interface for creating motion graphics videos can provide a useful and usable means to overcome their challenges.

Keywords: Motion Graphics; Computer Animation; Example-based Learning; Informational Videos; Explainer Videos

Acknowledgements

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivation

In today's increasingly digital world, the importance of effectively communicating complex ideas and sharing knowledge across various disciplines cannot be understated. One of the most compelling mediums for this form of communication is motion graphics videos. Motion graphics videos use movement to convey ideas through computer-generated graphics, and imagery [76]. The phrase motion graphics emerged from graphic design in motion ('motion design') [69] and has a long history in film and television production [177]. These videos have gained considerable popularity due to their ability to condense and present complex topics in an engaging and comprehensible manner. Motion graphics videos are finding wide applications in numerous areas such as data visualization [15], education [74, 46, 73], healthcare [117, 176], and finance [189]. Despite the growing demand for these dynamic and impactful videos, the creation process is often seen as expert-driven [1, 183, 68], necessitating a broad range of skills and knowledge encompassing content creation, animation, and sequence editing [201, 78, 105].

In response to these complex authoring needs, professional designers often turn to mainstream applications such as Adobe After Effects [11], Autodesk Maya [140], and Maxon Cinema4D [54] to facilitate the production of motion graphics videos. However, these tools, while exhaustive in their capabilities, come with feature-rich interfaces that can prove overwhelming for casual users. As a result, many users tend to gravitate towards simpler alternatives like Microsoft PowerPoint [164]. Although easier to navigate, these alternatives are not necessarily designed for the specialized task of creating motion graphics videos, leading to sub-optimal results [104].

Newer technologies in the field, such as the web-based applications Rive [170], Lumen [136], and Animaker [16], are designed with the intention of simplifying the authoring process. They offer more intuitive interfaces, and the added advantage of being web-based eliminates the need for software installation, maintenance, and file storage. However, the trade-off for this simplicity is often a reduction in features and capabilities, which may not

adequately serve the diverse requirements of a wide user base, encompassing both casual and professional users. In addition, AI-powered tools like Runway ML [171], Topaz Labs [123], and Synthesia [186] are bringing innovative automation techniques to the table, but the integration of these tools into the unique workflows of different users is not fully understood and remains a challenge.

In this research, we were motivated to delve into the challenges of creating motion graphics videos and investigating the design of a motion graphics authoring tool that can lower the barriers to entry. We designed and implemented *Katika* [110], a novel system that addresses the disconnect between the growing demand for motion graphics videos and the abilities of an non-trained individual to produce them. Katika offers an end-to-end solution to the design process, covering aspects such as scriptwriting, example-based artworks and animations, and sequence editing, effectively lowering the entry barriers for casual users.

By studying the usage of Katika by both casual and professional designers in real-world scenarios, our research provides a deeper understanding of the workflows, challenges, and future expectations of users. We intend to leverage these insights to foster the evolution of more efficient and user-friendly motion graphics authoring tools [105, 12, 15, 28]. Therefore, the ultimate motivation behind our research extends beyond the creation of a single tool and towards a more expansive goal: the democratization of the process of creating motion graphics videos.

At the core of this dissertation is the following thesis:

In the rapidly evolving domain of motion graphics software, there is a new class of users that are casual motion designers; this population is not equipped with a thorough understanding of various aspects of authoring such as content creation, animation or video editing. An end-to-end and example-based interface for creating motion graphics videos can provide a useful and usable means to overcome their challenges.

1.2   Research Overview

We conceptualized and developed Katika via a user-centered design process. This iterative process was underpinned by two exploratory studies probing casual and professional users' needs in motion graphics design and the general challenges of knowledge workers switching among many software applications. As the journey progressed, it involved several rounds of design revisions, a lab-based usability study, and ultimately culminated in a comprehensive implementation and deployment of the Katika system.

The structure of this dissertation adopts a design-based inquiry approach [207]. This investigative method is utilized to address an array of research questions concerning users' technology usage habits, as well as their unique authoring needs within the specific context of motion graphics. These questions include:

1. What are the challenges and obstacles users encounter when adopting and integrating new motion graphics software tools into their workflows?

2. What distinct and varied software tools do users rely upon to complete a new motion graphics project?

3. To what degree do users harness the myriad features and take advantage of the affordances of feature-rich software applications?

4. How do knowledge workers switch among multiple feature-rich applications and what are the reasons behind such application switching behavior?

5. What difficulties or roadblocks do users face when operating multiple tools and transitioning seamlessly between these applications?

6. What are the specific design goals that new tools to empower casual motion designers should prioritize?

7. How can the design of motion graphics software be optimized or modified to better support the unique and varied authoring needs of users?

8. To what extent can an example-based end-to-end motion graphics video authoring tool improve the workflows for casual and professional motion designers?

9. How do casual and professional motion designers perceive an example-based motion design tool within the lab settings and their own contexts?

10. To what extent and for what purposes can casual and professional motion designers embed an end-to-end motion design tool into their daily workflows?

The initial stage of this research was characterized by interviewing 19 casual and professional motion graphics designers and following up with a survey with 207 respondents. These initial studies unveiled a diverse range of software tools that designers integrate into their creative workflows, and also shed light on how the challenges that they faced. Interestingly, despite the availability of capable design tools, a significant portion of users expressed hesitation to engage with these advanced tools due to their perceived intricacy and complexity. Another key observation from this study was that designers were often relying on and switching between multiple tools to complete even a short motion graphics video.

In the next study, I investigated the phenomenon of how knowledge workers switch between multiple applications for a single task more deeply. The primary objective was to understand the practices and reasons for application switching, the related challenges, and any workarounds utilized by knowledge workers. I carried out in-depth interviews with 15 knowledge workers and also gathered invaluable insights from nine individuals working across five different software product teams. This multifaceted perspective allowed for an

understanding of the differences between user expectations and the perspectives of software vendors.

Drawing from these valuable insights, I embarked on the design and development of Katika. To tackle the challenges of creation motion graphics and to minimize application switching, my goal for Katika was to be an end-to-end tool for motion graphics design that empowers users to explore and learn from the motion graphics designs of others. A unique feature of Katika (stemmed from its end-to-end nature) is its exploratory mode, which provides users with an environment to experiment with various design options and understand their potential implications. Furthermore, Katika eases the application of these shared designs, effectively eliminating the need for users to sift through complex settings menus or seek external guidance. After passing through several iterations and reaching a stable prototype, we initiated a usability evaluation of Katika. The aim was to assess its intuitive design and the usefulness of its features. After a few studies, we learned that basing Katika on existing animation techniques, such as keyframing or interpolation, has the drawback of making it complex for newcomers. For that reason, we re-designed the animation feature of Katika using a novel technique of motion bundles. In this approach, we separate the animation from its visual look and can recycle the keyframes to be used by other objects. After reaching a degree of stability, we embarked on another set of usability studies. The feedback was largely positive, with users finding Katika's interface easy to navigate, appreciating its example-based approach and motion bundles, and expressing their value for the opportunity to experiment with potential designs without the typically associated risks.

Although the initial usability insights were useful, the lab-based usability studies could not provide a comprehensive picture of how Katika would integrate into users' day-to-day workflows in realistic settings. To gain a more grounded understanding of these aspects, we performed an in-context inquiry of Katika with another group of 14 users over a one-week period. This allowed for the collection of detailed usage data and rich qualitative insights from the follow-up interviews. The results from the deployment study strongly indicated that users felt empowered by quickly on-boarding a new tool (in the case of casual motion designers) or effectively include the tool within their workflows (for professional motion designers).

These findings open the door to exciting future design opportunities. They encourage HCI researchers and practitioners to consider harnessing end-to-end and example-based authoring as a tool to better support users' practices in authoring motion graphics. Specifically, the findings gathered from the perspectives of both casual and professional motion designers also suggest that this approach to design sharing could be explored in other domains of creativity, extending beyond just motion graphics.

1.3 Contributions and dissertation structure

The main contributions presented in this dissertation are:

1. Empirical insights into motion designers processes and learning strategies in authoring motion graphics videos using contemporary software applications (Chapter 3),

2. Empirical insights into the processes, perceptions and challenges of switching among many feature-rich software applications by knowledge workers (Chapter 4)

3. Synthesis of motion designers challenges and difficulties in getting started with authoring motion graphics videos (Chapters 3 & 4)

4. Implications for improving the design of motion graphics software applications for casual users (Chapters 3 & 4)

5. The design goals of Katika as an end-to-end and example-based motion graphics authoring software to support casual motion designers in creating their first video (Chapters 5 & 6)

6. The implementation and evaluation of an end-to-end example-based motion graphics authoring software within both lab and in-context studies (Chapters 5 & 6)

7. Insights from an initial evaluation of Katika's usability, providing details of the usefulness of its novel design elements (Chapter 5)

8. Empirical insights from an in-context deployment of Katika in users' practices, its potential to be incorporated into the workflows of either casual and professional motion designers (Chapter 6).

The remainder of this dissertation is structured as follows:

Chapter 2 provides an outline of relevant research literature and other works forming the background for this dissertation. In particular, it contextualizes this work among research into the state-of-the-art in motion graphics authoring, software learnability and approaches for authoring selected by various user groups.

Chapter 3 describes my exploratory research into how beginner motion graphics video creators learn, use, and benefit from tools that are designed for motion graphics video authoring as well as how they customize tools intended for other purposes.

Chapter 4 presents an analysis of the reasons, processes and challenges of switching between multiple feature-rich software applications among knowledge workers.

Chapter 5 introduces Katika and describes the user-centered design process that I followed to create it and evaluate its usability for amateur motion designers.

Chapter 6 describes an in-context deployment of Katika that I conducted to evaluate how both casual and professional motion designers might use it in their day-to-day motion graphics video authoring.

Chapters 7 and 8 reflect upon the broader takeaways for designing tools of motion graphics video authoring and creativity in general, the potential for example-based authoring as a general approach for improving the learnability of feature-rich software, and the main research contributions of my work.

Parts of this dissertation have previously been included in peer-reviewed publications (or have been accepted for publication). In particular, the content in Chapters 3 through 6 consists primarily of conference papers that I authored or co-authored, with the original publication listed at the head of each chapter.

1.4   The components of this thesis

There are multiple sections in this dissertation. Fig. B.4 represents these various stages.

Chapter 1	Chapter 2	Chapter 3	Chapter 4
Motion Graphics Processes and Challenges: This study explores various challenges of getting started with motion graphics design. Key contributions include design implications for future motion design tools to simplify the process for beginner users, as well as finding ways to limit the number of applications used.	**Cost of Task Switching for Knowledge Workers:** Following previous study, this part explores the difficulties introduced by switching among multiple software applications and the impact it might have on the efficiency and performance of users in a variety of knowledge domains, particularly creative and motion design.	**Katika System of Animation for Beginners:** After the first two studies, I looked into developing a new system for creating motion graphics for beginners. The system provides a minimal interface for creating videos using built-in examples. It also introduces an entirely new paradigm of animation using motion bundles.	**Katika Beyond the Lab:** In this final chapter, the system developed in the previous chapter (validated within a lab study) was taken to the field to assess its effectiveness within the contexts of the users and for specific requirements in different domains. It contributes insights into its efficacy for different use-cases of users in specific contexts.

Components of this thesis

Figure 1.1: The four components of this dissertation.

Chapter 2

Background and related work

In this chapter, I position my work within the broader research literature surrounding the use of motion graphics tools, the general learnability of software applications, the tools and technology for creative content authoring and the challenges of adapting new technology for novices. Though individual chapters 3–6 each provide a more focused discussion of the literature most relevant to their respective studies, here I offer an overview of the major research areas that best contextualize my work as a whole.

2.1   Motion Graphics; Definitions and Approaches to Authoring

Situated within the broader domain of animation, motion graphics emerges as a unique subset that serves a fundamentally different purpose. While animation tends to narrate a hero's journey [37, 48], motion graphics primarily serve as a visual medium of communication [69, 195]. The essential nature of motion graphics, complemented by the impactful influence of animation on content memorability [37], facilitates the visual representation of various concepts across diverse sectors, including but not limited to education [74] and geography [46].

Originating from the time-honored practice of employing illustrations in textbooks to augment understanding, the field of motion graphics expands upon this tradition by integrating dynamic elements. Evidence from educational research points to the capacity of animations to simplify comprehension [30], stimulate attention [39], and foster the long-term retention of information [36]. This potency is in part due to the instinctual human reaction to movement, positioning motion graphics as a rich, universal language [89] that can surmount cultural divides. By leveraging movement as a communication medium, motion graphics videos possess the ability to convey layered messages [39], demonstrating beneficial effects on learning in specialized fields like physics [63], geography [46], dental care [176], and even educational motivation [30].

Notably, motion graphics videos tap into the power of storytelling to establish common understanding [206]. The pedagogical benefits of storytelling have been affirmed in areas such as management education [155], the early years of higher education [73], the enhancement of knowledge interpretation [29, 206], and the simplification of complex subjects by introducing components sequentially [23].

However, despite its remarkable potentials, the creation of motion graphics remains a complex activity necessitating proficiency in specific tools [56]. This complexity presents a myriad of challenges for novice designers, spanning from the initial stages of getting acquainted with the process to locating suitable tools that can provide comprehensive support throughout the creation process. This scenario aligns with the broader trend in HCI research, which focuses on studying designers from non-traditional backgrounds to optimize their creative practices, processes, and experiences [82, 198].

In light of these observations, a detailed discussion of motion graphics within the larger context of design is warranted. While previous HCI studies have explored various types of design activities, including interactive software design [51, 82, 98] and graphic design [183], motion design presents a unique set of challenges. It shares commonalities with these domains, yet it also exhibits distinctive features such as an emphasis on movement and storytelling. These unique attributes necessitate a more targeted focus compared to interaction or graphic design, centering on the creation of engaging videos to articulate complex information or to unfold a captivating story.

The study of video authoring tools design also illuminates the domain of motion graphics. However, there is a clear divergence between motion graphic design and traditional video authoring. The latter often starts with pre-existing footage or templates, while the former requires designers to create all content from scratch using a purely software-driven process. Even though automation in video authoring has been a topic of research, motion design remains a largely human-centric process. It caters to specific informational needs, particularly in domains where capturing subject-matter expertise is crucial.

Lastly, the hurdles related to using and learning sophisticated software applications have been a recurring theme in HCI. Challenges span from the significant commitment required for training [64], to issues related to software usability [51], and to the intricacies of learning multiple feature-rich software applications [116]. These challenges have hindered the full realization of the potential of motion graphics, particularly in non-business domains such as education, due to these technical and pedagogical obstacles [162, 196].

In conclusion, this work seeks to illuminate both software-related and non-software-related obstacles to creating motion graphics videos, especially for those new to the field. It offers a comprehensive examination of the entire creation process from the vantage point of motion designers, both professionals and beginners alike. The study uncovers challenges faced not only during the active use of software, but also in the pre-production phases of weaving a compelling narrative. Finally, it puts forth recommendations to simplify the

creative process for casual motion designers and to bridge the gap between their aspirations and their technical capabilities.

2.2 Knowledge Workers and Using Multiple Feature-rich Software Tools

In the ever-evolving world of software tools, users frequently grapple with the challenging process of learning and mastering the myriad of features that accompany each new software release. This issue, often referred to as "featurism", constitutes a critical problem that the field of Human-Computer Interaction (HCI) research has continually strived to address [141, 88, 116]. The ongoing discussion has yielded several proposed strategies aimed at mitigating these issues, including enhancing task flow, augmenting user interface awareness, improving feature findability, and minimizing unnecessary functionality [88, 141, 71, 44].

Moreover, the contemporary landscape of knowledge work, which is hallmarked by a high degree of collaboration and multitasking across a wide range of software tools, exponentially magnifies the complexities associated with software learnability [94, 22, 156, 80, 167]. Although certain efforts have been directed towards amalgamating various functionalities into larger, more encompassing applications in order to alleviate the need for constant switching between different software tools [125, 77, 121], the overarching challenge of learning and mastering complex software remains a formidable obstacle [51, 116].

The phenomenon of multitasking has commanded considerable attention within the HCI research community, with focal points of study being elements such as work fragmentation, task management, and task-switching behaviors [145, 175, 187, 93]. In tandem with this, research has delved into exploring the role of larger screen real estate and the use of multiple displays in enhancing multitasking capabilities [93, 98], as well as the benefits and challenges of activity-centric computing systems [34], and strategies for managing window overlap [107]. Despite these efforts, seamless multitasking continues to be an elusive goal due to the inherent disruptions caused by frequent application switching [80, 174]. It is crucial to note the difference between intentional task switching and task-switching that is triggered by external interruptions, as this distinction is central to understanding and mitigating the associated challenges [3].

From the perspective of productivity and task management, the field of HCI has consistently embarked on exploring avenues to better support software users, with the aim of augmenting efficiency and enhancing overall effectiveness [22, 6, 31, 102, 141]. Although task-centric application switching can serve as a catalyst for knowledge transfer and thus potentially boost productivity [4], it also entails cognitive costs and can lead to performance reductions due to resumption lags [175]. In some cases, users may resort to less optimal techniques due to their familiarity with these methods, resulting in long-term impediments

to productivity [156]. Past studies have also indicated the potential benefits of employing project-specific task reminders as a strategy to bolster user productivity [60].

In this study, we seek to delve deeper into the intricacies that arise from the interaction between the tools being used and the workflows of users, alongside the consideration of individual characteristics and environmental settings of knowledge workers [146]. In doing so, we hope to build upon previous research by providing a detailed analysis of the specific challenges brought about by both software and non-software related issues pertaining to application switching. By doing so, we aim to spotlight potential areas of opportunity for further research and development within the HCI community [6, 31, 102].

2.3   Authoring Motion Graphics Video Authoring

With the growing demand for videos across various domains, innovation in amateur video production has seen an upward trend. This shift has necessitated further research to develop and refine interfaces that can improve and streamline the methodologies employed in video production [35, 59, 73]. One key area that has been the focus of much research is the simplification of post-production processes, such as video editing, where the core content typically stems from raw footage captured by cameras. This ongoing research has paved the way for the development of novel approaches that cater to generating screen recordings, facilitating home video creation, reusing content, and addressing the challenge of effectively sequencing video content [49, 78, 130]. Yet, these approaches generally target a single aspect of the complex video production process, thereby posing difficulties for beginners who are trying to grasp the entire process from pre- to post-production.

In addition, the rise in automation has extended its influence to amateur video production, prompting the exploration of automatic event timelines, video creation through recommendations, and the automation of text-to-video content generation. However, it is important to strike a balance between the convenience of automation and the need for user control, especially when authoring informational motion graphics videos [41, 190, 91, 184, 127, 95, 204, 203, 78]. Distinguishing itself from prior solutions, Katika presents an end-to-end authoring solution for amateurs that bridges the gap between pre- and post-production processes within a unified user interface.

In the context of authoring informational videos, industry norms lean heavily towards the utilization of motion graphics. This task can be quite challenging, often taking up to two days to create a minute-long video. As the number of amateurs attempting to author such videos grows, researchers are venturing into various aspects of the process, including the exploration of techniques for data videos with motion graphics, methodologies to create engaging content, and a deep analysis of the motion graphics design space [15, 97, 179, 28, 14, 135]. However, amateur motion designers often face challenges due to the sophisticated nature of the tools at their disposal, such as Adobe After Effects and Autodesk Maya,

which not only have steep learning curves but also require the use of multiple such tools to complete a video [76, 11, 140, 76, 105].

Additionally, the process of authoring informational videos often starts from scratch with a blank canvas. While template-based authoring techniques have been suggested as a way to alleviate this challenge, adapting these templates to meet the specific requirements of informational videos is not a straightforward task. Besides, the act of collaboration and acquiring suitable feedback continues to be a considerable challenge [16, 197, 66, 105, 64]. To address these issues, our approach with Katika incorporates a simplified middle-level interface design. This design simplifies the steps involved in creating explainer motion graphics videos, making it easier for a wider range of amateurs to understand and follow the process of motion graphics design.

A central aspect of simplifying the animation process, as researched extensively, involves the integration of time and space, which forms the core of animation production. Various techniques, such as K-sketch, the use of hand gestures to record animations, sketching, and motion transfer, have been employed, though they each come with their own limitations [62, 63, 112, 111]. To overcome these constraints, Katika introduces a new paradigm termed as 'motion bundles.' These example-based animations offer users the opportunity to experiment with predefined motions, thus simplifying the task of envisaging the subsequent stage of animation and assisting in the creation of their motion graphics videos.

2.4 Motion Graphics Authoring in User Contexts: An Extended Perspective

The field of motion graphics video authoring has undergone considerable advancements in recent years, with a marked contribution from sophisticated tools and technologies. The roles of renowned software suites such as Adobe After Effects [11], Autodesk Maya [140], and Maxon Cinema4D [54] in this transformative progress cannot be underestimated. These comprehensive tools provide a wide array of functionalities to cater to diverse user needs, including capabilities for creating both 2D and 3D animations, facilitating text animation, providing visual effects, and enabling integration with other software utilities for a seamless design experience.

Simultaneously, a new wave of motion graphics tools that specifically target non-professional users has emerged. These tools, including web-based applications such as Rive [170], Jitter [109], and Animaker [16], strive to lower the entry barrier by offering drag-and-drop interfaces and template-based workflows. Their ease of use and accessibility make them particularly suitable for educators, casual users, or other individuals who may not have a background in professional graphics design. Moreover, the platform-independent nature of these tools and their no-installation-required convenience add to their appeal for the aforementioned user demographic.

The recent integration of cutting-edge artificial intelligence (AI) and machine learning technologies [147, 182] into the domain of graphics authoring denotes a promising future direction in this field. By leveraging these technologies, graphics authoring software could offer intelligent recommendations, automate repetitive tasks, and ultimately enhance the overall user experience.

However, despite the variety and abundance of these tools, gaps in accessibility and usability persist. It is important to note that many of these tools are designed with professionals in mind. Consequently, they often present a steep learning curve for casual users or those without a background in design [183, 142]. Thus, while advanced tools have the capability to create visually complex and impressive graphics, they may not completely cater to the diverse needs of different user groups.

The process of creating motion graphics videos encapsulates a multitude of challenges, ranging from technical complexities to industry-specific hurdles and issues related to user experience [183]. Challenges can include the demanding tasks of breaking down a lengthy script into separate shots, sourcing and editing necessary graphical assets, animating these assets through keyframing and interpolation techniques, editing sequences into a cohesive video, adding suitable music, and converting the projects into appropriate video formats [105]. Given these complex processes, creating even a short video can be time-consuming and arduous for a casual user [161]. The need for collaboration and effective communication within many motion graphics projects further highlights the requirement for robust workflow management strategies, which are often lacking in current authoring tools [183, 105].

In spite of the progress that has been made, there remain persistent challenges that need to be addressed, such as the daunting learning curve of advanced tools, software integration issues, and the delicate balance of user expectations with learning difficulty. These challenges emphasize the need for ongoing improvement in the motion graphics industry and a constant evolution of the tools supporting it. An understanding of the unique challenges faced by different user groups—casual users, professionals, and individuals with specific industry needs— is paramount in developing tailored tools and strategies.

The concept of learnability is especially crucial in the context of complex tools like motion graphics authoring software [65]. The task of striking a delicate balance between simplicity and functionality in these tools is a challenging design problem, as these tools must cater to a wide spectrum of users ranging from professional designers to amateurs [183]. For novice users, the complexity and feature-rich nature of these tools can create significant hurdles [142]. However, by incorporating user-centered design principles, clear documentation, tutorials, and potentially adaptive interfaces, the learnability of software can be improved, thus encouraging a more inclusive design practice [105, 183, 40].

The technique of example-based learning has been shown to be a fruitful approach that enables users to learn from real-world examples in a variety of domains, including programming [40], 3D modeling [144], and image editing [87]. Despite its numerous advantages,

the domain of motion graphics has not seen substantial research assessing the efficacy of example-based learning. Exploring this approach in the context of motion graphics can lead to the development of more intuitive and accessible tools, which can cater to users of varying levels of expertise and experience.

2.5 Summary

The domain of motion graphics video authoring has been characterized by significant advancements in recent years, thanks largely to the proliferation of state-of-the-art tools and technologies. Key software suites like Adobe After Effects, Autodesk Maya, and Maxon Cinema4D have played pivotal roles in this transformation, providing an extensive array of features designed to create both 2D and 3D animations, facilitate text animation, offer visual effects, and enable integration with other software tools. In addition to these industry-leading tools, there's a surge of user-friendly motion graphics tools such as Rive, Jitter, and Animaker, designed to appeal to non-professional users by offering intuitive interfaces and template-based workflows. The inclusion of AI and machine learning technologies further adds to the appeal of these tools, potentially transforming the future of graphics authoring by automating tasks and improving user experience.

However, despite these strides, there remain considerable challenges, particularly in addressing the diverse needs of users from different backgrounds and proficiency levels. Many advanced tools, while equipped to create visually impressive graphics, pose a steep learning curve that may deter casual users. The process of creating motion graphics videos can be complex and time-consuming, involving multiple stages from scripting to animation and editing. Further complications arise when considering the need for effective collaboration and communication in many projects, aspects that are often insufficiently catered to by existing tools. To navigate these issues, future advancements must focus on enhancing learnability, incorporating user-centered design principles, and exploring alternative learning approaches, such as example-based learning, to make the authoring process more inclusive and accessible.

Chapter 3

Challenges in Getting Started in Motion Graphic Design: Perspectives from Casual and Professional Motion Designers

In my first study[1], my goal was to delve into the process of creating motion graphics videos, a powerful means for articulating complex ideas in a compelling manner through the interplay of dynamic visuals and animation. These videos are effective storytelling tools extensively utilized in various industries, including filmmaking, marketing, and video gaming. Nonetheless, when it comes to users outside these sectors, such as educators or domain specialists, the path to creating motion graphics videos is fraught with challenges. They encounter a steep learning curve which often deter their attempts. The seemingly daunting task of crafting even a brief motion graphics video demands many competencies. This includes scriptwriting and graphic design, a proficient understanding of animation, and the ability to navigate various feature-rich software applications.

To gain insights into the design processes and the associated challenges that prevail in this domain, we conducted interviews with 19 individuals and gathered 207 respondents to a survey. Our participants were a mix of casual and professional motion designers actively engaged in various motion graphics projects. The findings from our study shed light on the numerous obstacles that budding motion designers face as they venture into the field, illuminating how they grapple with conceiving workable solutions and improvising when faced with these barriers. We suggest designing user-centric tools to simplify the intricate process of motion design. These tools should integrate strategies like example-based learning

---

[1]Portions of this chapter were originally published in Jahanlou, A., Odom, W. and Chilana, P., 2020, December. Challenges in getting started in motion graphic design: Perspectives from casual and professional motion designers. In Graphics Interface 2021. [105]

and collaborative methodologies, significantly contributing to a more accessible and user-friendly motion graphics video creation process.

Figure 3.1: Examples of motion graphics in education (left), advertising (middle), and UI onboarding (right).

3.1 Introduction

Motion graphics videos use a combination of sound and movement to convey ideas through computer-generated graphics, icons, and pictograms [76] (Figure 1). The phrase motion graphics emerged from graphic design in motion ('motion design') [69] and has a long history in film and television production [177]. With the growth of video as a communication medium and the access to video publishing platforms [15], these videos have become popular to create lively and engaging content [59, 69, 201] in marketing and advertising contexts. Motion graphics videos are also being used in informational contexts to illustrate complex phenomena in physics [138], geography [46], healthcare [139] and other domains [46, 189].

Despite the promise of motion graphics videos in informational fields, creating such videos can be a laborious and expensive process. For example, surveys suggest that it can cost up to $8000 USD to make a short 60-second motion graphics video [201]. Unlike camera or screen-recorded videos where the captured footage serves as a starting point, the creation of motion graphics videos is an entirely software-driven process that begins with a blank canvas: the content and the movement are usually generated from scratch (Draper 2019). Furthermore, creating a motion graphics video requires scriptwriting, animation and graphics design, familiarity with feature-rich software, and the knowledge of color, typography, sound, or imagery [76]. Given the high costs and array of skills needed to create a motion graphics video, traditionally, only large organizations have been able to afford motion designers' expertise. As a result, motion graphics has primarily catered to sectors such as marketing, video games, film, or advertising. Domain experts, such as scientists or instructors (who could benefit from using this information-sharing medium), face a high barrier of entry in creating motion graphics on their own [69].

Although research in HCI and media studies has a long history of exploring visual storytelling [69], approaches to animation [76], interactive software design [161], and graphic

15

design [183] surprisingly, not much is known about the processes and workflows used by motion designers. In this paper, we argue that there is a rich opportunity for the HCI community to better understand and facilitate the process of creating motion graphics to empower more individuals to use this medium in information-sharing contexts. This has parallels to existing efforts in HCI to increase the participation of casual users and domain experts in various technical and design tasks, such as programming [69], website creation [199], interaction design and UX [69], and generating visualizations [69].

To investigate the work of motion designers, we conducted interviews with 19 professional and casual motion designers working on projects such as explainer videos, advertisements, education, or UI animations. We focused on understanding how these designers approached the overall design process and the types of tools and techniques they used. In particular, we probed into any barriers that they faced in getting started with motion graphics, challenges in the process, or the difficulties in communicating their progress. We found that professional motion designers invested significantly more in the preparatory stages of creating a video. In contrast, casual motion designers had difficulty understanding the "big picture" of a project and often skipped pre-production steps. Moreover, the software tools used in motion design were not necessarily developed for motion graphics as they catered to a variety of creative processes, leading to a mismatch of expectations. This paper makes two main contributions:

- we offer empirical insights detailing the work of motion designers, including the diverse range of challenges that they face as well as their learning approaches, issues in understanding the processes, and the experiences of working with software tools.

- we interpret these findings to offer opportunities where research in HCI can lower the barriers to entry and democratize motion graphics creation. For example, there is a key opportunity for supporting the onboarding process by helping individuals understand the bigger context of an entire motion graphics project. Furthermore, we advocate that future tools should support the end-to-end process of designing motion graphics. In doing so, new opportunities would be generated to address key challenges occurring in the preliminary steps of creating motion graphics projects.

3.2 Related Work

To situate our findings in the broader literature, we highlight the efficacy of using animation and graphics in different domains, how motion graphics videos fit in with the larger context of design, and insights from works in understanding the challenges of learning and working with feature-rich software applications.

3.2.1 The Efficacy of Graphics, and Animation

While motion graphics can be considered a subgenre of animation [37, 48], they are yet primarily motivated by communicating a phenomenon rather than following the story of a hero [69, 195]. The positive impact of animation on the memorability of content [37] enables motion graphics to visually present concepts in different domains [69], such as education [74] or geography [46].

While motion graphics is a relatively new phenomenon, decades of textbooks with illustrations enforces the notion of using imagery for better explanations. Using animation in education has been shown to simplify understanding [30], increase attention [39], and leave a lasting memory [36]. Furthermore, human reaction to motion is partly our instinct, making motion graphic a rich common language [89] understood across different cultures. Moreover, movement as a means of communication empowers motion graphics videos to affect different levels of the message [39]. Previous studies have highlighted the positive impact of animation in learning physics [63], geography [46], dental care [176], and even increasing motivation among educators [30]. Motion graphics videos also enable a degree of storytelling that makes them an effective method of establishing common grounds [206]. Storytelling has been found useful in the education of management [155], early years of higher education [73], as well as improving knowledge interpretation [29, 206], and making complex topics more approachable by introducing them sequentially [23].

Despite the illustrated effectiveness of using animation and graphics, motion graphics creation is yet a complex activity requiring tool-specific expertise [56]. Our study highlights the challenges in getting started with motion graphics and the lack of appropriate tools that support the entire creation ecosystem. More broadly, our work contributes to a trajectory of research in the HCI community aimed at studying designers (especially from non-traditional backgrounds) to better support their creative practices, processes, and experiences (e.g., [82, 205]).

3.2.2 Motion Graphics in the Larger Context of Design

Previous studies in HCI have explored various types of design activities. For example, studies have examined the processes of doing interactive software design [51, 82, 98] or graphic design (e.g., [183]). Although these domains of design bear some similarities to motion design, there also are key differences. For example, compared to graphics design, motion graphics entail movement and far more elaborate storytelling. In that, while graphic designers work on a static image, motion designers need to complete (and relate) many different parts of the content to create compelling stories. Also, unlike interaction design, where the goal is to produce a compelling task-specific user experience for an interactive artifact, motion design has a more nuanced focus on creating a video to convey complex information or tell a story.

Perhaps more relevant to the domain of motion graphics is research on the design of video authoring tools. However, unlike camera- or screen-captured videos that primarily rely on pre-recorded video footage to begin the authoring process, motion graphic design begins with a blank canvas, and designers have to create all of the content in an entirely software-driven process. Prior work has also explored the use of automation in video authoring – for example, by creating event timelines [41] and producing storybooks from text [91, 184], or simplifying the video production using recommendation [120]- and example-based video editing [69]. In contrast to these approaches, motion design is a process driven by people and their information needs (as opposed to automated content creation). Furthermore, motion graphics in informational domains is primarily concerned with capturing the nuances of subject-matter experts and imparting specific knowledge that is not achievable in automatic content creation. Our work complements previous studies and insights into graphics design, interaction design, animation, and video authoring, and illustrates the specific challenges faced by motion designers in the process of creating videos using computer-generated graphics, sound, and movement.

3.2.3 Challenges in Using and Learning Feature-Rich Software

HCI has a long history of exploring the difficulties of using and learning feature-rich software applications (e.g., novices and 3D printing [69], prototyping approaches by beginners [102], or graphic designers using digital tools [183]). In particular, it has been shown that it is problematic for subject-matter experts and educators (Cone 2015) to work with such software as it requires a substantial commitment to training on top of their day jobs [64]. As a side-effect of these difficulties in the usability of complex software [51] or the challenges in learning multiple feature-rich software applications [115], advanced tools often get ignored in their entirety [184]. On the other hand, specific tools with limited functionality also suffer from being useful for particular conditions only. As a result of such challenges, the time required to create motion graphics videos is longer than making slideshows or even video tutorials. Such issues compound the traditional challenges of integrating complex technology and pedagogy [162, 196] and is another reason why the potential of motion graphics is not fully realized in non-business domains, such as education.

Our work complements the previous studies and highlights the software and non-software related barriers for creating motion graphics videos for novices. We explore the entire creation process from the perspective of motion designers (both professional and beginners). Furthermore, our study reveals insights into the difficulties that occur not only during software use but also during the preparatory steps (pre-production) in creating a compelling narrative, as well as challenges that the novices face in understanding the context of the production. We suggest ways in which the process can be simplified for casual motion designers.

Table 3.1: Overview of participants in our interviews.

P#	Training	Domain	Gender	Location	Education	Expertise
1	Design	Marketing	Female	Australia	Undergraduate	Professional
2	Animation	Advertising	Male	Brazil	Some College	Professional
3	Design	Advertising	Male	Brazil	Undergraduate	Professional
4	Branding	UI Design	Male	Ukraine	Undergraduate	Casual
5	Animation	Advertising	Female	USA	Some College	Professional
6	Science	Research	Male	Colombia	Some College	Casual
7	Animation	Fitness	Male	Singapore	Some College	Casual
8	Science	Advertising	Male	Canada	Undergraduate	Professional
9	Design	Advertising	Female	USA	Undergraduate	Professional
10	Music	Film	Female	USA	Undergraduate	Casual
11	Film	Graphics	Male	Australia	Some College	Professional
12	Film	Health	Male	Brazil	Undergraduate	Professional
13	Animation	Marketing	Male	Germany	Undergraduate	Professional
14	Design	Marketing	Male	Lithuania	Some College	Professional
15	Science	Astrophysics	Male	Canada	Graduate	Casual
16	Animation	Advertising	Male	USA	Some College	Professional
17	Design	Documentaries	Female	Canada	Undergraduate	Casual
18	Design	Advertising	Male	Greece	Some College	Professional
19	Education	Education	Male	USA	Graduate	Casual

3.3 Method

To better understand the practice of motion graphics design and motion designers' work, we carried out interviews with a range of motion designers. Below, we present our research approach, details of recruiting participants who represented a range of motivations and skills, and our data analysis strategies. 3.1 Semi-Structured Interviews with Motion Designers We carried out semi-structured interviews with motion designers from a variety of backgrounds and levels of expertise. We recruited 19 motion designers (14 males, 5 females summarized in Table 1) with a range of expertise (2 to 12 years of experience) through searching for online motion graphics videos on various video publishing websites (e.g., vimeo.com ([197])). In doing so, we searched for recently published explainer or motion graphics videos and reached out to the creators, and invited them for an interview. When possible, we conducted face-to-face interviews, and for remote participants, we completed the interviews through Skype. Each interview lasted about 45 minutes, and participants received a $15 Amazon gift card for their participation.

Before starting the interview, we used a questionnaire to collect demographic information such as gender, age, occupation, education, experiences with motion graphics, and motivation for working with motion graphics. We then started the interview with warm-up questions asking participants to recall their most recent project and encouraged them to describe their typical process and any challenges that they faced. We then sought to understand the designers' processes by asking questions about their approach to learning new techniques, workarounds, and the types of resources they consulted to support their practice. We wanted to understand what kinds of situations triggered the need for particular kinds of resources, how they could locate them, and how useful they found these different resources. We used common resources to prompt the participants (e.g., online courses, books, tutorials, in-person training, etc.). After the first few interviews, we updated this list with more informal resources that have been mentioned, such as blogs, magazines, or forums.

During our interviews, we observed that most participants (12/19) were professional motion designers in that they had sufficient training and routinely worked on commercial projects. However, we also saw a class of participants (7/19) that could be described as casual motion designers working on a single project and mostly new to the field. This realization encouraged us to prompt all participants to reflect back on their early days of working with motion graphics and speak of the challenges they encountered in getting started. These insights helped us better understand the differences in casual vs. professional motion designers' perspectives and the prevalence of the key themes from our interviews across a larger pool of motion designers from different domains, motivations, and expertise. Lastly, we ended the interview by a conversation about the trends, tools, and the future of motion graphics design. In doing that, we sought to learn about motion designers' pain

20

points and any solutions they had devised to resolve them. These insights support us in synthesizing the challenges and solutions into design implications and guidelines.

3.3.1 Data Analysis

All interview recordings were transcribed and coded using Atlas.TI data analysis software. The coded data were analyzed to illustrate the different processes, challenges in learning or software interactions, and the workarounds that motion designers had devised to navigate their field. We used an inductive analysis approach to explore the themes around our main research question. To ensure the coded data's validity, the primary author performed the first open coding pass and consulted with another researcher to discuss and develop an initial list of codes. The initial pool included codes such as processes, pre-production, UI, and motivation. All of the researchers had prior experience with qualitative analysis and continually checked the legitimacy of the scheme. Upon completing the first phase, three researchers collectively discussed the emerging themes and finalized the coding scheme.

3.4 RESULTS: KEY INSIGHTS FROM MOTION DESIGN-ERS

In this section, we present the key results that we synthesized from our interviews, highlighting the diversity in the backgrounds of motion designers and their projects, the challenges they face during the design process, how they approach learning, and the role of collaboration and online communities.

3.4.1 Challenges in Using Motion Graphics Across Different Domains

Our participants (summarized in Table 1) worked in various domains, such as product marketing, user interface design, fitness content creation, music video production, astrophysics, health educational videos, and documentary filmmaking. Those classified as professionals (12/19) often had formal training in creative domains such as animation, design, or filmmaking. In contrast, casual motion designers (7/12) -who did not have formal training in design or animation- learned motion graphics to meet specific demands of their domains (e.g., to create explainer videos, educational videos, or physical fitness). Furthermore, while some participants had the option to solicit professional motion graphics services, this was costly, and the end-product did not accurately reflect the subtleties of the subject matter.

The diversity of content forced both casual and professional motion designers to continuously learn new techniques or regularly create new and unfamiliar content. For example, P12 working on medical field projects described how they had to learn to create artworks that were specific to a particular domain to illustrate various procedures. Another participant who worked on educational projects explained how they had to calibrate their content's complexity depending on the target audience: *"If it's for kids... I would just try something*

21

simpler. So, I will try to use many colors. . . For Science, I can go as complicated as I want with motion graphics because the audience is [scientists] who already know the subject. . . I can put all these weird names that for me don't make sense, and no one will get it. But, these people [scientists], this special audience will understand it." (P19)

In addition to participants who worked on informational content, some individuals worked on more creative productions, such as exploring motion graphics for music videos, which added to the complexity of interpreting the briefs and coming up with relevant content and animation:

"I am contacted by independent artists, mainly musicians who ask me to animate their music videos or things like that. Or some people such as independent filmmakers ask: "Can you animate this?"." (P10)

Despite the challenges in navigating a diverse field, we found that all motion designers in our study enjoyed the freedom to create motion graphics independently and to express a variety of ideas using motion graphics in their "own voice": *"With motion graphics, I guess my main idea is that you can experiment a lot and that was really appealing whereas if you go video production, you [have] very little creative freedom and I don't really like. So, my thinking was that I want to be more independent and with the motion graphics, it offered that option." (P17)*

3.4.2 Challenges in Creating a Motion Graphics Video

Video production can typically be described as a process with three stages of pre-production, production, and post-production [88]. We use these phases as a lens to analyze the process of creating a motion graphics video and highlight a range of practices and challenges that participants faced across different stages. Each phase of the production process includes multiple steps that add to the complexity of the process for non-experts. Furthermore, despite the seemingly sequential nature of the process, motion designers often had to work on several steps at the same time and equip themselves with the know-how of adequately allocating time to each stage or reducing the length of others.

3.4.3 Pre-Production Challenges

Pre-production supports an understanding of the project's scope, which results in better planning for the video. During pre-production, motion designers usually convert the project's brief to a script, create a storyboard, prepare a mood-board, or edit animatics (draft videos). However, most participants perceived this process to be time-consuming and expressed that they could not initially appreciate how their work benefited from pre-production. As a result, casual motion designers would often skip the pre-production steps and prematurely begin working on the final video, only having to stop the production later and re-start the process:

"[Beginners] usually skip [pre-production] phase, and that makes them spend much more

time on the animation stage. Because they don't want to plan. They open the software and start an animation." (P3)

Over time, motion designers had learned to invest more in pre-production. The first step usually was developing a brief for the project and converting that into a script. Participants spoke of a myriad of techniques for writing the scripts. As shown in other works [35, 150], the script included details such as action and the voice over the videos. However, scripts were not always easy to create. A participant working on educational content pointed out how they needed to hire someone else to write it: *"Most of the time if I'm building the script, it's not really simple, I need to hire somebody to write it. Just so it has everything fit. So, writing the script and finding the visual is the first challenge." (P16)*

Motion designers used the script to break down the video into separate shots. This breakdown helped them work on each shot independently while keeping the overall project in mind. Among the key factors for producing a shot-by-shot was the use of language structures such as sentences or paragraphs. These shot breakdowns supported the more experienced motion designers to create separate shots and edit them back together. However, motion designers expressed two challenges with creating these breakdowns; a) the first one was that there is no approach for breaking a long script into separate shots within the current software tools, and b) how existing tools did not provide a method of placing each shot within the context of an entire video. Such lack of context was rather challenging for casual motion designers as they had difficulty imagining the final video from separate shots.

While the script usually provides a scene breakdown, storyboards break up the script into key moments in time [183] using visual frames. Professionals spoke of how they resorted to basic mediums such as paper and pencil to draw low-fidelity storyboards that convey the necessary information. Casual motion designers, on the other hand, felt like they should create advanced storyboards. This attitude, coupled with their lack of skills, meant that, at times, they did not feel confident in their storyboards. Professional motion designers were mainly concerned about conveying their ideas through the storyboards. They expressed how doing that makes it easier to adjust in future steps: *"My storyboards are embarrassingly low fidelity. If I draw a circle and an arrow and I make a note at the bottom of what it's going to be, I'll know, but I don't bother fleshing them out too much because I like to keep that part of the process a little broad." (P9)*

3.4.4 Production and Post-Production Challenges

The interdisciplinary nature of the process (graphics, text, images, sound, movement, editing, etc.) means that several software applications are needed to create a motion graphics video. Participants, notably casual motion designers, expressed their frustration in learning many (often complex and feature-rich) tools. Our analysis revealed that at least two distinct software tools (Adobe Photoshop and Adobe After Effects [52]) were used to create a motion graphics video. While participants benefited from transferring their knowledge

from one software to another, even professional motion designers expressed the difficulty of working with many tools. For example, a participant working on UI videos shared their frustration: *". . . it creates a lot of difficulties when you switch between software. But I know that there is no perfect software. So, each software has its own strong parts like there are advantages and disadvantages because we have to use a lot of software." (P4)*

Furthermore, to make the video's motion, all (or some) of the artwork had to be animated. Animating the artworks is a software-based process that is a primary difference between camera recorded and motion graphics videos. While in camera-recorded videos, the motion is created using actors and camera movements, in motion graphics, designers invest their time in creating and animating graphical elements in software applications. This step was described as being "labor-intensive" and "time-consuming." Beyond the difficulty of animating, while all participants mentioned that knowing the animation principles had little to no bearing on their ability to create appealing videos, they acknowledged that the current tools are not intuitive for individuals from non-animation domains. Participants expressed how the underlying terminologies were challenging to understand for those from fields other than animation or design: *"The hardest thing for beginners [is] that they don't understand the interface from the first sight because it's not user-friendly. . . And there are no tips. No easy ways to tell about this software, to tell about this interface in a really quick way." (P8)*

Moreover, participants described how the state-of-the-art tools had been appropriated from other creative domains and were not necessarily designed for motion graphics workflows. As a result, motion designers had to devise a range of workarounds (e.g., macros and scripts, copying keyframes, external plugins, and re-using previous animations) for simplifying their creation process. Some had even invested time in learning programming to develop their own solutions. They expressed how using these third-party plugins was necessary for having a functioning software: *". . . without extensions, scripts and plugins, After Effects wouldn't be half as powerful . . . It becomes a requirement for people to know those plugins in order to be able to work in the industry and the higher league." (P13)*

While participants liked the control over their message, they also spoke of challenges with the size of the process. For instance, they expressed difficulties in choosing from many motion graphics styles or the need for a working knowledge of different software tools to create different content types: *"Motion graphics is a very broad thing. For example, character animation or visual effects. . . So, I hope that there is software that will be readily understood for [everyone]. Because we do not know the terms for visual effects or animation. We are not specialized in it, and we hope that there's a software that can make it easier." (P7)*

Finally, as most software tools were designed for those with training in animation or design, it was tricky for casual motion designers to begin with the software tools on their own. The lack of such DIY tools widens the gap between where video publishing has reached

(where non-experts can easily publish their videos) and motion graphics creation (which is still an expert-driven activity).

3.4.5 Challenges in Learning and Locating Examples

Learning the techniques of motion graphics can be a challenging task for new motion designers. While numerous online resources offer training materials, our participants indicated that the relevant instructions and tutorials were not always easy to locate. Furthermore, we learned that casual designers often had difficulty understanding where to even begin or select appropriate techniques for their own projects. A recurring theme in the interviews was that participants usually learned the skills by watching motion graphics videos created by other designers. All participants identified YouTube as a key resource for learning. They further noted the importance of viewing complete projects and cited the benefits of watching "making-of" videos in which the process of creating a video is demonstrated. Participants explained how watching such making-of videos was more useful than click-and-follow tutorials as it helped them view the workflow of other designers: *"I mean if you watch a lot of motion graphics videos, you kind of get that sense for how other people do it too, and then, getting inspired from those making the scene." (P11)*

Despite the benefits of watching making-of videos, we learned that casual motion designers usually struggled to understand the overall design process. Casual motion designers often had limited experience with the software applications shown in the demonstrations and the video authors sometimes assumed more advanced knowledge, which in turn resulted in casual designers being unable to apply the instructions to their needs: *"Even [if] the examples did exist, finding the precise thing you want is harder than just making it. How do I use it, I want something to fly in from the left to the right and jiggle and fly off the screen? You can't find the specific thing you need. . . And these issues waste a lot of time." (P6)*

Furthermore, participants also spoke of the importance of looking at examples to select the style for their videos. The style referred to the video's presentation mode, such as two- or three-dimensional (Figure 3). Each style would need a different set of skills and software applications, and some styles were more demanding to create than others (e.g., 3D was more complex and laborious than 2D). While professional motion designers could infer the complexity of either style and assess the time (and skills) required to create a video in either style, casual motion designers found this challenging. They struggled to understand the technical difficulties of making videos in each style or even the software tools required for them: *". . . there [are] definitely many styles and artistic types of motion graphics. I know that that's only two categories, and it's very not descriptive, but it's … that's a very hard task I think any motion graphics artist would have a difficult time coming up with plans for the right one [style] and the software he needs." (P10)*

Participants pointed out that a trend in motion graphics was to use premade content, such as graphical artwork, visual effects, or animated artifacts used directly within new

video productions. The access to such premade content was essential for casual motion designers as they struggled with creating pieces of the artworks on their own. Furthermore, while casual motion designers relied on such premade content to feel confident and overcome the issue of the "blank canvas," the software tools alone did not provide sufficient artworks or examples. The presence of such premade content (e.g., icons, stock imagery, animated objects, etc.) would support designers to speed their process up and skip the parts that were more difficult to understand: *"The actual animating is definitely the most labor-intensive part. I would say, for creating that stuff, I think the way to speed that up [process] is to use premade templates and premade graphics." (P9)*

However, such premade content came with its own challenges. Motion designers spoke of various hurdles such as the cost associated with such content, the ability to edit them, and the challenges with locating the appropriate content within numerous providers, websites, and software: *"If there were templates that fit everything, how do you search for that? It's virtually impossible to find what you want because the space is so big." (P15)*

3.4.6 Challenges in Communicating and Collaborating

The motion designers in our study repeatedly identified the importance of visual communication as part of motion graphics creation. For these designers, communicating with stakeholders such as colleagues, managers, and clients was an important part of the job. Most of our participants (11/19) usually had to work with stakeholders remotely. They expressed many challenges in communicating the visual aspects of the artifacts through traditional communication means (i.e., text, phone, or email) as they are not designed to convey visual content:

"It's email, or the worst is the comments added a Word document that gets emailed around. That still drives me crazy. I'll still deal with studios that will send me a marked-up Word document, and I'm like, "How do you guys function?" People seem very hesitant to transition to collaborative tools." (P16)

Motion designers resorted to approaches such as video calls with screen-sharing, cloud spaces, or even using storage disks to address issues in establishing a common ground. The introduction of new tools such as video annotation software helped simplify some parts of the process. However, participants expressed concerns over using such software as their tool-set of creating motion graphics was already crowded, and every new tool presented a steep learning curve and more management:

"In interacting with the client, I use different software to get their feedback, mainly Google drive, and Frame.io. But those two may get confusing. But not one software does everything that I would like it to have." (P5)

A primary workaround for participants was visual content, such as low-fidelity storyboards, sample images from the Internet, or online videos that help them convey their

intentions easier. As found in other works, visual communication is essential as not all stakeholders (e.g., clients) have the imagination to foresee the future of video from text-based descriptions:

"We have half an hour conversation, two days later I send them some images, and then if they say, "I love this one, but I hate this one." But can't explain why they hate the one or love another one, and the images are kind of similar." (P9)

3.4.7 Key Takeaways

The key takeaways from this study (summarized in table 2) illustrate that all motion designers agreed that the language and terminology used in most tools were challenging to understand for users without a background in design or animation. The visual nature of motion graphics led motion designers to devise various visual communication strategies, some of which were not always successful. Furthermore, for many motion designers who worked remotely, online communities were essential in creating and learning by example. However, navigating the way around these communities and locating relevant content was another challenge for motion designers.

3.5 Survey

Our interviews with motion designers from a variety of different backgrounds established challenges and workarounds in creating motion graphics videos. To understand the prevalence of these findings more widely and to further understand differences in casual and professional motion designers, we decided to conduct an online survey. In this section, we provide an analysis of the results from our survey of 207 motion designers from 71 countries. We had almost an even split in responses from casual (49.8%) and professional (50.2%) motion designers.

3.5.1 Participants, Tools, and Styles

During our interviews, participants mentioned a few different software tools. Our survey showed that, in fact, some tools were much more dominant among motion designers than others. An overwhelming majority of survey respondents were using Adobe After Effects (92.7%), Adobe Photoshop (84.0%), Adobe Illustrator (73.9%), Adobe Premiere (60.3%), and Maxon Cinema4D (42.5%). Interestingly, survey respondents indicated that the main determining factor in choosing a motion graphics software was the availability of learning resources (63.7%), and compatibility with existing tools (58.9%).

During our interview we had learned that there are various styles of motion graphics (examples shown in Fig.). Style referred to the presentation mode of the video such as 2D or 3D. To further understand the prevalence of these styles in practice, we created a list of the popular styles (from the interviews) and asked our survey respondents to rank them

based on what they use in their own work (or to suggest new ones). We found out that our respondents who were casual motion designers utilized 2D (86.4%) or typography (65.8%) videos (these tend to be simpler to create, as we learned in our interviews). Creating videos in other styles would either require artistic skills (i.e., traditional animation) or expertise in other software (i.e., 3D tools). Figure 1 shows an example of popular styles as well as their prevalence among professional and casual motion designers.

3.5.2 Motion Graphics Videos Motivation and Content

One of our research goals was to understand the different content that is presented through motion graphics. Our survey results revealed that while professional motion designers had more experience in content such as documentaries (26.8% more than casual motion designers), casual motion designers primarily worked on explainer videos (56.3%), advertising (45.6%) and educational content (39.8%). Even for professional motion designers the most exercised content was working on explainer videos.

3.5.3 Challenges of the Design Process

Given the process challenges that we discovered in our interviews, a few motion designers had raised the idea of using premade content. We wondered how common this practice was among motion designers. We also wanted to further understand the prevalence of the many software-related challenges that had emerged in our interviews and wanted to investigate how these challenges manifested for casual vs. professional motion designers. Table 4 presents a summary of the key challenges faced by both professional and casual motion designers.

3.5.4 Use of Premade Content

In our interviews, we learned that casual motion designers often used premade content to create their videos. In contrast, professional motion designers avoided using premade content. They attributed this to the difficulty of editing them (66.1%) and the cost associated with purchasing premade content (65.7%). These findings are similar to those of our interview phase as interview participants suggested that premade content would often take longer to adjust than creating them from scratch. Professional motion designers' disinterest in using premade content, means that there is little investment in continually improving them and would subsequently affect casual motion designers.

3.5.5 Software Tools Challenges

Based on the challenges that participants described in using complex feature-rich software for motion design, we probed into the prevalence of specific reasons. Our survey showed that the major problems were that interactions were complex, user interfaces demanded a

lot of repetitive tasks, or the menu systems were old and difficult to understand. For casual motion designers, the primary challenges were seen in overwhelming user interfaces (46.9%), and the repetitive nature of tasks (39.0%) such animating. While the preference for working with UIs depended on individual motion designer's experience with the software, complex feature-rich user interfaces made the learning curves rather steep for all motion designers. A key challenge in interacting with the software was that most of the tools used by motion designers were not designed for motion graphics. They were, in fact, adapted from other domains such as video editing or traditional and 3D animation over the past decade [3]. Such software has grown to cater to many disciplines, including motion graphics, and often have feature-rich complex user interfaces. Challenges with Learning

Our survey probed into any differences in learning approaches and challenges experienced by professional vs. casual motion designers. Our results established that for casual motion designers, the biggest challenge was finding resources for learning new techniques (67.2%). We had previously found the same result in our interviews as motion designers expressed the importance of receiving in-person support during the early stages of their learning journey. However, instructor-led training in motion graphics is not available in most countries around the world [56]. As a result, for many of our survey participants from 71 countries, online learning was the only path of learning. This dependence on online content coupled with the fact that more than half (50.2%) of participants worked as freelancers, implies that a large body of motion designers lacked any kind of access to formal face-to-face learning or guidance.

3.5.6   Communication Challenges

During our interviews, participants spoke of a variety of bottlenecks they faced in communicating their creation with various stakeholders. In our survey we sought to understand the prevalence of these communication challenges. Professional motion designers highlighted that receiving the correct requirements for a project was their biggest communication challenge (57.8%). This finding was similar to our interview phase where we learned that most stakeholders had difficulty articulating the visual results of a video through text-based communications. Table 5 presents the prominent communication challenges.

3.5.7   Key Takeaways

In summary, we found that the choice among a wide range of content and styles depended on motion designers experience. We also found that casual motion designers faced difficulties in interacting with complex software user interfaces while professional motion designers saw less value in using premade content. Finally, communicating with stakeholders was a challenging task using the current generation tools.

3.6   Discussion

Our results indicate that despite the promise of motion graphics for informational domains, casual motion designers, who were domain experts in fields such as science or education, faced an uphill battle trying to create a motion graphics video. Although some domain experts had solicited professional motion graphics services, this is not a viable option in the long run because of high costs. Furthermore, the subject matter's nuances were often not accurately captured by professional designers, forcing domain experts to tinker with complex software on their own.

Our study has illustrated critical barriers for entering the motion graphics field (summarized in Table 2) shaped by the need for interdisciplinary skills, availability and a working knowledge of feature-rich expensive software applications, and continued access to a strong community of professionals. For casual motion designers, learning the role of different steps was understandably challenging as none of the current software tools integrate all stages of the process (pre-production, production, and post-production) in one package. Furthermore, choosing a video style and interacting with feature-rich software applications were among other significant challenges. Although even professional motion designers were not immune to difficulties, such as labor-intensive animations, some had developed successful workarounds over time. However, even this group struggled to share their ideas, drafts, and creations with stakeholders.

While there is an opportunity for the HCI community to better support all motion designers, the need to understand casual motion designers who could benefit from these videos for their domain-specific needs is more urgent. Future initiatives can democratize creating motion graphics to empower more users to convey their ideas and complex concepts using this rich information-sharing medium. Similar challenges have been tackled in HCI for many years to close the gap between difficult technical tasks and content creation by casual users (e.g., complex domains and UX, supporting software programming participation, visualizations for everyone, or website creation.) We take inspiration from these efforts and offer future directions specific to lowering the barriers-to-entry for new motion designers.

3.6.1   Quick and Dirty Content Creation

In our study, we found out that one of the challenges of beginning with motion graphics for novices is the issue of a "blank canvas." Having no content to begin with deters novices' confidence. Future works can explore options for creating content "quick and dirty" to address that. Using libraries or premade content, icons, pictograms, automated text to image synthesis [169], or dynamic illustration, such tools can enable designers to turn their ideas into visual artifacts immediately. Furthermore, content-aware tools can synthesize users' ideas and offer relevant imagery from previously created content to inspire new videos. Such automation techniques can simplify the content creation process and increase beginners'

confidence in the process. Doing so will enable those who are currently using low-level tools with limited motion graphics capabilities (e.g., slideshow tools) to leverage the benefits of using motion graphics while at the same time providing interfaces that simplify the content creation for more advanced users.

3.6.2 Contextualizing the Design Process

One of the main findings of this study was that new and casual motion designers faced difficulty in understanding the role of each of the pre-production and production steps in the broader context of their projects. For instance, they did not appreciate the value of creating a shot-by-shot breakdown, storyboard, or designing a style frame before directly tinkering with a design or animation software. Furthermore, the need for multiple software applications to complete a project made the process rather complex and dispersed for casual motion designers. This area introduces an opportunity for HCI and visualization research to draw upon the successes of approaches used to enhance the design of interactive user interfaces [152]. Future work can explore techniques to support presenting each step of the process within a broader context of the entire project to help casual motion designers understand how different stages relate to one another. Such tools can allow users to navigate the distinct steps, relate the outcomes of the various steps in one software, and subsequently, better contextualize the process. The key goal here would be for beginner motion designers to not only better understand but also actively embrace the crucial pre-production steps of motion design.

3.6.3 Facilitating Content SaaS-ification

We learned that most casual motion designers were concerned with producing their videos as opposed to earning mastery in the trade, which was not surprising given that they already had day jobs in other domains. There is an opportunity for designing tools that could better support the "Software as a Service" approach. Such democratization and disintermediation have already been used in domains such as web development and 3D printing. In doing so, technical tasks are delegated to experts, and casual users focus on creating content. Some recent efforts in online motion graphics tools [16] have also begun working towards this strategy. Utilizing such a philosophy in motion graphics can enable casual motion designers to benefit from shared artwork, scripts, visual references, and pre-animated artworks to simplify the creation process. Pre-made content in such an environment can be adjusted to other users' specific demands on a shared platform. Doing so will eliminate the need for the tedious, time-consuming, or repetitive tasks for casual motion designers (e.g., animating). In such settings, the techniques of a community of experts can be harnessed to enable motion designers to focus on their messages.

3.6.4 Enhancing Learnability of Motion Graphics Tools

The sheer number of tools, menus, and user interfaces' complexity was a common challenge noted by motion designers. HCI has long focused on improving user interaction with feature-rich applications through appropriate user interface design or investigating approaches for improving software learnability. Building on this work, there is an opportunity for future research to address the usability and learnability difficulties of current motion graphics tools. For example, one approach could be embedding interface divisions in which aspects of the software related to creating motion graphics are separated from other techniques such as video editing or compositing. Allowing further customization of the software tools using strategies such as user-controlled adaptable menus [150], drag and drop node-based systems, and encouraging an environment of sharing customized user interfaces can support and casual motion designers. In our view, any or a combination of these approaches will enable individuals across many domains, such as education or science, to benefit from sharing their knowledge using motion graphics videos. Involving more individuals will also result in more creative ways to be developed and the tools to be utilized in unforeseeable ways and further support information sharing democratization.

3.7 Limitations

Although our study included perspectives from a broad range of motion designers, our focus was mainly on the processes and challenges of getting started in the motion graphics field. Future work can include studies of motion design professionals and the challenges they face when working on more advanced projects. Moreover, our results showed that Adobe After Effects has a significant prominence in the industry, and our participants suggested that they do not have much to compare with it. Future work can consider other research methods, such as case studies, to contrast multiple software applications within a smaller organizational or team setting and even consider controlled studies for comparing different tool interventions and approaches.

3.8 Conclusion

Motion graphics were traditionally created by professional artists and mainly used in film-making or advertising productions. However, they are being used in various informational contexts, and there is an increased interest from domain experts to use motion graphics to explain complex concepts, such as in education. The issues of high costs encourage many individuals to become casual motion designers and take it upon themselves to create their own motion graphics videos. Our study is among the first to contribute empirical insights into motion designers' current practices and workflows. Motion designers face several challenges: from the initial training and onboarding to completing their first motion graphics video

to using feature-rich software and communicating and collaborating with stakeholders. We have identified several opportunities for future work in HCI to build on this knowledge and innovate on tools and processes that can lower the barriers-to-entry to the field of motion graphics. By doing so, we can help casual designers participate in creating motion graphics and communicate complex informational concepts using this rich medium. We imagine a world where a domain expert (such as an educator) can leverage motion graphics videos as part of their daily routine with the same ease with which they can integrate slideshows. We further show how there is a strong motivation to create motion graphics videos independently. However, the complexity and the sheer number of software tools required deter most non-professionals from making their own videos. This phenomena, however, is not unique to motion design. In chapter 4 of this dissertation, we introduce the reasons behind switching among many software applications in the broader context of knowledge work. We further elaborate on the challenges that such application switching introduce and out forth opportunities where the HCI community can streamline the processes.
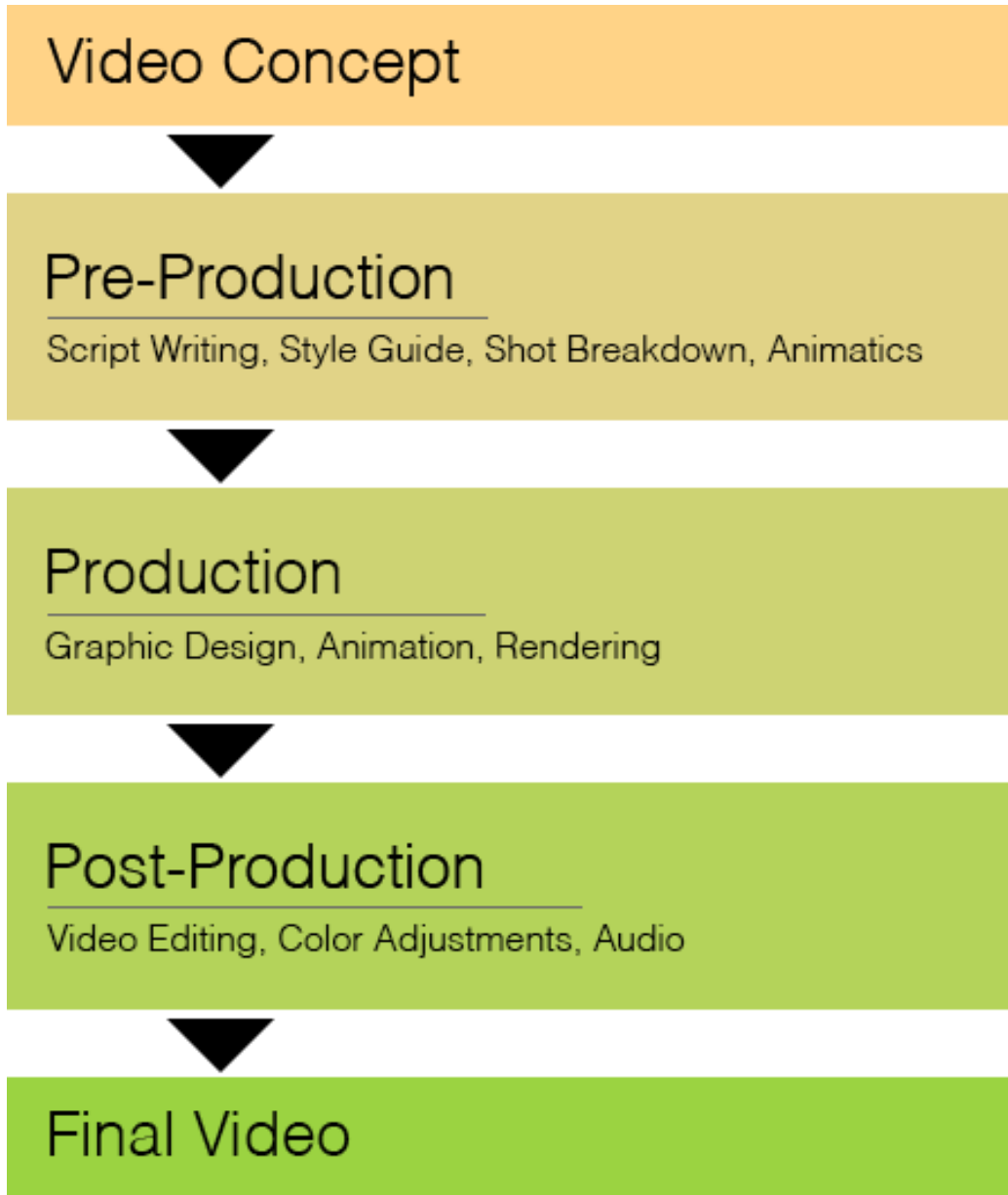
Figure 3.2: A typical process of creating a motion graphics video.

2D
| Novice: 86.4%
| Experienced: 75.6%

3D
| Novice: 38.5%
| Experienced: 67.3%

CHARACTER ANIMATION
| Novice: 27.6%
| Experienced: 60.6%

ISOMETRIC
| Novice: 30.1%
| Experienced: 59.6%

KINETIC TYPOGRAPHY
| Novice: 65.8%
| Experienced: 44.2%

CEL ANIMATION
| Novice: 15.5%
| Experienced: 35.2%
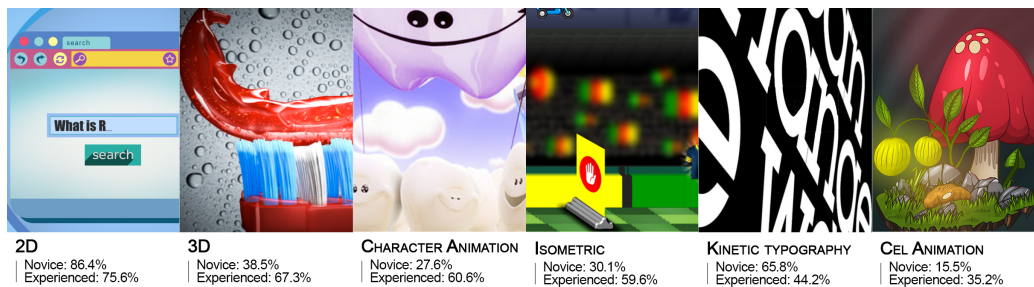
Figure 3.3: Examples of popular styles of motion graphics videos.

1 - Motion designers come from many different backgrounds and often struggle with feature-rich software tools that assume previous training in design or animation.

2 - There are multiple different stages (e.g., script-writing, animating) in creating a motion graphics video. Motion designers (particularly casuals) struggle to understand the role of each step of the process. Furthermore, none of the current tools provides a method of contextualizing this process.

3 - Current software tools do not directly support the pre-production stage of creating motion graphics videos, which many professional motion designers believe is the most critical phase in creating a video.

4 - Motion design requires access to large repositories of examples and references (images or videos). However, novices struggle to locate relevant content and adjust them to the needs of their projects.

5 - Motion designers find it most useful to look at examples and relevant step-by-step instructions, but most current tools lack such example-based learning opportunities for creating motion graphics.

6 - Multiple tools are required to create even a short motion graphics video, making it challenging for casual motion designers to keep up with learning and using multiple (often feature-rich) software tools

7 - Motion graphic design is a collaborative activity that requires back-and-forth between designers and stakeholders using different tools. Motion designers desire support for in-context communication and collaboration within the tools that they use for design.

Figure 3.4: Key takeaways of motion graphics video auhtoring processes and challenges.

Chapter 4

Task-Centric Application Switching: How and Why Knowledge Workers Switch Software Applications for a Single Task

One of the key findings from my study in Chapter 3 was that motion graphics designers were working with dozens of distinct tools and constantly switching between them. In my second study[1], we delved deeply into the phenomenon of application switching, particularly exploring the reasons and triggers that prompt not only motion designers, but knowledge workers more broadly to switch between multiple software applications. This chapter is not just about understanding the why and how of application switching; it's about unraveling the underlying complexities and inefficiencies in the workflows of knowledge workers, especially those related to motion graphics design. This investigation allowed us to understand the broader context within which motion graphics tools operate and for identifying key areas where new software solutions could enhance efficiency and creativity.

Through comprehensive interviews with 15 knowledge workers and five product teams, each deeply immersed in diverse, tool-intensive tasks, we sought to unravel the intricacies of their digital workflows. Our goal was to uncover the rationale behind their choices to alternate between different software tools for single tasks. The insights gained from these interviews allowed us to construct an initial taxonomy of reasons for application switching. This taxonomy not only highlights the varied motivations behind such behavior but also underscores the critical role of collaboration and the impact of external factors in this dy-

---

[1]Portions of this chapter were originally published in Jahanlou, A., Vermeulen, J., Grossman, T., Chilana, P., Fitzmaurice, G. and Matejka, J., 2022, December. Task-Centric Application Switching: How and Why Knowledge Workers Switch Software Applications for a Single Task. In Graphics Interface 2023. [106]
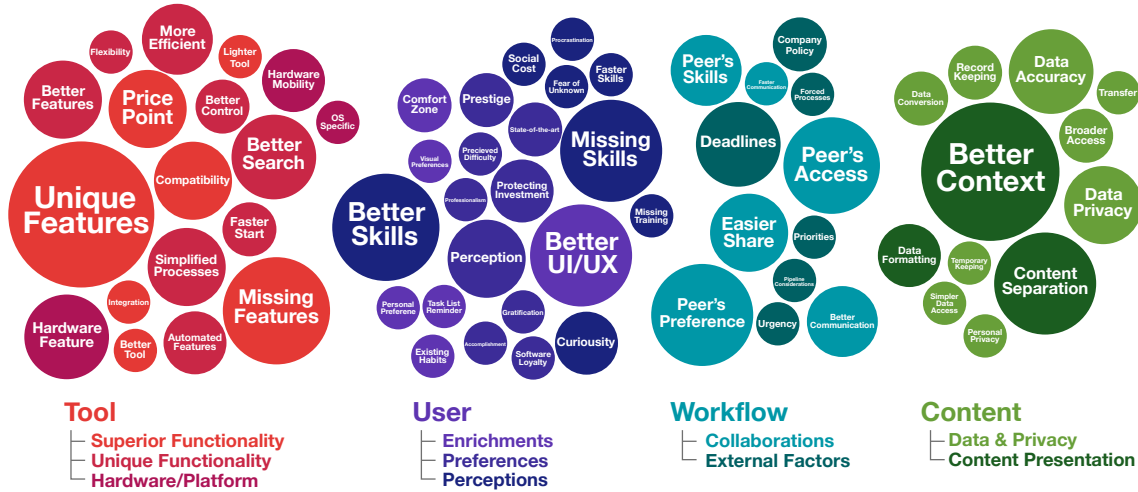
Figure 4.1: The low-level reasons for application switching captured during this research (colored bubbles) divided into four primary categories (Tool, User, Workflow, and Content), each with sub-categories (listed below the bubbles). The size of a bubble represents the number of responses for that low-level reason on a logarithmic scale. The largest bubble size represents 16 responses, and the smallest size represents a single response.

namic. We illuminate the challenges arising from frequent application switching and propose how these insights can inform the design of more integrated and streamlined workflows. Our findings are pivotal in understanding the digital ecosystem of motion graphics design, leading to the development of a novel animation software that addresses these identified gaps and fosters a more seamless and productive creative process for both casual and professional motion designers.

4.1 Introduction

Application switching occurs when a user navigates from one application to another [174] and is common in everyday software use [187]. Research has shown that knowledge workers have eight or more windows open [98] in most situations (78.1%) and make hundreds of switches within a single hour [158, 129]. Such switches come at the cost of launching a new tool, waiting for the tool to load, and in some cases converting and transferring data to another application.

Prior work has explored application switching from multiple causes, such as interruptions [60, 81], multitasking [4], and window switching [187] (more details in sections 4.2.2 and 4.2.3). While the literature has documented a high incidence of application switching when interruptions occur [3], this type of switching is often not a conscious decision by the user. A phenomenon that is not as well understood is why users *deliberately* switch applications while completing *a single task* – i.e., when the user switches between applications with the goal of finishing the same task. We define *task* as a specific software-based activity that a user performs to achieve a particular goal, such as editing a video. In this case, a video

author might make a conscious decision to switch among different applications for image manipulation (Adobe Photoshop [9]), title creation (Adobe Premiere [10]), sequence editing (Final Cut Pro [101]), or color correction (DaVinci Resolve [133]) to complete the same *task* of creating a video. We characterize the act of switching between multiple applications with the goal of finishing the same task as *task-centric application switching*. While previous works have explored the mechanics of switching behaviors (e.g., the use of shortcuts [181]), the reasons behind these switches and resulting difficulties are open questions, particularly from the perspective of knowledge workers.

Knowledge workers are individuals whose primary function is to create, share, and analyze information [167, 67]. Previous studies [60, 81] have shown that knowledge work is characterized by multiple ongoing and often disjoint tasks [22]. Although switching between applications provides knowledge workers the opportunity to learn and transfer their knowledge [3], it also increases the cognitive load [4, 158] that could impact overall productivity. Moreover, frequent switching between tools may also prevent users from developing fluency or expertise in any of the individual tools [93]. This paper establishes an initial understanding of the phenomenon of self-initiated [60] task-centric application switching within sample domains of knowledge work. We focus on understanding the reasons for *why* knowledge workers switch between tools and the potential challenges that result from these switches.

We conducted semi-structured interviews with 15 knowledge workers to understand the practices of switching between applications. We also had the unique opportunity to explore this question from the perspective of product teams that design software tools. We conducted group interviews with five product development teams to understand their decisions to embed a new feature, integrate tools within larger applications, and opinions towards learning how to use their software along with other applications. Our findings suggest that users deliberately switch applications due to tool-specific functionality, the need to collaborate with others, company policies, users' attitudes towards using (and learning) feature-rich tools, individual needs for data conversions, or privacy concerns. We synthesized our observations into an initial taxonomy of why knowledge workers switch between different applications for the same task.

Although application switching was perceived to be helpful for the task at hand in some cases, it also introduced challenges that impacted users' productivity. The cost of switching between applications due to data transfer, the time required to re-focus on the task, and the extra cognitive load the users must endure are some of the difficulties that application switching bears. We discuss the implications of how software tools can better support task-centric application switching and help knowledge workers be more productive and efficient in their tasks. In summary, our paper makes the following contributions:

- Initial insights into task-centric application switching, including causes, benefits, and challenges for knowledge workers.

- An initial taxonomy of reasons for application switching.

- Design implications for developers and learning content creators to better support task-centric application switching.

4.2   Related Work

To situate our findings, we draw upon research on software learnability, multitasking, and productivity support tools.

4.2.1   General Software Learnability

Software application users are challenged by featurism, the growing list of application features with each release, and the training that is often focused on a single software tool. Previous HCI research [141, 88, 116] has explored this from the perspective of software learnability. These works have made recommendations for improving the task flow [88], user awareness of the UI [141], improving feature findability [71], and understanding and reducing functionality [44]. Personalizing user interfaces [67] and adding customizability have also been explored. Moreover, contemporary knowledge work is rarely an individual activity [94, 22]. These collaborative environments are typically composed of several arrangements among multiple users and software tools, and collaboration itself causes difficulties with learnability [156]. Many challenges stem from the fact that each software company designs the tool and the training of their respective software, and the practices of knowledge workers—who often utilize several tools together with one another [80, 167]—are not fully considered.

Previous work on supporting individuals with small units of tasks (e.g., document production, email, and communication [80]) has largely focused on individual software tools and rarely considers the difficulties in utilizing multiple applications to complete a task. To address this need, initiatives in industry and research have resulted in larger software applications instead of multiple small applications that work in concert with each other [125, 77, 121]. While limited functionality tools are only useful for particular conditions, the difficulty of learning complex software [51] and the challenges of working with feature-rich tools [116] make users hesitant to use them.

In this paper, we highlight application switches resulting from challenges of feature findability, collaboration, and the need for appropriate tools that reflect and inform users of their practices. More broadly, our work contributes to a trajectory of research in the HCI community [6, 31, 102] aimed at studying the needs of knowledge workers to better support their practices, procedures, and experiences.

4.2.2 Supporting Users in Multitasking

Previous studies in HCI have explored approaches to support users in multitasking, such as software work interruptions [185] or situations when users lose their flow of work [145]. Much of the multitasking efforts look at "work fragmentation" [175]. In doing so, scholars have explored the role of task management and window switching [187, 93], understanding values that influence the adoption of creativity support tools [160] as well as the inclusion of devices such as mobile phones [172] for work purposes. A notable focus has been on improving application management [157, 108], and in understanding and providing a definition of a unit of task [103, 153, 67]. Furthermore, the utility of larger screens [93], and the benefits of utilizing multiple displays for work [98] as well as some of their difficulties [93] have been explored. Another area of study can be seen in activity-centric computing systems [34], and the analyses of the different window overlapping techniques [107] while simultaneously having a high-level understanding of the process [92].

While prior work demonstrates a few instances in which multitasking has been positive (such as enabling better creativity [134]), most literature on multitasking attempts to address its challenges. Among other difficulties is that for seamless multitasking [158], application switching must happen in near-simultaneous execution [174]. However, switching between applications results in interruption that often taxes the process [80]. It is worth noting that there is a difference between task switching and an interruption, as the former is generally a conscious choice for a more extended period while the latter is usually a temporary shift in attention caused by external factors [3]. Our work complements these previous studies by investigating knowledge workers' specific reasons for task-centric application switching and the challenges they face as a result.

4.2.3 Task Management and Productivity Support Tools

HCI has a long history of exploring productivity measurement and support for software users [22, 6, 31, 102, 141]. Switching between application contexts provides knowledge workers the opportunity to learn and transfer their knowledge [4]. Yet, changing contexts for completing the task might come at a cost [4, 158]. It produces an increased cognitive cost from fragmenting the work and a resumption lag that reduces performance [175]. Previous studies have illustrated how users may opt for sub-optimal techniques because they preferred working with tools they were familiar with rather than exploring (potentially) better alternatives [156]. Such constraints posed by knowledge deficits are a deterrent to long-term productivity. While there have been some efforts in measuring software productivity [145], concrete methods are needed to understand the processes and support users. A suggestion in that direction has been to provide users with project-specific task reminders [60].

Our study reveals insights into the difficulties that occur due to the tools or the users' workflow. We further explore the challenges shaped by individual traits of knowledge workers

Table 4.1: Overview of the participants in this study representing their domains, age, and years of experience they had in their work.

P#/PT#	Gender	Position	Age	Exp
P1	Female	Office Assistant	25-34	1-9
P2	Female	Software Project Manager	35-44	10-19
P3	Male	3D Content Creator	45-54	20-29
P4	Male	Software Developer	35-44	10-19
P5	Female	Executive Assistant	35-44	10-19
P6	Female	Community Manager	25-34	1-9
P7	Female	Architecture Manager	45-54	10-19
P8	Male	College Manager	45-54	20-29
P9	Female	Visualization Researcher	25-34	1-9
P10	Male	Business Manager	35-44	10-19
P11	Female	Book Author	45-54	10-19
P12	Female	Research Scientist	25-34	1-9
P13	Male	Personal Trainer	35-44	10-19
P14	Female	User Experience Designer	25-34	1-9
P15	Female	Life Coach	55-64	20-29
PT1-1	Male	Program Manager	55-64	20-29
PT1-2	Male	UX Designer	25-34	10-19
PT1-3	Male	Product Manager	25-34	1-9
PT2	Male	Product Manager	45-54	10-19
PT3-1	Male	Product Manager	35-44	10-19
PT3-2	Male	Product Designer	45-54	10-19
PT4-1	Male	Product Manager	35-44	10-19
PT4-2	Female	Design Manager	25-34	10-19
PT4-3	Male	Product Manager	35-44	10-19
PT4-4	Male	Product Manager	35-44	10-19
PT5	Male	UX Manager	55-64	20-29

(such as willingness to learn new tools or the presence of transferable skills) [146] or their settings (such as working against deadlines and in individual or collaborative environments.) Furthermore, our work complements prior studies by highlighting the software and non-software challenges that result from application switching.

4.3   Method

To establish an initial understanding of knowledge workers' processes, successes, challenges, and workarounds in application switching, we conducted semi-structured interviews with knowledge workers from several different domains. Moreover, to understand the perspectives of product managers and interaction designers, we conducted group interviews with members of five product teams (a total of 11 individuals) from a multinational software company. These individuals were actively working on designing new features or integrating previously standalone software tools into a larger application and could share observations of their users' feedback and potential struggles. This paper refers to these teams as PT1–PT5 (product team). All sessions were conducted remotely (via Zoom [208]), recorded, and later transcribed. Sessions lasted 45–60 minutes, and all participants were awarded a $50 gift card.

4.3.1 Recruitment and Participants

Knowledge Workers: We interviewed 15 participants (10 female, 5 male), spanning different age groups (P1–P15 in Table 4.1). We sought to interview individuals who used computer software tools for day-to-day office operations, architectural or UX design, software programming, or scientific research. We tried to obtain a reasonable representation from a variety of domains of knowledge work. Participants were recruited using personal contacts, email advertisements, and snowball sampling over two months in 2022.

Product Teams: Gathering perspectives from commercial product teams about software design decisions can be challenging due to privacy and intellectual property concerns. We had the unique opportunity of being the research division of a large software company (over 10,000 employees) with over 100 software products in its portfolio. We contacted several groups working on different products and conducted group interviews to allow people in various roles to add to each other's answers and arrive at a shared understanding. In five sessions, we interviewed 11 individuals (see participants with prefix PT1–PT5 in Table 4.1 for details on their experience and positions). For 2/5 product teams, we only had access to one team member.

4.3.2 Data Collection and Analysis

Knowledge Workers: During the interviews, we asked questions about application switching and their opinion on the benefits or challenges of switching among different tools. We also asked whether they could recall a situation in which switching has been easy or difficult and whether they switched between platforms. Once we established their application switching behavior, we explored the reasons behind their switching and how they felt about these switches. We encouraged our participants to take notes of the switches between applications in the days leading up to the interview sessions. Several people came prepared with notes, and two interviewees also produced diagrams of their software tool usage.

Product Teams: We sought to understand the teams' decisions of embedding a feature into their applications, the learnability of the tools, considerations for integrating tools within larger applications, the processes of switching between their applications and other tools, as well as their opinion towards learning how to use their software along with other applications. We further inquired about the role of branding and how users perceive a tool.

All interview transcriptions were coded using the *Atlas.TI* [79] data analysis software. We explored the data from the two groups of participants separately, and the coded data were analyzed to illustrate the different processes, challenges, and workarounds that participants had expressed. We used an inductive analysis, and axial coding [58] approach to explore the themes around our main research question. To ensure the validity of the coded data, the primary author performed the first open coding pass and consulted with other

researchers to discuss and develop an initial list of codes. Upon completing the first phase, researchers collectively examined the emerging themes and finalized the coding scheme.

4.4 Reasons for Application Switching

Based on our analysis, we were able to classify the reasons for task-centric application switching into four broad categories: *tool*, *content*, *workflow*, and *user*. We synthesized these reasons into an initial taxonomy and explored each category in detail based on the perceived impact of application switching on participants. This taxonomy of reasons for application switching can help future scholars and software developers provide systems by which users' performances can be measured and improved in specific categories.
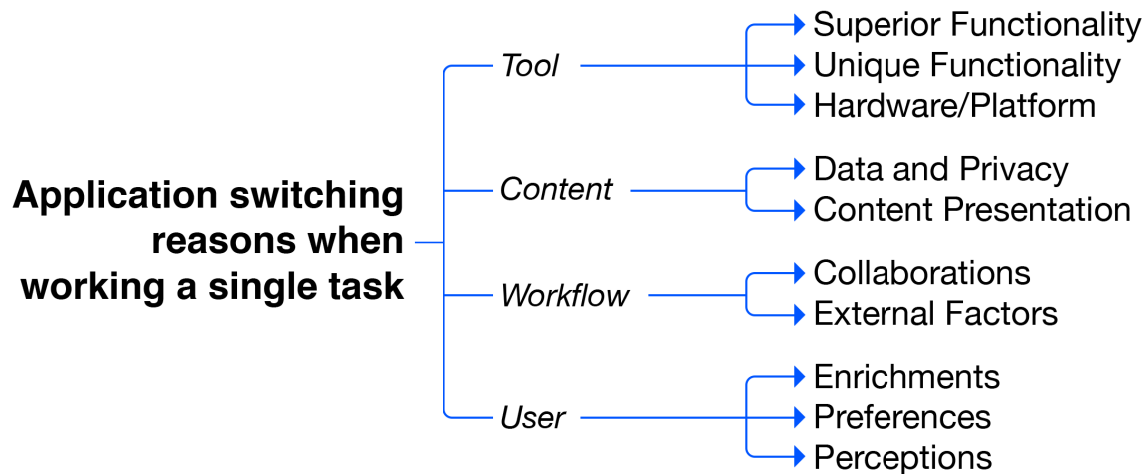
Figure 4.2: The initial taxonomy of task-centric application switching with categories and subcategories.

4.4.1 Taxonomy of Reasons for Task-Centric Application Switching

Our data analysis initially revealed 67 reasons for task-centric application switching. We did another coding pass to work towards clustering these reasons into categories and subcategories (Fig. 4.2). These categories include tool, content, workflow, and user. We synthesized an initial taxonomy of reasons for task-centric application switching based on these categories. Fig. 4.2 represents this breakdown and the various categories and subcategories of reasons.

4.4.2 Tool-Specific Reasons

In this section, we explore the reasons for application switching that pertain to the nature of software tools, such as the superiority of one tool's features, the uniqueness of its features, or the mere availability of an application on a specific platform.

Superior Functionality: Our participants worked in different domains with varying levels of software tool training. Some (such as P3 or P8) had spent years earning mastery of their applications. Others had to learn many aspects of the software work on the job. Regardless of the skill level, all participants had to continually work against deadlines, and for that, finding superior functionality was a concern. For example, P2 working as a project manager, described how they preferred switching to a tool over starting a new document in the current one simply because the other tool would fire up faster. This view was echoed by PT4: *"When we integrate a tool, we add new capabilities. That's great, but we also should work on making the interaction seamless and keep the tool fast enough. If the software becomes too heavy, we lose users."* Another participant—who worked on 3D content creation—explained how they switch between different tools to maintain multiple records for easier sharing:

"A lot of inter-company discussions are happening on Slack. But I like sending emails. Because it is easier to find data. So, if I say something on Slack, I back it up in an email as well. I guess Outlook and email are easier to find information. They do a better job of information retrieval." (P3)

Tool Specific Reasons → *Superior Functionality*
→ *Unique Functionality*
→ *Hardware/Platform*
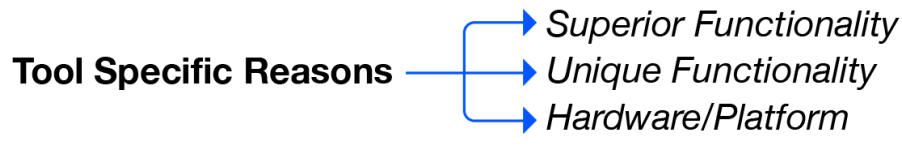
" *We use Slack for communicating in our office, but it doesn't allow us to send large files. So, in the middle of a conversation, I have to go to email or the [Google] Drive. (P2)*

Figure 4.3: The high-level reasons for application switching that relate to the nature of the tools.

Unique Functionality: Beyond superior functionality, participants also spoke of many situations where they had to switch to a different application because of its unique functionality. An example was a user wishing to edit an SVG figure before using it in a presentation in Google Slides [83], which they could only do with a particular software tool (Adobe Illustrator [8]). This was the only software that offered this functionality (to the participant's knowledge.) While this approach results in further application switching, PT2 viewed it as something positive: *The constant demand by users for adding new features is only making the tool complicated. Our tool is very simple, it's on the web, doesn't need installation or maintenance. It's supposed to help users move just one step forward. They can do the rest in a different tool." (PT2)*

Hardware/Platform: While participants spoke of various superior or unique functionality in their tools, there were also situations when they had to switch to different hardware

or operating system. We learned that some of these switches had developed organically, and the users had no recollection of how they had adopted such practices. Similar to a previous study on practitioner values for the adoption of creativity support tools [160], some participants deliberately chose to switch to a different hardware platform (e.g., P13 for the more accessible keyboard on their phone, P6 using their phone for taking a quick photo, or P15 to get up from their desk.) Some of these switches were also related to the specifics of hardware:

"When I want to do [a] freehand drawing, I use my tablet for quick ideation. Now, my laptop has a touch screen that I can draw. But it's not very convenient. So I go back to my tablet instead." (P4)

While some of these were a conscious choice, some had other reasons. P8, for instance, expressed how they occasionally do sound editing and have to switch to a different operating system (Mac OS) because their sound editing tool is only available for Mac. Others constantly navigated back and forth between hardware and preferred to switch when absolutely necessary:

"I usually get the notifications on my phone, but then I go to the desktop application. I use the phone just to get notified that something requires my attention. Beyond that, it rarely happens that I use my phone for actual work." (P9)

A summary of the tool-related high-level reasons for application switching can be found in Fig. 4.3.

4.4.3   User-Specific Reasons

In this section, we explore application switching reasons that stem from the individual differences between the users. Examples of such differences can be found in personal preferences, the training they had received, and the perceptions of the difficulty (or the ease) of using a tool. The user-specific reasons can be summarized in Fig. 4.4).

It would be easier if we could do everything right within Unity. But, none of us knows how to do it, so we end up creating the UI mockups once in Figma and then in Unity. (P14)

Figure 4.4: The high-level reasons for application switching that relate to the characteristics of the users.

Subcategories of User-Specific Reasons

Individual Preferences: Individual preferences factored in users' application switching practices. Participants such as P15 simply did not want to step outside their *comfort zone.* They would switch to a different, more familiar application, despite knowing there is a way to do that task in their current tool. As found in previous literature [45], they could not be "bothered" with learning the feature. P7, for instance, suggested that they prioritize productivity: *"I take a longer route just because I know how to do that. It's the frustration of having to search for the answer, as opposed to knowing you gonna get there."* Our product teams spoke about how users' habits impact their application use which was not always a positive outcome:

"We have users who use our tool for years, now, there's also a new generation of customers. They use iPads and touch phones. For them, they expect the UI to work similarly. Now, we have to align our UX with their preferences as well." (PT5)

Others, such as P8, had reservations regarding application switching, particularly because of the user experience. This was an area brought up by product teams as well:

"If we integrate a tool to our system, it should follow what users expect. So, that is an area that we spend a lot of time on aligning hotkeys and mouse-keyboard interactions." (PT5)

User Enrichment: Beyond the users' preferences, another major reason was learnability and user enrichment. As seen in previous work [141, 105, 45], for most participants, such as P12, the tool was viewed as mere means of tackling a task, and the mastery of the software was not a high priority: *"I want to know [how] to accomplish what I am trying to accomplish. I am not interested in learning many new tools. I want to be able to Google something and just find the answer."* Our product teams somewhat reluctantly agreed with such assessments of feature convolution:

"We invest countless hours integrating various features. Most users don't use these and would prefer to switch to an application with fewer [features] where they feel more comfortable. This way, they finish their work, instead of learning better features." (PT4)

The disinterest in learning new tools was also impacted by the number of applications. P1, for instance, said: *". . . in our onboarding training, they were showing the new hires the hundreds of applications that we have in the company, like for HR or for expenses or for other things. It's mind-blowing, can you imagine how a new hire deals with that? Wow!"* Multiple other participants (P4, P5, P9, or P11) spoke of their lack of knowledge in using a software tool. P1, for instance, mentioned not receiving the right training: *"So, right now, I can't send a sheet automatically to any of three services we communicate with. The guy who knows how to do it doesn't have the time to teach me. So, I end up going to three tools one by one"* Our product teams had mixed beliefs about this.

On the other hand, we also spoke to participants who welcomed personal enrichment through learning new things. For instance, we spoke to a participant who had taken the radical route of developing a tool to reduce the amount of application switching by combining multiple functions of different tools:

"..., you have a conversation, then you realize it needs to be recorded. Sometimes it becomes the project requirement. I needed a tool to transfer casual chats into actual project details." (P4)

Users' Perceptions: We should make a further distinction of user preferences based on what they perceive. An unexpected finding was when we spoke to P1, who suggested that they switched between two applications merely to have the second software used *somewhere in the pipeline.* Their rationale was that the other software is viewed as a more prestigious application and helps them look more professional. Another example is that we would generally assume that users might wish for better software applications. However, participants such as P9 challenged this notion by wishing that they could protect their time investment: *"If I put the time to learn something, then I'd like to be able to use it again. I feel the energy that went into [learning] it, shouldn't go to waste."* We found an interesting confirmation of this view from product teams:

"..., our users have invested in learning the highly specialized software. The integration is somewhat 'cheapening' their efforts. I've heard users say, 'now everyone can use this'." (PT1)

While more experienced participants view this as a way to capitalize on what they have already learned and focus on the task at hand, others, such as P14, viewed it as a bottleneck for new and better ways: *"I don't know the reason. But I think it comes from their background or places in careers. New people are willing to make changes. The more experienced wish to stay within their comfort zone. They force old ways and old tools." (P14)*

The challenges of learning new tools were a more prominent situation for experienced users. PT4, for instance, spoke of an interesting aspect: *"Users now [after integration] have to learn the UI of a new software only to be able to do what they already have been using. For new users, this is okay. For existing ones, they have to first un-learn and then re-learn the new interface." (PT4)*

The initial encounter with the software tools shaped some of the users' perceptions. P11 suggested that: *"If my first interaction has been rewarding, I prefer to go back to that tool because it gives me the feeling of success."* Product teams also spoke of such behavior by their users. PT4, for instance, reiterated *"Some tools are useful for some edge cases. Yet, the user goes there once, gets used to it, if they like [it], they use it for everything."*

Groups of Users and Application Switching Behaviors

As we conducted interviews and later in our analysis, we saw general "groups" of users appear. We organized these as broad groups of software-conservative, software-neutral, and
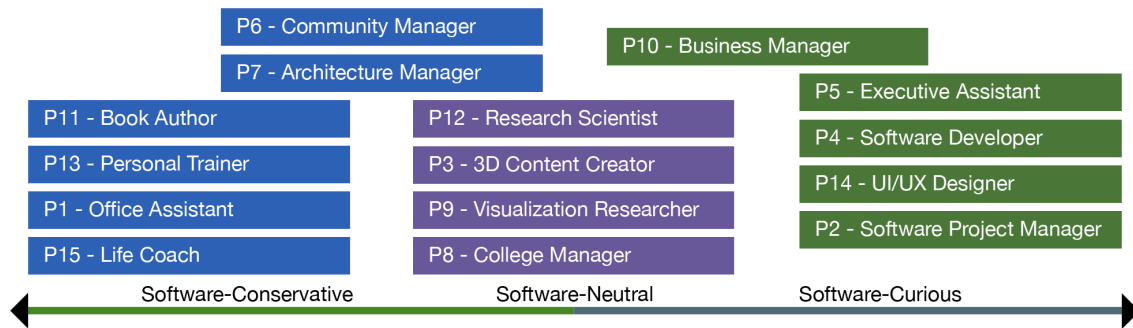
Figure 4.5: Three groups of users emerged in this research.

software-curious users. This categorization was influenced by various parameters such as their willingness to explore new software applications, ability to balance deadlines with onboarding new tools, or desire to support and train other individuals.

Fig. 4.5 illustrates the summary of these groups across our participants. Our *software-curious* participants usually had a genuine interest in learning new applications. They often welcomed the challenges of learning or onboarding new software and were seen as technical people in their workplaces. They would express their feeling towards applications or switching using phrases such as *"I'll try it on my own first," "In my free time, I'll play around with it,"* or *"I look for a better solution."* From this category of users, we met P5, who expressed: *"If I have to learn a new tool and switch to it, I don't mind it. I am very tech-savvy, so I don't mind the trouble. I see the issues as a new opportunity for learning and improving myself."*

At the other end, we saw the group of *software-conservative* users. This group behaves similarly to what is described in the *paradox of the active user* [45]. These users are mainly motivated by productivity and have little interest in learning software or exploring new features or software tools. Participants in this group often expressed their feeling towards applications or switching using phrases such as *"I don't know how to do that," "It seems very difficult,"* or *"I can't be bothered."* Unlike the curious group that saw the benefits of mastering software applications, this group viewed the software as an extra task and was concerned about how they had to overcome the challenges of the software on top of their daily work. P15, for instance, expressed: *"I can't try new tools. I'm happy to do things the way I do. I love learning new stuff, but tech isn't one of them."*

As we went through our analysis, we noticed a few participants that could not be categorized as either curious or conservative. This group was neither intimidated by the software switching nor enthusiastic about the tools. They would express their feeling towards applications or switches using phrases such as *"If I have to," "I wouldn't try on my own"* or *"If someone shows me how to do it."* They would simply use the software applications to move forward to the next level. If they had to, they could easily get started with new

tools and try to learn their intricacies, but that would be when they clearly felt a necessity. We categorized these individuals as *software-neutral* users that can be represented by P9: *"I told myself you have to relax about technology. It's a [matter of] necessity. If it is not absolutely necessary, I will not learn it."*

4.4.4 Workflow Specific Reasons

This section explores the reasons for application switching driven by users' workflows. While the workflows of individuals are vastly different, we observed distinct trends within the contexts of collaboration and external factors (Fig. 4.6).

Workflow Specific Reasons — → *Collaborations*
→ *External Factors*

" *We use Discord for all of our communications. Then there are clients we work with who only have Slack, so I have to move between these two to maintain conversations. (P10)*

Figure 4.6: The high-level reasons for application switching that are based on the various workflows.

Collaborations: Collaboration was perhaps the most influential aspect. In line with prior findings [160], we observed multiple instances in which users would radically change their processes depending on whether they worked with others or on their own. A participant, for example, expressed:

"I have to make distinct decisions about my own use versus my students. Whenever we have to add a new tool for our students, I'm rather hesitant. I want them to focus on what they have to learn as opposed to focus on their tool instead." (P8)

PT1, also encouraged software tool integration for a similar reason: *"The larger application means everyone in the pipeline, designer, engineer, managers, or anyone else shares the project by opening the same project within the same software tool. Then, everyone is talking about the same thing."*

Beyond the necessities of switching (such as using a different tool in the latter stage of a pipeline), we also observed several examples of switching directly related to stakeholders' use of different tools. P9, for instance, suggested: *"I am for adding a new tool. These are little tools, so their learning curve is pretty straightforward. My colleagues don't want to make such changes. We end up staying with the challenging workflow."* While some product teams believed in bringing everything into one platform to simplify the viewing context, this was not necessarily true from users' perspective:

"In our office, the biggest challenge is the perception of integration, not the actual technology. We have tools that provide all [the] different functionalities required, but our col-

leagues don't like them or don't believe that they can do everything. So, they end up [switching to] alternative tools." (P5)

Similar resistance to change, particularly when it required training, was observed by other participants as well. P14, for instance, noted: *"Sometimes, if you invest two hours this week, you'll save 20 hours by the end of the month. But most of my colleagues, almost none of them, will ever do that."* This view was also echoed by PT3: *"With each release, we ship an entirely new set of documentation and training material. Most users, unfortunately, skip those."*

Our product teams spoke of how they rely on the communities as an *extension* to the software. Our participants, such as P10, had a similar view: *"I make quick decisions about whether I like something or not. It's based on usability and the community. Can I jump into Google and quickly get answers without getting frustrated?" (P10)*

External Factors: Beyond colleagues' and stakeholders' preferences, some factors were built into the processes. Among others, regarding dealing with deadlines, many participants spoke of having to switch between applications simply because of time. P7, for instance, suggested that they had faced a situation where they switched (from one CAD software to another) to add some details. Still, they were unsure if the switch was necessary because it was only doable in one software or if they lacked the knowledge to use the first software. Working against a clock also meant that participants ended up on certain paths. P1, for instance, suggested that: *"Most often, a lot of things are happening fast, and we are busy. So, I end up just doing things manually that I know will finish the task."*

Content Specific Reasons → *Data and Privacy* / *Data Presentation*

" *I type up a contract in Word, but when I want to share it with our contractors, I want to make sure they can't change it. So, I convert it to .pdf and then add a password in Adobe [Reader]. (P2)*

Figure 4.7: The high-level reasons for application switching that are based on the content.

4.4.5   Content-Specific Reasons

Finally, we observed patterns in how participants dealt with their data and content (summarized in Fig. 4.7). A lot of these were affected by the need for privacy. P6, for instance, suggested: *"When I'm done in Word, I save the file as .pdf. I can then open it in Adobe [Reader]. It has really great features for setting various levels of access for contractors."* Other participants, such as P4, also echoed the need to switch between applications for various privacy reasons. In their view, the lack of privacy settings might result in them eventually migrating entirely from one tool to another. Beyond privacy, participants also

spoke of the ways that they had to separate their content. Some participants spoke of how they use different software solutions for different aspects of their operations that enabled a better degree of content separations:

"For the goto market planning and ideation and the documentation, we use Miro. For me, everything is there. Everything is altogether over there. If I need something in UX, I go to Figma. This separation is good. This allows me to give access to the right people in sales, marketing, etc." (P14)

Particularly in collaborative settings, we encountered examples when participants would switch from their current tool simply to view things the same way their stakeholders would. P3, who works on 3D content, suggested that depending on the next person in the pipeline, they use different applications to ensure that the data can be opened and manipulated correctly. Product teams also mentioned the viewing context as an important driver for merging applications. PT1, for instance, was pleasantly surprised with a similar situation: *"The visual context is something we didn't expect. Previously, users had to separate part of the data and execute a series of tasks with different tools. After the integration, they get a visual context of all the content at once that makes it more intuitive."*

4.5 Challenges Caused by Switching Applications

Although our participants had many reasons for switching between tools, and some willingly switched for more successful workflows, most faced substantive challenges.

4.5.1 Challenges of Learning the Many Tools of a Pipeline

Most of our participants had to employ multiple unrelated software tools. P5, for instance, brought a diagram (Fig. 4.8) that represented nearly 15 steps and multiple tools required to complete a contract. Others, such as P12—who worked as a research scientist—sometimes did not even know if they were using the right tool: *"I don't like having this burden of search to see if I can do something in that software, as opposed to first identifying the software that I need. I prefer knowing that I am in the right tool."*

Another challenge was the increasing feature-richness of tools, which created skill gaps for users. Participants spoke of how they had received initial training on using a software tool, yet they were unaware of the features added in each new release. PT1 also spoke of similar challenges: *"With each release, we have to add many new features. If we only fix the bugs and make the current tool better, we risk users dismissing it because we haven't made much innovation."*

The lack of awareness of features makes users' skills obsolete. The increasing richness of software applications also meant that the tools had become so complex that participants wished to access lighter, limited versions. P8, for instance, suggested that they would welcome an alternative to their 3D tool. Similarly, P7 expressed: *"You know, the software grew*

because we needed functionality. The problem is it's harder to learn. Needs much more investment in terms of time... there are functions that I'd never use." As observed in prior studies [116], for many participants, trying to employ new tools or features required looking to others for help. This approach comes with its own challenges. P4, for instance, articulated: "*Right now we have a challenge that many things are just the knowledge of an individual. The process shouldn't be based on the knowledge of one person, rather the software that is helping the user.*"
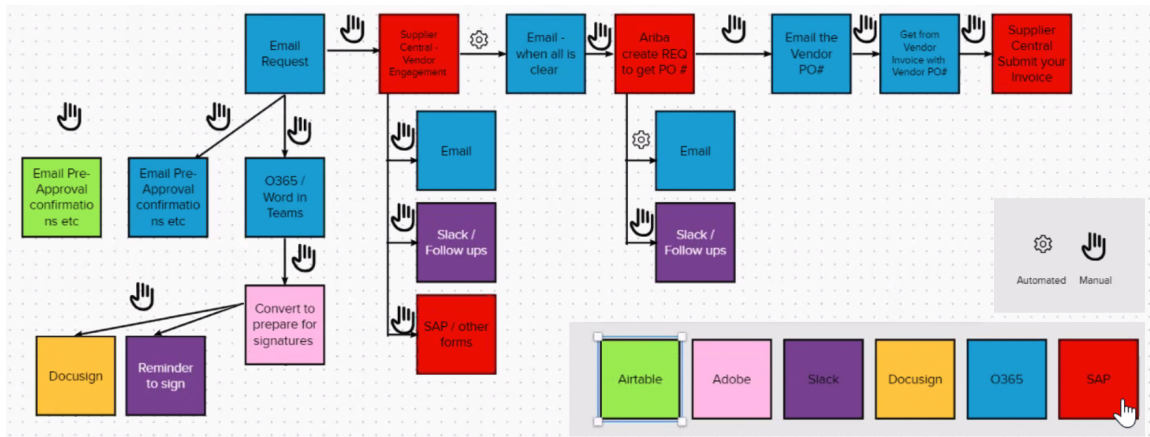
Figure 4.8: An example of switching among multiple applications illustrated by P5 to complete the task of getting a contract signed. The user has to navigate back and forth among numerous application.

4.5.2   Missing Interoperability Between File Formats

Although the technical capabilities of tools were important drivers of application switching, we observed that application switching also happened because users needed more customizability, better viewing of the content (e.g., in a simpler interface), or the ability to isolate specific parts of the content. P9, working on scientific data visualization, for instance, spoke of how their tools did not communicate with one another using standard file formats. Another participant, who worked on 3D content authoring, expressed:

"*I work with people who use MotionBuilder [26]. So, I have to open files in Motion Builder because I need to make sure I see what they see. This way, I can be sure that we both are talking about the same thing. We have that common frame of reference. But if there is another team that uses Maya, then I test the file on Maya.*" (P3)

The challenges of viewing data in different representation modes were not unique to 3D authoring. In fact, our participants recounted many situations where they needed to isolate their content to view them within a different interface (e.g., pieces of a text being edited in a separate view, isolating parts of graphics, reviewing sections of numeric data).

These content transfers often resulted in multiple copies of the files, the need for saving and re-opening the data, and occasionally feeling lost between multiple copies:

"There's a copy of the files on the server, but not everyone keeps that up to date. So, I maintain a spreadsheet that is manually updated, and I compare it to the one on the server. So, I have two files that have to be manually compared" (P7)

Participants complained that their software tools did not allow such content separations or that the user interfaces were rather crowded. In essence, these are capabilities currently built into the tool that don't benefit users due to sub-optimal interaction design. Some of these could be addressed by better discoverability of the features, others by designing better user interfaces.

4.5.3   Understanding and Measuring Productivity

Our participants also spoke of the difficulties of keeping track of many tools and the numerous times they switched between them throughout the day. P1, for instance, suggested: *"Because there are many tools, I can't remember which one of use for what. So, I have written a list for myself. I switch there to find out what tool is used for what. It's like a list of tricks."* In our participants' view, the sheer number of tools and the ongoing introduction of new ones were deterrents to productivity. P7, for instance, advocated for reducing the number of communication tools, while P3 suggested: *"Sometimes it [switching] is a pain because I have to go back to another software application. It'd be nice to have a merged application because you could do everything in one package. It breaks up your rhythm of working."* Despite this interest, participants were wary of the customizability trade-offs that might come with larger applications:

"[Microsoft] Teams is a good example. Even though it's available to us, we don't use it because it doesn't allow customization. It's everything for everybody, so it's not all that customizable. Then some of the things that we really need, we need IT intervention, which doesn't make sense. The workflow shouldn't involve IT." (P8)

P6 viewed customizability from a different perspective: *"I think using many small applications gives me more flexibility. They are separate, but [I] can switch between them more flexibly. Many small apps give me a bird's eye view."* In summary, our participants, spoke of challenges such as:

- Software learnability is often concerned with specific tools; the entire pipeline of software applications needed to finish a task is not considered. Moreover, new releases of tools require re-training.

- Participants were challenged with the lack of interoperability among applications and the constant need to change file formats and open data files in separate applications only to view their content in a certain way.

- Users complained about having to make many switches without any real way of reflecting or visualizing their processes. They advocated for reducing the number of tools while enabling further customizability.

While, for many, these application switches developed organically, and participants were perhaps impervious to them, some, such as P8, were aware of the costs: *"I have that high context-switching penalty. It would have to be a seamless integration for me to use integrated tools; otherwise, I prefer switching to the standalone one."*

4.6 Positive Aspects of Application Switching

Although knowledge workers faced several challenges when switching applications for the same task, they also discussed many positive aspects that facilitated switching through which they could extend their current workspace and seamlessly work with the same data on different applications.

One area our participants appreciated about switching applications was data availability and access. P2, for instance, spoke highly of the default applications installed on Apple devices and the availability of the AirDrop [18] feature and cloud storage in iCloud [21] across different devices: *"Notes [19], for instance, when I enter something in it on my phone, it's automatically on my laptop. Also, sometimes, when I copy text, Siri [20] makes suggestions for what I should do with that. Like if it's for a calendar entry, maps, or text messaging someone."* Some participants also spoke of the benefits of using single sign-in applications that host their data on the cloud. For these users, the backend of the cloud provided the necessary functions for accessing their data, and the single sign-in nature meant that they had immediate access:

"..., like if I have a presentation, I sometimes copy tables directly from Google Sheets into a presentation. It's just there, and it's very convenient. I click on the button, I'm automatically logged in, and my tables get connected." (P8)

Enabling such single-login cloud servers was also a major effort for our product teams. PT1, for instance, suggested: *"By integrating our tools, we also moved to the cloud. That means login, storage, and transfer are activities that the user no longer needs. They can now only focus on their actual job."*

Another approach that facilitated switching was live data integration. These could be seen when users chained a series of files from different applications while still keeping them interconnected. P3, for instance, spoke of how they used the file *"reference"* feature in Maya [25] while others were still working on those files. Changes applied to such files are immediately reflected on their scene. P14 spoke of the benefits of this approach in motion graphics: *"Sometimes, when I make simple animations, I make a file in [Adobe] Photoshop [9] and then open its layers in [Adobe] After Effects [7]. I can then go back to Photoshop and make changes. They show up immediately in After Effects.*

Finally, participants sometimes switched between applications to remind themselves of the tasks ahead. P2, for instance, talked about how launching a tool (that was very resource-intensive) meant they could take a little break. P9 had a similar take over doing repetitious tasks: *"If I want to turn my mind off, I go to a task that is mainly legwork. I take a break like that."* We also spoke to participants who benefited from multiple software tools to divide their work and segregate their content. P14, for instance, divided the creative and marketing work using two different software while P4 suggested: *"I use each software specific to one application. Like one IDE for Java, etc. This helps me have a division between my work and benefit from knowing when one work ends and the next starts."*

4.7 Discussion and Future Work

One of our key contributions of this paper is in providing an initial taxonomy of why users switch applications for the same task, highlighting key themes of events such as individual preferences or the requirement of collaborative work. Our participants mentioned many reasons why they switched to another tool while completing the same task (Fig. 5.1), such as individual skills, preferences, team constraints, or the nature of the work (e.g., having an important deadline). Although many users expressed that application switches were costly and detrimental to productivity, these users acknowledged that using multiple software tools was integral to how knowledge workers performed tasks.

In fact, a key lesson learned from this study is that **task-centric application switching appears to be here to stay.** This has important implications for researchers and software vendors who need to recognize that users *will* indeed switch between software tools, likely to those outside of a single vendor's suite of applications or even tools on different hardware platforms. If the software is designed with this behavior in mind, perhaps some of the challenges our participants experienced can be alleviated. With this understanding, we have three main takeaways for researchers, application designers, and learning content developers:

4.7.1 Development of Multi-Tool Learning Materials

Many application-switching challenges can be traced back to the difficulties in learning and understanding how multiple tools work together. Learning material focused on using a single product will continue to play an important role. However, given the prevalence of using multiple programs (often from different vendors) to complete a task, we encourage developers and learning content creators to further focus on creating learning materials that show how a piece of software can be used as a part of a larger, diverse, pipeline of tools for completing particular tasks. Such efforts can further enable users to focus on their tasks instead of learning the tools and increase knowledge workers' productivity. In doing so, it is also important to ensure these efforts consider the specific user. To provide customized

training specific to each user, the system can rely on the user's experience, background, and previous training, rather than using generic practices that may only apply to some users.

4.7.2 Enabling Self-Reflection on Switching Behaviors

During this research, we met participants who faced difficulty understanding (let alone navigating) the many software tools they had to utilize to complete a single task. Several individuals expressed how they had to keep notes of tools and their functionality to remember which software they should use for which part of the task. We also came across a participant who had developed a reminder system to inform them of the next software they should use in their work. This can be attributed to the greater number of back-and-forth switches between the different tools that make application switching nearly invisible for most users. As we learned from the product managers, new tool additions are happening organically over periods of time, and users need to be made aware of these. A suggestion for future work is to develop tools that can track and personalize individual users' workflows. Currently, a limited number of tools (e.g., [193, 90]) are able to track different applications and contextualize them for the user. Developing applications that can document individual workflows can enable a degree of self-reflection. We can draw upon previous self-reflection approaches introduced in HCI and visualization research. For example, future work can explore techniques to support presenting each step of the process within a broader context of the entire pipeline to help users understand how different stages relate to one another. The key goal here would be for knowledge workers to not only better understand their tool use but also actively reflect on each of the steps. An example of such self-reflection could be users viewing their processes in deadline-driven settings and assessing their effectiveness.

4.7.3 Application Switching Behavior Analytics

Many software vendors rely on using analytics to understand feature usage and gauge performance. However, this is usually only focused on a single tool. Our findings suggest the need to expand the scope of analytics to include application switching behaviors to develop a more complete picture of how users accomplish tasks. For example, future research can explore tracking the frequency of different types of task-centric application switching based on our taxonomy. This could provide insights into where feature usage is dropping off and being compensated with another tool. This information could be useful for UX teams to improve the usability of such features. It can also help product teams in setting benchmarks, for example, aiming for less than 10% of switches that occur due to user-specific reasons.

4.8 Limitations

Our study relied on self-reports of 15 participants and interviews with five product teams to provide a first exploration in this space. While our taxonomy is based on one interpretation

of the data, it provides a starting point for isolating and tackling specific challenging areas. Once task-centric application switching can be tracked effectively, future research could build on our results to explore task-centric application switching using a data-driven approach with a larger number of users and over a larger period of time. This could also allow researchers to discover how prevalent certain categories of task-centric application switching in our taxonomy are for different tasks and software applications, and in different contexts. We studied task-centric application switching with a broad range of knowledge workers. Future work could expand on this by investigating application switching of individuals in particular domains and capture domain-specific practices.

4.9   Conclusion

Our work explored task-centric application switching, contributing new insights into why knowledge workers deliberately switch applications when completing *a single task*. Our findings demonstrate the processes, benefits, and challenges of application switching devised and exercised across many information domains. As a growing number of knowledge workers use multiple software tools in concert with each other, application switching is here to stay. Partitioning complex software features to support users' learnability is crucial to enhancing overall productivity. Our results also underscore the importance of a more nuanced understanding in HCI of the interoperability among different applications and how knowledge workers leverage both limited-feature standalone tools and larger feature-rich software applications for the same task.

In Chapters 5 and 6 of this dissertation, we explore an approach to address the key issues of using multiple feature-rich software applications by designing a software tool that acts as a minimal end-to-end interface of motion graphics video authoring. Such a tool should be equipped with built-in example to minimize the need for external resources and further empower casual motion designers in their independent authoring. Such casual users might be domain experts, instructors or scientist who otherwise have to resort to design tools not specific to motion graphics to create their videos. The HCI community has the opportunity to play an instrumental role here to begin creating interfaces where creating motion graphics videos would be done with the same ease as slideshows in tools such as Microsoft PowerPoint.

Chapter 5

Katika: An End-to-End System for Authoring Amateur Explainer Motion Graphics Videos

After learning about the complexities of beginning with motion graphics in specific and the difficulties in utilizing and switching among multiple software tools, my third study[1], focused on the design, implementation, and evaluation of an end-to-end software tool of motion graphics authoring. In this chapter, we introduce Katika, an end-to-end system tailored to overcome the challenges faced by motion designers.

The design and implementation of Katika provide an innovative approach to creating motion graphics videos. The system guides users through the process of creating shots based on a script, incorporating artwork and animation from a crowdsourced library, and editing the video using semi-automated transitions. This integrated process streamlines the video creation process and empowers amateur creators with advanced yet manageable motion graphics capabilities. We conducted an observational study to validate Katika's effectiveness and usability, with participants (N=11) expressing the effectiveness of using the system. In just a one-hour session, they were able to produce an explainer motion graphics video.

These findings underscore the potential of systems like Katika to democratize the creation of motion graphics videos and lower the entry barriers that currently hinder amateur creators. Such a shift is crucial, considering the growing role of motion graphics in various domains, from education to marketing and entertainment. We also identify and discuss potential avenues for future research in Human-Computer Interaction (HCI) to further enhance the accessibility and usability of motion graphics authoring tools. In the following sections of this chapter, we elaborate on the design, implementation, and assessment of Katika and how its success could inform future advancements in the field.

---

[1]Portions of this chapter were originally published in Jahanlou, A. and Chilana, P.K., 2022, April. Katika: An End-to-End System for Authoring Amateur Explainer Motion Graphics Videos. In Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems (pp. 1-14). [104]
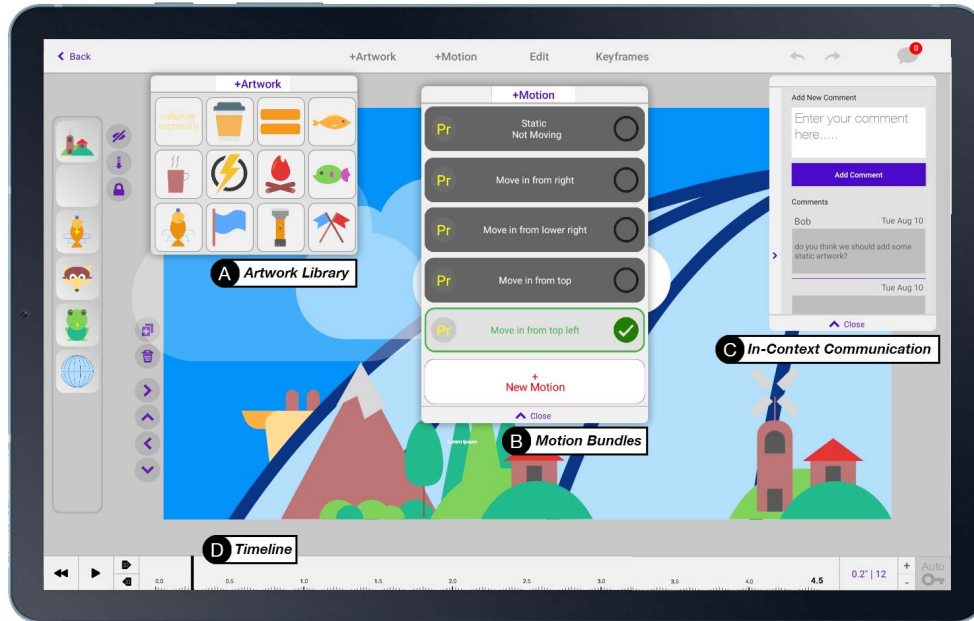
Figure 5.1: Katika is an end-to-end explainer motion graphics video authoring software that allows amateurs to understand and follow the key stages of motion design. This figure shows Katika's graphics editor view that includes: A) an embedded library that suggests artworks based on script , B) example motion bundles that can be directly applied on the artworks or further edited, C) in-context communication features and, D) a timeline for editing the animations.

An interface of a tablet app.

5.1 Introduction

Explainer motion graphics videos are short animated videos that demonstrate complex concepts using a combination of graphical elements, movement, and sound [13, 155, 176]. Unlike static forms of illustrations or infographics, these videos employ engaging artwork that can morph, shift, rotate, and transition in different ways to visually convey a message or a story [38]. Such videos can communicate topics in fields where videography is challenging (or perhaps even impossible) [33], making them popular in domains such as marketing, education, public health, and others [46, 189]. For instance, Figure 2 shows frames from an example explainer motion graphics video about vaccine production. These videos are usually authored by professional motion designers who have formal training in some aspect of animation, graphic design, or video production [28]. However, in recent years, the increasing demand for explainer videos —coupled with readily available online publishing platforms— has inspired more amateurs to tinker with tools of motion graphics video authoring [190]. Unfortunately, amateur motion designers often face a time-consuming and challenging learning curve with the currently available professional motion graphics tools [105].

59

There are several key reasons why motion graphics video authoring is particularly challenging for amateur motion designers. First, creating motion graphics videos involves multiple feature-rich tools across the pre-production to post-production stages that are usually disconnected from each other [179, 105]. Different tasks such as writing the initial script and dividing it into shots, finding the appropriate imagery and artwork, or combining the shots with transitions in a single video, are executed in separate software tools [105]. Moreover, unlike camera-recorded videos where the key content comes from the raw footage, motion designers face a blank canvas and have to populate the content of each frame using imagery or artworks, which can be challenging to make from scratch or locate online [28, 56]. Perhaps the most challenging aspect for amateurs is creating moving graphics that require competencies in animation techniques such as keyframing or interpolation [76, 105]. The combination of such barriers results in an environment in which amateurs get discouraged and even give up on their pursuit of creating explainer motion graphics videos.
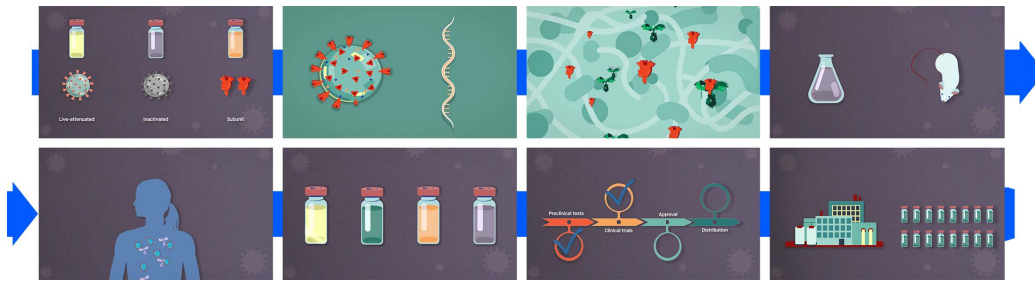
Figure 5.2: Sample frames from an explainer motion graphics video about the production and validation of vaccines. (Retrieved from: `https://www.youtube.com/watch?v=uWGTciX795o`)

<div align="center">Slide images from a Youtube video.</div>

Instead of working with feature-rich motion graphics software, some users may try to explore slide creation tools, such as Microsoft PowerPoint [164], Google Slides [180], or Apple Keynote [17], that have basic animation capabilities. However, the onus is on the user to repurpose such tools to create motion graphics videos and make decisions about how to segment their content into shots, locate relevant artwork externally, figure out if the desired motion effects are even possible, or assemble shots into a video. Moreover, slide creation tools mostly only offer animations in the form of transitions [53] that can only be applied to an artwork as a whole and there is limited room for editing the motions or adjusting timings.

In this paper, we present the design and implementation of *Katika*, a novel end-to-end tool for creating explainer motion graphic videos that bridges the authoring activities of pre-production, production, and post-production and allows amateurs to learn and understand the various steps within each stage. We are using a human-computer interaction (HCI) approach to lower the barriers to entry for amateur motion designers by taking into account

the actual workflows needed to produce a motion graphics video and offering a simplified walk-up-and-use interface. Katika addresses the key design challenge of end-to-end authoring by using a combination of semi-automated approaches (e.g., dividing a script into shots, assembling shots into a cohesive video) and community-based practices (e.g., creating content through examples shared by other users in an embedded library). Furthermore, Katika provides an in-context collaboration feature which makes it easy for amateurs to seek help and feedback at any stage of the process.

Another key challenge that Katika tackles is empowering amateurs to create animated graphics using a novel *motion bundles* approach. Current techniques of animation mainly depend on the notion of keyframing [28]. Such methods, however, are tricky for amateurs as they require an underlying understanding of concepts such as time/space integration, channels, frame rate, or interpolation [122, 176]. It can be particularly difficult for amateurs to anticipate the next state (such as position, shape, color, etc.) and animate the artwork between the current and the new state. With our design and implementation of motion bundles in Katika, motion is treated as an independent entity that can be shared across various artworks. This technique allows the animation to be applied to the artwork, a sub-object of the artwork, or copied from one artwork (or object) to another. Katika's motion bundles allow users to easily browse a library of example motions, apply a motion to their artwork and immediately view the animated artwork without struggling with keyframing or envisioning the possibilities. Moreover, users can also view the beginning and end of a motion on a timeline which provides a visual context for working with and adjusting the animation. Furthermore, users can create, edit, and share their own motions with others to grow Katika's example motion library.

To evaluate the concept of end-to-end explainer video authoring using motion bundles, we ran an observational study with 11 participants with varying degrees of familiarity with motion graphics, explainer videos, and video production. We found that within a one-hour session, all participants (even those without any previous experience) could understand the steps of the authoring process and create an amateur explainer motion graphics video. Furthermore, during the post-task questionnaire and interviews, participants appreciated the freedom that Katika offered for exploring different designs and animations and were keen to re-use the system to produce their own explainer videos for presentations, conferences, and other similar contexts.

In this paper, we make three new contributions:

- Katika as an end-to-end tool for authoring motion graphics videos that facilitates the process of understanding and creating a motion graphics video without requiring external applications. Its minimal interface supports key workflows for creating motion graphics videos and includes built-in guidance for generating shots, selecting artworks based on a script, adding and editing example animations, and producing the final video. Katika also allows users to see what steps they still have to complete, minimizing

the number of decisions that beginners have to make to produce their motion graphics videos.

- A novel approach to end-user animation using modular motion bundles that abstract away keyframing or interpolation and simplify the process of adding motion to artworks with a single click. These motion bundles can be applied to artworks as a whole or their constituent parts. They can be viewed and edited on a corresponding timeline (or curve editor), enabling highly detailed animations while not raising the usability threshold. Furthermore, Katika provides a library of motions that can grow with community contributions and allow for greater creative expression.

- Initial insights from an observational study demonstrating how people could learn the motion graphics design process and create amateur explainer videos in less than an hour using Katika and how they found the tool to be useful and intuitive.

Although recent works in HCI have been drawing attention to the importance of motion graphics and data videos in various informational domains [15, 105, 179], it is challenging to democratize the creation process unless we lower the barriers for amateurs. This paper argues that providing walk-up-and-use interfaces (with embedded content, example motions, built-in communications, and automated features) is one way of supporting amateurs to better understand and directly engage in the authoring process. We envision a future in which users from different backgrounds and skill levels can use such interfaces to create explainer motion graphics videos with ease.

5.2 Related Work

This research builds upon prior work related to amateur video production, the authoring of informational videos, and techniques for simplifying animation authoring.

5.2.1 Innovations in Amateur Video Production

The increasing demand for videos in various domains [35, 59, 73] has encouraged researchers to explore interfaces for improving video productions [178]. Much of the literature focuses on simplifying post-production efforts, such as video editing, where the main content comes from raw camera footage. However, such footage may not always be readily available. To tackle this, innovations have emerged in generating screen recordings [49], supporting home video creation [78], and supporting reuse of content [130]. While such methods focus on the initial video footage, a related challenge is the issue of sequencing the video content [143]. While almost all state-of-the-art tools (such as iMovie [100] or Adobe Premiere [165]) provide advanced editing timelines, amateurs have difficulty dividing their content into separate scenes and finding the appropriate order between them [95, 105]. To address such

challenges, systems have been developed to suggest candidate next scenes [95, 150], assemble a video based on examples [154], and edit recorded narratives [194]. While such approaches are helpful, they usually target only one specific aspect of video production, and it can be difficult for beginners to contextualize the complete pre- to post-production processes.

Another emerging innovation in amateur video production is the automation of content creation. For example, researchers have explored automatic event timelines [41], video creation using recommendations [120], or automating text-to-video content creation [91, 184, 169]. Many of these approaches rely on algorithms such as word concreteness [127] or content analysis [95, 204, 203]. While such systems automate the entire process by mining the text transcript for keywords, some argue that there should still be a balance between automation and user control, especially in the case of authoring informational motion graphics videos [78].

Unlike prior innovations, Katika provides end-to-end authoring for amateurs by bridging pre- to post-production processes within the same user interface. Although Katika relies on some semi-automated techniques to assist users with content creation, the user has full control in exploring and adjusting the suggestions made by the system as they finalize their design decisions. Using walk-up-and-use approaches in Katika, our goal is to enable informational video authoring with the same ease as some of the entertainment authoring tools (e.g., TikTok [192]).

5.2.2 Authoring Informational Videos

One industry standard for authoring informational videos is the use of motion graphics. Recent works [15, 97, 179] illustrate how authoring a motion graphics video is a challenging task, and creating a minute-long video can take up to two days [161]. With more amateurs attempting to author motion graphics videos [28], researchers have been exploring different dimensions such as techniques for motion graphics data videos [14], methods of creating more engaging content [135], or analyzing the space of motion graphics design [179, 190]. Trained individuals usually learn the intricacies of image/graphics editing or software applications [122] and the language and terminology used in such tools [76]. On the other hand, amateur motion designers are intrinsically different from expert motion designers [105] as they are often experts in different domains and expect that using the tools would not compete with their domain knowledge [128]. However, the state-of-the-art tools used in authoring these videos (Adobe After Effects [11], Autodesk Maya [140]) are not easy to use. Over several decades, these tools have evolved to serve high-end productions such as film or advertising [122], resulting in feature-rich applications with laborious learning curves [52]. To make matters worse, a combination of several such tools are often required to complete a motion graphics video [76], and amateurs face a challenge with the lack of integrated authoring environments [105].

Moreover, informational video authoring often begins with blank canvases [166]. Techniques of template-based authoring [122] have been suggested to mitigate such blank canvases and tools such as Animaker [16], Vimeo [197] or Doodly [66] provide users with complete scenes. However, research shows that adjusting such templates to the specifications of informational videos (such as presentation style, imagery, icons, and the overall space) is not always as straightforward [105, 179]. Another challenge is the issue of collaboration [64] and getting appropriate feedback. Despite research into collaborative video authoring in different environments [119, 188], such capabilities are yet to be widely available in the state-of-the-art video editing tools [64, 122].

Beyond tools specific to motion design, some users may tinker with slide creation tools (e.g., PowerPoint or Keynote) to author motion graphics videos. However, to create compelling videos, users have to know how to repurpose these tools for the different pre- to post-production steps and will still need to use external tools to make artworks or export slides into a video. Such tools lack provisions to view the content at the current time, which means that the user cannot know when an animation begins or ends. Moreover, the nature of the animations (e.g., keyframes) cannot be modified in such tools.

In summary, options for creating motion graphics videos are at the two far ends of a spectrum: users have face a steep learning curves with professional motion design tools or struggle with re-purposing slide sharing tools that are easier to access, but have limited animation and end-to-end authoring capabilities. Our approach in Katika has been to design a simplified middle-level interface to ease the steps of making explainer motion graphics videos. It enables a broader class of amateurs to understand and follow the steps of motion graphics design and tell compelling stories using explainer videos.

5.2.3   Techniques for Simplifying the Animation Process

The widespread adoption of animation in fields such as feature film, motion graphics, or video authoring has motivated researchers to investigate ways to simplify its creation. Producing an animation, in essence, is the integration of time and space [122, 176] which is an inherently complicated task. To address such difficulty, K-sketch [62], for example, explores the idea of an animation library. However, this technique is confined to a specific narrative, limiting its application to a broader scope. Moscovich et al. [149] employed hand gestures to record animations that depends on the user's understanding of the timing. Another popular method has been sketching [199], in which software-made frames expand a series of user-generated sketches. Other sketch-based attempts have explored defining the position of objects [63] or movement paths [166, 191]. Such methods have some application for directional (e.g., side to side) movements. Still, they have an evident deficit in the user's ability to draw and actively imagine the upcoming state of the animation.

Transferring motion is another popular approach in animation. Studies have explored transferring a pose from a stick-figure sketch [61] or using video footage [200] to convey a

pose for a character. Kazi et al. [113] explored how an amplified motion can be an approach to generate an animation from a very basic input. These techniques introduce their own limitations, such as understanding the application of basic animation and amplified motions. Other works have also explored animation created from a series of pre-built content [151] tailored to a specific narrative or relational animation [111] to make the dynamics between various objects.

Most of these prior efforts in simplifying the animation process rely on the user's ability to envision the next stage of animation, which is still a difficult task. To relieve amateurs from such burden in creating their motion graphics videos, we introduce a new paradigm of *motion bundles* in Katika. These example-based animations allow users to experiment with predefined motions (e.g., moving an object, morphing a shape) and help produce immediate gratification. Moreover, users can edit these predefined motion bundles, add new ones, and share their motion bundles with a community of users, ensuring the longevity of this example-based technique.

5.3   MOTIVATION AND DESIGN CONSIDERATIONS

Our motivation in this research is to lower the barriers for amateurs to both understand the process of motion design and create their own explainer motion graphics videos. Current state-of-the-art tools are feature-rich motion graphics design software (e.g., After Effects, Maya) that enable high-quality productions, but have a steep learning curve. Although it may be easier for amateurs to get started with slide-creation tools with basic animation capabilities (e.g., PowerPoint, Keynote), such tools can be difficult to repurpose to support all of the different stages of motion graphics design. Amateur motion designers need walk-up-and-use tools that support motion design workflows, minimize cognitive load, and allow amateurs to focus on conveying their informational message [76, 105]. We propose the following five key design considerations (DCs summarized in Table 1) for designing a tool that supports the process of motion graphic design for amateurs:

(DC1) Facilitate End-to-End Authoring: Producing any video consists of a three-stage process of pre-production, production, and post-production [75]. However, state-of-the-art video production tools often have limited or no provision for pre-production [28, 105]. As a result, creators have to rely on ad-hoc solutions to tackle pre-production tasks such as writing a script. Moreover, they need to employ separate feature-rich software applications [105, 122] for different stages of creating a motion graphics video, which increases the learning complexity and the cost of task switching. For example, a designer may write a script in a text editing tool, create artworks in an image editing software (Adobe Photoshop [163] or Illustrator [99] and similar) and assembles the artworks and animation in another application (e.g., After Effects). Integrating all of the key steps of producing an explainer motion graphics video within a single software application can help tackle such challenges.

(DC2) Minimize Complexity and Steep Learning Curves: Each stage of creating an explainer motion graphics video involves complex decisions and steps, such as dividing a script into separate scenes, coordinating color themes, adding content, or editing individual shots into a cohesive video [76, 105, 122]. New users (and even experienced designers) can face an overwhelming learning curve in completing such steps in feature-rich applications [105, 179]. The complexity of these steps can further discourage amateurs from experimentation and trial-and-error approaches for learning [148]. To empower beginners, tools should embrace minimalism and simplify the user interface to lessen the difficulties (e.g., using automated techniques) while still offering control over various steps.

(DC3) Facilitate Access to Example Content: Explainer videos depend on elements such as icons, images, or pictograms [76]. However, amateurs are usually not trained to create such content, and beginning with a blank canvas can negatively affect their confidence in getting started [166]. While some pre-made content is available in online repositories, finding and editing such content is not always easy [105]. To alleviate such challenges and facilitate access to content, it could be helpful to include content (e.g., artwork) within the software tools. The built-in content can be further expanded to ensure longevity and diversity by using crowdsourcing or community-based approaches [50, 124].

(DC4) Lower the Barriers for Animating by Using Examples: Animating content is a tedious task that requires understanding keyframing or interpolation techniques [176, 122]. While beginners have difficulty with such practices, even trained animators need long sessions to create their animations [113]. While trained animators can at least envision the animation, amateurs have difficulty imagining what to make in the first place. One approach to lowering the barriers to animating is using an example-based approach and offering predefined animations. In this technique, users could browse animation examples to learn what they could achieve. The durability of this approach can be expanded by using crowdsourcing or community-based practices for including further animation examples.

(DC5) Foster Easier Communication and Collaboration: When working on a video project, creators often need to communicate with others about the content or alternative designs. However, it can be difficult to use traditional communication methods (e.g., email or text) that are disconnected from their workspace [115, 105]. Amateurs can benefit from context-specific help or feedback but face arduous processes like exporting videos or sharing and seeking feedback on separate platforms. Similar to tools that allow collaborators to communicate directly within the application [50], it can be helpful for amateur motion designers to have such access and eliminate the need for 3rd party applications.

5.4 KATIKA

We designed Katika, a novel approach for the end-to-end authoring of amateur explainer motion graphics videos based on the above design considerations. Katika offers an integrated

Table 5.1: Key considerations for designing a tool that supports the process of motion graphic design for amateurs.

No.	Design Consideration
DC1	Facilitate End-to-End Authoring
DC2	Minimize Complexity and Steep Learning Curves
DC3	Facilitate Access to Example Content
DC4	Lower the Barriers for Animating by Using Examples
DC5	Foster Easier Communication and Collaboration

authoring system for writing scripts and creating shots, browsing an embedded library of built-in and crowdsourced artworks, applying animations through motion bundles, and collaborating in context. Moreover, we included various automated features such as dividing a script into separate shots, suggesting artworks based on a shot script, and editing shots into a cohesive video using transitions. In what follows, we describe the user interface design of Katika, a sample usage scenario, and the key features of Katika that tackle the challenges outlined above.
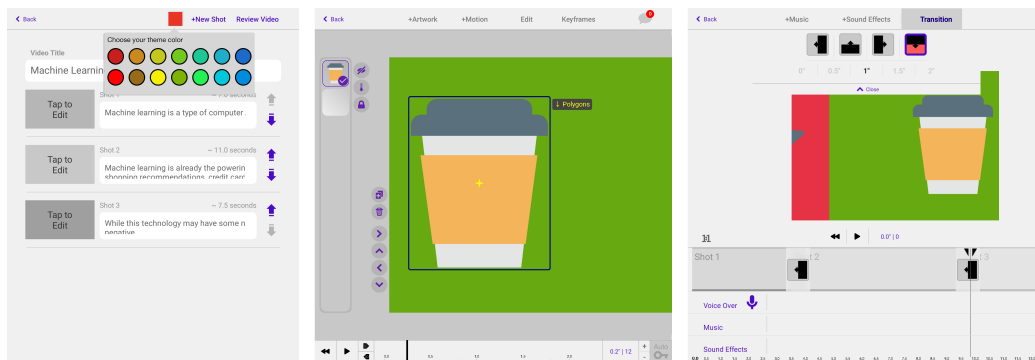
Figure 5.3: atika's main panels consists of A) a scriptwriting editor and shots view where users can enter the story of their videos in text and automatically generate shots (left); B) a main graphics and motion editor that includes various features for adding and editing artwork and motions (middle, shown in Figure 1 in more detail); and, C) the video editor that automatically assembles a timeline of all the shots with different transitions while still allowing the user to make adjustments (right).

Three images showing different screens of Katika.

5.4.1   Overview of Katika's User Interface Design

During the design process of Katika, we continually sought the opinion of several interface designers, HCI specialists, and motion designers. As a result, we opted for lowering the number of the "actionable" views to three (that reflect the industry-standard video authoring of pre-production, production, and post-production [75]). These include: the script editor (Figure 3-left), a graphics and motion editor (Figure 3-middle) and a video editor (Figure 3-right). To describe the workflow of creating an explainer motion graphics video using

Katika, imagine Emily, a journalist who has written a short script about electronic waste and wants to convert it into an explainer video. In a walk-up-and-use fashion, Emily opens Katika, starts a new project and can now enter a script for her video. She can then use Katika's built-in feature to divide the script into separate shots with a single click (Figure 3-left). This shot list contextualizes the project through thumbnails and allows Emily to choose a color theme that helps produce cohesion throughout the video. Next, Emily selects a shot and the canvas (Figure 1-A) provides her a wide array of editing tools and makes it easy for her to add artworks by browsing the embedded library and assembling a collage that statically represents the concept of this shot. Emily then chooses to animate (some of) the artworks. She explores different movements for each artwork by selecting motion bundles from the library (Figure 1-B). Emily repeats this process for all the shots and proceeds to the video editing (Figure 3-middle), where all shots are automatically connected using transitions, and a timeline represents the entirety of the video. In this view, she can also add background music, sound effects or record a voiceover for the video and view the video emerged from all shots. Finally, Emily can choose to view or export the video.

5.4.2 Key Features and Functionality of Katika

Katika addresses the key design challenge of end-to-end authoring by using a combination of semi-automated and community-based approaches. It further innovates on amateur animation techniques by introducing a novel concept of motion bundles and provides in-context communication and collaboration features.

Script Automation and Generation of Shots

With a single click, Katika breaks a text-based script into separate shots with continuous yet independent messages (Figure 3-left). These shots are automatically assigned a length (in seconds) and the timeline (Figure 1-D) adapts itself to their duration. To enable further control, users can adjust the text of each shot, and Katika automatically re-calculates the length of that shot. Moreover, since color plays a significant role in producing consistency, Katika automatically applies a background color to each shot of the video. The user selects the main theme color (Figure 3-left), and the software applies a color (by 15% variation from the central color) to all shots while still allowing the user to override each shot's color independently (DC2). Finally, a video is assembled by adding various shots using transitions. These transitions vary in style and length and are assigned automatically between adjacent shots. The user, however, can choose to adjust their style or length individually (Figure 3-right).

Embedded and Scalable Artwork

Previous research [76, 105, 122] and our own informal analysis of a corpus of more than 50 explainer motion graphics videos on YouTube and Vimeo, suggests that to author a motion

graphics video, access to an extensive repository of editable artworks is necessary (DC1). We embedded a library of artworks in Katika (DC3) based on a freely available repository [137]. Being scalable vector graphics (SVGs), these artworks (Figure 1-A) allow a high degree of editability. Moreover, users can choose to import any of their own SVG files as artworks or benefit from artworks added and shared by a community of users. We further tagged and enabled search for the artworks. Katika automatically runs a search against the shot's keywords and suggests relevant artworks (DC2). These keywords are extracted from the shot's script and, similar to previous systems [127, 95, 204, 203], connect the video content to the script. However, Katika only offers the artworks as recommendations and the user has control over choosing other artworks. This approach is essential as previous works [28, 56] have illustrated that beginning with a blank canvas negatively affects the confidence of video creators. It is worth noting that Katika's approach of examples differs from the templates approach [190]. While template-based authoring relies on adjusting large templates, Katika depends on small modular pieces. Such components can be assembled in different contexts or merged to produce new ones.

While having examples simplifies the authoring process, there are many times when the user would need to copy content from another project. In Katika, any content, such as artwork, polygon, motion, keyframe, shot, or even the whole project, can be cloned in its entirety (DC2). Using this approach, if the user finds something that could help with their project, they can clone it to their own library and adjust it to represent their intention.

Modular Motion Bundles

In our first iteration of Katika, we had followed the standards of adding animation, such as changing time, adding keyframes, and adjusting interpolations. Once we had a stable prototype, we observed four users, who had no experience with motion design, use Katika informally. We found that although the end-to-end nature of the tool was empowering, the process of animating was a significant bottleneck. The difficulty with the animation illustrated two crucial factors: 1) the animation approach should require minimal effort or input from users (DC4); and 2) while beginners face technical challenges with animating, they also find it difficult to imagine how a different state or duration can be used to represent a concept (DC2).

Such observations made us realize that we need a new approach to animating for beginners. Investigating the previous animation methods (such as animation sketching [63], motion amplifiers [114], and motion transfer [200]), we learned that such approaches are either dependent on user input or are challenging to scale to a broader scope. We then reflected on the practices of experienced motion designers [76] that illustrate how they continually watch examples created by others in pursuit of broadening their horizons. Combining this strategy with the requirements of the amateur creators, we invented the concept of motion bundles to enable users to explore animations by examples.

How Motion Bundles Work: We designed our novel approach of motion bundles such that the animation of an artwork (including those of the various hierarchy levels) is consolidated in a single entity (Figure 1-B). Upon request, this bundle can be applied to another artwork or its constituent parts. Using this strategy, users can apply a motion to artwork and animate it accordingly. There is no limitation to the complexity of the animation in this approach, and even advanced morph animations with hundreds of keyframes can be bundled.

A simple example would be animating a bouncing soccer ball and then applying that same motion to a basketball (Figure 4). In such a process, the user still has complete control over the duration (or start and end) of a motion, and can choose to clone a motion bundle into their private library to edit each keyframe or channel in its entirety. While this approach depends on the use of keyframes, it abstracts away the intricate parts of the process. For beginner users, the task is limited to selecting the names of motion bundles until they find one that matches their intention (DC3 and DC4).
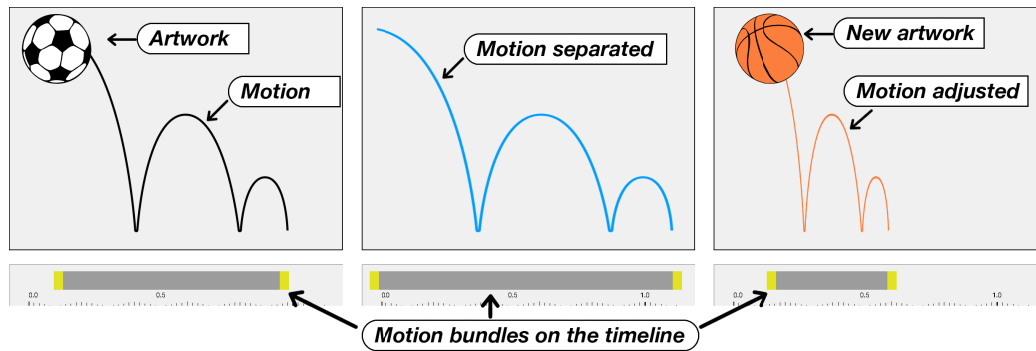
Figure 5.4: This example shows the process of separating the motion from one artwork (the soccer ball) and applying it to another one (the basketball). Users can choose to adjust the motion by simply dragging the motion (or its start/end) on timeline. This way, Katika treats the motion as an independent entity.
Three images side by side. In the left, there is a football with a motion path, in the middle, there is only the motion path and in the last one, there is the motion path applied to a basketball.

Built-in Motion Bundles: In designing the user interface options, we opted for terminology that could be more relatable for beginner users. This was done based on previous studies [5, 55, 105], highlighting the difficulty of understanding domain-specific terminology for beginners (DC1). For instance, instead of "alpha," we used "visibility," and instead of "stroke," we used "border." Below, we provide a list of the motion bundles offered by default in four categories of swipe, pop, travel, and shake:

- Swipe in from left, right, top, or bottom

- Appear by pop at top-left/right, bottom- left/right, middle-left/right, bottom-top/center

- Travel from left-right, right-left, bottom-top, or top-bottom (while choosing the starting point)

- Shake and appear in various locations
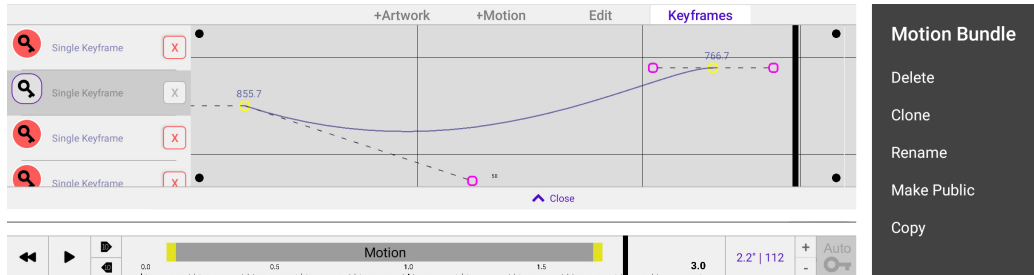
Figure 5.5: Katika's curve editor allows adjustments on keyframes such as changing their time, value and interpolation. Users can isolate various channels and work on them independently. Users also have the option to clone a motion bundle or make it public.

A curve editor where a curve is drawn and knobs allow changing its direction.

Katika's built-in motion bundles also make it possible to apply and merge multiple motions on the same artwork. This approach allows for a quick exploration of different ideas (DC2). Beyond using the built-in motion bundles, advanced users can also create their own motion bundles. To enable such creation, we have included various animate-able channels. Users can choose to make a new motion bundle (Figure 5-Right). Upon adding keyframes or adjusting curves (Figure 5-Left), they can decide to make their motion bundle public, which the system would make available to all users.

Built-in Collaboration and Communication

With DC5, our goal was to foster easier collaboration as sharing video projects is inherently a difficult task [64]. A video project often has numerous shots, artworks, transitions, and animations presented at different parts of the software tool. In Katika, everything is on a cloud server, and effectively all collaborators of a video are working on the same "live" project (DC5). This inclusive synchronized collaboration throughout all the steps is not presently available in any of the state-of-the-art software tools of motion design. Such a collaborative nature is beneficial for beginners who can delegate technically challenging tasks to others. Moreover, Katika allows users to add comments on various object levels such as artworks, motion, or shots and makes such comments discoverable in context (Figure 1-C). Katika provides non-intrusive notifications that inform users of the total number of adjustments or comments to make it easier to see responses or new comments.

5.4.3 Implementation

Katika was developed to work on tablet devices (with Android OS | SDK version 27 and above), which benefit from using various gestures. Doing so furthers the notion of amateur video authoring as many new creators may not have access to powerful desktop computers. We next describe the implementation of Katika's core features and functionality related to script-writing and shot breakdown, embedded artwork, modular motion bundles, and built-in collaboration.

Automatic Script Breakdown Algorithm

To implement the shot breakdown feature of Katika, we reflected on the practices of experienced motion designers [105, 179, 15]. We learned that they assign the beginning and end of shots upon parameters such as the length of the script for each shot, or the presence, complexity, and the number of keyword terms (that would require specific artwork and animation), and the location of various punctuation. We wrote a simple algorithm that first divides the story into pieces based on the placement of the punctuation, such as comma or period. It then assesses the complexity of each segment based on the number of words and their complexity. We included libraries of four word classes of pronouns, prepositions, conjunctions, and interjections to assess complexity as is common in other natural language processing algorithms [86, 42]. The algorithm removes all instances of such word classes and counts the remaining keywords. If the number of words in a shot is 50% lower than the average, the algorithm joins it with the previous or the next one. If the number of the words is 50% higher than the average, the algorithm keeps the shot but prompts the user to consider dividing it into two. Upon generating all of the shots, Katika assigns a length (in seconds) to each shot by evaluating the total number of the words multiplied by an average time to speak a word.

Scalable Artworks Details

The keywords extracted from the script and assigned to each shot, Katika runs an implicit search within the built-in artwork library. Artworks are tagged and the search algorithm matches the script keywords with them to suggest relevant artworks based on the highest similarity scores. To develop this artwork library, we considered several approaches such as image libraries, icons, PNG files with transparent backgrounds, or simple shapes (such as those in tools like PowerPoint or Keynote). After much consideration, we settled on highly customizable vector graphics and developed a parser that converts SVG files into an internal format to enable high editability. The implementation considered matters such as SVG groups and objects, referred to as "artwork" and "polygons." In this approach, the separate pieces of the SVG are represented as polygons (with corners and including body and stroke color/width). In essence, each artwork is a higher-level object that consists of

a combination of one or more polygons. The parser converts a stream of SVG files from a freely available repository (Magicons) into editable artworks and extracts the related tags. Moreover, the implementation trickles down an artwork's transformations (position, scale, rotation, transparency) on its polygons, while each polygon can have independent shape, color, stroke color, or stroke width. Since the artworks are editable, users can modify artworks (Figure 6) and share them back as new ones. Beyond this built-in and growing library, users can also choose to import any of their own SVG files as artwork.
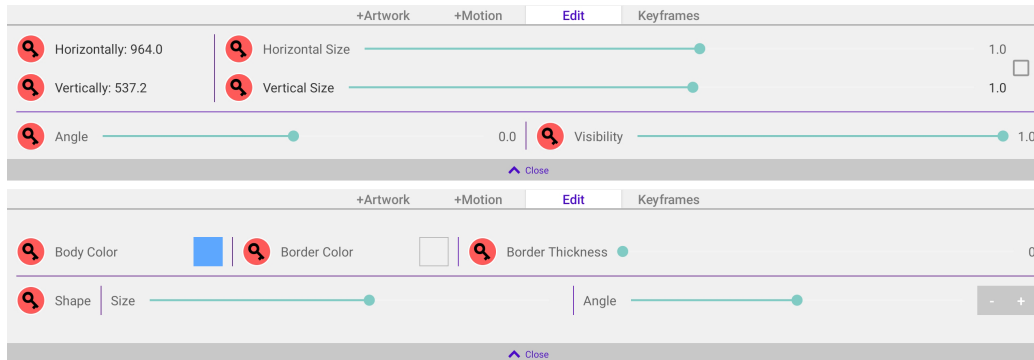
Figure 5.6: Katika offers various options for users to adjust the graphics. Beyond the on-screen adjustment (such as move or rotate), users have access to panels for artworks and polygons to edit specific attributes such as an artwork's scale, rotation or transparency (top), or a polygon's body or stroke color, stroke thickness, size, or rotation (bottom). Both panels also allow users to add keyframes on different attributes.

Attributes editor.

Motion Bundles Technical Implementation

We use the term motion bundle due to the multiple layers of motions in play. In Katika, each object (e.g., an artwork or an underlying polygon) has a "motion" attached to it. Depending on the type of object (artwork or polygon), the motion will include different animate-able channels. For artworks, channels include position (on x and y axes), rotation (on the z-axis), scale (on x and y axes), and transparency (as a float value between 0 and 1). On the other hand, the polygon channels consist of shape (represented by an array of corners and control points each represented on x and y axes), body or stroke color (represented in red, green, blue, and alpha floating values between 0 and 1), and a stroke width (as a float value equal or above to 0). In each channel, users can add as many keyframes, edit the value or the interpolation of the keyframes and position the keyframe against time. As the video plays, in each frame, the motion of the artwork is first calculated. The resulting values (formatted as x and y axes based on the combination of position, rotation, scale, and transparency multiplier) are then applied as a coefficient to all the underlying polygons. After that, each polygon is drawn based on its corners, the color/stroke values, and the parent coefficient. There is no limitation to the complexity of the animation in this approach. Combining

motions on various levels allows for more complex animations such as morph animations with hundreds of keyframes.

Collaboration and Communication Cloud Backend

In most applications, the project file maintains a link to the various artwork on a hard disk and how these files co-relate in the video. In Katika, we use the Firebase [72] technology so that every project, artwork, and motion are stored on the cloud and, as such, eliminate the notion of saving or loading projects. The working file of a project is a JSON file that maintains the relationship between resources. Having everything on a cloud server means that effectively all collaborators are working on the same "live" project (DC5). As several collaborators might work simultaneously, the software implements a first-in-first-out strategy that presents the latest adjustment at all times (similar to technologies such as Google Documents or Google Sheets).

5.5 Evaluation

We ran an observational study to assess how end-users could use Katika to create amateur explainer motion graphics videos. In this initial evaluation, we focused on observing the feasibility of our end-to-end authoring approach and whether users found Katika's features, such as motion bundles, to be helpful and intuitive.

5.5.1 Participants

We recruited 11 participants (6 female) aged 21 to 43 (average: 30.8) through personal and campus mailing lists. Among these participants, three individuals (P2, P3, and P10) had some previous experience with motion design or video production, and eight were entirely new. While we focused on how beginners could learn and use Katika to make an amateur motion graphics video, we included more experienced participants for comparison. We explored how they perceive Katika's more advanced features for creating new artworks or motion bundles.

5.5.2 Procedure

Initial Setup and Training:

We ran the study in person using an Android Tablet (SDK version 27). The study began with an initial demographic survey followed by an explanation of the goals of this study. Before starting the tasks, participants had the opportunity to watch a three-minute training video that illustrated how an explainer motion graphic video gets created. We prepared this video training so that viewers could gain a high-level understanding of concepts such as scripts, shots, artworks, animation, video editing, or transitions. The concepts were demonstrated

at a high level and the software was abstracted away. After the training, we gave our participants five initial tasks to get familiar with Katika's user interface. In these tasks, they had to a) find the final video of a completed project, b) divide a script into separate shots, c) add artworks, d) browse motions for an artwork, and e) respond to comments from a hypothetical colleague. Within the 15 minutes allotted time, all participants managed to finish these tasks.

Video Authoring Task:

After the initial tasks, we invited participants to complete the main task, replicating a 30-seconds public announcement explainer video (Figure 7) inspired by an online video [84]. Producing such a video would require common elements of an explainer motion graphics video, such as multiple shots, using artworks, animations, and editing the final video. This task had 25 minutes allocated to it. Upon its completion, we followed up with a post-task questionnaire by investigating metrics such as the tool's intuitiveness, the collaborative feature's efficiency, and the motion bundles. We then proceeded to the semi-structured interview that probed into the usefulness of the various components and the level of confidence participants felt upon completing the authoring.

Analysis:

Study sessions were audio and screen recorded while we also took notes. We analyzed participants' approaches of interacting with the tool and completing the tasks. We then explored our transcribed audios and notes to gain insights about the usage of the application, synthesize the strengths of the system, and discover difficulties that participants faced. We used an inductive analysis [57] approach to analyze the interview data and explored the themes around our main research question. To ensure the validity of the coded data, the primary author performed the first open coding pass and consulted with the other author to discuss and develop the list of codes. To determine the quality of the generated videos, we devised a matrix of completeness, use of artworks and animation, number of shots, and the overall length of the video. Using this matrix, we then asked an external examiner (who was not a member of the research group) to compare the produced videos against the sample video and provide a score out of 5.

5.6   Results

Our study investigated how users could employ Katika to create an explainer motion graphics video. The external examiner determined that all of our 11 participants successfully completed the main task of the study within the allotted time. During the post-task interviews, some participants reflected on their previous experience and expressed that creating the video in Katika was much faster. P3, for instance, suggested: *"If I had to make this on my*

Figure 5.7: The gallery shown is extracted from the motion graphics videos authored by participants in our usability study of Katika. These participants could utilize the different features and functionality of Katika to put together videos using a built-in library of artworks and animation.

Grid of sample images.

own, I would have to make the shots and artworks by myself. I think it would take me maybe 3 or 4 hours." Furthermore, the independent examiner's assessment reveals that videos produced by participants had a high similarity score compared to the sample video (average of 4.2 out of 5). We next provide insights into the overall experience with the end-to-end authoring software, details of the usefulness of the motion bundles, and the perceptions of Katika in actual practice.

5.6.1 End-to-End Authoring of Explainer Motion Graphics

Overall, integrating all steps of authoring an explainer motion graphics video into one system was encouraging for our participants, with P11 suggesting: *"Having everything in one platform is great. I myself don't like to go to so many apps or websites."* The post-task questionnaire illustrates that all participants found the user interface to be intuitive to some extent (very intuitive: 4/11, intuitive: 6/11, somewhat intuitive: 1), and it helped them in feeling more confident about experimenting with different features. One participant highlighted:

"The UX of this app is great. I could find everything that I wanted real fast. It really simplifies the process. It comes down to making animation now and finishing the video." (P9)

One of the challenges in authoring motion graphics videos is viewing the context of the different parts of the process [76, 105]. Our participants appreciated Katika's central view (where all shots are listed) as having "everything in one place" helped them contextualize the process and learn about what they should do next. P6, for example, commented: *"It [the process] was pretty obvious what I should do. Add the artwork, add motions and view the final video. It was very easy to know where to go next."* To convey the message within shots, our participants opted to use a variety of artworks (Figure 7) and even enjoyed exploring

the built-in library of content. Participants with experience in motion design appreciated the built-in artworks as normally they would have to spend a significant amount of time in finding content when working with state-of-the-art tools:

"[in other tools], it takes much longer. I can't be bothered to look [for artwork] online. Finding them is usually a disaster. It takes a long time and then you have to clean them and import. This [having the artwork built-in] is amazing." (P3)

Moreover, our results indicate that the built-in collaborative process of Katika was empowering for different reasons. For example, P1 compared it to being as easy as using Google Drive's shared documents with in-context commenting. Other participants, such as P10, commented: "*We always prefer to work with tools that offer collaboration. It's difficult to work with people overseas. I think that makes collaboration a major factor.*" We also noticed some interesting and unexpected usage of the collaboration feature. For example, P7 viewed the commenting tool as a mechanism by which they could take their own notes in context and reflect on their progress: *"I think it's useful if you have the chance to review all comments. I have a problem remembering things. I guess I could use it as note-taking for myself."*

5.6.2   Utility of Motion Bundles

One of our key innovations in Katika was our motion bundles feature for adding animations. We observed that participants explored different animations quickly and foresaw various uses for motion bundles. P8, for instance, stated: *"I was thinking about making animations, and I think this [motion bundles] is something that helps me [think] creatively."* Another participant commented on how this tool could also be useful for animating their own characters: *"I've been looking for something like this. I would particularly like to use a [motion bundles] to animate 2D characters. There are a few apps that I use to create characters, and it would be great if I could also animate them." (P3)*

We also observed that some participants were initially hesitant to explore many motions in our warm-up task. They would, for example, choose one of the motion bundles and continue with it for the remainder of the task. However, after attempting the main task of the study and trying different examples, participants organically learned about the low cost of the explorations and were, in fact, intrigued by the idea of examining different motions. Some participants considered every single motion bundle before settling on one that best represented their intended concept. Others even attempted to merge different motions to make advanced animations, such as changing colors and shapes while moving objects around. While participants were not expected to change the duration of a motion, several participants explored that and were encouraged by the simplicity of moving a motion or adjusting its size:

"[An] advantage of the [motion bundle] template is that they are just there and ready to use. I could use [the slider in timeline] to extend the time to make them faster or slower. That

really helps." (P8)

Such results indicate that participants with some experience could also produce more complex animations using Katika (beyond simple click-and-apply). Overall, these results are encouraging as they illustrate the effectiveness of presenting users with predefined animations. For many beginners, such examples mitigate the challenge of envisioning animations and encourage them to think about different movements representing an idea. Others, such as P10, who attempted to create their own motions, commended the streamlined process: *"Adding (new) motion was simple. The curve editor helped. If I needed to, I'd make my own motions, but I think I could use existing ones for most work."*

5.6.3   Perceptions about Using Katika in Practice

During our study, we consistently observed that participants were not intimidated by Katika's various features or functionality and some participants even attempted to push the application features further within the allotted study time. In fact, our questionnaire results show that our participants enjoyed using Katika (6/11 strongly agree, 5/11 agree), and their interview responses highlight how Katika enabled them to explore different ideas. Several participants commented that they wished to re-use the tool once they got used to it for their projects. P11, for instance, commented that: *"I would use this [Katika] to potentially make presentations for my conference. It's great to show visual stuff in presentations."* Another participant (P5) spoke of how they would prefer to watch videos as opposed to read content, and a different participant further elaborated on the value of democratizing motion graphics video authoring:

"I think this is a great tool...especially in today's world, there are so many videos on YouTube...this way, many [people] can make videos on their own." (P3)

Our participants appreciated some of the technical approaches of Katika as well. For instance, P3 enjoyed the cloud-based strategy of saving content as in the past they had trouble with moving files around. Another participant commended Katika's minimalist approach and that even the less-experienced team members would be able to use it effectively:

"I think this is a very straightforward system. I could see how I would use this for making video tutorials. Because I know how to use Adobe, but I have colleagues in marketing and sales that for them it's very difficult. We could add PNGs of our software and just make the tutorial. For that, this would be ideal." (P10)

Our participants had a few suggestions for improvement as well. Most of these ideas were cosmetic (or engineering) suggestions such as colors, size of the windows, or the type of interaction (such as drag and drop instead of click and act). However, we did receive some substantive suggestions for making Katika easier to use in practice. For instance, three participants suggested having a method to preview motion bundles. P2, for example, indicated that decision-making would have been easier with such an approach: *"Motions are great. They can be applied by a simple click. It would be great if I could see a thumbnail*

of them before applying them. It makes it really easy to quickly decide." Other participants proposed that commenting should also be possible at a specific point in time to enable more substantive discussions.

5.7   DISCUSSION

Our paper contributes the design and implementation of Katika, an integrated tool for authoring amateur explainer motion graphics videos. Current practices of creating motion graphics videos are at the two far ends of a spectrum. Amateurs either have to face a steep learning curves with professional feature-rich tools (e.g., After Effects or Maya) or struggle with re-purposing slide sharing tools (e.g, PowerPoint or Keynote) that are easier to access, but have limited animation and end-to-end authoring capabilities. The key novelty of Katika is in providing a bridge for pre- to post-production activities of motion graphic design and in enabling end-user animation using motion bundles.

Informed by actual motion design workflows, Katika provides a simplified, walk-up-and-use interface that helps amateurs understand the different stages of the motion graphic design process. This was inspired by dedicated authoring tools emerging in other design domains, such as user interface design and interactive prototyping. For example, although UI designers initially had to re-purpose and learn feature-rich image editing or slide sharing tools to create their prototypes, the industry standard now is the use of dedicated UI prototyping tools (e.g., Figma [70], Adobe XD [202], Axure [27]). Such tools not only provide a better user experience for designers as they have been informed by actual workflows of UI designers and UX researchers, but they also allow novices to learn the steps and terminologies of UI design. Our vision for Katika is similar in that we believe more designers will be able to learn about the process of motion graphic design and produce amateur motion graphics videos using our simplified end-to-end approach.

The initial results from our study demonstrate that amateurs from different backgrounds could use Katika to successfully create their first explainer motion graphics video within an hour. The responses from our participants also appear to confirm that the use of motion bundles simplified the process of animating and that they even enjoyed using the system. Furthermore, it was encouraging to see that our participants expressed interest in using Katika beyond the study for their own purposes (such as product demonstrations or conference presentations). These results suggest that Katika is complementary to other HCI approaches for simplifying video authoring for non-experts [78], authoring of informational videos [15, 59], and streamlining animation techniques [113, 166]. Finally, our participants were also enthusiastic about our open, community-shared content approach. Our vision is that user contributions for new artworks and motions will enable the creation of more advanced and creative motion graphics videos over time, ensuring the longevity of this author-

ing approach. Next, we reflect on our design and study insights to highlight opportunities for future HCI research.

5.7.1 Towards Complete Storytelling

A key novelty of Katika is that it integrates all steps of the motion graphics design process (breaking down a script into shots, adding graphics and animation, and video editing) into a single end-to-end system. Katika's underlying assumption is that a user would provide a story or a script on their own. While this may be possible for some users from creative fields, it may not necessarily be the case for other amateur creators. Given innovations in storytelling across various domains [73, 155], an exciting future direction would be helping users convert their ideas into stories. For example, we foresee a future in which a series of keywords can be utilized to develop the overall concept and use of semi-automated approaches to generate a corresponding script. This design can conform with our tool's current approaches, providing a starting color theme, shot breakdown, or video transitions. Using such as approach, a scientist, for instance, can generate their own visual stories and push information sharing through explainer videos a leap ahead.

5.7.2 Opportunities for Learning Video Production

A significant challenge in creating motion graphics videos is understanding the role of the smaller pieces in making the final video. During our study, we observed that some of our participants opted to go back and forth between the training video and Katika to better understand the subtleties. Given this behavior, future work can consider learning through reverse engineering as an approach for helping beginners dissect an example video to understand how it was created. Although prior works in learnability have explored techniques such as contextual help [50], on-demand support [47], and expert patterns [118], another direction could be to allow users to dig deeper into the different sections of a video and learn about the role of different building blocks within the video. We believe that having a way to learn by reverse engineering the video content will enhance the learnability of not only motion design, but other types of video productions as well. Moreover, in our initial studies, we learned that for many amateurs, envisioning the next state of animation was difficult or impossible. Using motion bundles, we supported them in browsing a library of animations. However, similar to prior work in example-based programming [32] or 3D design [115], we can use this technique for teaching animation even further as our participants were enthusiastic about using examples. In such an environment, beginners can learn about different (creative) approaches by merely watching and replicating examples on their own.

5.7.3 Quick and Dirty Video Prototyping

Our implementation of Katika is centered around video authoring for amateurs that binds scriptwriting with shot-by-shot breakdown, artwork, animations, collaboration, video transitions, and editing. This approach offers numerous advantages, including lowering the cost of task switching between different tools that can be useful for professionals as well. Our approach is similar to the use of end-to-end and lightweight approaches in design fields such as UI/UX (e.g., XD or Figma), 3D printing [96], or CAD [131] to produce rapid prototypes. Moreover, during our usability study, participants with prior experience in motion design saw the possibility of using Katika as a lightweight tool for quick authoring tasks that required feedback and iteration. Taking inspiration from such work, we envision a future in which users can employ lightweight solutions such as Katika in realtime and leverage repositories of pre-made artworks, motions, and automated features, combined with built-in collaborations to prototype motion graphics videos rapidly. These "quick and dirty" prototypes could perhaps be utilized in environments such as classrooms or even professional meetings.

5.8 LIMITATIONS

This research has had a few limitations, and as such, we put forth our contributions with caution. The design of our tool currently only supports the two-dimensional style of motion graphics. It is worth expanding this approach into other motion graphics design styles. As with any qualitative study, the small sample size warrants further evaluation and future work can directly compare Katika with other software tools used to create motion graphics videos. The main challenge in doing such a comparative study is that the available tools are at either end of the spectrum (such as high-quality productions with After Effects or rudimentary content with tools such as PowerPoint or Keynote) and do not focus on the end-to-end production process as we have in Katika. In addition, our study's main task asked the participants to re-produce an existing video given a particular script and we did not evaluate the cognitive processes of storytelling and scriptwriting. This aspect should be further investigated in real-world deployments or longitudinal studies of Katika.

5.9 CONCLUSION

The interest in creating explainer motion graphics videos is increasing across several domains. Our research contributes the design and evaluation of Katika, an end-to-end software application that supports amateurs in authoring motion graphics videos. Katika offers an embedded library of artworks, a novel approach of motion bundles for animation, and built-in communication features. Moreover, it lowers the complexity of various tasks through automated solutions and uses terminology that is easier to understand for beginners. Our

observational study illustrates how amateurs who had no prior experience with motion graphics could begin making explainer motion graphics videos using Katika within a few minutes.

Overall, our insights provide fodder for compelling opportunities in HCI to lower barriers to entry for amateur motion graphics designers. However, every lab study comes with its own limitations. Among other things, in this study we prescribed the tasks to our participants and they had a specific set time to finish their authoring. To maximize our understanding of the efficacy of the system, we did a longitudinal study (discussed in Chapter 6) that analyzes how users might incorporate such a tool within their own workflows, their specific use cases and the differences of perceptions between casual and professional motion designers.

Chapter 6

End-to-End Motion Graphics Authoring in Practice: Perceptions and Expectations of Casual and Professional Motion Designers

In my fourth study[1], I wanted to further investigate real life practices of both casual and professional motion designers, focusing on how they would incorporate an end-to-end motion design software such as Katika within their workflows. As shown in Chapter 5 users enjoyed creating prescribed videos within a lab study. However, to achieve ecological validity, we wanted to examine the software within users' own contexts and for their own specific tasks. Such in-context evaluation would provide insights into the real-life practices, usage, differences and the nuances of motion graphics authoring among different user groups.

6.1   Introduction

Informational videos that present visual content using motion graphics are gaining popularity in domains such as data visualization [15], education [74, 46, 73], healthcare [117, 176] and finance [189]. However, motion graphics authoring remains an expert-driven process [1, 183, 68], often necessitating skills and knowledge in content creation, animation, and sequence editing [201, 78, 105]. Advanced motion design applications such as Adobe After Effects [11], Autodesk Maya [140], or Maxon Cinema4D [54] offer the capabilities needed to produce high-caliber motion graphics videos. However, these applications introduce an steep learning curve due to their specialized and feature-rich interfaces, especially for casual motion designers [[154]] who have not received prior training in motion design and are

---

[1]Portions of this chapter are currently submitted for peer review in Jahanlou, A. and Chilana, P.K., 2024, April. End-to-End Motion Graphics Authoring in Practice: Perceptions, and Expectations of Casual and Professional Motion Designers. In Submission to the 2024 CHI Conference on Human Factors in Computing Systems.

Figure 6.1: Example frames from the animation made by one of the casual participants in this study. This was perhaps the first time this participant could actually explore making an motion graphics video on their own. For these users, the ability to create animated content independently was a rewarding experience.

exploring motion graphics for a domain-specific need. Such casual designer often resort to creating animated videos using general slide-creation tools like Microsoft PowerPoint [164] or Google Slides [180] that are simpler to use but require extensive modification to meet the requirements of motion graphics video creation [104].

To simplify the motion graphics authoring process, several web-based applications are emerging (e.g., Rive [170], Lumen [136], Animaker [16]) that offer fewer features but streamline the overall video creation process. These tools strive to provide minimal interfaces in which limited features are easier to find and use. Moreover, being on cloud servers, such tools eliminate the need for software installation and maintenance or file storage. Recently, AI-powered tools have also been emerging to further advance content creation using techniques such as automated green screen removal [171], image quality enhancement [123], text to avatar conversion [186], or direct text to video creation [127]. As the industry of motion design tools continues to innovate and grow, the availability of advanced tools and automation will offer an abundance of new authoring opportunities for creators.

Despite the promising innovations in motion graphics, we argue that the technical prowess of new motion design tools alone will not necessarily translate to more efficient workflows [106, 2]. To increase adoption and promote retention, it is important for researchers and industry practitioners to step back and understand how motion design is used in different contexts, how design tools fit within different users' workflows, and how users learn and use such tools to create motion graphics videos on their own. This is particularly important as recent studies (e.g., [105, 12, 15, 28]) show that the end-users of these

tools include both casual and professional motion designers who vary in their technical expertise, backgrounds, learning approaches, and have diverse needs and domain-specific constraints.

In this paper, we investigate how professional and casual motion designers across different domains use an end-to-end motion graphics design tool in their own contexts for one week. We recruited 14 participants and provided them access to Katika [104], an end-to-end motion design tool encompassing scriptwriting, example-based artworks, motion presets, and sequence editing. Our participants used this tool to experiment with different animation tasks at their own pace within their own contexts. Fig. 6.1 represents frames from an animation created by a casual and a professional motion designer. Using questionnaires and follow-up interviews, we aimed to understand how users perceived the authoring process, the types of videos they envisioned to create, and the extent to which such tools could be integrated into different workflows. Additionally, we tried to understand users' perspectives on the future of motion design tools employing automated and novel interaction techniques.

Our key results indicate that end-to-end and minimalist authoring was perceived to be useful by both groups; professionals found value in using an end-to-end tool for "quick-and-dirt" prototyping of more complex videos at earlier stages and appreciated having control over design choices. On the other hand, while casual motion designers found the example-based approach valuable for learning about the different stages of the design process. Interestingly, the casual designers were less interested in having control and wanted to see more automated options to further simplify the authoring process. Our results are consistent with findings from other creative domains [] where the experienced practitioners appreciate incremental improvements and prefer more control, while newcomers are influenced by advances in other domains that have potentially set unrealistic expectations. By illustrating users' experiences in real-life contexts and examining the perception of future use, we discuss several design guidelines for facilitating the development of more efficient and usable motion graphics authoring tools that cater to the diverse needs. Overall, our paper makes the following contributions:

- An in-context evaluation of end-to-end motion design workflows of 14 end-users (casual and professional motion designers), including insights into real-life practices, comparisons with current approaches of authoring, and the nuances of each group;

- A synthesis of users' workflows, challenges, and the perceptions shaped for future solutions and interaction transformed into design implications;

- An analysis of various features and the variation in usage that can help future developers of similar applications measure the need for different features by various user groups.
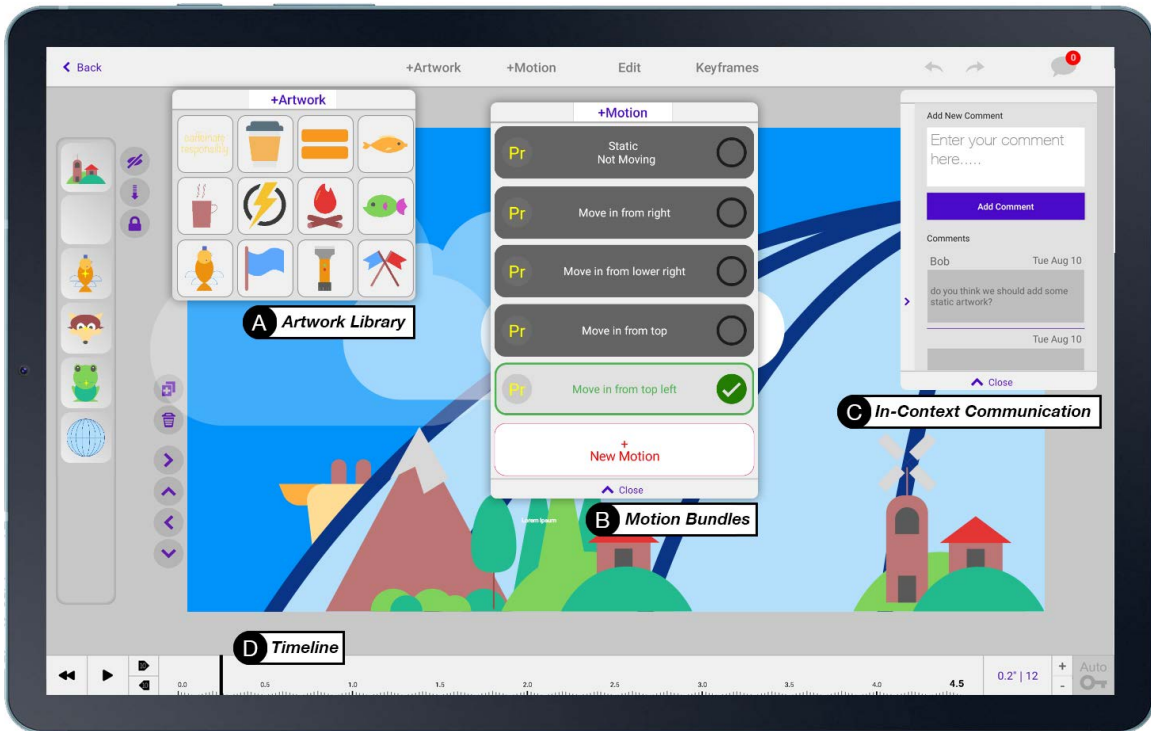
Figure 6.2: For this study, we utilized Katika; an end-to-end example-based research motion graphics video authoring software that includes a library of editable artworks, example animations (motion presets), video sequence editing, animation curve editor and a timeline for editing the animations. It's versatility and research-oriented prototype nature, made it an ideal candidate for this study.

6.2 Related Work

To contextualize our findings, we draw upon literature on the state-of-the-art tools of motion graphics authoring, challenges of authoring motion graphics videos faced by different users, and broader issues of software learnability in creative software domains such as graphics design or video authoring.

6.2.1 State-of-the-art Motion Graphics Video Authoring

Motion graphics video authoring has seen significant advancements in recent years, with various state-of-the-art tools and technologies shaping the industry. Notable motion graphics software such as Adobe After Effects [11], Autodesk Maya [140], or Maxon Cinema4D [54] offer countless capabilities, catering to various user preferences and requirements. Their comprehensive features include 2D and 3D animation, visual effects, text animation, and integration with other software tools. The versatility of these tools allows users to create complex and visually stunning motion graphics for various purposes. In recent years, we have also witnessed emerging tools with low entry barriers targeted at casual motion designers. Web-based motion graphics applications, such as Rive [170], Jitter [109], and Animaker [16], target non-professionals and educators offering drag-and-drop interfaces and template-based workflows. These tools provide a range of features and capabilities, with the added benefit of some being platform-independent and requiring no installation. Recently, there have also been advances in computer graphics authoring using artificial intelligence (AI) and machine learning technologies [147, 182].

While numerous motion graphics tools are available, these tools may not adequately cater to the diverse requirements of varying audiences. For instance, many motion graphics tools are primarily designed for professional use [183], making them less intuitive for casual motion designers who could benefit from more approachable interfaces. Such applications often come with complex features and functionalities, which can be challenging for casual motion designers and non-professionals to master [142].

6.2.2 Challenges Creating Motion Graphics Videos

Creating motion graphics videos involves challenges ranging from technical difficulties to industry-specific and user experience issues [183]. Such challenges can be seen in breaking a long script into separate shots, finding and editing the necessary graphical assets, using keyframing and interpolation techniques to animate assets, editing a sequence of shots into a cohesive video, and finally adding music and converting the projects into appropriate video formats [105]. In this environment, even a short video could take days for a casual motion designers [161] to create. While advanced tools of motion graphics (e.g., Adobe After Effects [11], Autodesk Maya [140], or Maxon Cinema4D [54]) provide features for creating high-caliber videos, there is an entirely different class of casual motion designers who could

benefit from making videos on their own. Moreover, creating a motion graphics video often requires collaboration between multiple individuals, necessitating effective communication and workflow management strategies that currently do not exist as a built-in feature in most tools of motion graphics authoring [183, 105].

Despite the significant advancements in the field, users still face challenges such as steep learning curves associated with advanced tools, integration between different software, and balancing user expectations with learning difficulty. Addressing these difficulties is essential for the continued growth and development of the motion graphics industry and the tools that support it. Understanding the challenges different user groups face, such as casual motion designers, professionals, and those with specific industry needs, is also essential for developing tools and strategies catering to unique requirements.

6.2.3 Software Learnability and Tools of Motion Graphics

Software learnability is a crucial aspect of user experience, particularly for complex tools such as motion graphics authoring software [65]. The design of motion graphics authoring tools often involves balancing simplicity and functionality to cater to diverse user groups, ranging from professional designers to amateurs [183]. This is particularly important as the complexity and feature-rich nature of motion graphics tools can pose significant challenges for casual motion designers [142]. Research has suggested that incorporating user-centered design principles and providing clear documentation and tutorials can improve the learnability of motion graphics software and promote more inclusive design practices [105, 183, 40]. A popular approach in learnability has been example-based learning that allows users to learn from real-world examples in domains such as programming [40], 3D modeling [144], and image editing [87].

Despite the advantages of this method, there needs to be more literature evaluating the efficacy of example-based learning in motion graphics. Investigating its application could provide insights into developing intuitive tools for users with varying levels of expertise.

6.3 Studying End-to-End Motion Graphics Authoring in-Context

Creating motion graphics videos involves pre-production, production, and post-production stages [30]. However, current mainstream video production tools often lack pre-production features [105], and designers have to use separate applications for different creation tasks, increasing the learning complexity and task switching cost [183, 106]. Beyond pre-production, to create even a simple motion graphics video, multiple, often feature-rich software tools have to be employed. In this study, we investigate the real-life practices, perceptions, and expectations of both casual and professional motion designers when they get a chance to work with a simplified tool that enables end-to-end authoring.

6.3.1 Katika: End-to-End Authoring Tool

We carried out an in-context deployment of *Katika* [110], an end-to-end research proto-type that uses semi-automated and example-based techniques for creating motion graphics videos. Built into the tool are features for automatically dividing a script into separate shots, adapting the timeline to each shot's duration, and applying a background color to each shot. Users have control over the color scheme, shot length, and transition style between shots. Additionally, it comes with an embedded library of scalable vector graphics (SVGs) that offers extensive editability, promoting creativity and confidence among users. The user interface is designed with beginner-friendly terminology and users can also clone content from other projects, and the software's approach emphasizes small modular pieces over large templates. Motion presets allow the animation of artwork (e.g., swipe, pop, travel, and shake) to be consolidated into a single entity that can be applied to other artworks or their constituent parts. Advanced users can create and share their own motion bundles, and multiple motions can be applied and merged on the same artwork. This approach abstracts away the complexities of animation, enabling even beginner users to animate with ease and flexibility. The features of Katika (represented in Fig. 6.2) could be divided into nine broad categories as explored in Table. 6.1

Table 6.1: Features of Katika in nine broad categories.

P #	Feature
01	**Script-writing:** allowing users to tackle the pre-production tasks.
02	**Shot breakdown:** automatically dividing a long script into separate shots.
03	**Built-in artworks:** a library of example assets that users could use.
04	**Library of Motion:** presets to apply to different artworks.
05	**New motion bundles:** allowing users to create an entirely new motion preset.
06	**Artwork editing:** enabling users to edit the underlying SVG of an artwork.
07	**Curve editor:** allowing advanced animation and keyframe interpolations.
08	**Video editing:** a sequence editor automatically stitching shots using transitions.
08	**Transitions:** adjustable transitions to make various effects between shots.

We aimed to understand how users integrate and perceive such a tool within their workflows. Moreover, we wanted to understand the type of content that the users will opt for creating and their usage of the various features.

6.3.2 Study Method and Data Collection

We conducted this research over four months in 2023 by allowing users to try out the Katika tool in their own contexts. Gathering data within users' contexts is an effective field method for evaluation that overcomes the limitations of lab-based studies, such as the observer effect [149]. We provided users with the software for a week and encouraged participants to use the

Table 6.2: List of participants, their gender, their group (professional or casual motion designer), the number of session, and the average time they spent per session during the deployment, their occupation, and the features they used when working with the application: a) scriptwriting, b) shot breakdown, c) artworks, d) motion bundles, e) new motion bundles f) artwork editing, g) curve editor, h) video editing, i) transitions.

P#	Gndr	Exp	Occupation	Ses	Mins	Features used	Tasks
01	Male	Pro	Animator	3	126	a,b,c,d,e,f,g,h,i,j	1
02	Male	Casual	Real Estate	2	106	a,b,c,d, g	1 and 2
03	Female	Casual	Office Manager	2	110	a,b,c,d	1 and 2
04	Female	Casual	Researcher	3	35	a,b,c,d, g	2
05	Male	Pro	Motion Designer	4	96	a,b,c,d,e,f,g,h	1
06	Male	Pro	3D Animator	3	90	a,b,c,d,e,f,i	none
07	Male	Pro	Animator	3	129	a,b,c,d,e,f,g,h,i	1 and 2
08	Female	Casual	Educator	4	117	a,b,c,d,f	1 and 2
09	Female	Casual	Visualization	3	104	a,b,c,d	1
10	Female	Casual	Educator	3	89	a,b,c,d	1 and 2
11	Male	Pro	Graphics Artist	3	144	a,b,c,d,e,f,h,i	none
12	Male	Pro	Animation Artist	4	165	a,b,c,d,e,f,g,h,i	1
13	Female	Casual	Real Estate	4	138	a,b,c,d	2
14	Male	Pro	Game Artist	3	141	a,b,c,d,e,f,g,h,i	1 and 2

software in multiple sessions spread across different days. This allowed us to examine their understanding of the tool even after a break and provided flexibility of use to participants who were all busy professionals in their respective fields with busy schedules. On average, participants used to tool for at least thee days, with one week being the maximum. During that time, they reported to have between 2 to 5 sessions of actively tinkering with the application. We asked participants to record the screen for each session while working with the software. Using these recordings and interviews, we collected raw data on the software usage, such as the number of sessions, the length of each session, and the total time spent with the software.

Questionnaire We gathered data using a demographic pre-test questionnaire and a post-task usability questionnaire. The pre-test questionnaire collected basic demographic information and data about participants' experience with motion graphics or video authoring tools. The post-task questionnaire was used to probe further into the areas explored in the interviews, including the participants' experiences and perceptions related to the motion graphics software.

Interviews We finally conducted a follow-up semi-structured interview to delve into participants' previous video authoring experiences, their understanding of the system's underlying model, evaluating the tool's primary objectives, the details of their work with the software such as their experience with using the tool, their approach for authoring a motion graphics video without the prescribed software, the extent to which they used various fea-

tures of the software, their perceptions of the software's UI, and their outlook on the future of tools of motion graphics authoring. To express gratitude for their time, each participant received a 50$giftcard$.

We gathered data using a demographic pre-test questionnaire, two suggested tasks (described below), a post-task usability questionnaire, and a follow-up semi-structured interview. The pre-test questionnaire collected basic demographic information and data about participants' experience with motion graphics or video authoring tools. In our post-task questionnaire and interviews, we probed into participants' previous video authoring experiences, their understanding of the system's underlying model, evaluating the tool's primary objectives, the details of their work with the software such as their experience with using the tool, their approach for authoring a motion graphics video without the prescribed software, the extent to which they used various features of the software, their perceptions of the software's UI and their outlook on the future of tools of motion graphics authoring. To express gratitude for their time, each participant received a $50 gift card.

6.3.3 Participants

Our study aimed to explore users' interactions and integration of software into their workflows and investigate their perceptions of its features, such as creating new artworks or animating an artwork. We recruited 14 participants (9 male, 6 female) from campus mailing lists and snowball sampling. The participants' ages ranged from 27 to 54. Table 6.2 illustrates the participants in this study.

Professional motion designers: Seven individuals had prior experience in motion design (we refer to them as professionals). These individuals worked as computer animation artists, motion designers, or game artists. With the professional participants that we recruited, our goal was to understand how they perceived an end-to-end tool and if they could use such a tool effectively along with their existing applications.

Casual motion designers: The other seven participants that we refer to them as casual participants worked in domains such as real estate, education, visualization, data science or office management. For our casual motion designers, we recruited individuals who shows an interest in utilizing a tool to create training material, advertising content, or videos that their clients, customers, or students could consume. Such users are often somewhat familiar with presenting content and usually well-versed in slide tools such as Microsoft PowerPoint [164] or Google Slides [180].

Such a sample allowed us to evaluate the extent to which end-to-end authoring can be embedded in the everyday workflows of users from the perspectives of both professional motion designers and casual motion designers. Including members from both groups further facilitated the comparison of workflows between users with different levels of expertise.
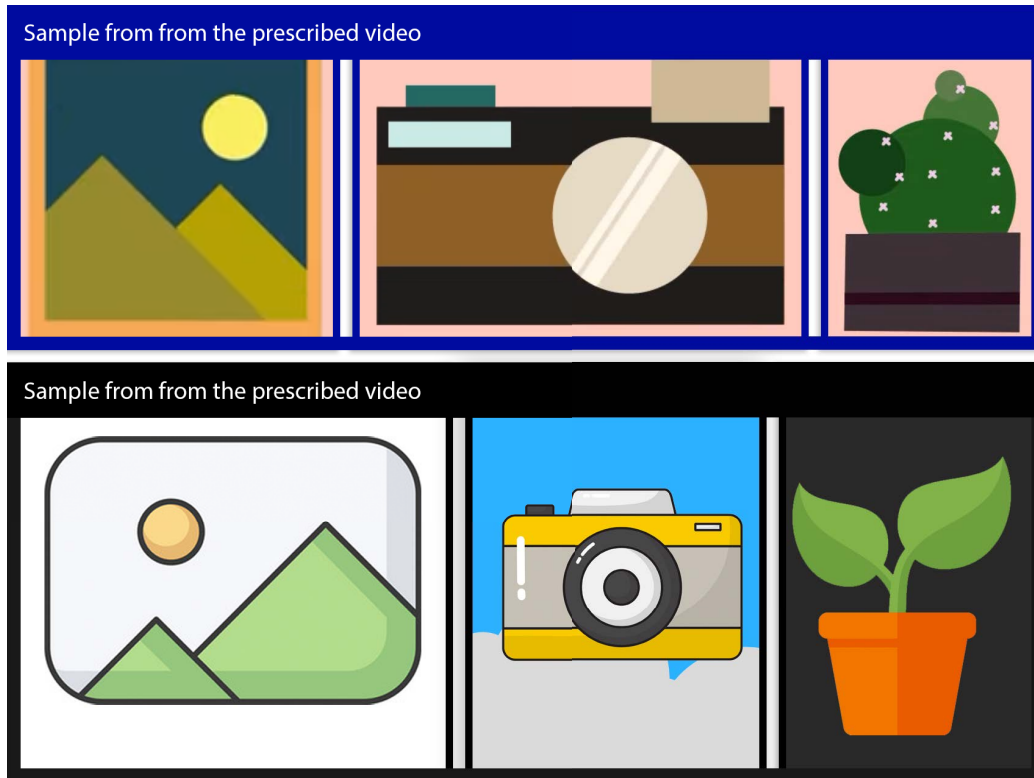
Figure 6.3: Examples images from task 1 that was prescribed to users. Participants were given a sample video to re-produce. Frames on top are taken from the sample video and the frames at the bottom represent their counterparts created by our participants. The majority of participants (11/14) completed this task with varying degrees of success.

6.3.4 Study Tasks

To conduct our study, we assigned two tasks to our participants. The first one (Example-Task) required the participants to reproduce an existing video (Fig. 6.3 above). This task allowed our participants to get acquainted to tool's user interface and was primarily designed to work as a warm-up for participants. By doing so, we made sure that our participants were sufficiently familiar with the software when they began working on the the second task. The second one (Freeform Task) was a motion graphics video authoring where participants could explore the different features of the tool and create a video on their own. In essence, we attempted to provide a degree of on-boarding for our participants during task one, while task two allowed them to explore the application on their own and learn about its efficiency and deficits without any constraints. Our goal for task two was to understand how users might utilize a tool such as this for their own authoring purposes, the features they might benefit fro and their perceptions towards the future of motion graphics authoring. Sample images of the videos made by our participants are provided in Fig. 6.5.

6.3.5 Analysis

Our research involved an examination of data collected from a multitude of sources: pre-test questionnaires designed to capture the profile of our test users, recorded screens for the visual representation of user-tool interaction, observations from user interactions, post-task usability questionnaires to capture immediate impressions, and follow-up interviews. In our analysis, our goal was to synthesize the insights from all of this data together to understand the incorporation of the tool within users' workflows, the applicability of the tool for quick-and-dirty authoring, the perceptions of use and the different practices observed in professional and casual participants.

To analyze our data, we initially synthesized the number of sessions users undertook and their time on each task. This helped us understand the overall usage pattern, time and also provided insights about the intuitiveness of the tool. Once we had this initial analysis in place, we employed a thematic analysis [126] approach and began by going through the interview transcripts, followed by coding relevant excerpts relevant to our research questions. The last step in our thematic analysis was to find patterns from the coded excerpts. The primary investigator identified recurring responses, grouped them, and analyzed them in light of the users' overall experiences, perceptions, and expectations of a motion graphics authoring tool. The second author reviewed and revised these themes and confirmed them against the collected data. This process revealed insights into areas where the the end-to-end authoring approach worked well and where there were tensions among the practices of different types of users.

Unused Quotes about features:

One notable area for our participants was the simplicity of the user interface. Casual motion designers, for instance, spoke highly of the simple interface of the software and how it was different than what they had experienced previously: *"I think the best thing here is that it's simple. I was initially scared, because I once tried [to use] Photoshop and it was very confusing. This is not like that. It's simple to learn each step." (P13)* These results speak to the interest of both groups of users (professionals and casuals) for interfaces where they can get their content created quickly and without so much cognitive expense.

"For the second task, I tried to put together a short clip with two shots. I quickly added some artworks and for a few of them I added [a] motion [bundle]. That's enough for me. It captures what I need for this video. It gives a very early understanding of what the final video will look like" P12

"For my work, it's very important to tell clients about the process. I was thinking that I could use this tool to make a tutorial. For clients to view every step" P13

"I tried pretty much everything. However, the one thing that stood out to me was the built-in library. I think many tools are going to that direction. However, this is editable which is new to me" P11

6.4 Findings

In introducing our findings, we first present data on how participants used the application and the type of content they attempted to create. We then turn to the variations in authoring workflows between professional and casual motion designers. Finally, we provide insights into participants' perceptions of control and flexibility.

6.4.1 Application Usage and the Type of Content

To analyze the application usage and content creation, we examined factors such as the users' number of sessions and the computed average time per session. Our participants were given up to a week to work on the tool, and, on average, they used the software for 3.2 sessions and spent 36.2 minutes in each session. The total logged time was 1589 minutes and the total number of sessions were 44 sessions among 14 participants. Table 6.2 summarizes the usage details of the participants.

To make any motion graphics videos, the user must have used script-writing, shot breakdown, artworks and motion bundles. Professional users and at least two of the casual participants reported that they attempted creating new motion bundles or experimented with the animation curve editor. While all professional participants explored how they could edit an artwork, only one casual motion designer reported of the same. Finally, changing the transitions between different shots was a feature that was only explored by four of the professional motion designers. While we encouraged our participants to spend as much time on the software as possible, we understood that all of them were professionals in various domains and were taking out time of their busy schedules to experiment with the tool and had varying degrees of task completions in this study:

Task Completion: During the study, nearly half of professional motion designers (3/7) reported that they skipped Freeform-Task to explore the software instead. On the other hand, all casual motion designers (7/7) said they invested their time working on and finishing both tasks. As a result, casual participants usually had more success completing the tasks than professional motion designers. For instance, six of the casual motion designers finished the Example-Task and five of them finished the Freeform-Task. At the same time, four of the professional motion designers finished Example-Task and only two of them completed the Freeform-Task. Table. 6.2 represents the task completion by different participants. While the casual motion designer enjoyed being able to create videos independently, professional motion designers often reported that most of their time went into investigating the tool's capabilities and gauging its potential for different scenarios (Fig. 6.4 represents some of these differences.). Professionals often had the ability to foresee future challenges or opportunities and wanted to assess the software's capabilities for their specific requirements such as asset generation, video editing, or pre-production: *"Yeah, I didn't really finish the videos. [I]*

basically tried to understand what it could do for me. I think it can be used for pre-viz or for storyboarding or other things." (P5)

Types of Content Created by Professionals: Among professional motion designers who attempted the Freeform-Task, they either opted for working on a very small piece (such as animating a piece of UI in the case of P14) or attempted to put together a very coarse version of a video that they would later create using more sophisticated applications. Overall, the professionals focused on specific elements or drafts that would be enhanced later (P11, P7).

Types of Content Created by Casuals: Our casual motion designers, however, worked on a variety of different ideas. P2, a casual motion designer who was a realtor, tried to create a video describing the steps of preparing a property for a viewing. P3, an office manager, tried to create the initial steps of on-boarding a new employee to their workplace. Casual motion designers viewed the software as a tool to create final pixel videos for various purposes (e.g., employee onboarding for P3, teaching content for P8 and P10, or making a logo animation for P13). One unforeseen finding of the software was that some participants found value in the library of artworks for static imagery. P13, for instance, was drawn to the sheer number of assets and had tried to use them as a tool to create a logo for their company: *"Animation is one part of it. But I think for me [it] was useful because I could create a logo. I have been looking for something like this [tool] before and never found a tool that is so simple.""*

6.4.2   Variation in the Practice of End-to-end Authoring

As prior works have highlighted differences in their motivations and skill levels [105] of casual and professional motion designers, we focused on observing how different users integrated a new end-to-end authoring tool within their workflows and how they perceived its capabilities. We provide an overview of the themes that emerged from observing the authoring practices of casual and professional designers.

Perceptions of Example-Based Authoring and Bult-in Content

Our participants expressed an appreciation for how the built-in examples (Fig. 6.2 A) helped them convert their shots from written text to visual graphics using adaptable and customizable content. Compared to searching for content on external libraries or from online resources, participants overwhelmingly (14/14) found that this approach significantly facilitated a faster process, contributing to the overall usability of the tool. P3, for instance, who worked as an office manager and attempted to create an onboarding animation, explained: *"If I were to create a video, I would probably use PowerPoint, but then one issue is that I would have to search for all the images myself. Here [in Katika], everything is ready for me."*

Participants further applauded the editability of the example content available within the tool such as the editable SVG graphics that simplified graphics editing tasks. They expressed that this adaptability to alter and adjust pre-existing assets streamlined the creative journey, translating to saved time and reduced effort. P4, who was working as a researcher and tried to create a graph animation, emphasized this aspect, noting: *"the prefabs [artworks] that were there gave me the option to disassemble and work with them in a more creative way. I haven't seen that in any other tool. Even when adding animation, that definitely helped save a lot of time."*

Similar to casual motion designers, the professional motion designers also found the built-in library to be a useful and efficient feature: *"I am putting together some icons together as we speak. If these [examples] weren't here, I would have to search for them. This [built-in library] really saves a lot of time. It's also very handy and easy to use." (P6)* Beyond being able to create content more quickly, professional motion designers also spoke of how having the editable built-in examples meant that they could invest their time on more creative aspects of the work: *"Usually when [we] create videos, we have to create loads of assets. Even at the early stages. Here, everything is in-built...which allows me to focus on more important things." (P1)*

Another interesting area was the implementation of shot breakdown within the software. This feature provides the users with a series of initial examples for how they could break long scripts into independent shots. While both groups of users benefited from this features, professional motion designers found it particularly helpful: *"I actually really liked the scriptwriting. I could copy a script to it; it made the shots and assembled them. This is something we have to do ourselves manually....which is in-built here." (P5)*

The key insight here is that both groups found it useful to have access to a simplified example-based authoring features for different reasons. Although each group was interested in a different level of quality, they yet enjoyed having access to tools that can simplify their creation process.

Efficiency of Animating without Keyframing

Another prominent aspect of the end-to-end authoring tool used in this study was the motion presets (Fig. 6.2 B). Animating is perhaps the most difficult aspect of authoring any motion graphics videos. Animation is theoretically the integration of time and space, while in practice, it requires a thorough understanding of topics such as time, frame, keyframes, interpolation, direction, and similar. While professional motion designers have received training (or gathered sufficient experience) in understanding these concepts, casual motion designers often lack any understanding of such topics

Beyond the built-in motion presets, the tool also offered features of keyframing, interpolation and curve editing. Our analysis reveals that some casual motion designers spent time learning about the keyframing process and how they could manipulate keyframes using the

curve editor: *"I tried the curves. It's a little confusing at first, but also it seems like I could make anything that comes to mind. If I had more time, I would play around with it more."* *(P3)* Other casual participants (such as P4 or P9) expressed that they tinkered with animation (keyframing) with varying degrees of success: *"I made a new motion [bundle]. First time it took like 20 minutes to understand how I make something move…Then, when I tried again, I could make a color change as well." (P9)* For this population, finding the right balance between the complexities of keyframing and the simplification that the software provides was a key criterion.

On the other hand, professional motion designers had access to and the know-how of using more advanced tools for keyframing and curve editing and were not interested in editing animations with this tool. Professional motion designers, such as P1, noted this view by suggesting that they were only interested in the most basics that allowed them to create their initial animations and would not encourage the expansion of the software: *"For me, beyond the basic animations, everything else would be useless. Because why would I waste time animating here if I could do it in After Effects?"* For this population, the basic animations in the motion presets were usually more than enough to create their videos and discouraged expansions beyond the core capabilities. This aspect speaks to the notion of *"quick-and-dirty"* authoring tools for professional users.

Video Prototyping and Pre-visualization

Our analysis reveals that for professional motion designers, it was crucial to invest their time in crafting a compelling story instead of creating detailed initial content. This early-stage content, often referred to as animatics or pre-visualization, is almost entirely discarded by the end of the production and only serves as a guide to the production. Although our professional motion designers were equipped with tools of advanced motion design, it was interesting to see them find value in the simplicity that the end-to-end authoring approach offered. For example, participants such as P6, could see the value of such a tool in creating their quick-and-dirty pre-visualizations quickly and how it could help unify their style: *"I guess for animatics, it would be great. Right now, we make [it] usually with a combination of hand-drawn sketches and some colored moodboards. This [Katika] could be a useful tool to make sure that we have some consistency in the animatics."*

These professionals also appreciated how the built-in features supported them in creating their animatics rapidly: *"The [motion] presets are great. I think they need to be limited to 10 or maybe 20. Then I can easily find something that roughly tells what I have in mind. I think that's the key, finding something easily and without having to search." (P11)* For these participants, a library of preset animations was a feature that could help them capture their intent quickly. This way, they could get a rough representation of their ideas right at the beginning. Some professional motion designers, took the prototyping to a step further and viewed the end-to-end tool as a choice for managing the entire production: *"I feel like this*

Figure 6.4: Example frames from the wokrs of a casual participant (top) and a professional one (bottom). As it can be seen, casual participants attempted to more accurately capture their design intent, while professionals mainly explored different aspects of the software, examples and motion presets.

could work as my project management tool. I could have the script here, I could see the breakdown of shots, I could put together some very basic art and motion, and if I need to, I will edit them easily. It can work like a very interesting guideline for what we are doing." (P1) The professional motion designers were only keen in the most basic of the feature in the software and found the limited tool would, in fact, be more effective to tackle their requirements: *"I tried to make a little character, it's basically an avatar. It's not the shape I had in mind, but it's fairly easy to animate it here to see how it looks. [A] more complex one, I can do it later." (P14)*

Casual motion designers, for the most part, skipped the concepts of storytelling and creativity in the interest of authoring animations and visual content. This can be attributed to the fact that they had, perhaps for the first time, an ability to create motion graphics and were excited to explore how they could make things animated on their own. These participants —who were less familiar with matters such as animatics—, were instead advocating for the expansion of the software into new capabilities (such as augmented reality (AR) by P2, AI by P3, or similar). P2, for instance suggested: *"Something I was looking for was if I could add some information over the camera. This way, I could ask my clients to view a property and know its details." (P2)*

Learning Motion Graphics by Doing

By going through the end-to-end authoring tool, our casual participants could learn the steps required to author motion graphics videos. These participants, were pleasantly surprised to learn that within a few short sessions, they had learned the basics of motion design and could, in fact, create some animated videos for the first time. The end-to-end nature of

the tool was a significant factor as previous studies have shown that the need for multiple software tools is often a deterrent to learnability for newcomers [105, 106]. A casual participant, for instance, expressed: *"I actually thought that I'd need to do a lot more on my own. But I was surprised that everything was there. Like when I saw the first task, I was quite scared that I wouldn't be able to make it. But then I saw after like an hour I had made it. Something I had never experienced before." (P9)*

Our analysis further reveals that by removing several steps and tools involved in motion graphics design, the end-to-end approach has helped participants streamline the authoring process (pre to post production) and helped them with learning the steps of authoring easier. P6, who was a professional 3D animator, for instance, spoke of how there are many tools used in their domain and a desire for simplicity: *"...If I could use this like Frame.IO to directly comment and talk to others, it would be great. I am, in general, for any tool that cuts some of the side tasks and helps me focus on the main work."* Such an approach means that casual motion designers could learn the process *by doing* and could explore how far they could get in creating the videos.

It was also interesting to see how the casual participants felt empowered to create motion graphics without relying on external support, allowing them to better capture and express their intent. These participants viewed their own ability to author (or at least understand the authoring process) as a method to capture their intent better. P2, who worked in real estate, for instance, spoke of how they think they could benefit from being involved in the authoring: *"A lot of times, I have to talk to the marketing person, and I have to go over the details. Here, if I could make things myself, then those specific things that I wanted were also there. Sometimes, they get lost when I work with others."*

We encountered another interesting example of learnability among the casual participants as well. P4, who worked as a researcher, spoke of how this approach had allowed them to *learn how to think* about the motion design process: *"I actually learned a little about making animation as well. I think, for example, that initial screen for writing the script; I might have a story in mind, but I might not think of it that specifically. That screen helped me. I had never thought of that before."*

These results speak to the need for re-evaluating the learning approaches for tools of motion graphics authoring and considering features (such as end-to-end authoring) to support newcomer to motion graphics video creation.

6.4.3 Expectations for Automation, Control, and Creativity

Throughout our interviews, we learned about different views among professional and casual motion designers about their expectations for how much control vs. automation an application should offer. Professional designers expressed preferences for specific tools and techniques, which could influence their willingness to adopt new technologies. On the other hand, casual motion designers were more open to exploring new tools and techniques out of
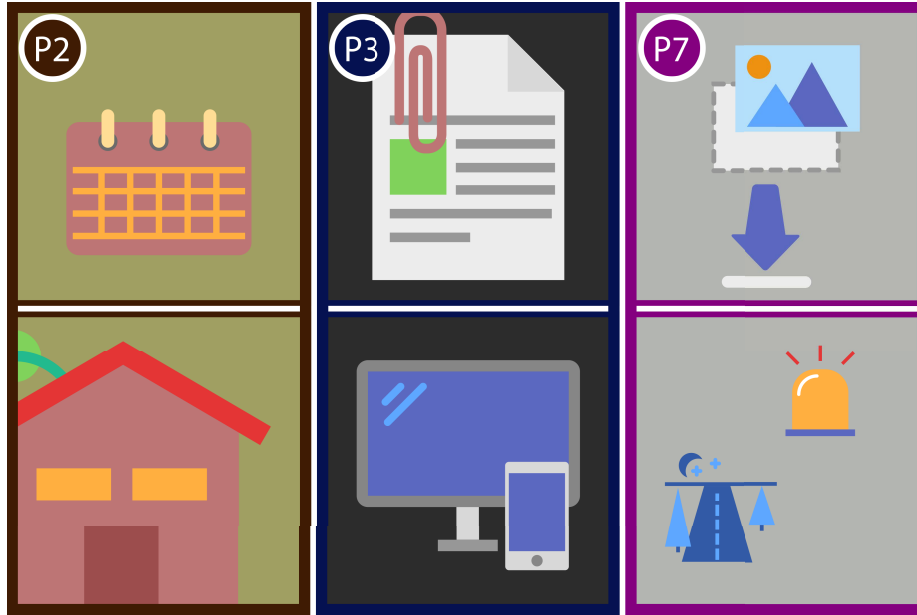
Figure 6.5: Example frames from different topics that participants attempted to create for their Freeform-Task. P2 attempted to create a video exploring "how to prepare a property for viewing", P3 explored the creation of an "on-boarding video for a new employee", and P7 attempted to create an "animatics videos".

curiosity. However, such curiosity came with the cost of being more susceptible to influence from external resources. This population would, for instance, begin working with a software merely based on the word of mouth or being exposed to advertising about it. One unforeseen aspect of this was observed when casual motion designers seemed to look over some of the novelties of the tools and expect even more advanced functionality. P4, for instance, explained how he had higher expectations from the tool: *"When I went into the application, I expected to see a lot of the work done based on my text [script] with the imagery. I like to see this app go in that direction where I can just type the text into it, and it would make the animation."* In fact, many of our casual participants compared opportunities available in emerging AI tools and expressed excitement about expressed prompt-based authoring as the future of motion graphics. This realization suggests that widely recognized tools (such as ChatGPT [159]) set a benchmark for user expectations, potentially influencing those new to a field. For instance, P2, immediately compared this process to those of new AI approaches: *"When you first told me about this software, I thought it'd be an easy interface. I thought it was like GPT that I talk to it, and then it does things for me."*

Professional motion designers, on the hand, were more cautious about the automation process. P12, for instance, highlighted: *"I'm being cautiously optimistic. There are so many changes coming. So much is being automated. You click and a whole image appears. But if you think about it, it's not my creation. Someone else has done the work."* The extent to which they desired such control was affected by their background, skill levels, and general perceptions human creativity. For example, P5 explained: *"I want to be able to create the*

scene the way I want with a starting point, without having to do everything by myself. [The] tool should help me get started and give me the control to adjust." We found this to be a recurring theme among professional motion designers who wished to use the tool as a means to augment their work. This perspective can be attributed to the fact that professional motion designers possess a better understanding and expertise in working with such tools, enabling them to fully grasp the significance of creative input. Casual motion designers, conversely, appeared to view this feature as an expected standard rather than an innovative addition and, consequently, were less impressed by it.

While professional motion designers recognized the benefits of advantages of augmenting their technical skills with software automation, they were yet foreseeing a future in which new technologies allow authoring through entirely new interactions: *"I think we are going toward a situation where knowing the technical things is less and less important. It's more important to be able to be creative and make better videos. In a few years, the tools will be so simple that anyone can use them, then it's more important to be creative." (P5)* For these motion designers, the distinction between technical and creative expertise was the deciding factor for how they viewed their work and practice: *"I have two experiences; one is knowing how to use, say, After Effects. I think that is less important. Another experience is, for instance, knowing what colors work better together. Of course, that will be very important. Or, for someone else, it might be how to tell an interesting story. I think these will be more important than knowing the tools." (P6)*

Another concern raised by professional motion designers was over automation limits. Participants, such as P5, voiced concerns over whether the actions done by the software could infringe on individual creativity: *"There needs to be a balance between what the tool makes and how much creative [input] we get from the user."* This view was echoed by another professional motion designer who advocated for human oversight over the creative process: *"Even if AI becomes more widespread, my role would be the same. It'll be about using newer and better tools. Someone has to be there to control the storyline and make sure he gets the right output from AI. I think that is a job for people." (P7)*

6.4.4 Summary of Key Takeaways

Our findings highlight the need for clearly communicating the capabilities of software tools, providing sufficient learning support and human oversight in tools of motion design. To that end, our key takeaways are as follows:

1. **End-to-end authoring** enabled professional participants to create quick versions of their artifacts quickly whereas the casual participants found it beneficial to learn the process of creating a motion graphics video and where able to capture their intents more thoroughly by authoring themselves as opposed to asking 3rd party content creators.

2. **Animations was an area of tension** between casual and professional motion designers. While professionals only expected the bare minimum, casual motion designers, expected to have more advanced controls by abstracting complexities.

3. **Professional motion designers** were hesitant in adding automation technique as they were unsure how it might affect their creativity. Casual motion designers, however, expected more automation and simplifications. This highlights the need to cater and communicate software tools to different audiences with diverse needs and perspectives.

4. **The balance between automation and human oversight** is a demanding front. Professional motion designers prefer specific tools and techniques, while casual motion designers are more open to exploring novel tools. Both groups, however, agreed on having human oversight on content. The degree of sigh control is yet to be determined.

These findings underscore the importance of considering users' contexts, and providing sufficient control while maintaining human oversight in the motion graphics authoring process.

6.5 Discussion

This research explored how professional and casual motion designers use an example-based, end-to-end application of motion graphics authoring within their own contexts. The ongoing tension between automation and control in motion graphics authoring tools is an intriguing dynamic that will likely continue as the technology evolves. There currently is a race towards making new and automated tools in the industry. However, before implementing such applications, we need to understand the role and expectations of the increasingly diverse user base. As these approaches gather pace, they are set to impact future software development by reshaping users' mental models and expectations. The challenge of reconciling human control with the potentials offered by software presents new areas ripe for further exploration and study. Our results contribute initial insights into real-life practices of both professional and casual motion designers, analysis of the features by usage and the expectations of various user groups when creating withing their own contexts. Conducting the studies within user's context [149] provides an opportunity to overcome the limits of observation in the lab and allows users to more freely explore the software tools for their own specific needs. Below we highlight a summary of our key takeaways from the study and synthesize design implications that can contribute to developing more efficient, and usable motion graphics authoring tools that fit within a range of casual to professional design workflows.

6.5.1 Finding a Balance Between Automation and User Control

A key finding from our study is the importance of striking a balance between automation and human control in motion graphics authoring tools. The advantages of automation, including enhanced productivity and streamlined workflows, are deemed to be significant by most users. However, it is equally important to ensure that users, especially professionals, can retain ample control over the creative process to accurately depict their envisioned intent. Striking such balance is intrinsically complicated, given the vast potential for variance among individual users, the diversity of content types, and the unique contexts they each bring. A casual motion designer prefers a different level of automation compared to a seasoned professional. For instance, creating a simple animation by a casual motion designers might require tasks such as keyframing or content selection to be automated. However, for a professional user attempting to craft an elaborate story, the needs for automation might be in optimization, color calibrations, or dynamically executing repetitive tasks. As a result, the level of experience can significantly influence the desired balance between automation and user control.

To more accurately understand and establish the ideal balance between automation and user control, we propose further research. We aim to delve into a more granular analysis of a range of tasks, such as pre-loading content, the various aspects of animation, complex content editing, and the creative nuances of storytelling. Exploring these areas promises to deliver valuable insights into precisely which tasks should be user-centric and what should be assigned to be managed by software applications. Furthermore, a detailed examination of the needs, preferences, and expectations of users from myriad backgrounds and experiences will provide a more nuanced understanding of the user-in-the-loop concept. This, in turn, can guide us in designing future tools that are not only more efficient and user-friendly but also ensure the valuable touch of human creativity is not lost in the process.

6.5.2 Innovating in Example-based and Vector and Modular Graphics

As we learned in our study, editable assets are vital in empowering creators to maintain control over their narratives while working with motion graphics. Our findings indicate that users, particularly professionals, expect more control over the authoring process. Current generative methods often focus on pixel-based graphics, which may limit the level of control and flexibility that creators require to fine-tune their work. As a result, there is a growing need for developing new technologies that can expand the generative nature of motion graphics tools beyond pixel graphics. Few recent works have begun looking into this domain to convert pixel images into vector graphics or create vector graphics by prompts [132, 168]. Vector and modular graphics offer several advantages over pixel-based graphics, including improved scalability, resolution independence, and easier manipulation [43]. These benefits make vector graphics more suitable for motion graphics authoring, allowing creators to

maintain high-quality visuals while adjusting their work. As such, this is a fruitful avenue for future research to explore the development of generative tools based on vector and modular graphics. A suggestion in that direction is to explore machine learning models to analyze vector graphics data that facilitate seamless interaction between creators and editable generated content.

6.5.3 Simplifying Keyframing Using Animation Abstraction

During this research, we learned that users (both professional and casual motion designers) tend to enjoy tools that limited features and simple user interfaces. In our work, we provided users with access to a tool with built-in libraries of preset animations. However, modifying such animation would still require skills in time-space integration; a skill that most casual motion designers lack. One suggestion to remedy this situation –inspired by participants continuous reference to tools with large language models (such as ChatGPT)– is to train language models to understand the nuances of animation and subsequently further simplify the authoring process for casual motion designers. A rudimentary example of such system would be a model that can synthesize the textual vector graphics content and the animation data to infer the animation in a scene. The model could, for instance, understand that there is a bouncing ball in the scene with certain number of bounces. The user can then direct the model to increase the number of bounces or the height of each bounce. Such a model can provide another layer of abstraction on top of the animation presets to make animating further accessible for casual motion designers.

Any or a combination of these suggestions can yield insights and technologies to improve the processes of authoring motion graphics, ultimately leading to the development of more usable and effective tools for users with varying skills.

6.6 Limitations

This study provides insights into incorporating an end-to-end motion design tool within the workflows of both professional and casual motion designers. However, certain limitations warrant a cautious interpretation of our findings. A broader, longitudinal deployment may be necessary to thoroughly evaluate the tool's impact on both user groups' performance. Additionally, while the example-based interaction introduced constitutes a significant advancement compared to traditional keyframing techniques, it deviates from the emerging trend of creativity tools focused on prompt-based interactions. At the time of this study, to the best of our knowledge, there was no software capable of generating motion graphics using prompts, which precluded direct comparisons. Future research could also investigate participants' team-based interactions, which could provide further insights into the tool's effectiveness in collaborative settings. Finally, as our study was done within the contexts of users and they each could have their own pace and produced different content, a direct

comparison between the works of different groups or individuals might be difficult. Future work can explore comparing specific prescribed tasks for gaining insights into the success rate of various tasks among different users.

6.7 Conclusion

This work provides insights into how users with varying expertise navigate the landscape of motion graphics authoring within their own contexts. Our findings underscore the importance of considering users' expectations, perceptions, and experiences in tool development. The participants in our research envisage a future of authoring where customization is amplified, and industry-specific tools emerge, offering a higher level of flexibility and efficiency. Moreover, despite the growing capabilities of automation technologies, our findings indicate that human oversight and control remain integral to the creative process and storytelling. Striking a balance between content automation, usability, and creative user input remains a pressing issue for both professional motion designers and casual motion designers.

Overall, the results from chapters 3 to 6, shed light on the challenges of motion designers, difficulties in using and switching among multiple software tools, the inception of Katika for authoring a variety of content for both casual and professional motion designers, and its evaluation within lab and user context. In what follows, I provide a reflection on this journey.

Chapter 7

Reflections, implications, and future work

7.0.1 Key takeaways

In this dissertation, I have presented the design of Katika and insights from various studies that illustrate how this end-to-end, example-based motion graphics tool can be used by users of different backgrounds and skill levels to create and customize their motion graphics. Katika offers a promising direction to aid a variety of users in creating the animated motion graphics videos. In particular, we found that the tool facilitated greater autonomy in learning and creation processes, opening new opportunities for serendipitous learning and enhancing overall understanding of motion graphics. Further, the practical understanding developed from formative studies discussed in Chapter 3, offers insights into the diversity of approaches to motion graphics creation, along with the challenges with utilizing the state-of-the-art feature-rich software applications of motion design. These insights, while informing the design of future motion graphics authoring tools, also underscore the role of motion graphics customization in users' practices and struggles in an increasingly complex visual content creation landscape. In this chapter, I reflect on the broader implications of this research, discuss potential improvements and tradeoffs in the end-to-end motion graphics tool's design, and explore connections to other HCI research areas that relate to my work.

7.1 Implications and promising directions for future work

7.1.1 Extending Katika for professional users

In this section, I highlight the potential avenues for extending Katika, the end-to-end system for creating explainer motion graphics videos, specifically with professional users in mind. Given the rise and increasing demands of professional motion designers, there is a discernible need to augment the features and capabilities of Katika to cater to this audience segment. Despite its initial design catering primarily to casual users, we observed how professional users could foresee a future where they could benefit from incorporating Katika

within their day to day work. These users often viewed the tool as a potential solution of prototyping and animatics that is currently missing in their pipelines. The lack of such a solution results in utilizing multiple, often disjointed applications to tackle the early stage tasks (pre-production) or the creation of the initial animatics.

At present, Katika simplifies the motion graphics creation process through pre-configured templates, semi-automated transitions, and a library of artwork and animation. As seen in chapter 6, for professional users, these features can be expanded to include more granular control over story elements, improved animation techniques, and integration with professional-grade software tools for a seamless workflow. Future work could also involve incorporating features like raster image input, multi-story development, or motion sketching to providing users with greater creative autonomy. Collaborative functionalities could also be expanded, enabling multiple users to work simultaneously on a project, a feature that aligns well with the team-oriented workflows common in professional settings. Ultimately, by extending Katika's functionality to accommodate the complex needs of power users, we can potentially unlock a new realm of possibilities in creating prototype motion graphics videos.

7.1.2 Personalizing Motion Graphics Authoring for Different Domains

Another interesting avenue is exploring the potential of tailoring the process of motion graphics authoring to meet the unique requirements of diverse domains and audiences. Considering the increasing utilization of motion graphics across various fields, it is important to understand how a tool like Katika can be customized to effectively address the unique needs of different user groups. As discussed in chapters 5 and 6 these groups can encompass professional motion designers with training in animation as well as other domain experts such as instructors, scientists, subject-matter experts, UI/UX designers, among others. In our studies, we came across scientists who wanted to see features specific to data visualization and animation or instructors who wished to have more educational content built into the software. Catering to all of these domains in the same software runs the risk of bloating the software with too many features and assets. Each of these domains carries its own set of needs, work processes, and communication styles, which may require distinct features or interfaces within a motion graphics tool. Traditionally, advanced tools of motion graphics (sch as Adobe After Effects [11] have been catering to various domains by adding all necessary features to the same interface. Such an approach while increasing the extensibility of the software, it adds to the feature-richness of the tool and further adds to its complexity. For that reason, we advocate for approaches that treat the software in a modular model and cater specific features to specific users.

For instructors, for instance, the focus may be on creating educational content that is both engaging and easy to understand. Therefore, they could benefit from simplified design workflows, a wide array of educational graphics, and functionalities that allow for the easy

incorporation of textual and verbal explanations. Scientists and subject-matter experts, on the other hand, often deal with complex data and concepts, hence, they might require data visualization capabilities and the option to create custom graphical assets that precisely represent their theories. For UI/UX designers, the emphasis is likely to be on crafting compelling, interactive digital experiences, thereby necessitating features like prototyping, animation controls for interaction design, and integration with UI design tools. By carefully considering these individual needs, we can begin to outline a roadmap for the development of domain-specific functionalities in motion graphics authoring tools, thereby enhancing their utility and usability across a wider spectrum of users.

7.1.3 Incentivizing Users to Collaborate on the Authoring Process

Katika has some minimal collaborative features. During our initial exploratory studies and the later in the interviews following the usability studies, our participants often spoke about the needs for collaboration in motion graphics video authoring. While professional users were interesting in synchronous collaboration with their teams, casual users wanted to see ways to connect with other users to benefit from their expertise. Similar to collaborative platforms such as Wikipedia, we can further explore the prospect of facilitating an environment of shared expertise and collective contribution in motion graphics authoring. The aim here is to foster a community wherein advanced users can share their knowledge, thereby benefiting those new to the field. By encouraging collaboration among users with varying skill levels, we could engender a supportive ecosystem that enhances the learning and authoring process, making it more efficient and less intimidating for beginners.

Creating this collaborative atmosphere could entail the development of features that enable knowledge sharing and cooperative work within the authoring tool itself. For instance, implementing a platform for sharing tutorial videos, template designs, or user-created content can provide novices with readily available resources and inspiration. Advanced users could be incentivized to contribute through mechanisms such as reputation systems, recognition within the community, or even tangible rewards. Moreover, incorporating real-time collaboration functionalities could facilitate teamwork on shared projects, allowing users to learn from each other directly. Enabling forums or chat features for problem-solving discussions and feedback exchange could create an active community around the tool. Thus, by harnessing the collective knowledge and expertise of its users, a motion graphics authoring tool could effectively lower the barriers to entry and nurture a thriving community of designers at all skill levels.

7.1.4 Extending the End-to-end Nature of the Tool to Include Story-telling

The proposition of Katika has been to serve as an end-to-end tool for motion graphics video authoring. While Katika currently supports the visual authoring stages, it presupposes that

users arrive equipped with a pre-defined storyline or script. During our usability studies (chapters 5 and 6) we came across multiple participants who —although professionals in their own domains— expected the software to support them with the storytelling aspect of the video as well. As such, this is a promising avenue for future research. For users who might struggle with this initial creative stage, we can integrate technologies such as machine learning to help generate narratives based on user-provided overviews or prompts, further bolstering the end-to-end nature of the tool.

By leveraging such techniques, we can create features that assist users in crafting engaging narratives. For instance, upon receiving a set of prompts or a basic synopsis from the user, the software could automatically generate a detailed script. Furthermore, it could provide suggestions for plot development, dialogue, and pacing, among other storytelling elements. In addition to narrative generation, we can also envisage features for assessing and improving story quality —providing feedback and recommendations to refine the narrative and ensure it effectively conveys the intended message. Such enhancements would render the tool more comprehensive, allowing it to support users from the very inception of their ideas through to the final stages of motion graphics authoring. Ultimately, this would create a truly end-to-end solution that not only lowers the barriers to entry into motion graphics design but also enriches the entire video creation process.

7.1.5 Similarities and Differences in Motion Graphics Authoring and Other Video Creation Types

Motion graphics authoring, while sharing some commonalities with other forms of video creation, presents unique challenges and opportunities that set it apart. The distinctions between motion graphics and other video authoring processes, such as live-action filming, traditional animation, and interactive media production, particularly stand out.

Unlike live-action video creation, which relies heavily on physical filming and actors, motion graphics authoring is predominantly software-driven, focusing on animated elements and visual effects. This distinction significantly influences the required skill set, production timelines, and the nature of storytelling. While live-action videos are constrained by the physical realities of sets and actors, motion graphics offer more flexibility in representing abstract concepts and stylized visuals. However, this flexibility in motion graphics comes with the challenge of creating engaging narratives purely through visual elements, without the support of real-world cues.

Traditional animation, like motion graphics, involves creating frame-by-frame visuals. However, motion graphics often leverage more sophisticated software capabilities, including 3D modeling and complex animation effects, which are less commonly used in traditional animation [85]. The design principles in motion graphics also differ, focusing more on conveying information and less on character-driven storytelling typical in traditional animation.

The intersection of graphic design principles with animation techniques in motion graphics sets it apart from the narrative-focused approach of traditional animation.

Interactive media production, such as video game design, shares some similarities with motion graphics in the use of digital assets and animation [24]. However, motion graphics authoring is typically linear and non-interactive, whereas interactive media demands a design approach that accounts for user input and dynamic content changes. The skills and tools required for creating interactive experiences are more complex, often involving programming and game engine utilization, which are not typically central to motion graphics work [173].

The skills and techniques developed in motion graphics authoring can be valuable in other video creation domains and vice versa. For example, the emphasis on visual storytelling and effective communication in motion graphics can enhance live-action video production. Similarly, principles learned in interactive media regarding user engagement can inform more dynamic motion graphics design.

By understanding these similarities and differences, professionals across various video creation fields can cross-pollinate ideas and techniques, leading to more innovative and effective visual content creation. This analysis not only helps in understanding the unique place of motion graphics in the broader video production landscape but also opens opportunities for integrating best practices across different domains.

7.2 Limitations of the research methodology

This thesis has advanced our knowledge in understanding the process and challenges associated with motion graphics design and adopting end-to-end authoring approaches. However, it does have its limitations, and the contributions are offered cautiously. The study primarily focused on the journey of motion designers, when working on entry-level animations (in the case of casual motion designers) or creating quick-and-dirty content (for professional motion designers). Future research could expand on this by investigating the difficulties experienced by professional motion designers when they tackle more advanced projects.

Our work also revealed the predominant role of Adobe After Effects in the industry, with few comparable alternatives, according to our participants. There's room for future investigations to undertake case studies comparing multiple software tools within a smaller team or organizational setting or even implement controlled studies to assess different tool interventions and approaches. Moreover, authoring motion graphics videos is often a collaborative process. Although, we had built some degree of collaboration into the Katika, this was an area that we did not get to fully explore and future research can evaluate and introduce novel approaches of collaboration within team settings.

Another limitation lies in the design of our tool, which currently only supports two-dimensional styles of motion graphics. As a result, future work could benefit from expanding

this approach to cater to other styles of motion graphics. A comparative study between our tool, Katika, and other software tools used to create motion graphics videos would also be advantageous. This is in the light of the fact that contemporary tools tend to lie at opposite ends of the spectrum —from high-quality production software such as Adobe After Effects to rudimentary content authoring tools like Microsoft PowerPoint or Google Slides.

Conducting research during the COVID-19 pandemic presented unique challenges that potentially impacted the limitations of this thesis. Restrictions on in-person interactions led to a heavier reliance on digital platforms for data collection, which might have limited the depth of understanding, reduced spontaneity, and potentially introduced biases in the gathered information. Lack of physical access to resources and research participants also likely influenced the sample size and diversity. Additionally, the global shift to remote work and the changes in software usage patterns that ensued could have influenced the interpretation and applicability of our results. The psychological stress and uncertainty related to the pandemic may have impacted both participants' responses and researchers' productivity. Moreover, the pandemic-induced changes might have affected the design practices, tool adoption, and collaborative patterns among motion designers, potentially making some of our findings less relevant in a post-pandemic world. Future work should take these aspects into account, possibly reconsidering the research design to adjust for these potential biases and influences.

Finally, while our study has illuminated the potential of an innovative motion design tool within the workflows of both professional and casual designers, a broader, more extended deployment may be necessary to thoroughly evaluate the tool's impact on both user groups' performance. This research has also introduced example-based interactions in our tool, marking a significant advancement from traditional keyframing techniques. However, it deviates from the emerging trend of creativity tools focused on prompt-based interactions. As no software capable of generating motion graphics using prompts was available at the time of the study, direct comparisons were impossible. Future research could also investigate team-based interactions with the tool, possibly unveiling its effectiveness in collaborative settings.

Chapter 8

Conclusion

The overarching aim of this dissertation was to investigate and evaluate innovative tools and methodologies for enhancing the workflow and creativity of both professional and casual motion designers. The development and evaluation of our design tool, Katika, through a user-centered design process has demonstrated that an in-context example-based interaction system can empower motion designers to discover and apply unfamiliar features within their daily workflows. Our initial usability studies showed that Katika's design is intuitive for motion designers to learn and use, offering valuable insight into the range of use cases envisioned for the design concept (Chapter 5). Building upon this promising start, the deployment of Katika in real-world scenarios provided vital insights into how motion designers quickly utilize example-based interaction to gain self-sufficiency in discovering and learning new motion graphics features, feel comfortable experimenting with new ideas instantaneously, and ensure consistency within their team (Chapter 6). Moreover, the design of Katika is informed and justified by exploratory work offering an empirical understanding of the diversity in motion designers' needs, attitudes, and practices. This includes the types of techniques they commonly employ, variations in how proactively they approach design, and the challenges they encounter in their workflow (Chapter 3). In this chapter, I conclude the dissertation by reflecting on my research contributions and how they can support further advancements in the design of motion graphics tools. Furthermore, these findings can potentially inform the development of more general classes of software, thus having a broader impact on the field of digital creativity tools.

8.1   Core research contributions

8.1.1   A synthesis of processes and challenges of authoring motion graphics videos for novices

One of the main contributions of this dissertation is the synthesis of the processes and challenges associated with the creation of motion graphics videos, particularly for casual users. This comprehensive study is the result of careful observation and in-depth interviews

with beginners in the motion graphics field. Our findings contribute to the burgeoning field of user interface design and interaction for motion graphics authoring tools.

Our research identifies the step-by-step process that novices undertake when authoring motion graphics videos, beginning from conception, through design and execution, and finally, to the evaluation and refining of the final product. Each stage has its intricacies and hurdles, and understanding these stages allows for a more in-depth look into where most novices struggle and where the most significant opportunities for improvements exist.

The challenges that beginners face are highlighted in our research, including issues such as understanding the software interface, mastering the timing and layering of animations, and dealing with software limitations. There are also broader challenges in switching among multiple feature-rich software applications knowledge workers face. In specific to motion graphics authoring, we particularly emphasize the steep learning curve of widely-used tools, such as Adobe After Effects, which often become a barrier to entry for beginners.

This synthesized perspective has profound implications on the design of new software and educational resources in the field of motion graphics. With an understanding of the pain points and stumbling blocks for casuals, developers and educators can better tailor their products and programs to support these individuals in overcoming such challenges and more effectively express their creativity through the medium of motion graphics. The insights generated from this synthesis thus play a pivotal role in informing and shaping the future of motion graphics authoring tools.

8.1.2 Empirical insights into how knowledge workers switch between a multitude of feature-rich software and its implications on the design of future tools

Since we observed the usage and switching among multiple feature-rich applications in motion designers, we further embarked on a deep, empirical understanding of how knowledge workers navigate and switch between numerous feature-rich software applications in their daily workflows. This detailed exploration into the practical complexities of multi-tool management has shed new light on an often-overlooked aspect of modern-day productivity.

The data gathered from our observational study, combined with in-depth interviews, have revealed the rich diversity of ways in which professionals manage their workflows across various applications, with strategies often varying based on their expertise, task at hand, and the specific nature of the software tools they use. We also illuminated the triggers and reasons that precipitate application switching, providing a comprehensive taxonomy that could aid in more targeted tool development.

The implications of these findings extend beyond mere observation, informing the design and functionality of future productivity software. By understanding the challenges and nuances of task-centric application switching, developers can design more intuitive and efficient software solutions that cater to the real-world needs of knowledge workers. The

goal would be to minimize disruptive context switches, foster smoother transitions, and possibly integrate functions across applications to reduce the need for frequent application switches.

These insights also have broader implications on how we conceptualize the work of knowledge workers, highlighting the intricate balance of focus and flexibility required in their dynamic work environments. Overall, this empirical investigation not only fills a critical gap in our understanding of modern work practices but also provides actionable intelligence for improving the design of future software tools.

8.1.3 Design of a new end-to-end motion graphics authoring software tool

The most significant technical contribution of this thesis lies in the design of a new end-to-end motion graphics authoring software tool, which we have named Katika. This comprehensive tool has been designed to accommodate the needs of both beginner and professional motion designers, providing an integrated environment that simplifies the process of creating intricate motion graphics.

Katika's design started with a careful analysis of the current industry standards and practices, the inherent challenges in existing tools, and the diverse needs of motion graphics designers at various skill levels. This analysis guided the feature selection for Katika, ensuring it encapsulated a wide range of necessary functionalities while remaining user-friendly and accessible.

Uniquely, Katika integrates an example-based approach to animation, utilizing our novel concept of motion-bundles; pre-packaged animation segments that can be applied to objects within a scene. This feature simplifies the creation process significantly, allowing users to bypass the complexity of traditional keyframing techniques. Additionally, Katika has been designed to support an end-to-end workflow, providing features that support the motion design process from ideation to final render.

The subsequent evaluation of Katika in real-world settings further emphasizes the success of its design. Both professional and novice users were able to seamlessly incorporate the software into their workflows, creating complex animations with greater efficiency and ease. These evaluations highlight Katika's potential to revolutionize the way motion graphics are authored, providing a comprehensive, user-friendly platform that supports the creative process from start to finish.

Overall, the design and realization of Katika represent a significant contribution to the field of motion graphics authoring, providing an innovative and powerful tool that could shape the future of the industry.

8.1.4 Simplifying animation and keyframing via motion-bundle abstraction

The implementation and evaluation of a novel approach to example-based animation through the introduction of the "motion-bundle" technique forms a significant contribution of this thesis. This unique approach marks a new approach in the field of motion graphics authoring, providing an intuitive and efficient means for animators to create dynamic motion graphics.

In our research, we designed and implemented a motion graphics tool that utilized the concept of motion-bundles —pre-packaged chunks of animation that could be applied to objects in a scene. This marked a departure from the traditional keyframing techniques, providing an innovative approach that allows for the efficient creation of complex animations with relative ease.

During our evaluations, we learned that users, especially those with minimal experience in motion graphics, were able to grasp the concept quickly, and they were able to produce complex animations in a shorter time than would have been impossible with traditional keyframing methods.

Our real-world deployment involved giving access to the tool to a group of professional and casual motion designers, and observing how they incorporated it into their workflows over a period of time. Feedback from the deployment was overwhelmingly positive, with many users praising the the animation capabilities that the motion bundles provided, and its potential to disrupt the way they approach their work.

Overall, the development and evaluation of this novel approach to example-based animation through motion-bundles offers a substantial contribution to the field. By providing empirical evidence of its efficacy and potential for adoption within professional and casual workflows, this research lays the groundwork for further advancements in the field of motion graphics authoring.

8.1.5 Empirical insights into how individuals with various levels of experience incorporate example-based motion graphics authoring into their daily workflows

Another key contribution of this thesis is identifying design implications for developers and learning content creators to better support task-centric application switching. This is an extension of our understanding of the reasons for application switching and represents practical recommendations for how the user experience can be enhanced.

In our research, we closely studied the workflows of motion designers, noting the common tasks for which they were compelled to switch applications. Analyzing these points of transition provided us with rich insights into the tensions between the software ecosystem, such as feature gaps, usability challenges, or workflow inefficiencies that compelled designers to switch between different tools.

Drawing from these insights, we propose a set of design implications that software developers and learning content creators should consider in their work. These implications range from creating more integrated software environments, improving interoperability between different applications, developing features that reduce the need for switching, to creating instructional materials that acknowledge the realities of task-centric application switching. This set of recommendations acts as a roadmap for those seeking to improve the experiences of motion designers.

Overall, these design implications contribute to our broader goal of making the design process more seamless for motion designers. They offer tangible strategies for reducing disruptions and inefficiencies caused by application switching, thereby helping designers maintain their creative flow and ultimately produce higher quality work. Our proposed design implications offer a valuable starting point for software developers and learning content creators aiming to provide more supportive, fluid, and intuitive digital environments for motion design professionals.

8.2 Secondary Contributions

8.2.1 An initial taxonomy of reasons for application switching

A secondary contribution of this thesis is the development of an initial taxonomy of reasons for application switching. This novel framework offers a structured lens for understanding the reasons why knowledge workers (including motion designers) may alternate between different software applications in the course of their work. This taxonomy was conceived from meticulous data analysis of our interviews and observations of motion designers at various stages of their professional journey.

In the fast-paced and evolving field of motion design, it is not uncommon for designers to use multiple applications to fulfill their creative vision. This could be due to the unique capabilities of different applications, or their comfort level with certain tools, or even the specific requirements of a project. Our taxonomy breaks down these various reasons into comprehensive categories, thereby providing a more nuanced understanding of application switching behavior.

This categorization is not just theoretically interesting but also has practical implications. By illuminating why designers switch between applications, we can make more informed decisions about how to design and implement future motion design tools. Our taxonomy can help software developers and HCI researchers design tools that better suit the workflow of designers, potentially reducing the need for disruptive application switching.

Therefore, our taxonomy of reasons for application switching presents a considerable stride forward in understanding and improving the work processes of motion designers. It lays the foundation for future work to further refine these categories and explore their implications in other digital creative industries.

8.2.2 Closing Remarks

This dissertation delved into diverse facets of understanding how motion designers with varying backgrounds experience the process of discovering, integrating, troubleshooting, and customizing their motion graphics software. Reflecting on these formative insights, we established concrete design goals that culminated in the design, implementation, and evaluation of Katika, a novel in-context platform to facilitate example-based interaction among motion designers.

Katika's features and design offer a promising path forward, helping motion designers learn and customize their motion graphics tools by leveraging the collective wisdom of their peers. I am convinced that these designs and insights are strategically positioned to underpin future efforts by HCI researchers and practitioners. The aim is to better understand and support motion designers as they navigate the increasingly complex world of customizing and using digital creativity tools, particularly in the realm of motion graphics.

Bibliography

[1] How much does animated video cost?

[2] The 2019 motion design survey, 2019.

[3] Zahra Shakeri Hossein Abad, Mohammad Noaeen, Didar Zowghi, Behrouz H Far, and Ken Barker. Two sides of the same coin: Software developers' perceptions of task switching and task interruption. In *Proceedings of the 22nd International Conference on Evaluation and Assessment in Software Engineering 2018*, Ease'18, pages 175–180, 2018.

[4] Zahra Shakeri Hossein Abad, Guenther Ruhe, and Mike Bauer. Task interruptions in requirements engineering: reality versus perceptions! In *2017 IEEE 25th International Requirements Engineering Conference (RE)*, pages 342–351. Ieee, 2017.

[5] Mark S. Ackerman, Juri Dachtera, Volkmar Pipek, and Volker Wulf. Sharing knowledge and expertise: The cscw view of knowledge management. *Computer Supported Cooperative Work (CSCW)*, 22(4-6):531–573, 2013.

[6] Piotr D Adamczyk and Brian P Bailey. If not now, when? the effects of interruption at different moments within task execution. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Chi '04, pages 271–278. Acm, 2004.

[7] Inc. Adobe. Adobe after effects. `https://www.adobe.com/products/aftereffects.html`, 2022. Accessed: 2022-08-18.

[8] Inc. Adobe. Adobe illustrator. `https://www.adobe.com/products/illustrator.html`, 2022. Accessed: 2022-08-18.

[9] Inc. Adobe. Adobe photoshop. `https://www.adobe.com/products/photoshop.html`, 2022. Accessed: 2022-08-18.

[10] Inc. Adobe. Adobe premiere. `https://www.adobe.com/products/premiere.html`, 2022. Accessed: 2022-08-18.

[11] Adobe After Effects. Adobe after effects, 2021.

[12] Lanto Ningrayati Amali, Nurtianingrat Zees, and Sitti Suhada. Motion graphic animation video as alternative learning media. *Jambura Journal of Informatics*, 2(1), 2020.

[13] Fereshteh Amini, Nathalie Henry Riche, Bongshin Lee, Christophe Hurter, and Pourang Irani. Understanding data videos: Looking at narrative visualization through the cinematography lens. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems - CHI '15*, pages 1459–1468. ACM Press, 2015.

[14] Fereshteh Amini, Nathalie Henry Riche, Bongshin Lee, Jason Leboe-McGowan, and Pourang Irani. Hooked on data videos: assessing the effect of animation and pictographs on viewer engagement. In *Proceedings of the 2018 International Conference on Advanced Visual Interfaces - AVI '18*, pages 1–9. ACM Press, 2018.

[15] Fereshteh Amini, Nathalie Henry Riche, Bongshin Lee, Andres Monroy-Hernandez, and Pourang Irani. Authoring data-driven videos with dataclips. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):501–510, 2017.

[16] Animaker. Animaker, 2021.

[17] Keynote Aple. Apple keynote, 2021.

[18] Inc. Apple. Apple airdrop. `https://support.apple.com/en-us/HT203106`, 2022. Accessed: 2022-08-18.

[19] Inc. Apple. Apple notes - icloud. `https://www.icloud.com/notes`, 2022. Accessed: 2022-08-18.

[20] Inc. Apple. Apple siri. `https://www.apple.com/siri/`, 2022. Accessed: 2022-08-18.

[21] Inc. Apple. icloud. `https://icloud.com/`, 2022. Accessed: 2022-08-18.

[22] Sinan Aral, Erik Brynjolfsson, and Marshall Van Alstyne. Information, technology, and information worker productivity. *Information Systems Research*, 23(3-part-2):849–867, 2012.

[23] Agnaldo Arroio. Comics as a narrative in natural science education. page 6, 2011.

[24] Fredrik Augustsson. *They did IT: The formation and organisation of interactive media production in Sweden.* PhD thesis, AlbaNova universitetscentrum (Tills m KTH), 2005.

[25] Inc. Autodesk. Autodesk maya. `https://www.autodesk.com/products/maya/overview`, 1989-2022.

[26] Inc. Autodesk. Autodesk motion builder. `https://www.autodesk.com/products/motionbuilder/`, 2022. Accessed: 2022-08-18.

[27] Axure Axure. Axure, 2021.

[28] Niksa Babic, Jesenka Pibernik, and Nikola Mrvac. Media study: Motion graphics. page 4, 2008.

[29] Benjamin Bach, Zezhong Wang, Matteo Farinella, Dave Murray-Rust, and Nathalie Henry Riche. Design patterns for data comics. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems - CHI '18*, pages 1–12. ACM Press, 2018.

[30] Hana Jamal Badi, Akram M Zeki, Waleed F Faris, and Roslina Bt Othman. Animation as a problem solving technique in mechanical engineering education. 4(5):4, 2013.

[31] Brian P Bailey and Joseph A Konstan. On the need for attention-aware systems: Measuring effects of interruption on task performance, error rate, and affective state. *Computers in Human Behavior*, 22(4):685–708, 2006.

[32] Lingfeng Bao, Zhenchang Xing, Xin Xia, and David Lo. Vt-revolution: Interactive programming video tutorial authoring and watching system. *IEEE Transactions on Software Engineering*, 45(8):823–838, 2019.

[33] Miri Barak, Tamar Ashkar, and Yehudit J. Dori. Learning science via animated movies: Its effect on students' thinking and motivation. *Computers & Education*, 56(3):839–846, 2011.

[34] Jakob E Bardram, Steven Jeuris, Paolo Tell, Steven Houben, and Stephen Voida. Activity-centric computing systems. *Commun. ACM*, 62(8):72–81, 2019.

[35] Spencer Barnes. Studies in the efficacy of motion graphics: The impact of narrative structure on exposition. *Digital Journalism*, 5(10):1260–1280, 2017.

[36] Scott Bateman, Regan L. Mandryk, Carl Gutwin, Aaron Genest, David McDine, and Christopher Brooks. Useful junk?: the effects of visual embellishment on comprehension and memorability of charts. In *Proceedings of the 28th international conference on Human factors in computing systems - CHI '10*, page 2573. ACM Press, 2010.

[37] Travis Bergwall. 7 reasons students learn better with video, 2015.

[38] Biteable. Biteable, 2019.

[39] João Aranda Brandão. Motion graphics ergonomics: Animated semantic system, for typographical communication efficiency. *Procedia Manufacturing*, 3:6376–6379, 2015.

[40] Joel Brandt, Philip J Guo, Joel Lewenstein, Mira Dontcheva, and Scott R Klemmer. Two studies of opportunistic programming: interleaving web foraging, learning, and writing code. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1589–1598. Acm, 2009.

[41] Matthew Brehmer, Bongshin Lee, Benjamin Bach, Nathalie Henry Riche, and Tamara Munzner. Timelines revisited: A design space and considerations for expressive storytelling. *IEEE Transactions on Visualization and Computer Graphics*, 23(9):2151–2164, 2017.

[42] Eric Brill and Raymond J Mooney. An overview of empirical natural language processing. page 12, 1997.

[43] Alexandre Carlier, Martin Danelljan, Alexandre Alahi, and Radu Timofte. Deepsvg: A hierarchical generative network for vector graphics animation. *Advances in Neural Information Processing Systems*, 33:16351–16361, 2020.

[44] John M Carroll and Caroline Carrithers. Training wheels in a user interface. *Commun. ACM*, 27(8):800–806, 1984.

[45] John M Carroll and Mary Beth Rosson. Paradox of the active user. In *Interfacing Thought: Cognitive Aspects of Human-Computer Interaction*, pages 80–111. MIT Press, 1987.

[46] Taner Çifçi. Effects of infographics on students achievement and attitude towards geography lessons. *Journal of Education and Learning*, 5(1):154, 2016.

[47] Yan Chen, Sang Won Lee, Yin Xie, YiWei Yang, Walter S. Lasecki, and Steve Oney. Codeon: On-demand software development assistance. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, pages 6220–6231. Acm, 2017.

[48] Chew Chew. What are motion graphics?, 2019.

[49] Peggy Chi, Zheng Sun, Katrina Panovich, and Irfan Essa. Automatic video creation from a web page. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology*, pages 279–292. Acm, 2020.

[50] Parmit K. Chilana, Amy J. Ko, and Jacob O. Wobbrock. Lemonaid: selection-based crowdsourced contextual help for web applications. In *Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems - CHI '12*, page 1549. ACM Press, 2012.

[51] Parmit K. Chilana, Jacob O. Wobbrock, and Amy J. Ko. Understanding usability practices in complex domains. In *Proceedings of the 28th international conference on Human factors in computing systems - CHI '10*, page 2337. ACM Press, 2010.

[52] Mark Christiansen. *Adobe® after effects® cs3 professional studio techniques.* 2007.

[53] Devan Ciccarelli. Google slides vs powerpoint vs keynote: Everything you need to know, 2020.

[54] Cinema4D. Cinema4d.

[55] Andy Cockburn, Carl Gutwin, Joey Scarr, and Sylvain Malacria. Supporting novice to expert transitions in user interfaces. *ACM Computing Surveys*, 47(2):1–36, 2015.

[56] Justin Cone. 6 challenges facing motion design education, 2015.

[57] Juliet Corbin and Anselm Strauss. Basics of qualitative research: techniques and procedures for developing grounded theory., 2007.

[58] Juliet Corbin and Anselm Strauss. *Basics of Qualitative Research.* Sage, 2014.

[59] Carman Coto. What are explainer videos, and why are they so popular, 2018.

[60] Mary Czerwinski, Eric Horvitz, and Susan Wilhite. A diary study of task switching and interruptions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Chi '04, pages 175–182. Acm, 2004.

[61] James Davis, Maneesh Agrawala, Erika Chuang, and Zoran Popovi. A sketching interface for articulated figure animation. page 10, 2003.

[62] Richard C. Davis, Brien Colwell, and James A. Landay. K-sketch: a 'kinetic' sketch pad for novice animators. In *Proceeding of the twenty-sixth annual CHI conference on Human factors in computing systems - CHI '08*, page 413. ACM Press, 2008.

[63] Richard C Davis and James A Landay. Informal animation sketching: Requirements and design. *AAAI Technical Report*, page 7, 2004.

[64] Marco de Sá, David A. Shamma, and Elizabeth F. Churchill. Live mobile collaboration for video production: design, guidelines, and requirements. *Personal and Ubiquitous Computing*, 18(3):693–707, 2014.

[65] Alan Dix, Janet Finlay, Gregory Abowd, and Russell Beale. *Human-Computer Interaction*. Prentice Hall, 2004.

[66] Doodly. Doodly, 2021.

[67] Anton N Dragunov, Thomas G Dietterich, Kevin Johnsrude, Matthew McLaughlin, Lida Li, and Jonathan L Herlocker. Tasktracer: a desktop environment to support multi-tasking knowledge workers. In *Proceedings of the 10th International Conference on Intelligent User Interfaces*, Iui '05, pages 75–82. Acm, 2005.

[68] Jon Draper. Live action vs animation: A comparison, 2019.

[69] Matt Ellis. Motion graphics vs. animation: what's the difference, 2019.

[70] Figma. Figma. `https://www.figma.com/`, 2016-2022.

[71] Leah Findlater and Joanna McGrenere. Evaluating reduced-functionality interfaces according to feature findability and awareness. In *IFIP Conference on Human-Computer Interaction*, Interact '07, pages 592–605. Springer-Verlag, 2007.

[72] Firebase. Firebase, 2021.

[73] Sarah Flanagan. How does storytelling within higher education contribute to the learning experience of early years students? *Journal of Practice Teaching and Learning,*, page 23, 2015.

[74] Roberto L. Flores, Patrick DeMoss, Carrie Klene, Robert J. Havlik, and Sunil Tholpady. Digital animation versus textbook in teaching plastic surgery techniques to novice learners:. *Plastic and Reconstructive Surgery*, 132(1):101e–109e, 2013.

[75] Dustin Freeman, Stephanie Santosa, Fanny Chevalier, Ravin Balakrishnan, and Karan Singh. Laces: live authoring through compositing and editing of streaming video. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1207–1216. Acm, 2014.

[76] Lu Geng. Study of the motion graphic design at the digital age. In *Proceedings of the 2016 International Conference on Arts, Design and Contemporary Education*. Atlantis Press, 2016.

[77] Andrea Giessmann and Katarina Stanoevska. Platform as a service–a conjoint study on consumers' preferences. In *33rd International Conference on Information Systems*, Icis '12. Association for Information Systems, 2012.

[78] Andreas Girgensohn, Sara Bly, Frank Shipman, John Boreczky, and Lynn Wilcox. Home video editing made easy – balancing automation and user control. *Interact*, page 8, 2001.

[79] ATLAS.ti Scientific Software Development GmbH. Atlas.ti. `https://atlasti.com/`, 1993-2022.

[80] Victor M González and Gloria Mark. "constant, constant, multi-tasking craziness": Managing multiple working spheres. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Chi '04, pages 113–120. Acm, 2004.

[81] Victor M González and Gloria Mark. Managing currents of work: Multi-tasking among multiple collaborations. In *Ecscw 2005*, pages 143–162. Springer, 2005.

[82] Elizabeth Goodman, Erik Stolterman, and Ron Wakkary. Understanding interaction design practices. In *Proceedings of the 2011 annual conference on Human factors in computing systems - CHI '11*, page 1061. ACM Press, 2011.

[83] Google. Google slides. `https://slides.google.com/`, 2022. Accessed: 2022-08-18.

[84] Brandon Gorthy. Coffee cup proper disposal psa, 2013.

[85] Randy Gossman. The animation production process. *Recuperado de http://nir3d. com/handouts/Handouts% 20History% 20of% 20Animation*, 2011.

[86] Chirag Goyal. Part 11: Step by step guide to master nlp – syntactic analysis, 2021.

[87] Connor Gramazio, Rundolph Linder, James Farrow, Jeffrey Heer, and Steven Dow. Learning from examples in the wild: The role of optional metadata. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, pages 1–12. Acm, 2018.

[88] Tovi Grossman, George Fitzmaurice, and Ramtin Attar. A survey of software learnability: Metrics, methodologies and guidelines. page 10, 2009.

[89] Mieke Haesen, Jan Meskens, Kris Luyten, and Karin Coninx. Draw me a storyboard: Incorporating principles and techniques of comics to ease communication and artefact creation in user-centred design. page 8, 2010.

[90] Harvest. Harvest time tracking. `https://www.getharvest.com/`, 2013-2022.

[91] Masaki Hayashi, Seiki Inoue, Mamoru Douke, Narichika Hamaguchi, Hiroyuki Kaneko, Steven Bachelder, and Masayuki Nakajima. T2v new technology of converting text to cg animation. 2(1):8, 2014.

[92] D. Austin Henderson and Stuart Card. Rooms: The use of multiple virtual workspaces to reduce space contention in a window-based graphical user interface. *ACM Trans. Graph.*, 5(3):211–243, 1986.

[93] Raphael Hoffmann, Patrick Baudisch, and Daniel S. Weld. Evaluating visual cues for window switching on large screens. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Chi '08, pages 929–938. Acm, 2008.

[94] Steven Houben, Jakob E Bardram, Jo Vermeulen, Kris Luyten, and Karin Coninx. Activity-centric support for ad hoc knowledge work: A case study of co-activity manager. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Chi '13, pages 2263–2272. Acm, 2013.

[95] Xian-Sheng Hua, Zengzhi Wang, and Shipeng Li. Lazycut content-aware template-based video authoring. In *Proceedings of the 13th annual ACM international conference on Multimedia - MULTIMEDIA '05*, page 792. ACM Press, 2005.

[96] Nathaniel Hudson, Celena Alcock, and Parmit K. Chilana. Understanding newcomers to 3d printing: Motivations, workflows, and barriers of casual makers. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems - CHI '16*, pages 384–396. ACM Press, 2016.

[97] Jessica Hullman, Steven Drucker, Nathalie Henry Riche, Bongshin Lee, Danyel Fisher, and Eytan Adar. A deeper understanding of sequence in narrative visualization. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2406–2415, 2013.

[98] Dugald Ralph Hutchings, Greg Smith, Brian Meyers, Mary Czerwinski, and George Robertson. Display space usage and window management operation comparisons between single monitor and multiple monitor users. In *Proceedings of the Working Conference on Advanced Visual Interfaces*, Avi '04, pages 32–39. Acm, 2004.

[99] Adobe Illustrator. Adobe illustrator, 2021.

[100] iMovie iMovie. imovie, 2021.

[101] Apple Inc. Final cut pro. `https://www.apple.com/ca/final-cut-pro/`, 2022. Accessed: 2022-12-13.

[102] Shamsi T Iqbal and Brian P Bailey. Leveraging characteristics of task structure to predict the cost of interruption. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Chi '06, pages 741–750. Acm, 2006.

[103] Shamsi T Iqbal and Brian P Bailey. Understanding and developing models for detecting and differentiating breakpoints during interactive tasks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Chi '07, pages 697–706. Acm, 2007.

[104] Amir Jahanlou and Parmit K Chilana. Katika: An end-to-end system for authoring amateur explainer motion graphics videos. In *CHI Conference on Human Factors in Computing Systems*, Chi '22, pages 1–14. Acm, 2022.

[105] Amir Jahanlou, William Odom, and Parmit Chilana. Challenges in getting started in motion graphic design: Perspectives from casual and professional motion designers. *Graphics Interface 2021*, page 11, 2021.

[106] Amir Jahanlou, Jo Vermeulen, Tovi Grossman, Parmit Chilana, George Fitzmaurice, and Justin Matejka. Task-centric application switching: How and why knowledge workers switch software applications for a single task. In *Graphics Interface 2023*.

[107] Mikkel RJakobsen and Kasper Hornb. Piles, tabs and overlaps in navigation among documents. In *Proceedings of the 6th Nordic Conference on Human-Computer Interaction: Extending Boundaries*, NordiCHI '10, pages 246–255. Acm, 2010.

[108] Steven Jeuris, Paolo Tell, Steven Houben, and Jakob E Bardram. The hidden cost of window management. *arXiv preprint arXiv:1810.04673*, 2018.

[109] jitter.video. Jitter app, 2023.

[110] Katika. Katika app, 2021.

[111] Rubaiat Habib Kazi, Fanny Chevalier, Tovi Grossman, and George Fitzmaurice. Kitty: sketching dynamic and interactive illustrations. In *Proceedings of the 27th annual ACM symposium on User interface software and technology - UIST '14*, pages 395–405. ACM Press, 2014.

[112] Rubaiat Habib Kazi, Fanny Chevalier, Tovi Grossman, Shengdong Zhao, and George Fitzmaurice. Draco: bringing life to illustrations with kinetic textures. In *Proceedings of the 32nd annual ACM conference on Human factors in computing systems - CHI '14*, pages 351–360. ACM Press, 2014.

[113] Rubaiat Habib Kazi, Tovi Grossman, Cory Mogk, Ryan Schmidt, and George Fitzmaurice. Chronofab: Fabricating motion. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems - CHI '16*, pages 908–918. ACM Press, 2016.

[114] Rubaiat Habib Kazi, Tovi Grossman, Nobuyuki Umetani, and George Fitzmaurice. Motion amplifiers: Sketching dynamic illustrations using the principles of 2d animation. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pages 4599–4609. Acm, 2016.

[115] Kimia Kiani, Parmit K. Chilana, Andrea Bunt, Tovi Grossman, and George Fitzmaurice. "i would just ask someone": Learning feature-rich design software in the modern workplace. In *2020 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*, pages 1–10. Ieee, 2020.

[116] Kimia Kiani, George Cui, Andrea Bunt, Joanna McGrenere, and Parmit K. Chilana. Beyond "one-size-fits-all": Understanding the diversity in how software newcomers discover and make use of help resources. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems - CHI '19*, pages 1–14. ACM Press, 2019.

[117] Jiyoon Kim, Min Suk Chung, Hae Gwon Jang, and Beom Sun Chung. The use of educational comics in learning anatomy among multiple student groups: Anatomy learning comics. *Anatomical Sciences Education*, 10(1):79–86, 2017.

[118] Juho Kim. Learnersourcing: Improving learning with collective learner activity. *Doctoral dissertation, Massachusetts Institute of Technology*, page 213, 2015.

[119] Juho Kim, Phu Tran Nguyen, Sarah Weir, Philip J. Guo, Robert C. Miller, and Krzysztof Z. Gajos. Crowdsourcing step-by-step information extraction to enhance existing how-to videos. In *Proceedings of the 32nd annual ACM conference on Human factors in computing systems - CHI '14*, pages 4017–4026. ACM Press, 2014.

[120] Nam Wook Kim, Hanspeter Pfister, Nathalie Henry Riche, Benjamin Bach, Guanpeng Xu, Matthew Brehmer, Ken Hinckley, Michel Pahud, Haijun Xia, and Michael J. McGuffin. Datatoon drawing dynamic network comics with pen + touch interaction. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems - CHI '19*, pages 1–12. ACM Press, 2019.

[121] Oliver Krancher, Pascal Luther, and Marc Jost. Key affordances of platform-as-a-service: Self-organization and continuous feedback. *Journal of Management Information Systems*, 35(3):776–812, 2018.

[122] Jon S. Krasner. *Motion graphic design: applied history and aesthetics.* Focal Press, 2nd ed. edition, 2008.

[123] Topaz Labs. Topazlabs, 2021.

[124] Benjamin Lafreniere, Tovi Grossman, and George Fitzmaurice. Community enhanced tutorials: improving tutorials with multiple demonstrations. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems - CHI '13*, page 1779. ACM Press, 2013.

[125] George Lawton. Developing software online with platform-as-a-service technology. *Computer*, 41(6):13–15, 2008.

[126] Jonathan Lazar and Jinjuan Heidi Feng. Research methods in human-computer interaction. page 53, 2010.

[127] Mackenzie Leake, Hijung Valentina Shin, Joy O. Kim, and Maneesh Agrawala. Generating audio-visual slideshows from text articles using word concreteness. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–11. Acm, 2020.

[128] Ghang Lee, Charles M. Eastman, Tarang Taunk, and Chun-Heng Ho. Usability principles and best practices for the user interface design of complex 3d architectural design and engineering tools. *International Journal of Human-Computer Studies*, 68(1-2):90–104, 2010.

[129] Mariëlle Leijten, Luuk Van Waes, Karen Schriver, and John R Hayes. Writing in the workplace: Constructing documents using multiple digital sources. *Journal of Writing Research*, 5(3):285–337, 2014.

[130] Germán Leiva and Michel Beaudouin-Lafon. Montage: A video prototyping system to reduce re-shooting and increase re-usability. In *The 31st Annual ACM Symposium on User Interface Software and Technology - UIST '18*, pages 675–682. ACM Press, 2018.

[131] Wei Li, Tovi Grossman, and George Fitzmaurice. Gamicad: a gamified tutorial system for first time autocad users. In *Proceedings of the 25th annual ACM symposium on User interface software and technology - UIST '12*, page 103. ACM Press, 2012.

[132] Raphael Gontijo Lopes, David Ha, Douglas Eck, and Jonathon Shlens. A learned representation for scalable vector graphics. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7930–7939, 2019.

126

[133] Black Magic Design Pty Ltd. Davinci resolve. `https://www.blackmagicdesign.com/products/davinciresolve`, 2022. Accessed: 2022-12-13.

[134] Jackson G Lu, Modupe Akinola, and Malia F Mason. "switching on" creativity: Task switching can increase creativity by reducing cognitive fixation. *Organizational Behavior and Human Decision Processes*, 139:63–75, 2017.

[135] Min Lu, Chufeng Wang, Joel Lanir, Nanxuan Zhao, Hanspeter Pfister, Daniel Cohen-Or, and Hui Huang. Exploring visual information flows in infographics. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–12. Acm, 2020.

[136] Lumen. Lumen 5.

[137] Magicons. Magicons, 2021.

[138] Elvira Martinez, Victoria Carbonell, Mercedes Florez, and Jose Amaya. Simulations as a new physics teaching tool. *Computer Applications in Engineering Education*, 18(4):757–761, 2010.

[139] Justin Matejka, Wei Li, Tovi Grossman, and George Fitzmaurice. Community-commands: command recommendations for software applications. In *Proceedings of the 22nd annual ACM symposium on User interface software and technology - UIST '09*, page 193. ACM Press, 2009.

[140] Autodesk Maya. Autodesk maya, 2021.

[141] Joanna McGrenere, Ronald M Baecker, and Kellogg S Booth. An evaluation of a multiple interface design solution for bloated software. page 8, 2002.

[142] Joanna McGrenere and Maria Klawe. Field evaluation of the continuous zoom: revisiting zoom and pick rate. *Journal of Visual Languages & Computing*, 18(5):502–525, 2007.

[143] Britta Meixner, Katarzyna Matusik, Christoph Grill, and Harald Kosch. Towards an easy to use authoring tool for interactive non-linear video. *Multimedia Tools and Applications*, 70(2):1251–1276, 2014.

[144] Valeriu Mercea, Florica Moldoveanu, and Anca Moldoveanu. Learning in context: using situated cognition and connectionism for 3d modeling. *Educational Technology & Society*, 14(1):56–66, 2011.

[145] André N Meyer, Laura E Barton, Gail C Murphy, Thomas Zimmermann, and Thomas Fritz. The work life of developers: Activities, switches and perceived productivity. *IEEE Trans. Softw. Eng.*, 43(12):1178–1193, 2017.

[146] André N Meyer, Gail C Murphy, Thomas Fritz, and Thomas Zimmermann. Developers' diverging perceptions of productivity. In *Rethinking Productivity in Software Engineering*, pages 137–146. Apress, 2019.

[147] midjourney.com. Midjourney, 2023.

[148] Justin Mifsud. The difference (and relationship) between usability and learnability, 2018.

[149] Tomer Moscovich and John F Hughes. Animation sketching: An approach to accessible animation. *Unpublished Master's Thesis, CS Department, Brown University*, page 5, 2001.

[150] Tsai-Yun Mou. Creative story design method in animation production pipeline. page 8, 2015.

[151] Santiago Negrete-Yankelevich and Nora Morales-Zaragoza. e-motion: a system for the development of creative animatics. *Proceedings of the Fourth International Conference on Computational Creativity (p. 184)*, page 5, 2013.

[152] Mark Newman, James Lin, Jason Hong, and James Landay. Denim: An informal web site design tool inspired by observations of practice. *Human-Computer Interaction*, 18(3):259–324, 2003.

[153] Darren Newtson. Attribution and the unit of perception of ongoing behavior. *Journal of personality and social psychology*, 28(1):28, 1973.

[154] Naoko Nitta and Noboru Babaguchi. Example-based video remixing for home videos. In *2011 IEEE International Conference on Multimedia and Expo*, pages 1–6. Ieee, 2011.

[155] Susanna Nordmark and Marcelo Milrad. Mobile digital storytelling for promoting creative collaborative learning. In *2012 IEEE Seventh International Conference on Wireless, Mobile and Ubiquitous Technology in Education*, pages 9–16. Ieee, 2012.

[156] Midas Nouwens and Clemens Nylandsted Klokmose. The application and its consequences for non-standard knowledge work. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, Chi '18, pages 1–12. Association for Computing Machinery, 2018.

[157] Nuria Oliver, Mary Czerwinski, Greg Smith, and Kristof Roomp. Relalttab: Assisting users in switching windows. In *Proceedings of the 13th International Conference on Intelligent User Interfaces*, Iui '08, pages 385–388. Acm, 2008.

[158] Nuria Oliver, Greg Smith, Chintan Thakkar, and Arun C Surendran. Swish: Semantic analysis of window titles and switching history. In *Proceedings of the 11th International Conference on Intelligent User Interfaces*, Iui '06, pages 194–201. Acm, 2006.

[159] OpenAI. Chat gpt, 2022.

[160] Srishti Palani, David Ledo, George Fitzmaurice, and Fraser Anderson. "i don't want to feel like i'm working in a 1960s factory": The practitioner perspective on creativity support tool adoption. In *CHI Conference on Human Factors in Computing Systems*, Chi '22. Association for Computing Machinery, 2022.

[161] Lucia Pechersky. How long does it takes to produce a customized explainer video?, 2018.

[162] Francesc Pedró. The new millennium learners. page 17, 2006.

[163] Adobe Photoshop. Adobe photoshop, 2021.

[164] Point Power. Powerpoint, 2021.

[165] Adobe Premiere. Adobe premiere, 2021.

[166] J Quevedo-Fernández. idanimate a general-purpose animation sketching tool for multi-touch devices. page 10, 2013.

[167] Tye Rattenbury and John Canny. Caad: An automatic task support system. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Chi '07, pages 687–696. Acm, 2007.

[168] Pradyumna Reddy, Michael Gharbi, Michal Lukac, and Niloy J Mitra. Im2vec: Synthesizing vector graphics without vector supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7342–7351, 2021.

[169] Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. Generative adversarial text to image synthesis. page 10, 2016.

[170] rive.app. Rive app, 2023.

[171] runwayml. Runway, 2020.

[172] Markus Salo and Markus Makkonen. Why do users switch mobile applications? trialing behavior as a predecessor of switching behavior. *Commun. Assoc. Inf. Syst.*, 42, 2018.

[173] David Z Saltz. Live media: Interactive technology and theatre. *Theatre Topics*, 11(2):107–130, 2001.

[174] Dario D Salvucci, Niels A Taatgen, and Jelmer P Borst. Toward a unified theory of the multitasking continuum: From concurrent performance to task switching, interruption, and resumption. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Chi '09, pages 1819–1828. Acm, 2009.

[175] Heider Sanchez, Romain Robbes, and Victor M Gonzalez. An empirical study of work fragmentation in software evolution tasks. In *2015 IEEE 22nd International Conference on Software Analysis, Evolution, and Reengineering (SANER)*, pages 251–260. Ieee, 2015.

[176] Widi Sarinastiti, Dwi Susanto, and Rodhiyah Mutammimah. Skill level animation technique on dental care motion graphic for children. In *2016 International Electronics Symposium (IES)*, pages 389–394. Ieee, 2016.

[177] Amaral Schlittler. Motion_graphics_and_animation, 2014.

[178] Klaus Schoeffmann, Marco A. Hudelist, and Jochen Huber. Video interaction tools: A survey of recent work. *ACM Computing Surveys*, 48(1):1–34, 2015.

[179] Yang Shi, Xingyu Lan, Jingwen Li, Zhaorui Li, and Nan Cao. Communicating with motion: A design space for animated visual narratives in data videos. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pages 1–13. Acm, 2021.

[180] Google Slides. Google slides, 2021.

[181] Greg Smith, Patrick Baudisch, George Robertson, Mary Czerwinski, Brian Meyers, Daniel Robbins, and Donna Andrews. Groupbar: The taskbar evolved. In *OZCHI 2003 Conference for the Computer-Human Interaction Special Interest Group of the Human Factors Society of Australia*, Ozchi '03, 2003.

[182] stablediffusionweb.com. Stable diffusion, 2023.

[183] Alex Stone, Phoebe Sengers, Daniela K Merrill, and Mark Rouncefield. A theory of interaction: A treatise on the principles of exchange, cohesion, and general intellect in the age of hci. *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, 2018.

[184] Kaoru Sumi and Mizue Nagata. Animated storytelling system via text. In *Proceedings of the 2006 ACM SIGCHI international conference on Advances in computer entertainment technology - ACE '06*, page 55. ACM Press, 2006.

[185] Edward R Sykes. Interruptions in the workplace: A case study to reduce their effects. *Int. J. Inf. Manag.*, 31(4):385–394, 2011.

[186] synthesia video. Synthesia, 2020.

[187] Susanne Tak. *Understanding and Supporting Window Switching*. PhD thesis, University of Canterbury, Computer Science and Software Engineering, New Zealand, 2011.

[188] Naimdjon Takhirov and Fabien Duchateau. A cloud-based and social authoring tool for video. In *Proceedings of the 11th ACM symposium on Document engineering - DocEng '11*, page 65. ACM Press, 2011.

[189] TedEd TedEd. Build a lesson around any ted-ed animation, ted talk or youtube video, 2018.

[190] J. Thompson, Z. Liu, W. Li, and J. Stasko. Understanding the design space and authoring paradigms for animated data graphics. *Computer Graphics Forum*, 39(3):207–218, 2020.

[191] Matthew Thorne, David Burke, and Michiel van de Panne. Motion doodles: An interface for sketching character motion. page 8, 2004.

[192] TikTok TikTok. Tiktok, 2021.

[193] Rescue Time. Rescue time. `https://www.rescuetime.com/`, 2013-2022.

[194] Anh Truong, Floraine Berthouzoz, Wilmot Li, and Maneesh Agrawala. Quickcut: An interactive tool for editing narrated video. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*, pages 497–507. Acm, 2016.

[195] Orana Valerde. What is the difference between motion graphics and animation, 2018.

[196] Laton Vermette, Joanna McGrenere, Colin Birge, Adam Kelly, and Parmit K. Chilana. Freedom to personalize my digital classroom: Understanding teachers' practices and motivations. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems - CHI '19*, pages 1–14. ACM Press, 2019.

[197] Vimeo. Vimeo, 2019.

[198] Yun Wang, Haidong Zhang, He Huang, Xi Chen, Qiufeng Yin, Zhitao Hou, Dongmei Zhang, Qiong Luo, and Huamin Qu. Infonice: Easy creation of information graphics. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems - CHI '18*, pages 1–12. ACM Press, 2018.

[199] Daniel Wilches, Pablo Figueroa, Agustin Conde, and Faramarz Samavati. Sketch-based interface for animation for non-experts. In *2012 XXXVIII Conferencia Latinoamericana En Informatica (CLEI)*, pages 1–8. Ieee, 2012.

[200] Nora S Willett, Hijung Valentina Shin, Zeyu Jin, Wilmot Li, and Adam Finkelstein. Pose2pose: pose selection and transfer for 2d character animation. In *Proceedings of the 25th International Conference on Intelligent User Interfaces*, pages 88–99. Acm, 2020.

[201] Wyzowl. How much does animated video cost, 2019.

[202] Adobe XD. Adobe xd, 2021.

[203] Haijun Xia. Crosspower: Bridging graphics and linguistics. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology*, pages 722–734. Acm, 2020.

[204] Haijun Xia, Jennifer Jacobs, and Maneesh Agrawala. Crosscast adding visuals to audio travel podcasts. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology*, pages 735–746. Acm, 2020.

[205] Jiayi Eris Zhang, Nicole Sultanum, Anastasia Bezerianos, and Fanny Chevalier. Dataquilt: Extracting visual elements from images to craft pictorial visualizations. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–13. Acm, 2020.

[206] Zhenpeng Zhao, Rachael Marr, and Niklas Elmqvist. Data comics: Sequential art for data-driven storytelling. page 12, 2015.

[207] John Zimmerman, Jodi Forlizzi, and Shelley Evenson. Research through design as a method for interaction design research in hci. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems - CHI '07*, pages 493–502. ACM Press, 2007.

[208] Inc. Zoom Video Communications. Zoom. `https://zoom.us/`, 2022. Accessed: 2022-08-18.

Appendix A

Study materials

In this appendix, I include study instruments and related documents that were used to conduct and collect data from participants in my research.

Application Switching Study Procedure Document

Title:

An Empirical Study of Software Switching: Summary, Taxonomy, and Considerations

Introduction to this research:

As the number of software tools in users' arsenal expands, so do the challenges of interoperability between these different applications. We want to provide a taxonomy of the application switching for users in various domains and put forth considerations for future tools. Such insights can support future software designers in improving users' efficiency and lowering the cost of application switching plays a major role in that. To that end, we have identified the following problem statement, objective, and research questions:

Problem

Using multiple tools introduces challenges of interoperability and turning to large applications presents difficulties with learnability.

Objective

The objective of this research is to increase users' efficiency when using multiple software applications (or various modules of a large application) for completing a task, as well as producing an understanding of the challenges with trends toward greater integration (big applications), and the considerations for such integrations.

Research question

What are the processes of challenges of applications switching in various domains of knowledge work?

What is the taxonomy of switching between applications in different domains?

Sub-Questions:

What are the various reasons behind switching between applications and what are the challenges of doing so?

What is the attitude of users towards large applications with many features and functions?

What can we learn from application switching challenges to inform the design of future large tools?

In conducting this research, we are interested in:

- Unexpected insights into the process of integration,
- Challenges and areas in which the assumptions are contradicted,
- Insights into practical approaches to address challenges,
- Potential new directions for research.

The nature of the study:

This study will be an in-depth interview with representatives of two groups of individuals: a) individuals that use computers (either for casual/non-professional work or for knowledge work in a variety of domains), and b) product managers at software companies who decide about the feature/functions of tools and the interoperability of tools with one another. Representatives in each group will respond to questions that speak to their experience with decision-making, use of software, challenges, processes, and workarounds. As articulated in the consent form, there is no perceived risk or benefit for participants in this study.

Study protocol:

This study will be conducted remotely. Participants will be invited to a remote meeting platform (Zoom) and asked various questions regarding their use/development or integration of software tools. The interview will take 60 minutes to complete, and the sessions will be audio recorded (upon their consent) for future transcriptions.

Consent:

A consent form has been prepared, and participants will be given a verbal definition of its content. Upon participant's approval, they will be asked to review and sign the consent form online and submit it. Once the investigator receives the submitted form, s/he will continue with the next step.

Interview Questions:

Upon agreeing with the consent, we begin recording the Zoom session and begin asking participants questions, and probing into their practices and perceptions. The questions to be

asked in this interview will be from the following (although not specifically in the same order) list of questions:

1 – Users Interview Questions

Demographics and domain

1. What is your gender? (you can skip this question)
2. How old are you? (you can skip this question)
3. What is your highest level of education?
4. What do you commonly use a computer for?
5. Do you consider yourself a beginner or a professional in using computers to complete your tasks?

Software use

1. Tell me a little bit about your daily routine with software/computers (what type of content do you usually work on? e.g., graphic design, programming, financial data, spreadsheets, photo retouching, architecture, text, etc.)

2. What types of software do you work with on a day-to-day basis?

3. In general, are you able to find/ locate all of the needed functionality within the same application (e.g., Word) or do find that you use multiple tools at the same time (e.g., Word and Photoshop together)?

4. Think of a recent project/work that you finished. Was there a software tool that offers a variety of features, but that you only ever used for a few functions? Why do you only use those few functions?

5. How do you feel about a situation where a lot of applications are ingefrated (MS Teams). As opposed to limited functionality tools (Stickies or zoom)

6. In your work, you probably use many different software tools and switch between these applications. Can you think of why do you have to switch between these applications?What advantages do you think these switches cause?

   a. How does such switching make you feel?

7. When you use multiple tools, how do you move your content (data) between them? Are there examples you can think of that are easy or difficult?

8. When you switch from one application to another, are there situations where it takes a while for your data to be available in the following application?

9. Are there situations that you can think of when switching is challenging or easy and helps you?

10. Think of one of the primary software tools that you use. Is there a smaller software or a utility application you would like to see integrated within your primary tool? Why?

11. Think of two main tools that you continually use together. How would it feel if the two were merged into a single app? (What are those apps, why do you want not want to see them merged)

12. Can you think of situations where you know there is a way to accomplish a task in your current tool, but you choose to switch to a different one? Why do you do that?

13. Are there situations in which you know a better way but choose a sub-optimal tool? Why is that?

14. Prioritize these features when you choose a software tool for your work. (cite from some source) - (Value Framework Paper might have a good list)

    a. Price Point and Affordability
    b. Customizability and Flexibility
    c. Learning Curve
    d. Security of Data
    e. Customer Support
    f. User Interface
    g. Data transfer and Integration with Other Tools
    h. Word of Mouth and Branding

15. In your opinion, what could be improved to help you better navigate and switch between the different applications?

2 - Case Studies - Questionnaire

1. Who are your users? What levels of skills do they usually have?
2. How many different tools are usually involved in finishing a project in their domain?
3. What efforts do you do on your side to support switching between your application and other tools?
4. Can you think of situations where users know there is a way to accomplish a task in your current tool, but they choose to switch to a different one? Why do they do that?
5. Are there situations where you know users must switch to a different application to complete a part of the project?
6. What are some of the challenges of using your tool in conjunction with other software applications?
7. When users switch between your application and other tools, what benefits do they get?What are some of their challenges in doing that? (data, time, etc.)
8. Do you have an insight as to what would your users generally prefer? One large application that can address all your required functionalities, or would they rather have many small/specialized applications?
9. If your tool wasn't integrated (standalone), would you consider developing it as a standalone (integrated) software? Why?

10. Is there a limited-functionality software that you would like to see integrated into your tool? How about a larger tool that your software might get integrated into?
11. When you do an integration, does people's perception change? Are there users that are not happy with the decision of integration? Why?
12. What's the relationship between the price point and the feature expectation from users?
13. How do you see embedding a tool affecting your market position or branding of the tool?
14. How do you prioritize your decision-making for designing/developing your tool?
    a. Price Point and Affordability
    b. Customizability and Flexibility
    c. Learning Curve
    d. Security of Data
    e. Customer Support
    f. User Interface
    g. Data transfer and integration with Other Tools
    h. Word of Mouth and Branding
15. In your opinion what could be improved to help you better navigate and switch between the different applications?

Closing

The session will end by asking the participant for any further comments they might have.

Katika In-Context Follow-up Interview Questions - Non-Motion Designers

1. Show me what you did for the first task (replicating the given video). Did you manage to get the same results as you expected?

    a. What about the second task? What did you decide to create?

    b. From your perspective, how was your progress with both of the tasks?

2. Without Katika, what would be your process for creating these videos? What tools and methods would you use?

    a. Remembering those other approaches, how was your experience using Katika?What was the simple? What was difficult?

    b. Did you try the different features in Katika? [motion bundles, sequence editing, script writing, assets, etc.]

    c. Did you create these videos in one session or multiple sessions? If you used the tool over several sessions, did you notice any differences in usage or behavior over time?

3. **Novices:** How was your experience authoring your first motion graphics video with Katika? Can you see a path where you could embed Katika into your work?

4. Now that you've used Katika, would you want to reuse it for another project, or would you want to go back to your existing workflow? What would need to change?

5. Have you ever used automated/ AI-based tools to create videos/artwork? What's your experience with them? How does Katika compare? [examples such as synthesia.io/, lumen5.com/]

6. Do you have any other thoughts that you'd like to share?

Follow-up Interview Questions - Motion Designers

1. Show me what you did for the first task (replicating the given video). Did you manage to get the same results as you expected?

   a. What about the second task? What did you decide to create?

   b. From your perspective, how was your progress with both of the tasks?

2. Without Katika, what would be your process for creating these videos? What tools and methods would you use?

   a. Remembering those other approaches, how was your experience using Katika?What was the simple? What was difficult?

   b. Did you try the different features in Katika? [motion bundles, sequence editing, script writing, assets, etc.]

   c. Did you create these videos in one session or multiple sessions? If you used the tool over several sessions, did you notice any differences in usage or behavior over time?

3. **Experts:** To what extent could Katika be embedded into your daily workflow for quick content creation?

   a. To do that, what are some of its strengths or weaknesses? (content, motions, flexibility, customizability, Deadlines, UI, I/O, etc.)

4. Have you ever used automated/ AI-based tools to create videos/artwork? What's your experience with them? How does Katika compare? [examples such as synthesia.io/, lumen5.com/]

5. Do you have any other thoughts that you'd like to share?

Katika Follow-up Interview Questions

PRE

      How old are you?

      What is your gender?

      What is the highest level of education you have completed?

      What was your major?

      Have you ever created motion graphics videos by yourself?

            Yes (if yes -> direct to next questions 6, 7, 8, 9)

            No

            I'm not sure

      How many years of experience do you have in making motion graphics videos?

      How often do you create informational videos using motion graphics? •Frequently •
Occasionally

•Sometimes Rarely • Never

      How often do you create explainer videos using motion graphics? •Frequently •
Occasionally

•Sometimes Rarely • Never

      If you have answered "Yes" to either of the last two questions, how long do you think it
would take you to make an informational video from scratch?

POST

I felt empowered to author explainer motion graphics videos using Katika. Agree - Disagree

I felt that I could successfully author motion graphics videos using Katika. Agree - Disagree

    2. I felt that Katika offered me the freedom to learn and explore the process of authoring explainer

       motion graphics videos.

A great deal - Not At All

    3. I was satisfied with the information that the system gave me about where I was in the process of

       making an explainer motion graphics video.

Very Satisfied - Very Dissatisfied

    4. When looking for different features and functionality, I found the user interface of
Katika to be: Very High - Very Low

~~Very Intuitive - Intuiti~~ve - Neither Intuitive nor Confusing - Confusing- Very Confusing

5. ~~Overall, I found Katika to be the system flexible and efficient to use.~~
Overall, I found that Katika gave me different options for experimenting with the creation of motion graphics videos.
Agree - Disagree

6. I would use the collaborative aspect of Katika (e.g., commenting, editing projects with others)

   when authoring motion graphics videos?
Very Likely - Very Unlikely

7. How likely would it be for you to recommend Katika to others?
Very Likely - Very Unlikely

8. Overall, how useful was Katika in creating an explainer motion graphics video?
Extremely Useful - Not Useful At all

9. How likely would it be for you to reuse Katika in the future to create other motion graphics

   videos?
Very Likely - Very Unlikely

Katika in-context - All Questions

Initial Questionnaire

1) How old are you?
2) What is the highest level of education you have completed?
3) What was your major?
4) Have you ever created motion graphics videos by yourself?
    a) Yes (if yes -> direct to next questions 6, 7, 8, 9)
    b) No
5) How many years of experience do you have in making motion graphics videos?
6) How often do you create informational videos using motion graphics?
    a) •Frequently • Occasionally •Sometimes Rarely • Never
7) How often do you create explainer videos using motion graphics?
    a) •Frequently • Occasionally •Sometimes Rarely • Never
8) How familiar are you with the underlying concepts of motion graphics authoring?

Daily Survey Forms

I felt empowered to author explainer motion graphics videos using Katika.
- ○ Agree - Disagree

I felt that I could successfully author motion graphics videos using Katika.
- ○ Agree - Disagree

I felt that Katika offered me the freedom to learn and explore the process of authoring explainer motion graphics videos.
- ○ A great deal - Not At All

I was satisfied with the information that the system gave me about where I was in the process of making an explainer motion graphics video.
- ○ Very Satisfied - Very Dissatisfied

When looking for features and functionality, I found the user interface of Katika to be:
- ○ Very Intuitive - Intuitive - Neither Intuitive nor Confusing - Confusing- Very Confusing

Overall, I found that Katika gave me different options for experimenting with the creation of motion graphics videos.
- ○ Agree - Disagree

Was there something that didn't work up to your expectations today?
- ○ (provide a short answer)

Do you have any recommendations about the features that you had to use today?

10. How likely would it be for you to reuse Katika in the future to create other motion graphics videos?
    - ○ Very Likely - Very Unlikely