



HAL
open science

StreamSketch: Exploring Multi-Modal Interactions in Creative Live Streams

Zhicong Lu, Rubaiat Habib Kazi, Li-Yi Wei, Mira Dontcheva, Karrie Karahalios

► **To cite this version:**

Zhicong Lu, Rubaiat Habib Kazi, Li-Yi Wei, Mira Dontcheva, Karrie Karahalios. StreamSketch: Exploring Multi-Modal Interactions in Creative Live Streams. CHI 2021 Conference on Human Factors in Computing Systems, May 2021, Yokohama (virtual), Japan. hal-03155671

HAL Id: hal-03155671

<https://hal.science/hal-03155671v1>

Submitted on 12 Mar 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

StreamSketch: Exploring Multi-Modal Interactions in Creative Live Streams

ZHICONG LU, University of Toronto, Canada

RUBAIAT HABIB KAZI, LI-YI WEI, and MIRA DONTCHEVA, Adobe Research, USA

KARRIE KARAHALIOS, University of Illinois at Urbana-Champaign, USA

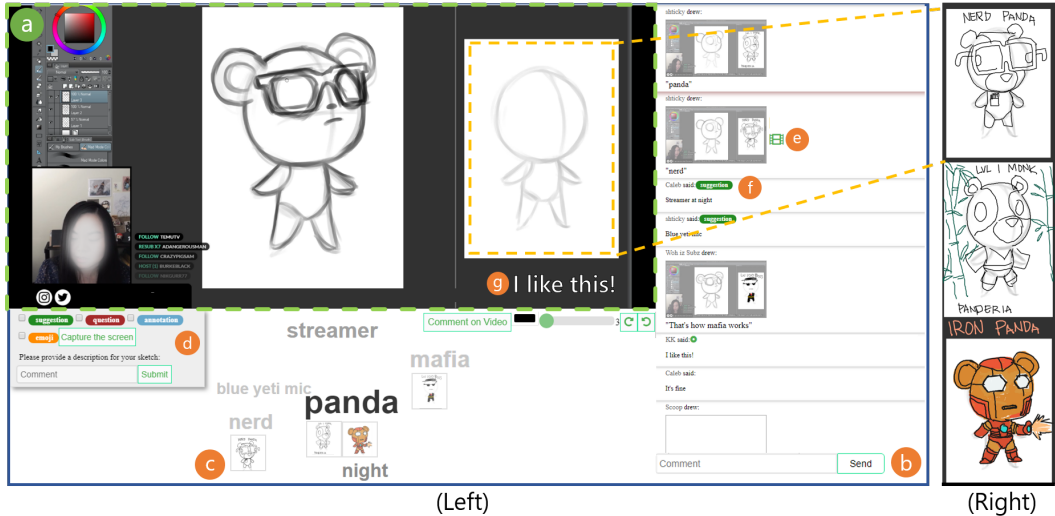


Fig. 1. Within StreamSketch’s user interface (Left), a streamer can share the progression of their artwork in real-time (a) and viewers can simultaneously comment and sketch over top of the streamer’s work in a web browser (b). In this example, the streamer began with a skeletal sketch (depicted by the yellow dashed rectangle) and asked her viewers to create their own variations (Right) using this starting point. To help with filtering and clustering (c), viewers can also tag their drawings using category tags or textual descriptions (d). All viewer sketches and comments can be spatially and temporally anchored (e, f, g) to facilitate browsing.

Creative live streams, where artists or designers demonstrate their creative process, have emerged as a unique and popular genre of live streams due to the real-time interactivity they afford. However, streamer-viewer interactions on most live streaming platforms only enable users to utilize text and emojis to communicate, which limits what viewers can convey and share in real time. To investigate the design space of potential visual and non-textual modalities within creative live streams, we first analyzed existing Twitch extensions and conducted a formative study with streamers who share creative activities to uncover key challenges that these streamers face. We then designed and implemented a prototype system, StreamSketch, which enables viewers and streamers to interact during live streams using multiple modalities, including freeform sketches and text. The prototype was evaluated by two professional artist streamers and their viewers during six streaming sessions. Overall, streamers and viewers found that StreamSketch provided increased engagement and new

Authors’ addresses: Zhicong Lu, University of Toronto, Toronto, ON, Canada, luzhc@dgp.toronto.edu; Rubaiat Habib Kazi; Li-Yi Wei; Mira Dontcheva, Adobe Research, USA; Karrie Karahalios, University of Illinois at Urbana-Champaign, USA.

© 2021 Copyright held by the owner/author(s). Publication rights licensed to ACM.

This is the author’s version of the work. It is posted here for your personal use. Not for redistribution. The definitive Version of Record was published in *Proceedings of the ACM on Human-Computer Interaction*, <https://doi.org/10.1145/1122445.1122456>.

affordances compared to the traditional text-only modality, and highlighted how efficiency, moderation, and tool integration were continued challenges.

CCS Concepts: • **Human-centered computing** → **Collaborative and social computing**; *Collaborative and social computing systems and tools*; • **Applied computing** → *Fine arts*.

Additional Key Words and Phrases: Live streaming, user engagement, sketching, visualization, creativity

ACM Reference Format:

Zhicong Lu, Rubaiat Habib Kazi, Li-Yi Wei, Mira Dontcheva, and Karrie Karahalios. 2021. StreamSketch: Exploring Multi-Modal Interactions in Creative Live Streams. *Proc. ACM Hum.-Comput. Interact.* 5, CSCW1, Article 58 (April 2021), 26 pages. <https://doi.org/10.1145/1122445.1122456>

1 INTRODUCTION

Due to the proliferation of live streaming platforms such as Twitch, many artists and designers have begun to adopt live streaming to showcase their creative processes [8, 28]. These *creative live streams* typically feature the streamer’s full screen of their computer or workspace (Figure 1a), a camera feed of their face, along with their narration while working [8, 28]. For creative live streamers and viewers, live streaming enables anyone to learn and be inspired by a streamer’s creative process, engage with each other during real-time shared creative activities, and connect with other creative professionals around the world [8, 28]. However, streamer-viewer interactions on most live streaming platforms today are confined to chatting with text and emojis [2, 27, 28, 52].

One way to better engage viewers in a live stream is to enable multi-modal communication through audio, video, and images [2, 19]. These rich modalities have been adopted in various genres of live streams including video gaming [27], live events [19], and learning [2, 18], and have been shown to effectively improve viewer engagement with live streams. However, video and audio may be intrusive and physically and cognitively demanding for both streamers and viewers, while image has limited flexibility [2]. The present work aims to explore how *sketches* and *text* can be incorporated into creative live streams to enrich viewer-streamer interactions. Sketching, as a primitive form of human expression, has been shown to support *visual thinking* to help organize one’s thoughts and improve one’s ability to think and communicate [1, 14]. Sketching is also a flexible form of visual communication that is familiar to creative streamers and viewers [8]. Integrating *sketches* and *text* into a live streaming interface may be worthwhile because multi-modal messages can convey richer information and enable new forms of expression during creative live streams. For example, a sketch can depict illustrative ideas such as composition and patterns, while such information is hard to convey through text only. Although sketching may be a more natural method of viewer-streamer interaction within creative live streams, design challenges and affordances of such modalities are relatively unexplored within the live streaming literature.

Through a formative study with creative live streamers and an analysis of available visual interaction modalities for streamers to engage viewers, the present work found that text-only chat and emojis impose constraints on what viewers can convey and share with streamers and other viewers in real time. For example, when trying to ask specific questions about the content in a live stream, viewers could not accurately or conveniently make reference to objects or points of interest. When making a suggestion to streamers about a drawing, viewers can only describe it textually, rather than visually, with illustrative details and examples. Based on the results of the formative study and the analysis, we designed and built a system, *StreamSketch*, as a technological probe.

StreamSketch enables viewers to engage with streamers through multiple modalities of input (i.e., text and sketch), and aids streamers and viewers in making sense of this rich data. With StreamSketch, viewers can sketch and place text comments directly into the live stream video to convey their thoughts, suggest ideas, express opinions, and annotate the video (Figure 1a). Sketches

and texts can also be anchored both spatially and temporally. These messages appear in a chat window (Figure 1b) as part of the chat conversation and are visible to the streamer and all of the viewers. Since the amount and complexity of multi-modal messages can be overwhelming, viewers can tag their text and drawings to facilitate filtering and clustering for increased visual clarity (Figure 1d). To provide an overview of the multi-modal chat conversation, StreamSketch displays a Hybrid Cloud visualization below the live streaming video that visualizes keywords from viewer comments and relevant sketches in real time (Figure 1c). To provide richer and more detailed contextual information to the streamer and their viewers, StreamSketch displays sketches and text alongside relevant contextual information (i.e., a snapshot of the video and position of sketches).

To better understand how the use of temporally and spatially anchored sketches and text facilitates new types of interactions and degrees of engagement between a streamer and their viewers, and how streamers and viewers navigate such rich, multi-modal data during live streams, we have conducted a user study with two professional creative streamers and their viewers (N=18) for six live streaming sessions that lasted up to an hour each. The findings demonstrated that viewers used sketches for interactions that were more relevant to art or self-expression and largely used text chat for socializing or non-art topics. However, the time required by viewers for ideation and sketching during the stream made sketching challenging for some. Both streamers and viewers also reported increased engagement while using StreamSketch.

This paper contributes:

- insights into the needs for supporting multi-modal interactions in creative live streams drawn from the literature, an analysis of existing tools, and formative study with streamers
- introducing spatially and temporally anchored sketches with text and labels
- the design of StreamSketch, a system which enables viewers to sketch over top of a live stream video while watching the live stream and supports the sense-making of multi-modal input from viewers
- evaluations with real users (i.e., streamers and viewers) to test the usability and provide more insights into future design of multi-modal viewer input within live streaming

2 RELATED WORK

2.1 Engaging Viewers through Live Streaming

Prior work has found that interaction and sociability are what make live streaming engaging [7, 16, 17, 31], and that streamers and viewers desire for deeper and richer interactions during live streams beyond text-only chatting [31]. To enable such interactions and encourage viewers' active participation during live streams, streamers, streaming platforms, and researchers have explored the use of various modalities of communication within live streams.

Many of these explorations have occurred within the domain of video game live streaming. Streamers on Twitch, for example, often use third-party tools or extensions to conduct polls during their live streams and make decisions about gameplay action using the polling data to satisfy more viewers [17]. For example, Lessel et al. designed several communication channels for live streams of the card game *Hearthstone* that enabled viewers to draw lines on the game UI to indicate their preferred player action [27]. They found that these additional communication channels were preferred because they enabled viewers to exert influence over streamers. Glickman et al. explored the design challenges inherent in live streamed audience participation games, and identified six challenges to streamers including latency, screen sharing, attention management, player agency, audience-streamer relationships, and shifting schedules [13].

Several projects have explored enriching communication channels in other genres of live streams, including live events, public environment, education, and knowledge sharing. Hamilton et al.

introduced push-to-talk audio input into their streaming interface for live events, and found that its instant and high-profile nature engaged viewers [19]. Reeves et al. studied a mixed reality game that supported live streaming from city streets and identified challenges facing streamers such as demands from remote viewers, situational challenges, and self-presentation desires [36]. Chen et al. explored the use of text, audio, video, image, and stickers to enrich interactions while live streaming for language learning, and found that these multimedia tools increased learners' engagement [2]. LiveMâché supported the sharing of context and participation in online learning through the collaborative and synchronous curation of multi-media, including live video, text, images, and sketches [18]. Snapstream enabled viewers to take snapshots of live streams for visual art, annotate them with basic shapes, and share the annotated snapshots with other users [52]. Although serving a similar group of live streaming users, StreamSketch differs from Snapstream in that it supports viewers in expressing themselves using free-form sketching rather than through annotations with simple shapes. StreamSketch also provides an overview visualization of all the viewer input for viewers and streamers to make sense of.

These prior projects have demonstrated that supporting multi-modal interaction during live streams is favorable, engaging, and beneficial to both streamers and viewers. To continue this line of research, we focus on a specific genre of live streaming – creative live streams. Viewers of creative live streams have different motivations and practices, and thus streamers have to use different strategies to engage viewers [8, 28]. With the ability to use sketch and text as communication mediums, viewers can potentially organize their thoughts better [1], communicate in a more expressive way, and thus engage more with creative live streams.

2.2 The Challenges with Text-Based Interaction

Live streaming has utilized text chat as its dominant input and feedback modality because text chat enables people to socialize with others, exchange information, and form communities when watching live streams together [17, 31, 42]. Chatting using a text back-channel when watching video together has also been found to lead to higher engagement with video content and better impressions of other participants [48]. Danmaku, a form of text commenting that is overlain on the live stream video, has been widely adopted on East Asian video sharing and live streaming platforms [29]. Danmaku has been shown to increase user engagement and affect user emotions and judgements [29, 50], though they are seldom adopted on Western platforms such as Twitch.

Text chat, however, does come with a number of drawbacks that have been identified within many Computer-Mediated Communication (CMC) tools. For example, managing a large volume of text messages has been a challenge for various text chat-based systems such as group chat for work [54]. With respect to live streams, text chat may become unusable or a distraction due to the real-time nature of the commenting and the fact that comments may be made by thousands of viewers at the same time [32]. Viewers joining such large scale streaming channels have also been found to be less willing or able to contribute and thus lurk more [33].

Keeping up with chat messages during highly popular live streams is especially challenging for streamers, because they have to make sense of viewers' preferences and reactions and adjust their content to better engage viewers in real time [31]. However, several tools and systems have been designed to address this. For example, Conversational Chat Circles [32] displays a subset of the chat stream to different subgroups of viewers, so that the number of messages sent to an individual viewer is smaller. StreamWiki [30] leveraged viewer-sourcing to summarize content and comments during knowledge sharing live streams. StreamSketch continues this line of research by exploring how massive amounts of information can be understood when there are multiple modalities of information in a live stream.

2.3 Multi-Modal Interaction in Online Communication and Remote Collaboration

Research in CMC has shown that text-based CMC lacks nonverbal cues that is inherent in face-to-face communication, which reduces the capacity of text-based CMC for deep interpersonal interaction [46]. To address this, several different modalities of communication have been proposed to improve CMC, including voice (audio), video, pointing gestures, and sketches.

The audio modality has been shown to be preferred over text when people chat with remote friends or family members while co-watching online videos [47] or interactive TV [11]. However, when incorporated in live event streams, it is not used by many, mostly because of concerns about interruption or trolling and inappropriate behavior encouraged by audio [19]. Videos have been shown to encourage communication [2], improve responsiveness [43], and elicit authentic and engaging conversations [45]. However, when incorporated in language learning live streams, it is rarely adopted because of the required time and efforts for making videos [2]. Because of the aforementioned challenges of audio and video modalities, they are not incorporated in StreamSketch.

Research in remote collaboration has explored various technologies that support sharing multi-modal communication cues (e.g., pointing gestures and sketches) between a local worker and a remote expert to improve performance. For example, Fussel et al. integrated a virtual pointer into a shared live video where the remote expert could sketch on to support remote collaboration [9, 34]. Kirk et al. proposed a method that captured the remote user's hand gestures and projected them into the local user's workspace [23]. Videodraw enabled users to sketch on their local displays and share their work by projecting it into the remote user's workspace, which worked in a symmetric manner [41]. The evaluation of these methods showed that sketches could be more descriptive and lasting in sharing information than a pointer, while a pointer facilitates quick participation [22]. Combining multiple visual cues (e.g., sketch and hand gesture) was shown to enable richer interactions [20]. Drawing inspiration from these methods, we focus on enabling multi-modal communication in creative live stream, a *one-to-many, asymmetric* communication channel between the streamer and viewers, which is different from most tasks and contexts explored in prior research.

StreamSketch also draws inspiration from prior research in video interfaces for online learning to complement text-based interaction during live streaming, including temporal anchored peer comments [25], spatio-temporal anchored collaboration [5] and references [12], fluid video annotation interaction techniques [35, 39], video collaboratory [38], and multi-modal referencing [4, 53]. These projects were mostly designed for asynchronous use, however, StreamSketch aims to explore the design space of multi-modal interactions in *synchronous* and *real-time* settings.

3 MULTI-MODAL INTERACTIONS DURING LIVE STREAMS: MOTIVATIONS & CHALLENGES

Creative live streams are mostly seen on platforms such as Twitch, YouTube Live, Behance, and Picarto. For example, Twitch has several distinct live stream categories that are closely related to creative activities, such as *Art, Makers & Crafting*, and *Music* [8]. Under the Art category, there are over 300 live streams and over 6,000 viewers at any moment [8]. Unlike other platforms where live streaming interfaces are almost non-customizable, Twitch enables streamers to use Twitch extensions to enrich the ways in which viewers can interact with them. These extensions are widgets, panels, or overlays that are developed by streamers, viewers, developers, and companies for use by streamers during their live stream. To better understand how viewers currently interact with streamers using modalities beyond text commenting, we reviewed a set of 111 existing Twitch extensions that were designed to improve viewer engagement in June 2019 [44]. At the time of the analysis, there were 122 extensions designed for viewer engagement that were available for Twitch users, however, 11 extensions were excluded from the analysis because they were

designed specifically for video game live streams and contained video gaming elements that were not generalizable to creative live streams. Among the 111 Twitch extensions that we analyzed, 41 (37%) extensions were related to interactions using visual input or output, 18 related to community building and interactions, 9 supported audio-related interactions, and 8 were for voting. Due to the present focus on multi-modal input, the 41 Twitch extensions that were about visual input and output were further analyzed.

3.1 Existing Visual Interaction Modalities in Live Streaming

Through the analysis, a diverse range of Twitch extensions were found to support visual input and output to engage viewers, including *graphic symbols*, *interactive overlays*, and *collaborative interaction space*. Almost all of such extensions overlaid visual input on top of the live video. However, they varied in level of interactivity they supported.

3.1.1 Graphic Symbols. The most common category of extensions supported the use of pre-defined graphic symbols for self expression, such as emotes, GIFs, emoticons, and stickers, similar to other social media platforms. Extensions that enabled viewers to apply animated visual effects to a live stream were also available, e.g., fireball, rain, throwing virtual objects to a streamer, applying masks or filters to a streamer's face, etc. These extensions enabled viewers to request visual effects to be overlain on the live video and visible to all users. Although streamers can customize the visual representations of these elements to some extent, viewers could only choose from a predefined set of visual elements. Most of these extensions, except for a few that enabled animated visual effects, did not allow viewers to specify where they wanted the visual elements to be shown on top of the video, providing limited *flexibility* in spatial anchoring.

3.1.2 Interactive Overlays. Extensions supporting interactive overlays enable streamers to make certain part of the live video 'clickable'. They thus allow streamers to dictate how viewers can interact with their video according to where they click. For example, Heat [10] enables streamers to show a real-time heat map of viewer clicks on the video, or show special visual effects at locations where viewers have clicked on the video, providing a novel way for viewers to interact with streamers and other viewers. Though providing more flexibility in spatial anchoring, these extensions fall short in terms of their *expressiveness*, because most interactions are designed on a case-by-case basis. The interactions are mostly ephemeral, which can cause *context loss* for streamers and other viewers if they have missed the moment the interaction took place.

3.1.3 Collaborative Interaction Space. Four extensions enabled viewers to collaboratively create art, for example, enabling them to draw pixels on top of a live stream or within a certain area of the stream, change the color of a matrix of tiles to design shapes or patterns, or draw directly on a canvas together. These extensions support better expressiveness and encourage collaboration, however, because they are a shared collaborative canvas, individual viewers can easily lose *agency* over their creations. Also, they do not have control over how other viewers collaborate, so the results of collaboration could be uncertain.

The analysis demonstrated that interactivity, expressiveness, flexibility, context, and agency all play an important role in viewer engagement. Enabling viewers to sketch or use pixel art to present themselves during live streams seems promising, especially for artists and designers who have experiences in these activities. However, because they regard highly of agency and expressiveness, the design of current visual input extensions on Twitch do not fit their needs well.

3.2 Formative Study with Creative Live Streamers

To further explore the potential of supporting visual input during creative live streams, informal interviews and observations of live streams and Discord channels were conducted with four creative live streamers. The four streamers included a freelancer (S01), an illustrator (S02), a web comic author (S03), and a concept artist (S04). Each interviewee had experience streaming on different platforms such as Twitch, YouTube Live, and Behance, and they regularly watched other creative streamers or moderated for them. Thus, their input can provide insights into viewer's perspective. The interviews focused on visual modalities to gain deeper insights into the design space. Interviewees were shown a low-fidelity sketch of a streaming interface that supported free-form sketching to gain their feedback about this idea. We identified several key themes regarding viewer engagement and viewer-streamer communication during creative live streams.

3.2.1 F1: Visual Input is Preferred over Text-Only Chat for Engaging Viewers. Creative live streamers and viewers preferred expressing themselves using visual language, or at least combining visual and textual information together, because they felt that visual input could be more expressive, e.g., *"During live streams, sometimes I prefer drawing a sketch and narrate while I am drawing to tell a story to my viewers. It just feels more natural to me"* (S03). Engaging viewers in a more expressive way through visual input would also be their *"competitive edge of being a streamer"* (S01), motivating more viewers to watch their streams. However, although some aforementioned Twitch extensions support visual input, streamers found it hard to integrate them in their streams, e.g., *"I think existing extensions are disconnected with my stream. Viewers use the tools, mostly in a separate window or website, to create their drawings, and then send them to me. It takes their attentions away. I'd like them to be fully engaged with my stream"* (S02). With current streaming interfaces, most viewers can only use emotes, emoticons, and stickers overlain on the video to express nuanced emotion, which have limited expressiveness, e.g., *"Sometimes when I want to express myself when watching a stream, I just wish I could draw something out myself instead of choosing from the list of emotes"* (S02).

When asked to reflect on how useful sketch input would be for their live streams, the streamers came up with several situations where visual input might be useful. They noted that viewers could provide detailed *ambient suggestions* during streams via visual input, e.g., drawing something interesting, helpful, and relevant to the streamer and other viewers (S02, S04). For example, viewers could draw perspective guidelines to indicate their preferred composition, and beautiful patterns for streamers to adopt in their drawing. Viewers could also annotate using visual input, e.g., pointing to or circling an object in the live stream, and asking specific questions or giving suggestions.

3.2.2 F2: Creating a Real-time Space for Sharing Ideas and Feedback. The interviewed creative streamers hosted design or art challenges regularly in their Discord channels. For example, S04 invited his viewers to submit their early-stage drawings or even rough sketches on different topics (e.g., fruits) every week. He then commented on the submissions from viewers on Discord. They noted that integrating such design/art challenges and critique into their live streams in real time would improve viewer engagement, however, no existing tools could readily support it, e.g., *"It'll be great if viewers submit their cool ideas in real time during my streams, and I comment on them. Other viewers can benefit from this as well. They can even build on each other's ideas"* (S03). S04 noted that when he wanted to reward his viewers by drawing customized illustrations that met their interests, he would like to collect more viewer input about their preferences, e.g., *"If they [viewers] can annotate on what I am drawing and give me feedback, I think my drawings will better satisfy them, and also mean more to them"*.

3.2.3 F3: Understanding the Context of Viewer Comments is Hard. The interviewees mentioned that when viewers ask questions or provide suggestions and feedback, they often refer to a point, an area,

an object, or part of an object within the artwork the streamer is working on, similar to findings in [52]. However, viewers can only make references via text commenting during live streams, which makes it hard for streamers and other viewers to understand the exact thing viewers are referring to, e.g., “*sometimes my viewers comment on specific details of the comic I am drawing, but I cannot accurately understand what they refer to*” (S03). For some streamers, sometimes when they want to have high productivity, they focus on their artwork during the stream and only occasionally pay attention to viewer comments (S01, S04). S01 noted that because the artwork changes constantly during the stream, by the time they see viewer comments referring to a part of the artwork, the artwork may have already changed, leading to *context loss*, similar to prior findings [30, 52].

3.2.4 F4: Making Sense of Viewer Contributed Content and Feedback. Streamers also noted there might be several challenges regarding the sensemaking of visual input from viewers. The volume of visual input and text comments can be overwhelming if there are many submissions from viewers, which is especially difficult for streamers if they want to focus on their work (S01, S04), e.g., “*Even if I do not have a lot of viewers, sometimes when I focused too much on my work and missed a lot of chat, I found it hard to respond to many comments*” (S04). Making sense of a large amount of viewer input can be more challenging because of the context loss, e.g., “*Sometimes there are just too many comments talking about different things, or referring to different parts of my work, for example, the color, the shape, the stroke...*” (S03). It is also hard for streamers to get a sense of the common themes and interests of viewers when there are too many comments during their streams (S01).

3.3 Design Goals

Based on our analysis, formative studies, and literature review, we identified the following design goals to guide the development of a system that supports visual input during live streaming :

- G1** *Support free-form sketching for richer expression and interactivity during streams.* Streamers value the interactivity during creative live streams and thus would like to incorporate visual input to enrich viewer-streamer interaction [F1] and to create a real-time space for sharing ideas and feedback [F2] within creative live streams. Sketching is a flexible and natural form of visual input that is familiar to creative live streaming communities [8]
- G2** *Provide spatial and temporal contextual information.* Streamers and viewers should be able to understand the context of viewer’s multi-modal input [52, F3], i.e., where the viewer is referring to and what the artwork was like when the viewer submitted their input.
- G3** *Facilitate the navigation and exploration of multi-modal information.* Streamers and viewers should be able to navigate, explore, and make sense of viewer input during live streams, and should not be overwhelmed if there are too many multi-modal messages [F4].

4 STREAMSKETCH USER INTERFACE

Based on the findings from the formative study and the analysis of existing Twitch extensions, StreamSketch was designed to support creative live streamers in engaging viewers using two modalities – sketch and text. StreamSketch consists of three main components that are always visible – a video player that shows the live video of the streamer’s artwork (Figure 1a) with a canvas overlain on it to support free-form sketching input from viewers (Figure 1a), a comment window that displays all of the text and sketches added by viewers (Figure 1b), and a Hybrid Cloud visualization that depicts keywords and sketches from the viewer submitted content (Figure 1c). Herein, StreamSketch’s key features and components are described, in addition to the design process and rationales that were followed. Within the prototype, the same interface is displayed to both the streamer and viewers to maintain the consistency of what they see on the interface.

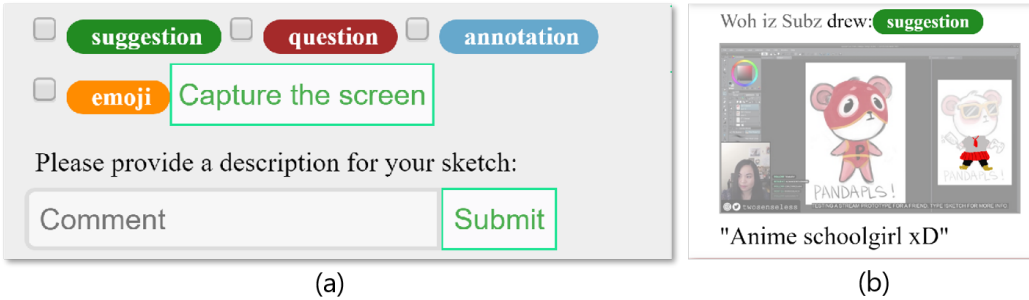


Fig. 2. (a) The pop-up window to collect tags and descriptions of a sketch. (b) An example of a sketch with a video snapshot that was captured, a description, and a tag ('suggestion').

4.1 Workflow when Sketching on a Live Video

To enable viewers to communicate using richer modalities (G1), StreamSketch supports *spatially* and *temporally anchored* sketch and text input (G2). When watching a live stream, viewers can sketch freely on top of the live video (Figure 1a) using a simple inking interface by clicking and dragging the mouse or other pointing devices of their choice. This enables viewers to sketch as on a layer of *tracing paper* over what the streamer is sketching live. The sketch is not synchronously shared with the streamer or other viewers before being submitted. While sketching, viewers can change the color, transparency, and size of the strokes. Before they submit their sketch, viewers can choose to provide some tags and/or more fine-grained textual descriptions of their sketch. After the sketch is submitted, it appears temporally overlain on the video to other viewers and the streamer (like laying the submitted tracing paper over the video), at the same position on top the live video (scaled if the size of the video is different), and will disappear after 6 seconds. The sketch, together with its tags and descriptions, will also be shown in the comment window (Figure 1b).

4.1.1 Viewer-Sourced Tags and Descriptions for Sketches. When the viewer is sketching on the video, a pop-up (Figure 2a) appears under the live video and enables the viewer to enter tags and/or descriptions about their own sketch. The names of the tags are customizable and can be changed by the streamer during their stream. Because free-form sketches can have high variance and be hard to interpret, sometimes streamers may have difficulty correctly understanding their meanings. Further, few mature AI tools are available to accurately analyze the theme or topic of sketches [15]. This pop-up is designed for viewer-sourcing more explicit, contextual, and structured information about their sketch, which can be leveraged to provide useful insights about viewers' sketches to streamers and to analyze viewer input (e.g., through analyzing tags and descriptions).

4.1.2 Spatial and Temporal Anchoring: Capturing Video Snapshots for Context. The formative study found that spatial and temporal anchoring of viewer input could be useful for streamers and viewers to understand contexts (G2), e.g., when what a viewer sketches is highly relevant to what is being shown in the video, or when a viewer annotates part of the video to ask questions or make suggestions. In this case, it is important to preserve the context and visual information so that the sketch can make sense when viewed by others. In another case, when the live video changes rapidly, it can be hard and distracting for viewers to sketch on it, so they may need to capture the current video as a static image and sketch on it.

To preserve the context for spatial and temporal anchoring, viewers can choose to capture the video frame by clicking the "Capture the screen" button on the pop-up. After the server finishes

processing the image the viewer captured, the image will be overlain on the live video and “screen captured” will be shown in place of the capture button on the pop-up, together with the option to hide or delete the captured image. When viewers do not actively choose to capture the video frame, the system still automatically saves a snapshot of the live video when a viewer submits a sketch on the video, in case it will be useful.

4.1.3 Spatially and Temporally Anchored Comments on the Video. Inspired by many existing viewer engagement tools in live streaming, especially Danmaku [30, 50], StreamSketch allows viewers to submit comments directly to streamers at a specific location in the video (G1, G2). Viewers can click the “Comment on video” button under the video player, and then click on a location in the video and type in the pop-up window to post a comment. Once submitted, the comment will be shown to all the users and last for 6 seconds (Figure 1g). Such comments also appear in the comment window with a green dot beside the author’s name to indicate that it is on the video. The video snapshot captured when the comment is submitted is also saved on the server.

4.1.4 Fork and Sketch on Existing Sketches. To further enable viewers to interact through visual input during live streams (G1), viewers can “fork” existing submitted sketches by other viewers by clicking on the + button shown beside an existing sketch when their cursor hovers over it. Once the button is clicked, the existing submitted sketch appears on top of the video, together with the snapshot captured when the sketch was submitted (if any). Viewers can then sketch on top of the video and the existing sketch in the same way as they sketch from scratch. They can also provide tags and descriptions for their new sketch, capture a new snapshot, or use existing snapshots through the pop-up window.

4.1.5 Contextual Information Available On-Demand. To provide viewers and streamers with more contextual information about viewer input (G2), when navigating the comment window, different information can be shown on-demand. When the mouse cursor hovers over a sketch or comment in the comment window, the sketch or the comment appears overlain on the live video. If the sketch or the comment had a snapshot captured when submitted, the video snapshot also appears overlain on the live video. The live video and audio keep going on below the snapshot, so viewers can easily switch back to the live stream. Even for sketches that did not have a snapshot captured by the viewer when submitted, the streamer and viewers can still see the video snapshot that appears overlain on the video to gain contextual information by hovering over the filmstrip icon (🎞️) shown to the right of the sketch in the comment window (Figure 1e).

4.2 Visualizing Viewer’s Textual Input and Sketches

To facilitate the navigation and exploration of multi-modal viewer input during live streams (G3), a visualization is used to depict key topics/themes of viewers’ input to provide high-level information to both viewers and streamers. Similar to [6], the visualization evolves as the live stream continues and updates in real time when new information is added. StreamSketch analyzes the descriptions of the sketches and all the textual input from viewers to update the visualization in real time.

4.2.1 A Hybrid Cloud of Sketches and Text. To provide streamers and viewers with a medium to mitigate the overload multi-modal messages may cause, keywords from the viewers’ input and related sketches are combined to form a Hybrid Cloud (Figure 3a). The keywords are extracted from viewers’ comments and the descriptions they provided for when submitting their sketches. Although different visual encodings may be more useful or helpful, such a ‘light weight’ and straightforward approach has potential because one does not want to overload streamers with a more complicated visualization.



Fig. 3. In this example, the user clicked on ‘panda’ in the (a) Hybrid Cloud visualization and then on the (b) tag ‘suggestion’ to filter the comments about this sketch.

To visualize the comment and sketch description keywords, the frequency of the appearance of the keywords within a 5 minute time window is computed. The frequency of each keyword is mapped to the font color of the keyword in the Hybrid Cloud, with darker colors indicating higher frequencies. To avoid visual clutter in the visualization, only the top 20 keywords are displayed. The time window can be adjusted during the stream according to the streamer’s needs, e.g., they can change the window to 30 minutes if they want to reflect on viewers’ input in the past 30 minutes.

In addition to keywords, sketches that are related to the keywords are also visualized within the Hybrid Cloud to provide streamers with an overview of what sketches viewers have submitted and their themes. The descriptions of the sketches are analyzed in real time, and if certain keywords that are already in the visualization match the descriptions, the sketches will be shown adjacent to the keywords in the cloud. The most recent five sketches corresponding to each keyword will be shown, with the newest shown on the left. Sketches without descriptions are clustered under a ‘Misc.’ keyword within the cloud.

4.2.2 Overview, Details, and Filtering. To enable viewers and streamers to navigate the details of individual comments or sketches (with spatial-temporal anchoring, G3), StreamSketch supports interaction between the Hybrid Cloud (Figure 1c) and the comment window (Figure 1b).

The keywords in the Hybrid Cloud can be used to filter the comment window by *keyword*. When the mouse cursor hovers over a keyword tag, the text chat and sketches in the comment window will only show comments with the keyword and sketches whose descriptions contain the keyword. When the cursor moves away, all the comments and sketches will reappear. This interaction technique enables viewers and streamers to quickly navigate the visualization. Streamers and viewers can also click on the keyword tag to apply a persistent filter to the comment window, allowing them to navigate the filtered comment window to find more detailed information (Figure 3b).

The tags provided by viewers can also be used to filter by tag. When the user clicks on a tag in the comment window, the comment window only show sketches and comments with the selected

tag. The tag and keyword filtering can also be combined such that streamers and viewers can see sketches and comments that have a specific tag and are related to a specific keyword (Figure 3b).

4.3 System Implementation and Setup

The StreamSketch system consists of a front-end web interface and a back-end server. The front-end web interface was built using React, Javascript, D3.js, HTML and CSS. The Twitch and YouTube Live APIs were used to embed the live video in the StreamSketch interface (Figure 1a) to provide a seamless experience for streamers to stream using a platform of their choice. The back-end server was built using Node.js and an Amazon S3 database. The front-end and the back-end communicate with each other using a WebSocket to transmit data and HTTP requests to access APIs. The web interface can run in modern web browsers including Chrome, Firefox, Safari, and Edge, which should enable many viewers to use StreamSketch. The keyword extraction feature was implemented with node-rake [37], an open source tool that implements keyword extraction algorithms within Node.js. Streamers use the drawing software of their choice to create art, and use the streaming software of their choice (e.g., Open Broadcaster Software) to broadcast their drawing process.

5 EVALUATION

StreamSketch was evaluated via deployment study of six streaming sessions to gain insights into the effectiveness and usefulness of the system. The goals of the study were to understand how viewers use temporally and spatially anchored visual and textual multi-modal input (i.e., sketch and text) to interact with streamers, and how streamers and viewers make sense of such multi-modal input during live streams. The technical and social challenges that arose when such input was made available were also of interest.

5.1 Participants

Two streamers (S1 and S2) who were frequent creative live streamers on Twitch were recruited through personal connections to participate in the study. S1 was a digital artist working on illustrations who normally streamed about digital painting and illustration and had over 7,000 followers. She had 4 years of experience with streaming and she usually streamed about 2 hours every day to 20-30 viewers. She usually mirrors her iPad screen to her PC screen and broadcasts through her PC. S2 also had over 4 years of experience with streaming and he usually streamed about 3 hours a day, 5 days a week to about 100 viewers. He had 23,000 followers and was a concept artist and illustrator who mostly streamed character illustrations using Adobe Photoshop and a Wacom Cintiq.

In addition to the streamers, 18 viewers were recruited to participate in the study. As reported in prior research, paid viewers from Amazon Mechanical Turk tend to behave differently than volunteers watching live events [40]. This work thus focused on recruiting viewers who were intrinsically motivated to watch creative live streams and engage with streamers' communities. Those who regularly watched creative live streams were invited to participate in the study via the invited streamers' Discord channels. Four of S1's regular viewers and five of S2's regular viewers accepted offers to participate. As the response rate was not high enough to recruit a significant number of viewers, other viewers were recruited via word-of-mouth and snowball sampling within the creative professional community. All the viewers had at least 3 months of experience with creative live streams and watched creative live streams at least once a week. Most participants were between the ages of 21 to 40, with 10 being male, 6 female, and 2 not disclosing their gender. Twelve voluntary viewers visited our website and interacted with StreamSketch freely during the sessions, although they were not asked to respond to the surveys after the sessions.

5.2 Method and Procedure

Similar to real-world creative live streaming [8], the study was a simulated situation with several ‘active’ viewers who participated by commenting and sketching when watching creative live streams and other viewers lurking or doing their own activities as they wish.

Previous studies have also indicated that only running the study once might not be able to gain enough insights, due to the ‘cold start’ problem (i.e., viewers might not participate using new interactions [30]). Thus, 3 sessions were run with each streamer, and viewers were also allowed to participate in multiple sessions if they would like. In the end, eight viewers joined only one session, eight viewers joined two sessions, and two viewers joined three sessions.

5.2.1 Pre-Study Training. Before the study, the research team demonstrated how to sketch on the video, how to provide tags and descriptions, and how StreamSketch’s visualizations worked to each streamer. This was done via video calls and a live remote demonstration. Before each session, participating viewers were given a demonstration of StreamSketch via video calls and then the research team shared a link to a 5-minute tutorial video that highlighted key features of the system. Viewers were then asked to visit the prototype website and were encouraged to freely explore the interface to get familiar with the key features of StreamSketch before the study.

5.2.2 Streaming Sessions. Before the first session, the streamers were instructed that they could stream using their normal topics and processes, such as streaming about character design, concept design, illustrations, etc. and asking viewers to give feedback or allow them to request something. The streamers were also encouraged to come up with their own tasks/strategies they thought might be appropriate and helpful to engage viewers using StreamSketch. Both streamers streamed for an hour for each session, with StreamSketch shown on a side screen. Participating viewers were asked to join the stream session using their own desktop or laptop computers and use StreamSketch to interact with the streamer or other viewers until the session ended. To allow a broader range of viewers to participate in the study, the types of pointing devices that viewers could use to sketch with StreamSketch was not restricted. Among all the eighteen viewers, five used a Wacom Cintiq tablet and the other thirteen used a mouse to sketch with StreamSketch.

5.3 Metrics

5.3.1 Post-Study Surveys. After each session, survey responses were collected about viewers’ ratings of the usability of various system features (i.e., sketching and text commenting on the video, the visualizations, and the filtering) using 5-point Likert Scales (i.e., 1-strongly disagree, and 5-strongly agree), their perceptions about the differences between watching live streams using StreamSketch and traditional live stream platforms, how StreamSketch influenced their watching experience, what they liked and disliked, and what they would like to improve about the experience. They were also asked to reflect on a sketch that they felt was most engaging and to provide more contextual information about that sketch through open-ended questions. The survey was modelled after questionnaires previously used by Lu et al. [30], Yang et al. [52], and Lessel et al. [27], which explored the usability and usefulness of novel live streaming interfaces. The questionnaire can be found in the supplementary material of this submission.

For those viewers who participated in multiple streams, only the survey responses from their first usage of StreamSketch were analyzed to ensure that their first impressions were captured. To provide richer insights about the user experience and potential improvements to StreamSketch, their responses to the open-ended questions were included in the analysis.

5.3.2 Interviews with Streamers. After each streaming session, the streamers were interviewed to gather feedback about StreamSketch. The interview questions probed how they interacted with

viewers using StreamSketch, how they used the visualizations and filtering, how they thought StreamSketch influenced the way they streamed and interacted, their positive and negative experiences with StreamSketch, and other thoughts and feedback on the system.

5.3.3 Analysis of Sketches. To understand how the viewers used sketches to interact in a more nuanced way, the research team also randomly sampled all the sketches during the 6 streaming sessions and yielded a sample of 80 sketches from viewers. Two researchers categorized all the sketches through a thematic analysis process. The recorded videos of the live streams and the descriptions provided for sketches were referenced during the analysis for context. If tags were provided by viewers, they were classified as self-reported categories. The two researchers first coded the sketches independently, and then discussed them together to reach consensus on categories. One researcher then coded the remaining sketches alone.

6 RESULTS

We now report on the main use cases of spatially and temporally-anchored sketches in creative live streams, and how StreamSketch enables different ways of interaction compared to how viewers and streamers interact with conventional live streaming interfaces. In general, both viewers and streamers noted that streaming sessions with StreamSketch were engaging.

6.1 Streamer's Strategies to Engage Viewers

Although the study did not have specific task requirements for streamers, S1 tried out different ways to engage with her viewers in her three sessions. During her first session, she started with a warm-up exercise. First she painted a random shape on the canvas and invited viewers to create sketches which built on the shape. She then drew half of a cartoon cat's head and asked the viewers to complete it. After this warm-up, she started an interactive "draw together" session, where she began to sketch a simple illustration from scratch, asking viewers to contribute ideas on what should be in the sketch. Her second session started with the same warm-up session, and she then started an interactive character design session, where she sketched the body and face of cartoon characters, and invited viewers to sketch facial expressions. Her third session was a 'paper doll' session, where she sketched a basic shape of a doll and asked viewers to sketch out how they thought it should look like. She then began to sketch out what she thought it should look like, taking viewers' suggestions into consideration and iterate on her sketch.

S2's sessions were more similar to one another. During his streams, he streamed about his process of designing fictional characters, and he asked viewers to contribute ideas about the visual design of the character, e.g., what shape should the head be, what color he should use, etc. He adopted a more iterative strategy, by first focusing on the rough shape of the character, and gradually focusing on facial and body parts, clothes, or accessories. He noted that he sometimes does streams exactly like this, though he only interacts with viewers through text comments.

6.2 Viewers' Use of StreamSketch

During the six streaming sessions, 30 unique viewers contributed a total of 451 sketches when watching live streams (Table 1). It took on average 72.60 seconds ($SD=132.01$) for viewers to finish a sketch that they submitted. In general, all viewers used both comments and sketches to interact during live streams. Considering the time and effort viewers must devote to sketching, the result is encouraging because it shows that viewers would like to use sketching as a primary channel for communication in creative live streams.

Overall, StreamSketch enabled viewers to interact with streamers and viewers in an engaging way, and supported their creative activities and idea sharing during live streams (Figure 4). Viewers

Table 1. Number of viewers and interactions per streaming session

Live stream session	S1-1	S1-2	S1-3	S2-1	S2-2	S2-3	All sessions
Total # of viewers using StreamSketch	9	9	12	11	8	8	30 (unique)
# of viewers responding to the survey	5	5	5	5	5	5	18 (unique)
Total # of viewers watching the stream	19	21	38	82	76	93	329
Total # of comments	111	132	104	121	178	130	776
Total # of sketches	74	56	70	87	92	72	451
Total # of sketches with tags	16	13	21	34	43	41	168
Total # of sketches with descriptions	31	35	42	64	73	51	296
Total # of sketches with snapshot captured	34	21	21	45	42	44	207

reported that using StreamSketch to watch creative streams and interact with streamers and viewers was fun and enjoyable ($Median=4$, $IQR=1$), and they felt more engaged in live streams using StreamSketch ($Median=4$, $IQR=1$). They also reported that sketching on the live video was helpful to communicate during live streams ($Median=4$, $IQR=1$), and they felt they interacted with the streamer more than they normally do ($Median=4$, $IQR=1$). Although it takes some time to master all the features of StreamSketch, viewers reported that they felt StreamSketch was easy to use ($Median=4$, $IQR=1$), easy to learn how to use ($Median=4$, $IQR=1$), and they would like to use StreamSketch when watching creative live streams in the future ($Median=4$, $IQR=1.75$). They noted that StreamSketch provided them with new experiences when watching creative live streams, e.g., “*With StreamSketch, I feel like the streamer interacts much more with viewers, the audience can contribute equally to the stream rather than being a passive recipient*”(V3) and “*It would be more actively engaging as opposed to something I might have in the background while I work on other things*” (V11). Several viewers who were less experienced in using digital drawing tools typically rated the ease of use of StreamSketch less than those dedicated and experienced users of digital drawing. They commented that sometimes they felt less need to provide tags or textual descriptions, or found it hard to sketch well with their mouse because they did not have other devices available. Two viewers rated ease of use low because of connection issues.

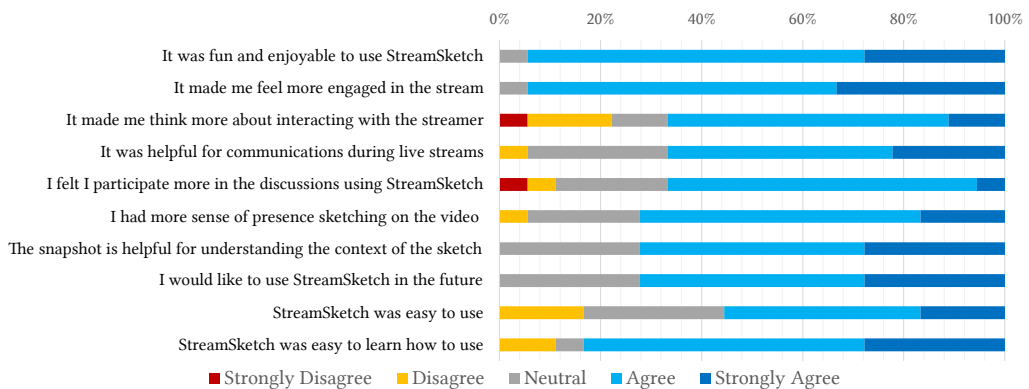


Fig. 4. Survey responses about the experience of using StreamSketch.

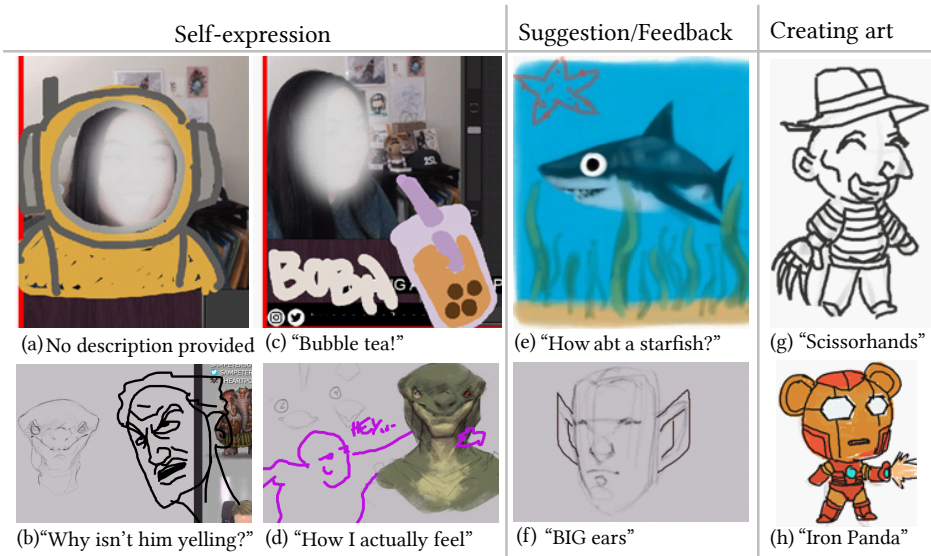


Fig. 5. Examples of viewers' sketches during the live streams: (a) a diving suit which the streamer could use as a 'costume', (b) a yelling face to make a joke, (c) a cup of bubble tea to treat the streamer, (d) a figure which leans on the character in the stream, (e) a viewer suggesting that the streamer sketch a starfish in the scene, (f) a viewer suggesting that the streamer sketch big ears for the character, (g) a character with scissorhands based on the 'paper doll' by S1, and (h) an 'Iron Panda' based on the 'paper doll' by S1.

6.3 The Use of Sketches in Creative Live Streams

The analysis of all the sketches (N=451) showed a variety of ways in which the sketches were used during the streaming sessions. Among all the sketches, 17 were illegible scribbles that were hard to categorize, and were thus excluded from the analysis. Generally speaking, sketches were mostly used for suggestions, self-expression, and collaborative art creation (Figure 5), while text comments were mostly used for chat and socialization, e.g., "Sketches are great for suggestions and explanations, but for more casual/chill talk, text seems better" (V11) and "Text works for stuff like coming up with names or such. Drawings are better for ideas" (V18).

6.3.1 Sketches for Self-Expression (N=203). The most frequent use of sketches was for self expression (Figure 5 a–d). Several viewers used sketches in a way similar to how they used stickers, but they enjoyed sketching because it provided them with more flexibility. Such sketches helped them express nuanced humor or personal meanings that were hard to convey through text only, e.g., "sketching is more personal compared to using text" (V10) and "Some humor can only be conveyed by drawing out rather than using text. The interaction with the streamer's drawings allows for some fun shenanigans" (V17). Several viewers also expressed their care and support for the streamer by sketching something with kind implications and used pleasant words as descriptions e.g., heart, thumb up, flowers, food, coffee, and bubble tea (Figure 5c). Other viewers expressed themselves by sketching an enhancement of the video rather than contributing to the concrete idea of the art work. For instance, several viewers drew costumes, sunglasses, hats, or accessories which were overlain on the camera feed of the streamer to make them seem 'cooler' or 'stylish' (Figure 5a). These sketches were usually submitted without a snapshot captured, so that they served as graphical overlays that streamers can 'interact' with during the live stream to create dramatic effects. A viewer in S2's

stream, for example, drew an alternated mascot overlain on the original one, expressing their wish for S2 to redesign the mascot because it had been there for too long.

6.3.2 Sketches to Provide Feedback and Suggestions (N=156). Viewers provided a variety of suggestions to streamers during the streaming sessions (Figure 5 e–f) due to the interactive nature of these sessions. They suggested different topics of content to sketch, e.g., accessories of a character, how to pose or position of the subject, the hairstyle of a character, background objects, etc. Some viewers also used annotations to make suggestions, for example, sketching a circle around certain area of the video and writing or typing text to make suggestions.

S2 mentioned that when using his traditional streaming platform, he found it useful for viewers to be able to provide him with suggestions directly on the content he is drawing. He noted that some of his viewers took a snapshot, edited it in Photoshop, and then sent it to him on Discord, in an asynchronous manner. He thought that StreamSketch had the potential to support this kind of interaction in a more real-time manner, and that he could change his drawing in real time and get feedback on the changes from viewers.

6.3.3 Sketches to Create Art Together and Share Ideas (N=75). Several viewers mentioned that StreamSketch transformed the live stream into a space to create art together (Figure 1 Right, Figure 5 g–h). They noted the benefits of getting inspiration not only from the streamer, but also from other viewers, e.g., “I prefer using sketches if I have an idea or picture in my head that I want to communicate so I can show it to others” (V3) and “Drawing a picture and posting it seems better for the whole chat. Everyone gets to participate in the drawing, since you can add on to drawings” (V16).

Both the streamers and viewers expressed that they would like to have the sketches created during the live stream archived or saved locally so that they could share them with other people, e.g., “I would like to save all the drawings from viewers on my PC and upload them to social media to share the ideas” (S1). These sentiments present an opportunity for live streaming services to develop better archiving methods for creative live streams and collective creative activities.

An unexpected use of StreamSketch was to recolor a work in progress sketch. For example, viewers in one of S2’s sessions discovered that StreamSketch’s palette supported changing stroke opacity, so all of the viewers began filling in the color of the characters that S2 was sketching to make new styles or visual feelings for the characters. They also used translucent strokes to build on other viewers’ sketches. This behavior suggests that it may be worthwhile to provide viewers with the opportunity to adjust more stroke attributes in the future.

6.4 Viewers’ Experiences with StreamSketch

6.4.1 Drawbacks of Sketching during Live Streaming. Although viewers used sketching for a variety of purposes, several limitations of sketching during live streaming were also found. The most critical problem with sketching might be with respect to timeliness, because it took viewers on average 72.60 seconds to finish and submit a sketch and they had to spend more time on coming up with what to sketch, e.g., “I sketched every 5–8 minutes. It takes around that time for me to think of an idea, plan how to draw it and actually draw it” (V3). This makes sketching a low throughput communication channel, limiting the potential it has to enhance communication. However, some viewers noted that the slow-pace nature of creative live streaming makes this practice more acceptable, e.g., “I think since the streamer is working on her drawing and does not reply in real time, it is fine that our sketches are not that timely” (V7).

6.4.2 Timeliness of Text Overlain on Videos. When timeliness is critical and when the information being provided is quite simple or close-ended, text was still the preferred communication channel between viewers and streamers, e.g., “for a simple question like “suggest a name”, writing is more

effective”(V13) and “text is used for making statements or answering non-art-related questions” (V2). Viewers also expressed their praise for other viewers’ creative ideas mostly with text comments, e.g., “text is more preferable when I comment on others’ work and I want to do it quickly” (V3).

Overlaying text comments on the video (Figure 1g) was still found to have a ‘cold start’ problem. During the first 2 sessions of both streamers, viewers seemed to be more engaged in sketching and few used it. Some even drew hand-written text on the video instead of using text comments overlain on the video. During the third session, S1 explicitly asked viewers to use text comments on the video to give her suggestions, thereby piquing viewers’ interest in using this feature. In this session, viewers said that “it is a convenient and visible way to make sure your idea was seen immediately”(V14). This shows the potential that free-form Danmaku might have to further engage viewers in creative live streams, though it is not widely adopted by streaming platforms today.

6.4.3 Free-form Personalization over Canned Responses. Although users can add tags to comments or sketches by typing a hashtag directly or choosing from a predefined list of canned tags, viewers generally preferred providing their own personalized descriptions (N=296) rather than simple tags to their sketches (N=168) (Table 1). “I always provide descriptions when submitting my drawings. I do it because if I don’t, others might not understand what I’m trying to draw as my sketches aren’t really that apparent on what they are”(V6). Because they already devote time to create sketches, they would like to use text to complement or enhance their communication, so it does not bother them to take a few extra seconds to provide more detailed text descriptions. As viewers were motivated to provide free-form text (i.e., descriptions for sketches) rather than structured text (i.e., tags), this additional data, while useful, is still hard to analyze using automatic algorithmic approaches, thus presenting an additional challenge to developers of live streaming platforms.

6.4.4 Video Snapshots Only When Necessary. Viewers were also found to be less likely to capture the snapshots than provide descriptions, with only 207 snapshots captured by viewers (Table 1). About half of the viewers expressed that if the snapshot was essential to understand the context of their sketch, e.g., when their sketch was built on the content in the video, they actively captured the snapshot to preserve the contextual information, e.g., “a screenshot allows me to add details and my own thoughts onto an existing sketch to create an image I want” (V4). However, when contextual information was not critical, they did not capture it, e.g., “When the sketch is not associated with the content on the screen, it doesn’t need the capture” (V7). V10 noted that sometimes his sketches related more to what the streamer was talking about, or the background music in the stream, rather than what was shown in the live video, so he chose not to capture a snapshot. V7, V10 and V13 also noted that they intentionally did not capture snapshots so that the streamer could ‘interact’ with the sketch or draw something new to respond to their sketches. For example, V10 drew a space suit overlain on the webcam feed of S1 and because the snapshot was not captured, it acted like a mask that the streamer could use to role play and perform with (through screen sharing), thus creating some dramatic moments during the stream (Figure 5a).

6.4.5 Making Sense of Multi-Modal Information. Viewers had mixed feelings about the visualizations and filtering. Some used the visualization feature during the streaming sessions because they perceived benefits in being able to use the visualization to navigate text and sketches to make them aware of other people’s thoughts, keep them updated about important information, and get an overview of comments and sketches, e.g., “Seeing sketch thumbnails mixed with the keywords makes it easy to browse and view them” (V18). They mentioned that keyword filtering in the Hybrid Cloud was useful because they were sometimes only interested in certain topics so filtering enabled them to focus on these specific topics. However, other viewers did not fully understand the visualizations and filtering, citing distraction and too much information as their main concerns. These viewers

only used the tag-based filtering when they were interested in certain types of information (e.g., suggestions). They noted that in a stream with more comments, such filtering might be useful, e.g., *“for more text-intensive stream, I may use visualization overview of the comments more”* (V17).

6.4.6 Tension between Versatility and Simplicity. Because creative live streams attract viewers with different experiences and expertise, it is challenging for streamers to satisfy the varying skill levels of viewers in one stream. The simple sketching functionality of StreamSketch was favored by several viewers who were amateur artists because it made them worry less about getting a full-fledged sketch and more about the creativity of ideas and sharing these ideas with others. The ability to capture snapshots and sketches based on the content in the snapshot or other viewers' sketches was also favored, e.g., *“I always capture the snapshot when sketching, because I need the background canvas to help me achieve what I want to say, as I can't really create art from scratch”* (V1). However, some professional artists who were skilled in using digital drawing tools were a little discontent with the inking limitations within StreamSketch. They desired more versatile sketching tools that would allow them to create sketches like they normally did within digital drawing software, e.g., *“I hope I can use some tools and brushes in Adobe Photoshop to sketch during the stream. For example, I wish there was an eye dropper tool”* (V9).

6.5 Streamers' Experiences with StreamSketch

Both streamers expressed that using StreamSketch to interact with viewers was a unique experience and they provided feedback on how to further improve its design.

6.5.1 Increased Interactivity through Multi-Model Viewer Input. They enjoyed viewers' active engagement during their live streams and appreciated the opportunity to gather inspiration from viewers. S1 was impressed by the engagement with their interactive character design session, e.g., *“Normally my streams are not as active and interactive as these sessions. The only interaction is just chatting. With StreamSketch they can draw and share their drawings, and participate more in my stream”*. She further added that,

“It actually ended up being really fun. It was unexpected. It felt like one of those paper dolls. When I had the base and then people kept adding like clothes, a hat, different things. And when I would ask them what color they wanted, they can color it in, just as an example to see what it would look like.”

S2 also expressed similar sentiment, and added that *“Viewers like the fun side of it, and I enjoy seeing how creative the viewers can get with StreamSketch. Even with a simple inking tool they can create amazing drawings”*. They also noted that some visual jokes (e.g., [Figure 5](#)) were fun and impressive, and the same effects could not be achieved using only text comments in their normal streams. StreamSketch was also found to help streamers present more content during streams to engage viewers, e.g., *“I think it adds a lot of fun to the stream, and a lot of content, which is what most streamers want. Entertaining content.”* (S1).

6.5.2 Influence on Inter-Personal Interaction. S1 also noted that she felt that the sketches sometimes revealed viewers' personalities, even more so than text, which could help her learn more about her viewers, e.g., *“Because I know some of my viewers well, I could see some patterns of their drawings and the nuances the drawings reflected about their preferences, tastes, or hobbies”*. The inter-personal interactions mediated through the sketches were also perceived to be unique by streamers and they wanted to be able to better archive them, e.g., *“it gives me another connection with my viewers. It's their personality. I'm proud of the drawings that they made, and I want to keep that somewhere.”* Even viewers who usually did not draw much felt motivated to sketch and engage with the stream,

largely influenced by the streamers, as noted by S2, “*I know some of my viewers didn’t draw on a normal basis. It was a cool experience to see them draw and engage in my stream*”.

6.5.3 Filtering Helped Streamers Make Sense of Viewers’ Multi-Modal Input. Both streamers thought filtering was useful, as noted by S2, “*there’s always chatting about random stuff. It’ll be cool to use that to sort the actual suggestions or questions from random chat*”. However, they preferred using discourse act, e.g., question, suggestion, request, etc., rather than keywords in the Hybrid Cloud, to filter the comments and sketches. The streamers seemed to be comfortable attending to a variety of different topics within their live streams, so they did not have strong preference for different topics in viewers’ comments. They cared more about the discourse within viewer comments because the discourse might be more actionable for them.

The visualizations of the sketches and text received mixed reviews. The streamers noted that the extracted keywords in the visualization might not be accurate, e.g., “*I think there were many irrelevant words in the visualization, which makes it less useful*” (S1). However, S1 did mention a situation when the visualization could be useful, e.g., “*I think when it’s for voting, probably the visualization is useful, because viewers can just type text to vote, and the visualization can make the winning one very visible with bigger size*”. Their mixed reviews about the visualization might have been due to the small number of viewers who actively participated in the streaming sessions and the time it took for viewers to finish their sketches.

6.5.4 Concerns about Moderation and Tool Integration. Streamers also expressed a few concerns about future adoptions of StreamSketch. Both streamers noted that moderation would be important because Twitch communities have trolls who might post inappropriate content [49]. As noted by S1, “*you’re going to get some artists that are more aggressive than criticism. They’re going to tell you like a teacher and mark your drawing all over the place, and say ‘you’re doing this wrong’*”. S2 thought that it might be better to have more delays before showing sketches to viewers so that moderators could decide which could be shown, and that preventative moderation might be more important. Another concern was that it might be hard to integrate such tools into the streamer’s workflow because they are already using a number of tools for streaming. Both streamers suggested that StreamSketch could be integrated into their existing streaming tools so that they could directly show interesting sketches to other viewers (e.g., those not using StreamSketch).

7 DISCUSSION AND IMPLICATIONS

The studies revealed a number of potential directions and challenges to further improve experiences when using sketching as an input modality during creative live streams.

7.1 Comparing StreamSketch with Other Systems

We first reflect on our results and compare with other relevant systems in HCI and CSCW. In the study of Snapstream [52], a snapshot-based interaction tool for creative live streams, many snapshots were used for referencing part of the artwork and making jokes with funny images, while only a small number of snapshots were used for providing suggestions. In our study, sketches with text were used mostly for self-expression, providing feedback or suggestions, and creating art together. There were also cases where viewers did not capture snapshots but used sketches as user-generated stickers that the streamer could ‘interact’ with. The different use cases brought by StreamSketch may result from the expressiveness and flexibility of sketches and the choice over whether to capture snapshots. In LiveMâché [18] which supports synchronous sketching interaction, sketching was not only used to illustrate ideas, but also as deictic gesture to call attention to objects in discussion, especially by instructors. Learners using LiveMâché may feel less comfortable to sketch when the instructor is sketching, because their sketching process is visible to others. In

StreamSketch, asynchronous sketching interaction makes viewers feel more comfortable using sketches to express themselves and share ideas. They can keep improving their own sketches until they are satisfied and then choose when to submit it to be seen by others. In sum, the *expressiveness*, *flexibility* regarding snapshots, and synchronous video interaction with *asynchronous* sketching interaction are what make StreamSketch different from other systems.

7.2 Balancing Versatility and Simplicity

Creative live streams attract viewers with diverse backgrounds and art skills, which introduces many challenges. Some professional artists preferred the sketching functionality of StreamSketch to be as feature-rich as possible, approaching the versatility of Adobe Photoshop and complete with layering, pressure sensitivity, different brushes, selection tools, etc. For these professional viewers, more feature-rich sketching functionality enables them to sketch in the same way as they normally work. However, this level of fidelity may discourage viewers who are not very skilled in using such tools because they may worry about the quality of their sketches and thus be reluctant to share any content or ideas. The interviewed streamers also preferred the user interface for viewers not to be too complicated so that viewers would not be distracted during their streams. As sketching tools may become even more integrated within streaming platforms, questions remain regarding how to engage both professional and casual viewers without discouraging professional viewers to utilize all of their skills and casual viewers to share ideas. A potential way to balance versatility and simplicity is to allow streamers to configure StreamSketch interface so that it can adapt to the specific needs of their streaming sessions. For example, when tags or textual descriptions of viewer sketch are unimportant or if viewers are unwilling to provide them, the streamer or viewers themselves could temporally switch off this feature to make sketch submitting more streamlined. When viewers care less about overview topics and content of comments and sketches, the Hybrid Cloud visualization could be temporally switched off to simplify the UI. Such feature setting should also have good defaults and be personalized to individual viewers based on their preferences and expertise.

7.3 Rethinking Modalities of Context in Live Streaming

In StreamSketch, comments and sketches are spatially and temporally anchored to snapshots that are captured either manually by the viewer or automatically by the system. However, the results of our study showed that some sketches were not directly related to visual elements shown in the snapshots but instead related to the streamer's speech or background music. In these cases, simply providing snapshots cannot help other viewers to fully understand the context. Recording a video clip with audio when viewers are sketching might be a solution, but it can be distracting and cognitively demanding for viewers to watch video clips for context while they are watching live streams. Future work should explore methods to extract and analyze multi-modal contextual information and visualize them in a less intrusive way for viewers to understand the context of other viewers' multi-modal input.

7.4 Making Sketching More Timely

The findings demonstrated that sketching while watching live streams takes time, resulting in the throughput of this channel being low and limiting its communication efficiency during creative live streams. This also undermines the benefits sketching brings to creative live streams, because timely exposure to ideas is important to improve creative work [24]. One possible solution may be to integrate automatic drawing guidance [26] or auto-complete functionality to help viewers reduce time spent performing tedious repetitions [51]. The contextual information present in live streams, e.g., current topics and sketches in the video, could be leveraged to fine-tune the results to

better support viewers during their sketching processes. Another solution may be to introduce some synchronous channels (rooms) for viewers during live streams, where viewers are grouped based on their interests and skills and they share a canvas where they can sketch collaboratively to generate ideas and create art. In this way, those who do not have ideas can help and get inspirations from those who already have ideas, and if coordinated well, they can sketch out the ideas faster and produce higher-quality sketches [24] and also get more chance for socializing.

7.5 Filtering for Different Information Needs

The feedback of streamers and viewers on visualizations and filtering revealed that viewers cared more about topics and content of comments and sketches during live streams, while streamers paid more attention to the intention of viewers' sketches and the discourse act of viewer comments. Streamers' focus may also change depending on their goals, for example, when they ask viewers to do a quick poll, they may care more about content/topics. It indicates that the visualization/filtering functions should have different emphasis for viewers and streamers, and should also be more dynamic and customizable. For example, the Hybrid Cloud visualization for streamers should highlight the discourse act of viewer input and could be reorganized dynamically based on streamers' focus. The visualization could also be simplified, showing only 'salient' and important keywords and sketches rather than displaying almost everything. Because viewers' descriptions could not provide accurate labels for their sketches, the current visualization might be error-prone when clustering sketches. Future work should consider leveraging machine learning algorithms to cluster sketches and analyzing descriptions to improve the accuracy of clustering. The current design of visualization and filtering is optimized for *communication* to provide contexts and overview for streamers and viewers to make sense of. However, to better support creative activities, these features should also be optimized for *creativity*. Future work can explore designing tools to help streamers and viewers get inspirations more efficiently through exploring sketches and comments.

7.6 Content Moderation of Multi-Modal Messages

Based on their prior experience with live streaming, both streamers and several viewers expressed concerns about content moderation with StreamSketch if it was to be deployed in the wild. They noted that trolls might abuse sketching and create inappropriate content on top of the video. Although the current implementation allowed the research team to have full control over what would be shown to everyone via a back-end administrator interface, when deployed in the wild, it will be important to determine how to support moderators and the streamers' community to moderate inappropriate content. As anyone could become a troll [3], creating inclusion or exclusion lists may not be viable. Even more challenging is the notion that visual content such as sketches are difficult to interpret due to their subjectivity, so it can be hard to determine what is or is not appropriate. Both streamers in the study also noted that they wanted more control over the areas in the video where viewers could sketch so that they could ensure that certain areas of the video were free from viewer sketches. Perhaps the introduction of community-based moderation [21] could be a solution, however, challenges arise regarding how to support streamers in communicating their moderation needs and how to develop strategies for community moderation teams.

7.7 Generalizeability to Other Streaming Genres

The current design of StreamSketch was grounded in creative live streams, and more specifically, drawing live streams. However, the streamers that were interviewed noted that there was potential for StreamSketch to be applied to other live streaming genres as well. Based on her experiences streaming video games, S1 thought that StreamSketch would be useful to ask viewers to answer visual polls by circling on the game interface or use sketches as a mini game when there were

idle moments during gameplay. These streamers also mentioned that because sketching supports self-expression, viewers could also use it to express their emotion or share personal stories in other genres of streams. Because sketching and other related art and design skills are found within traditional practices of intangible cultural heritage (ICH), StreamSketch also has the potential to enable streamers who share and promote cultural practices through live streaming to better engage their viewers [28]. However, care must be taken when generalizing StreamSketch to other genres, as the communities might be very different, trolling might be more or less prevalent, and their skills and perceptions of sketching could be different from what was focused on in the present work.

8 CONCLUSION

This work analyzed existing visual interaction modalities on Twitch and interviewed creative live streamers to better understand their practices and needs. Based on the limitations of current viewer-streamer interactions, a new tool, StreamSketch was designed, developed, and evaluated. StreamSketch enables viewers to engage with streamers using sketch and text, and facilitates the navigation and exploration of such multi-modal viewer input during live streams. Through the evaluation of StreamSketch with streamers and viewers, sketching was found to be used for a variety of purposes by viewers during live streams and such interaction was found to increase their engagement. However, the effort viewers need to exert to sketch, the challenges surrounding content moderation, and the integration of StreamSketch within existing tools are challenges that must be overcome before sketching and expressive visual inputs can become a widespread interaction modality in creative live streams.

REFERENCES

- [1] Rudolf Arnheim. 1997. *Visual thinking*. Univ of California Press.
- [2] Di (Laura) Chen, Dustin Freeman, and Ravin Balakrishnan. 2019. Integrating Multimedia Tools to Enrich Interactions in Live Streaming for Language Learning. In *CHI '19*. Article 438, 14 pages. <https://doi.org/10.1145/3290605.3300668>
- [3] Justin Cheng, Michael Bernstein, Cristian Danescu-Niculescu-Mizil, and Jure Leskovec. 2017. Anyone can become a troll: Causes of trolling behavior in online discussions. In *Proceedings of the 2017 ACM conference on computer supported cooperative work and social computing*. ACM, 1217–1230.
- [4] Soon Hau Chua, Toni-Jan Keith Palma Monserrat, Dongwook Yoon, Juho Kim, and Shengdong Zhao. 2017. Korero: Facilitating complex referencing of visual materials in asynchronous discussion interface. *Proceedings of the ACM on Human-Computer Interaction* 1, CSCW (2017), 1–19.
- [5] Brian Dorn, Larissa B. Schroeder, and Adam Stankiewicz. 2015. Piloting TrACE: Exploring Spatiotemporal Anchored Collaboration in Asynchronous Learning. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing (Vancouver, BC, Canada) (CSCW '15)*. Association for Computing Machinery, New York, NY, USA, 393–403. <https://doi.org/10.1145/2675133.2675178>
- [6] M. Dörk, D. Gruen, C. Williamson, and S. Carpendale. 2010. A Visual Backchannel for Large-Scale Events. *IEEE Transactions on Visualization and Computer Graphics* 16, 6 (Nov 2010), 1129–1138. <https://doi.org/10.1109/TVCG.2010.129>
- [7] Travis Faas, Lynn Dombrowski, Alyson Young, and Andrew D. Miller. 2018. Watch Me Code: Programming Mentorship Communities on Twitch.Tv. *Proc. ACM Hum.-Comput. Interact.* 2, CSCW, Article 50 (Nov. 2018), 18 pages. <https://doi.org/10.1145/3274319>
- [8] C. Ailie Fraser, Joy O. Kim, Alison Thornsberry, Scott Klemmer, and Mira Dontcheva. 2019. Sharing the Studio: How Creative Livestreaming Can Inspire, Educate, and Engage. In *Proceedings of the 2019 on Creativity and Cognition (San Diego, CA, USA) (C&C '19)*. ACM, New York, NY, USA, 144–155. <https://doi.org/10.1145/1182475.1182537>
- [9] Susan R Fussell, Leslie D Setlock, Jie Yang, Jiazhi Ou, Elizabeth Mauer, and Adam DI Kramer. 2004. Gestures over video streams to support remote collaboration on physical tasks. *Human-Computer Interaction* 19, 3 (2004), 273–309.
- [10] Scott Garner. 2019. Heat. <http://heat.j38.net/>
- [11] David Geerts. 2006. Comparing Voice Chat and Text Chat in a Communication Tool for Interactive Television. In *Proceedings of the 4th Nordic Conference on Human-Computer Interaction: Changing Roles (Oslo, Norway) (NordiCHI '06)*. Association for Computing Machinery, New York, NY, USA, 461–464. <https://doi.org/10.1145/1182475.1182537>
- [12] Elena L. Glassman, Juho Kim, Andrés Monroy-Hernández, and Meredith Ringel Morris. 2015. Mudslide: A Spatially Anchored Census of Student Confusion for Online Lecture Videos. In *Proceedings of the 33rd Annual ACM Conference*

- on *Human Factors in Computing Systems* (Seoul, Republic of Korea) (*CHI '15*). Association for Computing Machinery, New York, NY, USA, 1555–1564. <https://doi.org/10.1145/2702123.2702304>
- [13] Seth Glickman, Nathan McKenzie, Joseph Seering, Rachel Moeller, and Jessica Hammer. 2018. Design Challenges for Livestreamed Audience Participation Games. In *Proceedings of the 2018 Annual Symposium on Computer-Human Interaction in Play* (Melbourne, VIC, Australia) (*CHI PLAY '18*). ACM, New York, NY, USA, 187–199. <https://doi.org/10.1145/3242671.3242708>
- [14] Gabriela Goldschmidt. 1994. On visual design thinking: the vis kids of architecture. *Design studies* 15, 2 (1994), 158–174.
- [15] David Ha and Douglas Eck. 2017. A neural representation of sketch drawings. *arXiv preprint arXiv:1704.03477* (2017).
- [16] Oliver L Haimson and John C Tang. 2017. What makes live events engaging on Facebook Live, Periscope, and Snapchat. In *Proceedings of the 2017 CHI conference on human factors in computing systems*. 48–60.
- [17] William A Hamilton, Oliver Garretson, and Andruid Kerne. 2014. Streaming on twitch: fostering participatory communities of play within live mixed media. In *Proceedings of the 32nd annual ACM conference on Human factors in computing systems*. ACM, 1315–1324.
- [18] William A Hamilton, Nic Lupfer, Nicolas Botello, Tyler Tesch, Alex Stacy, Jeremy Merrill, Blake Williford, Frank R Bentley, and Andruid Kerne. 2018. Collaborative Live Media Curation: Shared Context for Participation in Online Learning. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. ACM, 555.
- [19] William A. Hamilton, John Tang, Gina Venolia, Kori Inkpen, Jakob Zillner, and Derek Huang. 2016. Rivulet: Exploring Participation in Live Events through Multi-Stream Experiences. In *Proceedings of the ACM International Conference on Interactive Experiences for TV and Online Video* (Chicago, Illinois, USA) (*TVX '16*). Association for Computing Machinery, New York, NY, USA, 31–42. <https://doi.org/10.1145/2932206.2932211>
- [20] Weidong Huang, Seungwon Kim, Mark Billinghurst, and Leila Alem. 2019. Sharing hand gesture and sketch cues in remote collaboration. *Journal of Visual Communication and Image Representation* 58 (2019), 428 – 438. <https://doi.org/10.1016/j.jvcir.2018.12.010>
- [21] Charles Kiene, Jialun Aaron Jiang, and Benjamin Mako Hill. 2019. Technological Frames and User Innovation: Exploring Technological Change in Community Moderation Teams. *Proceedings of the ACM on Human-Computer Interaction* 3, CSCW (2019), 1–23.
- [22] Seungwon Kim, Mark Billinghurst, and Kangsoo Kim. 2020. Multimodal interfaces and communication cues for remote collaboration. *Journal on Multimodal User Interfaces* (03 Oct 2020). <https://doi.org/10.1007/s12193-020-00346-8>
- [23] David Kirk, Tom Rodden, and Danaë Stanton Fraser. 2007. Turn It this Way: Grounding Collaborative Action with Remote Gestures. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) (*CHI '07*). Association for Computing Machinery, New York, NY, USA, 1039–1048. <https://doi.org/10.1145/1240624.1240782>
- [24] Chinmay Kulkarni, Steven P Dow, and Scott R Klemmer. 2014. Early and repeated exposure to examples improves creative work. In *Design thinking research*. Springer, 49–62.
- [25] Yi-Chieh Lee, Wen-Chieh Lin, Fu-Yin Cherng, Hao-Chuan Wang, Ching-Ying Sung, and Jung-Tai King. 2015. Using Time-Anchored Peer Comments to Enhance Social Interaction in Online Educational Videos. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (Seoul, Republic of Korea) (*CHI '15*). Association for Computing Machinery, New York, NY, USA, 689–698. <https://doi.org/10.1145/2702123.2702349>
- [26] Yong Jae Lee, C Lawrence Zitnick, and Michael F Cohen. 2011. Shadowdraw: real-time user guidance for freehand drawing. In *ACM Transactions on Graphics (TOG)*, Vol. 30. ACM, 27.
- [27] Pascal Lessel, Alexander Vielhauer, and Antonio Krüger. 2017. Expanding Video Game Live-Streams with Enhanced Communication Channels: A Case Study. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) (*CHI '17*). ACM, New York, NY, USA, 1571–1576. <https://doi.org/10.1145/3025453.3025708>
- [28] Zhicong Lu, Michelle Annett, Mingming Fan, and Daniel Wigdor. 2019. "I Feel It is My Responsibility to Stream": Streaming and Engaging with Intangible Cultural Heritage Through Livestreaming. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (*CHI '19*). ACM, New York, NY, USA, Article 229, 14 pages. <https://doi.org/10.1145/3290605.3300459>
- [29] Zhicong Lu, Michelle Annett, and Daniel Wigdor. 2019. Vicariously Experiencing it all without Going Outside: A Study of Outdoor Livestreaming in China. *Proceedings of the ACM on Human-Computer Interaction* 3, CSCW, Article 25 (Nov. 2019), 28 pages. <https://doi.org/10.1145/3359127>
- [30] Zhicong Lu, Seongkook Heo, and Daniel Wigdor. 2018. StreamWiki: Enabling Viewers of Knowledge Sharing Live Streams to Collaboratively Generate Archival Documentation for Effective In-Stream and Post Hoc Learning. *Proceedings of the ACM on Human-Computer Interaction* 2, CSCW, Article 112 (Nov. 2018), 26 pages. <https://doi.org/10.1145/3274381>
- [31] Zhicong Lu, Haijun Xia, Seongkook Heo, and Daniel Wigdor. 2018. You Watch, You Give, and You Engage: A Study of Live Streaming Practices in China. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*

- (CHI '18). Paper 466, 13 pages. <https://doi.org/10.1145/3173574.3174040>
- [32] Matthew K. Miller, John C. Tang, Gina Venolia, Gerard Wilkinson, and Kori Inkpen. 2017. Conversational Chat Circles: Being All Here Without Having to Hear It All. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) (CHI '17). ACM, New York, NY, USA, 2394–2404. <https://doi.org/10.1145/3025453.3025621>
- [33] Gustavo Nascimento, Manoel Ribeiro, Loïc Cerf, Natália Cesário, Mehdi Kaytoue, Chedy Raïssi, Thiago Vasconcelos, and Wagner Meira. 2014. Modeling and analyzing the video game live-streaming community. In *2014 9th Latin American Web Congress*. IEEE, 1–9.
- [34] Jiazhi Ou, Xilin Chen, Susan R. Fussell, and Jie Yang. 2003. DOVE: Drawing over Video Environment (*MULTIMEDIA '03*). Association for Computing Machinery, New York, NY, USA, 100–101. <https://doi.org/10.1145/957013.957034>
- [35] Gonzalo Ramos and Ravin Balakrishnan. 2003. Fluid interaction techniques for the control and annotation of digital video. In *Proceedings of the 16th annual ACM symposium on User interface software and technology*. 105–114.
- [36] Stuart Reeves, Christian Greiffenhagen, Martin Flintham, Steve Benford, Matt Adams, Ju Row Farr, and Nicholas Tandavantij. 2015. I'd Hide You: Performing Live Broadcasting in Public. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (Seoul, Republic of Korea) (CHI '15). Association for Computing Machinery, New York, NY, USA, 2573–2582. <https://doi.org/10.1145/2702123.2702257>
- [37] Stuart Rose, Dave Engel, Nick Cramer, and Wendy Cowley. 2010. Automatic Keyword Extraction from Individual Documents. *Text Mining: Applications and Theory* (2010), 1–20.
- [38] Vikash Singh, Sarah Abdellahi, Mary Lou Maher, and Celine Latulipe. 2016. The video collaboratory as a learning environment. In *Proceedings of the 47th ACM Technical Symposium on Computing Science Education*. 352–357.
- [39] Vikash Singh, Celine Latulipe, Erin Carroll, and Danielle Lottridge. 2011. The Choreographer's Notebook: A Video Annotation System for Dancers and Choreographers. In *Proceedings of the 8th ACM Conference on Creativity and Cognition* (Atlanta, Georgia, USA) (C&C '11). Association for Computing Machinery, New York, NY, USA, 197–206. <https://doi.org/10.1145/2069618.2069653>
- [40] John Tang, Gina Venolia, Kori Inkpen, Charles Parker, Robert Gruen, and Alicia Pelton. 2017. Crowdcasting: Remotely Participating in Live Events Through Multiple Live Streams. *Proceedings of the ACM on Human-Computer Interaction 1*, CSCW (2017), 1–18.
- [41] John C Tang and Scott L Minneman. 1991. VideoDraw: a video interface for collaborative drawing. *ACM Transactions on Information Systems (TOIS)* 9, 2 (1991), 170–184.
- [42] John C Tang, Gina Venolia, and Kori M Inkpen. 2016. Meerkat and periscope: I stream, you stream, apps stream for live streams. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. ACM, 4770–4780.
- [43] Pei-Yun Tu, Mei-Ling Chen, Chi-Lan Yang, and Hao-Chuan Wang. 2016. Co-Viewing Room: Mobile TV Content Sharing in Social Chat. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems* (San Jose, California, USA) (CHI EA '16). Association for Computing Machinery, New York, NY, USA, 1615–1621. <https://doi.org/10.1145/2851581.2892476>
- [44] Twitch.tv. 2019. Twitch Extensions. <https://www.twitch.tv/ext/categories/f3d9ce3c-696d-4c6f-b623-e76c06e33f1d>
- [45] Gina Venolia, John C. Tang, and Kori Inkpen. 2015. SeeSaw: I See You Saw My Video Message. In *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services* (Copenhagen, Denmark) (*MobileHCI '15*). Association for Computing Machinery, New York, NY, USA, 244–253. <https://doi.org/10.1145/2785830.2785847>
- [46] Joseph B Walther, Tracy Loh, and Laura Granka. 2005. Let me count the ways: The interchange of verbal and nonverbal cues in computer-mediated and face-to-face affinity. *Journal of language and social psychology* 24, 1 (2005), 36–65.
- [47] Justin D. Weisz and Sara Kiesler. 2008. How Text and Audio Chat Change the Online Video Experience. In *Proceedings of the 1st International Conference on Designing Interactive User Experiences for TV and Video* (Silicon Valley, California, USA) (*UXTV '08*). Association for Computing Machinery, New York, NY, USA, 9–18. <https://doi.org/10.1145/1453805.1453809>
- [48] Justin D. Weisz, Sara Kiesler, Hui Zhang, Yuqing Ren, Robert E. Kraut, and Joseph A. Konstan. 2007. Watching Together: Integrating Text Chat with Video. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) (CHI '07). ACM, New York, NY, USA, 877–886. <https://doi.org/10.1145/1240624.1240756>
- [49] Donghee Yvette Wohn. 2019. Volunteer moderators in twitch micro communities: How they get involved, the roles they play, and the emotional labor they experience. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–13.
- [50] Qunfang Wu, Yisi Sang, and Yun Huang. 2019. Danmaku: A New Paradigm of Social Interaction via Online Videos. *ACM Transactions on Social Computing* 2, 2 (2019), 1–24.
- [51] Jun Xing, Hsiang-Ting Chen, and Li-Yi Wei. 2014. Autocomplete painting repetitions. *ACM Transactions on Graphics (TOG)* 33, 6 (2014), 172.
- [52] Saelyne Yang, Changyoon Lee, Hijung Valentina Shin, and Juho Kim. 2020. Snapstream: Snapshot-Based Interaction in Live Streaming for Visual Art. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*

(Honolulu, HI, USA) (*CHI '20*). Association for Computing Machinery, New York, NY, USA, 1–12. <https://doi.org/10.1145/3313831.3376390>

- [53] Dongwook Yoon, Nicholas Chen, Bernie Randles, Amy Cheatle, Corinna E Löckenhoff, Steven J Jackson, Abigail Sellen, and François Guimbretière. 2016. RichReview++ Deployment of a Collaborative Multi-modal Annotation System for Instructor Feedback and Peer Discussion. In *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing*. 195–205.
- [54] Amy X Zhang and Justin Cranshaw. 2018. Making sense of group chat through collaborative tagging and summarization. *Proceedings of the ACM on Human-Computer Interaction 2*, CSCW (2018), 196.

Received June 2020; revised October 2020; accepted December 2020