Rivulet: Exploring Participation in Live Events through Multi-Stream Experiences

William A. Hamilton[1,2], John C. Tang[1], Gina Venolia[1], Kori Inkpen[1], Jakob Zillner[1,3], Derek Huang[1]

[1]Microsoft Research
Redmond, WA USA
{johntang,ginav,kori}@microsoft.com
derekahuang@hotmail.com

[2]Interface Ecology Lab
Texas A&M University
College Station, TX USA
bill@ecologylab.net

[3]VRVis Research Center
Vienna, Austria
jakob.zillner@vrvis.at

ABSTRACT
Live streaming has recently emerged as a growing form of participatory social media. While current live streaming practice focuses on single stream experiences, there are increasing instances of events covered by multiple live streams. In order to explore how to support communication and participation in multi-stream experiences, we present the design and evaluation of Rivulet, an end-to-end mobile live streaming system designed to support participatory multi-stream experiences. Rivulet affords simultaneously watching multiple live streams and incorporates existing feedback mechanisms of text chat and hearts with a novel push-to-talk audio modality. By recruiting viewers through Mechanical Turk, we were able to conduct a study of Rivulet at scale. We found that Rivulet afforded new engaging experiences for participants and led to an impromptu sense of community.

Author Keywords
live streaming, multi-stream, video, push-to-talk, telepresence

ACM Classification Keywords
H.5.2 User Interfaces

INTRODUCTION
During the past decade, live streaming has emerged as a new form of participatory social media. Live streaming has come to refer to live, streaming, video as well as a set of communication media that enable viewers to interact with each other and the streamer. The emerging popularity of live streams is attributed to their ability to enable remote viewers to engage and participate in shared live experiences [10].

The typical live streaming experience consists of a streamer broadcasting a single video stream accompanied by a dedicated chat channel. However, multiple, simultaneous, live streams provide an interesting opportunity to experience events. For example, on Periscope multiple streamers commonly stream simultaneously or within minutes of each other while attending events like concerts or conventions [24]. Similarly, on Twitch, streamers frequently play games together, while they both broadcast independent streams and their viewers' chat in separate chat channels [10].

Despite this trend there is minimal support for identifying and participating in these *multi-stream experiences*. There are a number of 3rd party sites that support embedding multiple live streams together, but do not provide much support beyond the visual aggregation of live streams and their separate chat channels. There are a number of research projects looking at combining multiple live streams [1, 3, 5, 6, 7, 8, 22, 23, 29], but these do not examine audience participation and the resulting experiences.

In this work, we explore how to support communication and participation in multi-stream experiences. In particular, we are interested in the following research questions.

1. How will people experience a collection of streams coming from a live event?
2. How will people use new and existing communication modalities to participate across different streams that are part of an event?

We designed and prototyped Rivulet, an end-to-end mobile live streaming system for multi-stream experiences, as a technology probe for investigating these questions [12]. Rivulet incorporates common live stream modalities including live video, text chat, and hearts (as seen in Periscope). However, we extended these modalities to specifically support a more integrated multi-stream experience, for example all of the streams share an event-wide chat channel. Rivulet also enabled us to explore push-to-talk (PTT) audio from any viewer to the stream, a higher fidelity communication modality that we hypothesized might be more engaging for participants.

To observe and explore realistic participation in multi-stream experiences through Rivulet, we conducted an at-scale field study with eight local Periscope streamers who streamed a local music event. Four participants streamed the event using

TVX'16, June 22 - 24, 2016, Chicago, IL, USA

ACM 978-1-4503-4067-0/16/06…$15.00
DOI: http://dx.doi.org/10.1145/2932206.2932211

Rivulet while the other four streamed the event using Periscope. We also recruited 226 viewers on Mechanical Turk to watch live on both Rivulet and Periscope. This led to a brief, but realistic, multi-stream experience.

We found that by aggregating multiple streams together Rivulet helped participants find interesting streams to watch and participate in. It also afforded new engaging live experiences for viewers and streamers, engendered a stronger sense of community, and helped participants better understand what was happening at the event as a whole. Finally, despite some technical issues, PTT audio proved to be an engaging communication modality, which afforded unique participatory opportunities for viewers.

We start with a discussion of related work around live streaming media and practice. We then present the motivations and a detailed description of Rivulet's design. Next, we present the design of our field study of Rivulet along with the results of the study and a discussion of their implications for the design of live streaming experiences.

RELATED WORK

We present related prior work in the context of live streaming, live participation modalities, and multi-stream environments. We also briefly present the sensitizing concept of hot and cool media.

Live Streaming

While there are currently a number of popular live streaming platforms including Periscope, Meerkat, YouTube Live, and Twitch, live streaming has been emerging as a new form of social media over the past decade. In 2010, Juhlin et al. presented a detailed investigation of some previously popular mobile live streaming services including Qik, Bambuser, Flixwagon, and kyte.com [15], most of which are now effectively defunct. At the time, it was clear that mobile live streaming was still in its infancy. They found that streamers had a difficult time finding interesting topics to stream and there were many technical issues around how to manage the camera. As a result, they prescribed a need for more support on mobile devices and the web for better production of live streams [15]. Live streaming practice has evolved significantly during the past six years. There are now thousands of both professional and amateur live streamers streaming every day across the various platforms.

Live streaming as a medium for civic engagement was studied by Dougherty [4] who did a qualitative analysis of live streams on Qik. She found that much of the civic content on Qik focused on political or activist topics [4]. We see emergent multi-stream experiences around events as a potentially new form of civic engagement for both streamers and viewers. For example, many live streams are often shared during political events like protests and debates. We expect multi-stream experiences to potentially increase the impact and reach of communities around streams.

In 2014, Hamilton et al. reported on findings around Twitch, a video game live streaming site, which emerged from the now shutdown Justin.tv [10]. They found that, while video game content was a major factor for the success of Twitch, what really defines Twitch streams is viewer participation and how that leads to forming communities around live steams. Given this finding, we aim to support audience participation in multi-stream experiences through the design of the Rivulet prototype.

Participatory Live Modalities

A number of research projects have explored interaction and communication modalities that afford audience agency and participation. Jo and Hwang [13] explored viewpoint control and direct sketching on video to support viewer communication and participation during live calls. Kim et al. [18] found that providing contextual information, such as maps and high resolution photographs, during live experiences enabled viewers to actively participate in the experience by pointing out things invisible to or unnoticed by a remote streamer. Yonezawa and Tokuda [30] designed a system which helped connect musical performers with their audience by enabling remote viewers to control the light and camera angle of the broadcast. They found that these modalities engaged viewers and increased the connection between performers and their audiences. In the Rivulet prototype, we explore how to augment and combine existing communication modalities within a multi-stream environment to afford greater viewer participation.

Blast Theory recently designed a participatory live streaming experience that takes the form of a game where performers simultaneously stream live video and engage with online viewers [21]. They prescribe the "thickening" of online connections between streamer and online connections by making viewer messages more prominent and incorporating new communication modalities [21]. Additionally, Webb et al. in their recent investigation of distributed live performances identified a need to develop new modalities to serve as subtle feedback mechanisms between audiences and performers [12]. Through Rivulet, we aim to thicken the connections between audiences and streamers by redesigning existing and including additional communication modalities.

Multi-Stream Environments

To support multiple live stream experiences around an event, Rivulet supports dynamically aggregating streams. We note that a number of third party sites exist, especially in the context of Twitch, that support the combination of multiple streams into a single aggregated view [11, 16]. However, these sites generally only allow the user to collect streams and their separate chat channels together visually. We argue that, to meaningfully support multi-stream experiences, communication modalities must be designed to support participation across as well as within individual streams.

Tazaki proposed one of the first multi-stream systems [25]. The system, although never implemented, was designed to engage participants in not viewing, but contributing and curating live video in a shared multi-stream experience. Bentley and Groble later designed TuVista, a live video

Figure 1. Screenshot of Rivulet viewer client during the Jazz Walk study. Viewers could focus on and listen to one stream at a time and see previews of the other live streams. Viewers shared an event-wide chat with usernames color coded by the stream they were watching. Viewers could send hearts to their focused stream, and see hearts sent to any stream. Viewers could send push-to-talk audio to their focused stream by clicking on the microphone icon in the upper right of the live stream.

production system meant to facilitate the real-time composition of multiple live streams from a sporting event [1]. Similarly, Engström et al. presented a multi-stream system that supported the collaborative contribution and composition of mobile video streams in a night club setting [5]. Engström went on to explore several projects which explored the live mixing and production of mobile live streams [6, 8]. Juhlin et al. also investigated the production practices of professional [7] and amateur [14] live broadcasters to inform the design of live video applications. Recently, Sa et al. designed a live streaming application that helped mobile live streamers collaboratively produce live experiences by providing awareness of other nearby streams [22]. Numerous other works have investigated how to enable crowds to compose and edit video both live [3, 23, 29] and after the fact [1, 9, 17, 27]. These works have focused on issues such as event coverage [9, 17], automated organization [3, 17, 27], collaborative orchestration and organization [1, 3, 9, 23, 27], and privacy [1] around event contexts like concerts [17] and sporting events [111, 9]. In the Rivulet prototype, we do not directly support the composition or production of multiple live streams, but rather we take the approach of supporting viewers in experiencing, selecting from, and participating in multiple live streams simultaneously.

Sensitizing Concept: Hot and Cool Media

In their analysis of live streaming media in the context of Twitch, Hamilton et al. drew on McLuhan's concept of Hot and Cool media to describe how text chat and live video afforded participatory live experiences [10, 19]. McLuhan described *cool media* as those which are typically low fidelity and afford high levels of participation. Inversely, he described *hot media* as high fidelity and affording little participation [19]. In the context of live streaming, Hamilton et al. describe live video as *hot*. It is high fidelity and affords the sharing of rich live experiences, but alone offers little opportunity for participation. Conversely, text chat is *cool*, affording much greater opportunity for participation through a lower fidelity medium. They argued that together, these *hot* and *cool* modalities afforded the shared history and participatory experiences at the core sense of community in many live streams [10]. We draw on this concept to discuss the qualities of the hearts and push-to-talk communication modalities and their resulting role in Rivulet.

RIVULET PROTOTYPE

The Rivulet prototype implements an end-to-end live streaming service. The prototype consists of a custom Android video streaming application, web based viewer client, and web service. By developing each of these components we were able to design a holistic multi-stream experience aimed at engaging participants through novel communication modalities. We present the design and motivations for each of the components of Rivulet.

Viewer Client

The viewer client (Figure 1) was implemented as an online web-based interface. This enabled us to recruit a large group of viewer participants through the web who could use the system by simply navigating to a URL. The client enables participants to watch multiple streams simultaneously, engage in a global chat, give feedback in the form of hearts to streamers, and broadcast PTT audio.

Broadcaster Client

We developed a custom Android application that enabled streamers to broadcast video from either the front- or back-facing camera (see Figure 2). Participants could also rotate the orientation of their phones while streaming as the viewing client dynamically rotated the streaming video for viewers. Video was broadcast at a resolution of 576 x 320 (the same resolution used by Periscope) and was encoded at a data rate between 1.5 – 2.0 Mbps using H.264. The encoded video was streamed to a cloud based Wowza streaming engine server using the Real Time Messaging Protocol (RTMP). During the course of the presented study, video was uploaded over cellular LTE connections. We will describe how the broadcaster interface integrates each of the explored communication modalities in the following sections.

Supporting Multi-Stream Experiences

To support participants in watching and participating in multiple streams, they first needed to be able see them all and choose which one to focus on. While viewers could see a live preview of all of the active streams in the experience, they could only focus on one. The focused stream appears in the middle of the interface and audio plays for that stream (see Figure 1). Previews of other streams appear smaller and darkened on the right side of the interface. To focus on another stream, viewers simply click on a preview to swap its place with the currently focused stream in the interface. At the same time, the new stream's audio is played instead of the previously focused stream's audio. As streams started and stopped broadcasting, they were dynamically added to and removed from the interface. When the client is first opened, and if the stream that the viewer is focused on ends, the system randomly selects a stream to play.

Each stream is labeled with the name of the streamer as well as a fraction indicating the portion of all viewers of the event who are watching this stream. We intended this to help viewers understand how other viewers were selecting which stream to watch. This fraction also shows the streamer how many viewers are watching them compared to participating in the event as a whole (see Figure 2). Additionally, each stream is algorithmically assigned a unique color, which is helps differentiate each stream's viewers in the global chat.

We used a custom Adobe Flash Player to stream and render each video stream. The total delay from broadcaster to viewer was typically between 2 and 5 seconds, which is equal to or less than most current live streaming platforms. While viewers could watch each of the streams, streamers were unable to see other live streams during the study. While streamers could not directly maintain awareness of other streams, they would be indirectly aware through viewers' comments in the provided communication modalities.

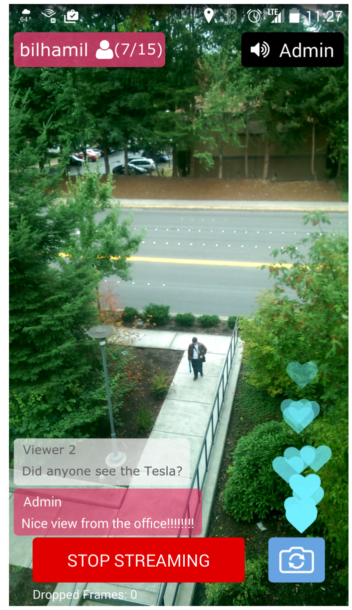

Figure 2. The mobile client enabled sharing live video and audio and monitoring viewers, hearts, chat, and PTT audio.

Event-Wide Text Chat

Our second research question focuses on how to participants experience communication modalities in multi-stream environments. We were particularly interested in how to support viewers using text chat in a multi-stream experience. Rivulet associates all of an event's live streams with a single event-wide text chat. This differs from associating a single stream with its own text chat, as seen on Twitch, Periscope, Meerkat, and many other streaming platforms. We expected that participants would discuss and experience the event as a whole instead of in disjointed conversations around each stream. However, we still wanted participants to be able to make comments localized to particular streams and make sense of who was watching what stream. In the chat, viewers' usernames were color coded with the live stream they were watching when they made the comment.

Similarly, we hypothesized, for streamers, text chat from viewers focused on their stream would have more immediate value than chats from other viewers. Thus, while all text chats would briefly appear on the streaming interface, chats from the streamer's viewers would be highlighted with the color associated with their stream (see Figure 2). Chats from viewers of other streams appeared with a gray background.

Hearts

For the Rivulet prototype, we adopted the hearts communication modality featured in Periscope. Hearts are an ephemeral mechanism that enable users to send lightweight feedback to streamers. A viewer simply clicks on the stream to send one heart, which appears on the video stream and briefly floats up before disappearing. This can be done rapidly, and often is, to send a stream of hearts. To help viewers maintain awareness of each stream, viewers can see hearts appearing on each stream separately (see Figure 1).

Hearts are roughly color coordinated with viewers. While the heart shape implies love, their exact meaning is ambiguous.

Hearts are an interesting emerging communication modality because they provide extremely ephemeral and localized feedback about a live stream. They provide quick, positive feedback to the streamer about their viewers. However, taken beyond the context of just one stream, hearts might help other viewers identify interesting activity in a multi-stream environment. Considering hearts within the framework of *hot* and *cool* media, they are extremely *cool*. They are very low fidelity in that they only have one particular form. At the same time they afford ample opportunity for participation, as any number of participants can send as many hearts as they wish at any point in time without being clearly identified.

Push-To-Talk Audio

Our goal with the incorporation of PTT audio was to further explore our second research question by examining how participants engage in a multi-stream experience using a relatively novel communication modality. The PTT modality is not common to live streaming practice and lies somewhere between *cool* text chat and *hot* live video on McLuhan's spectrum. We designed PTT to afford viewers a higher level of impact on the experience, while affording more opportunity to participate than live video.

In Rivulet, any viewer with a microphone can broadcast audio on the stream they are currently focused on by clicking and holding on the microphone icon displayed on the stream (see Figure 1). The audio is captured and encoded in the browser, streamed to the Rivulet web service, and then pushed to the streamer's broadcasting client. On the mobile client, the audio is played back immediately to the streamer and also mixed into the right channel of the outgoing stream's stereo audio. This allowed viewers of that stream to hear PTT audio from other viewers in sync with when the streamer heard it. An indicator appeared in the video stream during showing who was talking (see PTTs from *AJBBB* and *Sweet* in Figure 1). This indicator was also displayed on the broadcaster client (see PTT from *Admin* in Figure 2). To prevent feeding the PTT back to the person who spoke it, they heard only the left channel from the video stream for the duration of the PTT. Streamers wore headphones to prevent PTT audio from leaking into the left channel of the broadcast.

We limited PTT broadcasting to only one viewer at a time per stream. We also set a maximum of 10 seconds for PTTs to prevent any viewer from dominating the modality by continuously broadcasting. The system also ensures a 5 second break between every PTT to give the streamer a chance to respond. When a viewer tries to start a PTT, if the channel is clear, a start chime is played and a 10 second countdown starts. After 10 seconds, if the viewer has not stopped broadcasting the system plays a disconnect chime and stops the transmission. If the viewer tries to PTT when the channel is not clear, they see a wait signal until it is clear. If multiple viewers are trying to PTT simultaneously, the system places them into a wait queue.

We explicitly intended PTT to be a communication modality at the single stream level. Only viewers of a particular stream would be able to hear PTTs sent to that stream. We also expected PTT to be easier for streamers to pay attention to while still engaging in the shared event, since they did not have to look at their device to perceive the incoming audio.

STUDY DESIGN

We designed a study of Rivulet to explore our research questions around communication modalities and emergent behaviors in multi-stream experiences. Through the study, we aimed to create a multi-stream experience that was as ecologically valid as possible. To this end, we recruited experienced streamers to broadcast at a local event to an audience of live viewers. We also worked to recruit an online audience of reasonable scale. In the following sections, we describe our process for selecting and organizing an event, recruiting participants, and evaluating the experience.

The Jazz Walk Event

We wanted to find an event which would be interesting to the streamers and viewers and had multiple concurrent activities to provide ample opportunity for streamers to share different perspectives of the event. We also had to consider the availability of robust cellular network connections as a prior study failed due to cellular network issues. We chose a local jazz festival called The North City Jazz Walk, which historically attracts several hundred attendees. The event featured 10 local musical groups playing in different venues across a 3 block area. Venues included bars, parking lots, a coffee shop, a church, and a club house.

Live Streamers

Prior to the event, we recruited local Periscope streamers. By recruiting experienced streamers, we aimed to have participants who were comfortable conducting a live stream and interacting with viewers. We also expected that streamers would be able to provide insights into how their experience with Rivulet compared with Periscope.

We identified local Periscope streamers by collecting geocoded Periscope Tweets from the local area over a four-day period. From the resulting 250 streamers, we were able to contact approximately 50. We also asked these streamers to forward the study information to any local streamers they knew. We successfully recruited 7 participants to attend the Jazz Walk and added one personal contact who was familiar with live streaming. Participants were offered a 250 USD gratuity for taking part in the study.

Prior to the study, we met the streamer participants outside the event area where we administered a short pre-questionnaire and divided the participants into two groups of four. Four participants were asked to stream using Periscope [P1-P4], the other four were asked to use Rivulet [R1-R4]. The Rivulet streamers were given a brief tutorial on how the system worked. While the Periscope streamers used their own devices, we gave the Rivulet participants Android phones to use during the study. We asked that participants

	Likert Questions
Q1	I was aware of all the streams offered by the people streaming at the Jazz Walk today.
Q2	I enjoyed being able to choose different streams at the Jazz Walk.
Q3	I was aware of what the other streamers at the Jazz Walk were covering compared to what I was watching.
Q4	I felt like I was able to influence the live streams using the push-to-talk feature.
Q5	I felt like I was able to influence the live streams using text chat messages.
Q6	I felt like I was able to influence the live streams by sending hearts.
Q7	I was able to easily find a view that was interesting to watch.
Q8	Using [Periscope, this Prototype] to view the Jazz Walk event was fun.
Q9	I felt like I was part of a community of people enjoying the Jazz Walk.
Q10	I felt connected to the people streaming the the Jazz Walk.
Q11	I felt connected to the other people viewing the Jazz Walk event.
Q12	I felt like I could control what I viewed of the Jazz Walk event.

Table 1. Summary of Likert questions asked in each condition. Note that Q4 was only asked in the Rivulet condition.

attend the event for approximately an hour and a half and that they stream for at least a quarter of the time. The Periscope streamers were asked to publish a tweet with the hashtag #MSRJazzWalk anytime they started streaming, so they could be found. We placed no other restrictions on what or how they streamed. We only asked that they do what they would normally do. During the study, a researcher was available at the event for technical assistance. After the study, we met with the streamers again and briefly discussed the experience and asked them to complete a short survey.

Mechanical Turk Viewers

We aimed to recruit an audience of reasonable scale to observe during the study. We argue that this is critical for observing meaningful engagement and communication during a live streaming experience. Thus, we recruited viewer participants through Amazon's Mechanical Turk. Approximately 15 minutes after sending the streamer participants into the event, we published two Human Intelligence Tasks (HITs): one to recruit viewers to watch the experience on Rivulet, and the other for Periscope.

Participants in both conditions were shown a brief video explaining how either Rivulet or Periscope worked. Viewers were also given a link to the Jazz Walk website. Rivulet participants were directed to the viewer client through a link. Since the web-based Periscope client does not afford sending hearts or chats, Periscope participants were asked to use their smart phones (downloading the Twitter and Periscope mobile apps if needed). They were told they could locate streams by searching for #MSRJazzWalk on Twitter.

Participants in both conditions were asked to watch streams for at least 20 minutes and as long as they liked beyond that. After watching, participants in both conditions were asked to fill out a short questionnaire composed of a series of Likert questions (see Table 1). We also asked participants to rate

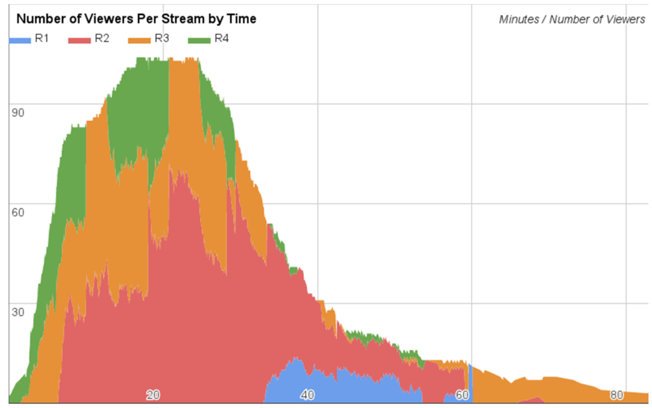

Figure 3. Viewers per stream in Rivulet over the course of the study, showing how viewers switched among streams.

the usefulness of the communication modalities in each condition using a semantic differential and answer a series of open-ended questions. We expected that viewer participants would be engaged in the task between 35 minutes to an hour. Thus, participants in each condition were offered an 8 USD compensation (in keeping with a 10 USD hourly wage). 120 HITs were published for each condition.

Data Logging

Besides serving the page content and managing real-time messaging, the Rivulet web server also logged user actions and relevant metadata in a database for later analysis. We were not able to log periscope user interactions, so we are not able to present a quantitative analysis comparing conditions.

RESULTS AND DISCUSSION

Despite the complex nature of the presented study, we experienced relatively few issues, resulting in an engaging and rewarding experience for both streamers and viewers. We present the results of the study and discuss the implications of our findings. We first provide a brief description of the recruited viewership and streams shared during the event. Next, we provide a discussion of how each of the communication modalities were used, and draw implications from our observations. We then discuss how participants engaged in multiple streams and the implications of multi-stream live experiences around events. We also discuss the emergent sense of community we observed. Finally, we discuss implications related to our study design.

Viewership

After publishing the Mechanical Turk HITs, participants quickly flooded into the study. Figure 3 illustrates the number of viewers over time in the Rivulet condition. Within 20 minutes over 100 viewers were watching on Rivulet. In total, we had 115 participants in the Rivulet condition [RV1-RV115] and 111 in the Periscope condition.

Live Streams

During the study, some of the streamers in different conditions decided to stream together. Consequently, there were similar streams in each condition. We provide a brief description of what was streamed by each of the streamers.

R2, P2, and P3 were a group of high school boys and were friends prior to the study. They walked and talked together while simultaneously streaming 3 different streams almost continuously for the duration of the study. Before entering the event, they first went to a nearby grocery store and purchased some rice cakes and water. This proved to be a fairly humorous diversion for many viewers. They then started walking around the event and stopped at several different musical performances. While they walked they focused on interacting with their viewers and with each other. R2's stream received the most chat messages per minute and the second most PTTs per minute.

P1 and R3 are a brother and sister in their thirties. For most of the study they streamed from a bar that was hosting one of the musical performances, later walking to another bar hosting a performance. They used their front-facing cameras for much of their streams to interact with their viewers with less focus on the jazz performances. R3 throughout the study made humorous faces and noises trying to get a reaction from her viewers. At one point, she started encouraging viewers to tell jokes on her stream. She also pretended to eat the hearts viewers were sending her. P1's stream was more subdued, and he streamed both the musical performances and himself while he interacted with viewers. P1 and R3 frequently interacted with and streamed each other during the study.

R1 and P4 were two men in their early twenties and were friends prior to the study. During the study they streamed at different outdoor performances and while walking between performances. They interacted with their viewers to a much lesser degree than the aforementioned streamers. At one point the pair got up and started dancing while a band played a cover of the Peanuts' theme song. While P4 streamed for most of the duration of the study, R1 only streamed for a short duration toward the end of the event.

R4 was by himself for most of the study. He frequently responded to viewer chats, but generally his stream focused on the musical performances at the event. He never showed his face on stream during the study, and often just streamed different performances. At one point, he did stream himself walking down the street, but minimal interaction occurred with viewers during this time.

Text Chat

During the study, a total of 862 chat messages were sent among the Rivulet viewers, and all but 17 of our 115 Rivulet viewers sent at least one chat message. Figure 4 illustrates the distribution of viewers based on how frequently they chatted. As may be expected given the tendencies of lurkers [20], a large number of viewers, 87 out of 115, chatted either not at all or less than twice every five minutes. However, the remaining 28 viewers chatted regularly, one as often as 3 times a minute.

Chat messages were of varying content including comments and questions about the event directed at the streamer or other viewers. Some viewers and streamers reported that at some points the chat was moving too fast for them to effectively read every message, a known issue within large live streaming chat channels [10]. We conducted a coding analysis of chat messages to build an understanding of the conversation. Codes emerged through the analysis relative to our research questions. The most common codes included viewer responses to other participants (13.8%), commentary on the experience (13.6%), discussing the prototype (13.6%), viewer reactions to events (10.9%), and questions about the experience (9.7%). Interestingly, viewers often made requests (7.4%) of the streamers via text chat suggesting how they should stream or participate in the event:

> *Turn the phone sideways* RV71, to R1.
> *Go to the nearest venue!* RV72, to R2.
> *haha...[R3], somebody else is livestreaming at the same venue as you, you should find them!!!!* RV94, to R3.
> *Can you ask him where he got that hat?* RV108, to R4.

These requests illustrate the level of engagement some viewers had with the streamers in shaping how the event was covered. It even included coordination among the streamers, as viewers recognized other streamers at the event.

Understanding Event-Wide Text Chat

We were particularly interested in evaluating how participants understood the global text chat and its impact on the experience. Many viewers indicated that, while they were able to understand which viewers were watching each stream, it was confusing. Many viewers also expressed wanting to see only chats from viewers on the same stream or at least be able to filter out chats from other streams. When asked to rate the usefulness of the text chat modality, 90% of viewers responded positively in the post questionnaire that being able to see chat messages from the same stream they were watching was useful, only 57% reported that seeing chat messages from viewers in other streams was useful. The event-wide chat did enable viewers and streamers to maintain awareness of what was going on in other streams. R4 reported he experienced *"greater awareness about the event as a whole and what other streamers were doing via event wide chat."* While an event-wide chat has clear benefits, viewers and steamers need to be able to quickly identify messages in their active stream.

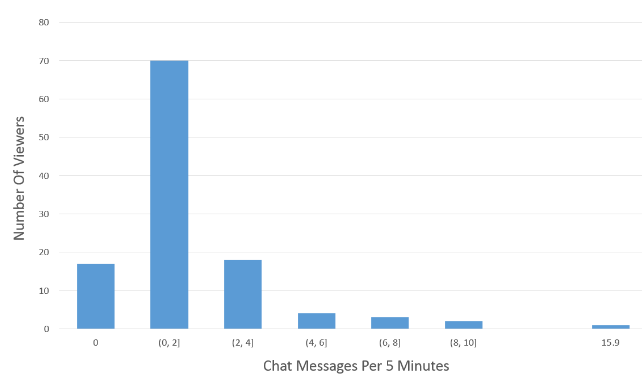

Figure 4. Distribution of chat frequency among viewers.

Despite some initial confusion, 3 of the 4 Rivulet streamers indicated that they could easily understand which chats were coming from their viewers. The binary nature (only 2 colors) of the chat visualization on the broadcasting client (Figure 2) made it easier to understand which chats were coming from viewers of their stream.

Push-To-Talk Enables High Profile Participation

PTT was used by significantly fewer viewers than text chat. Only 14 out of the 115 recruited viewers attempted to broadcast audio. Furthermore, 50% of the messages failed to be understood by the streamer or viewers. This was due to a number of issues including viewers' microphone configurations, the system prematurely cutting off participants' audio, or the audio being too quiet to hear. Since the Jazz Walk was a live music event, the ambient noise level at the event frequently drowned out incoming PTT audio.

For the half (42 out of 83) of the PTT messages that were comprehensible, messages ranged from asking questions, making jokes, commenting on the stream, asking if the speaker could be heard, or simply saying *"Hi!"*. In several instances a streamer and viewer were able to have a short conversation through the stream audio and PTT. Unlike chat, PTTs were mostly directed at the streamer, not other viewers.

When asked what they liked about PTT, many viewers indicated that they liked the instant, high profile communication with the streamer. According to RV18: *"It's loud and heard, so it's easily recognizable. It would be easy to make a point that stands out above the wall of text."* Other viewers seemed to appreciate others' use of PTT. RV45 indicated that: *"While I did not personally use it, listening to other people interact with the streamer was neat. Being able to influence their decision making was the best part."*

Other viewers had concerns about the value of PTT. RV63 felt like PTT would *"just encourage people to act out"*, and R3 indicated that she would like the option to mute particular viewers who were trolling her. While we did not explore this issue directly, there is a clear need to support boundaries of use for such a high-impact communication modality.

We transcribed and coded PTT messages for content and to whom they were directed. This revealed that PTT messages were integrated into the conversation in the stream where almost all messages either clearly implied a response from the streamer, or were in direct response to the streamer. This is in contrast to text chat, which more often was just commentary that did not respond to or imply a response.

While there does not appear to be a direct correlation between the number of chats a user sent and how often they used PTT, 12 of the 14 PTT users were in the top 35% of the most frequent text chatters during the event. This leads us to suspect that PTT appeals to already engaged viewers, who are looking for a more direct means to participate.

Our timing strategy of allowing only 10 seconds of speaking time seemed to keep people from dominating the channel and gave the streamers an opportunity to respond. None of the streamers indicated that they felt overwhelmed by the incoming audio. We also suspect that PTT audio may be socially intimidating and thus self-regulating. RV34, who sent the third most chats during the experience, but not any PTTs, indicated that s/he was scared to use the feature.

Despite technical issues, the results indicate that PTT audio provided new opportunities for participating in live streaming experiences. PTT proved to be *hotter* than text chat. It is high fidelity and affords a unique means for highly engaged participants to have impact. Furthermore, PTT is *cooler* than live video, with more space for participation.

Hearts are Noisy

The hearts feature was used extensively in both the Rivulet and Periscope conditions. While we do not have exact numbers for the Periscope streams, a total of 24,523 hearts were sent through Rivulet. While 22 viewers did not send any hearts, 21 viewers sent more than 200 hearts over the duration of the study. Ultimately, we observed it was very easy for the hearts modality to be dominated by a few viewers. For example, one outlier alone sent 8686 hearts.

When we asked Rivulet viewers if they thought hearts were useful to send or see using a five point Likert scale, responses averaged 3.26 (s=1.18) and 3.19 (s=1.32) respectively. Results in the context of Persicope were similar. This lukewarm perception of hearts seems counter-intuitive given the apparent popularity of the feature in Periscope. However, what we did find is that 6 of the 8 streamers thought hearts provided useful feedback to them about their streams (the other 2 were neutral). This leads us to suspect that hearts, at least in their current form, are more meaningful to streamers.

Despite this finding, hearts played a significant role in informing viewers when they should switch streams. RV82 reported that *"when people would hit hearts on other streams I would pop over and see what was going on."* However, since hearts were so easy to generate in rapid succession, it was easy for one viewer to create a potentially distracting signal with hearts. As RV102 reported *"I think the hearts were more the result of someone clicking for no reason than the video's content."* Similarly, in the case of R3's stream, when she pretended to eat incoming hearts, sending hearts became more of a game and less a signal of interesting content. Given the noisy nature of sheer heart throughput, a more valuable signal might be derived by normalizing the number of hearts by the viewer's typical heart sending rate or from the number of unique heart senders at one time.

Viewing Multiple Streams

We found that being able to view multiple live streams and readily switch between them had several immediate impacts on viewers' experience of the event. Drawing from participant responses to Likert questions in the post questionnaire, we found the significant benefits of the Rivulet prototype over Periscope (see Table 2). The results

Likert Question	Periscope-μ	Rivulet-μ	Mann-Whitney U	p-value
Q1	3.68	4.21	4320	<.001
Q2	3.45	4.15	3698	<.001
Q7	3.80	4.25	4630	<.001
Q9	3.71	4.10	5059	<.004
Q10	3.60	4.03	4788	<.001
Q11	3.47	4.15	3804	<.001
Q12	3.15	3.83	4081	<.001

Table 2: Comparison of selected Likert response means across conditions, with 1 = Strongly Disagree and 5 = Strongly Agree. The difference in responses across conditions were analyzed for significance using Mann-Whitney U non-parametric tests with Bonferroni correction applied.

for Q1, Q2, Q7, and Q12 indicate that Rivulet effectively enabled participants to watch several live streams simultaneously. We describe the impact of multiple streams through a discussion of how and when viewers switched streams, how viewers were able to find interesting streams to watch, and emergent multi-stream experiences.

Switching Streams

We logged how often participants voluntarily switched between different streams in Rivulet, removing automatic stream switches that occurred when a stream ended. Figure 5 shows the distribution of viewers by how often they switched streams. While many viewers switched infrequently or not at all, a significant portion switched streams regularly. On average, 32% of viewers switched streams at least every 2 minutes. One extreme viewer switched streams a total of 42 times. During the first 60 minutes of the study, we observed a diverse distribution of viewers across the streams. As shown in Figure 3, viewers actively switched to new streams when they appeared.

Being able to watch and switch between multiple streams enabled viewers to find and participate in streams that were of interest to them. Despite covering the same event, each stream was different in content and activity from the others. R1 and R4 focused more on the musical performances while R2 and R3 focused more discussing the event and interacting with viewers. Different viewers reported enjoying both of these types of streams and switching streams for an experience that was of most interest to them.

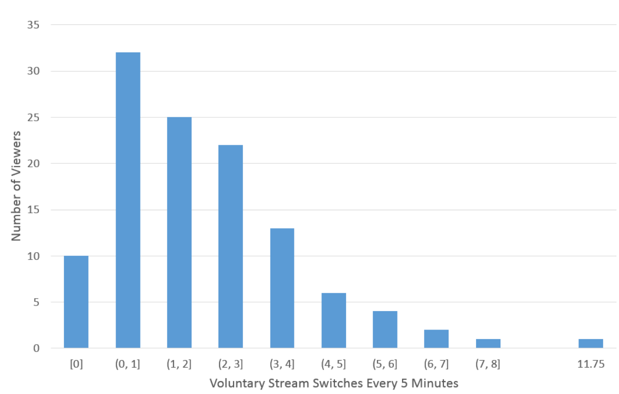

Figure 5: Distribution of Viewers by Rate of Stream Switches.

This observation that viewers' personal interests drove their varied viewing behavior is consistent with prior work such as Velt et al. [26] who also explored a music festival. Hamilton et al. [10] also observed that viewers are drawn to certain streams either for their content (such as live music) or to primarily interact with the stream and its community. We argue that by combining different kinds of streams and enabling viewers to explore and participate in them simultaneously, we can support live experiences that are more personally meaningful to individual viewers.

Viewers used a combination of signals to inform switching between the different streams. Many viewers indicated that they switched streams when they saw an interesting conversation occurring in chat. RV66 reported *"being able to see the different conversations from the different streams let you know which stream was the hottest at that moment"* (hottest meaning most interesting, not McLuhan's *hot)*. Other viewers reported monitoring the live previews to watch for interesting content. RV13 reported *"when there was a change of scenery, or when someone changed the camera angle to their face, it made me switch to see what was going on, to hear the audio."* Despite the ambiguity of hearts, many viewers reported choosing streams based on heart activity (see *Hearts are Noisy*). Viewers also returned to streams of viewers they had been watching previously. In several cases, we observed viewers exclaiming in chat *"[Streamer] is back!"* after a streamer restarted their stream.

Cross-Stream Experiences

The multi-stream nature of the Rivulet prototype enabled several experiences that viewers noted as exciting. In one case, viewers noticed that R3 and R4 were streaming at the same part of the event and they could see R4 through R3's stream. They pointed out R4 to R3, and one viewer switched over to R4's stream and suggested that he go over and talk to R3. In a similar case, R3 and R2 randomly encountered each other while walking down the sidewalk. They then streamed each other for a while and had a short discussion about the event. RV20 reported that *"the most interesting thing that happened while I was watching was when two "hosts" met each other. It was a little surreal."*

We note that without aggregating live streams together, these kinds of cross-stream experiences are virtually impossible in existing platforms. By coordinating streams and communication modalities together around an event, viewers are more aware of stream and can interact across streams.

Sense of Community

Over the course of the experience, it appeared as if a temporary sense of community emerged within the audience of the Rivulet experience. When answering Q9, 94 of the 115 Rivulet viewers agreed (37 strongly agreed) that they felt like they were part of a community. This feeling was significantly greater in Rivulet compared to Periscope (See Table 2: Q9).

We also found that viewers in Rivulet felt significantly more connected to the people streaming than those in Periscope (See Table 2: Q10), and more connected to other viewers (See Table 2: Q11). Many viewers also indicated in their free responses that they felt they were part of community during the experience. As RV110 said:

> *"What I liked best was how easy it was for the streamers to interact with viewers and the close-knit feeling that I gained from watching several streams. It felt like I was a part of the community."*

Study Design Implications

Recruiting a relatively large number of viewers from Mechanical Turk enabled us to observe a live experience at-scale through the Rivulet prototype. However, given that both streaming and viewing participants were compensated to participate in the event, the study cannot be considered an organically emerging experience. Thus, there are some inherent issues with the ecologic validity of the experience.

For example, while 100% of the participants watched for at least the requisite 20 minutes, only 12 Rivulet viewers watched longer than 30 minutes. It appears that most participants left after the minimum required viewing time in the HIT. By 45 minutes into the study, only about 20 viewers remained, resulting in a relatively short window of time when Rivulet had a reasonably sized audience. Future work could look at different ways to design this kind of study to engender more ecologically valid viewer behavior.

Additionally, looking at the length of streams shared during the study, almost all of the streamers (both on Rivulet and Periscope) were active for most of the study duration. This contrasts to the brief (5-10 minute) streams typically seen on Periscope. It is unclear if this was because they had more viewers than they were accustomed to or they felt like they were expected to because of the study.

We also note that during the study a significant amount of chats mentioned Mechanical Turk (6.7%). While these messages might have distracted from the shared experience, they may also have helped participants connect through their shared experiences on Mechanical Turk. Further work is needed to investigate the social implications of using turkers as participants in live social systems.

CONCLUSION

We built and field tested at-scale the Rivulet prototype for experiencing multiple streams of an event. Viewers used all modalities (text chat, PTT, and hearts) to engage with the streamers and with the viewers within and across streams in the event. Their engagement included shaping the way that streamers were covering the event and working to inform other viewers as streams started or stopped. Taken together, we see evidence that multi-stream experiences around events afford new opportunities for participating in and forming impromptu communities. We reflect here on our second research question, namely how people used the various communication modalities in Rivulet on the spectrum of *cool* to *hot* media [10, 19].

It is apparent that lightweight, *cool* signals, like hearts, are a compelling emerging participation modality. While we saw many people engaging through the hearts modality, displaying all those hearts may imply more importance than is warranted. When used in isolated live streams, as in Periscope, hearts may give meaningful feedback to the streamer and viewers. But in the context of multiple streams, people used them as a cue to switch to a stream, only to find out they did not indicate what they expected. We argue that work needs to be done further refine these types of modalities. For example, visualizing the proportion of people that give hearts, rather than the total number of hearts, may be a more useful signal of which streams are interesting.

Text chat is a warmer communication modality that is used less than hearts. We redesigned text chat as a communication modality to bridge across multiple streams, and foster an event-centric experience. We found that this approach had clear benefits, leading to interesting cross-stream interactions. However, there is a need to more clearly present which chats are from people watching the same stream versus other streams.

Additionally, we found that new modalities like PTT, a modality *hotter* than text chat and *cooler* than live video, supported compelling new participatory experiences. While only a small subset of highly engaged users sent PTTs, they engendered a higher level of engagement by immediately responding to or evoking responses from streamers. PTT afforded a new opportunity for higher impact participation. We argue that there is a need for continued investigation of new communication modalities to understand the roles they can play in participatory live experiences.

Finally, with regards to our first research question, we found that multi-stream experiences led to interesting cross-stream interactions. Viewers were excited about encounters involving multiple streamers. They were also able to easily find and participate in streams that addressed their interests and desire for engagement.

We note that we were only able to observe interactions in the context of this one event. Future work could examine multi-stream interactions around different types of events such as parades, conventions, sporting events, political debates, or protests at both larger and smaller scales than what we observed. We expect that different events at different scales will exercise communication modalities in different ways, helping us further learn how to support participation in multi-stream events. Rivulet also did not explore streamer-to-streamer communication, which could become more important in events with more streams. As live streaming continues to evolve and practices emerge, we believe that supporting interaction among multiple streams from the same event is an important, new form of social media communication that is ripe for future work.

REFERENCES

1. Edward Anstead, Steve Benford, and Robert Houghton. 2016. MarathOn Multiscreen: Group Television Watching and Interaction in a Viewing Ecology. In Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing (CSCW '16). ACM, New York, NY, USA, 405-417. DOI=http://dx.doi.org/10.1145/2818048.2820003
2. Frank R. Bentley and Michael Groble. 2009. TuVista: meeting the multimedia needs of mobile sports fans. In Proceedings of the 17th ACM international conference on Multimedia (MM '09). ACM, New York, NY, USA, 471-480. DOI=http://dx.doi.org/10.1145/1631272.1631337
3. Michael S. Bernstein, Joel Brandt, Robert C. Miller, and David R. Karger. 2011. Crowds in two seconds: enabling realtime crowd-powered interfaces. In Proceedings of the 24th annual ACM symposium on User interface software and technology (UIST '11). ACM, New York, NY, USA, 33-42. DOI=http://dx.doi.org/10.1145/2047196.2047201
4. Audubon Dougherty. 2011. Live-streaming mobile video: production as civic engagement. In Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services (MobileHCI '11). ACM, New York, NY, USA, 425-434. DOI=http://dx.doi.org/10.1145/2037373.2037437
5. A. Engström, M. Esbjörnsson, and O. Juhlin. 2008. Mobile collaborative live video mixing. In Proceedings of the 10th international conference on Human computer interaction with mobile devices and services (MobileHCI '08). ACM, New York, NY, USA, 157-166. DOI=http://dx.doi.org/10.1145/1409240.1409258
6. Arvid Engström, Mark Perry, and Oskar Juhlin. 2012. Amateur vision and recreational orientation:: creating live video together. In Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work (CSCW '12). ACM, New York, NY, USA, 651-660. DOI=http://dx.doi.org/10.1145/2145204.2145304
7. Arvid Engstrom, Mattias Esbjornsson, Oskar Juhlin, and Mark Perry. 2008. Producing collaborative video: developing an interactive user experience for mobile tv. In Proceedings of the 1st international conference on Designing interactive user experiences for TV and video (UXTV '08). ACM, New York, NY, USA, 115-124. DOI=http://dx.doi.org/10.1145/1453805.1453828
8. Arvid Engström, Goranka Zoric, Oskar Juhlin, and Ramin Toussi. 2012. The mobile vision mixer: a mobile network based live video broadcasting system in your mobile phone. In Proceedings of the 11th International Conference on Mobile and Ubiquitous Multimedia (MUM '12). ACM, New York, NY, USA, , Article 18 , 4 pages. DOI=http://dx.doi.org/10.1145/2406367.2406390
9. Martin D. Flintham, Raphael Velt, Max L. Wilson, Edward J. Anstead, Steve Benford, Anthony Brown, Timothy Pearce, Dominic Price, and James Sprinks. 2015. Run Spot Run: Capturing and Tagging Footage of a Race by Crowds of Spectators. In Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15). ACM, New York, NY, USA, 747-756. DOI=http://dx.doi.org/10.1145/2702123.2702463
10. 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 SIGCHI Conference on Human Factors in Computing Systems (CHI '14). ACM, New York, NY, USA, 1315-1324. DOI=http://dx.doi.org/10.1145/2556288.2557048
11. Hamrick, B. MultiTwitch. http://www.multitwitch.tv/.
12. Hilary Hutchinson, Wendy Mackay, Bo Westerlund, Benjamin B. Bederson, Allison Druin, Catherine Plaisant, Michel Beaudouin-Lafon, Stéphane Conversy, Helen Evans, Heiko Hansen, Nicolas Roussel, and Björn Eiderbäck. 2003. Technology probes: inspiring design for and with families. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '03). ACM, New York, NY, USA, 17-24. DOI=http://dx.doi.org/10.1145/642611.642616
13. Hyungeun Jo and Sungjae Hwang. 2013. Chili: viewpoint control and on-video drawing for mobile video calls. In CHI '13 Extended Abstracts on Human Factors in Computing Systems (CHI EA '13). ACM, New York, NY, USA, 1425-1430. DOI=http://dx.doi.org/10.1145/2468356.2468610
14. Oskar Juhlin, Arvid Engström, and Elin Önnevall. 2014. Long tail TV revisited: from ordinary camera phone use to pro-am video production. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14). ACM, New York, NY, USA, 1325-1334. DOI=http://dx.doi.org/10.1145/2556288.2557315
15. Oskar Juhlin, Arvid Engström, and Erika Reponen. 2010. Mobile broadcasting: the whats and hows of live video as a social medium. In Proceedings of the 12th international conference on Human computer interaction with mobile devices and services (MobileHCI '10). ACM, New York, NY, USA, 35-44. DOI=10.1145/1851600.1851610 http://doi.acm.org/10.1145/1851600.1851610
16. kbmod. kbmod multistream. http://kbmod.com/multistream/.
17. Lyndon Kennedy and Mor Naaman. 2009. Less talk, more rock: automated organization of community-contributed collections of concert videos. In

Proceedings of the 18th international conference on World wide web (WWW '09). ACM, New York, NY, USA, 311-320. DOI=http://dx.doi.org/10.1145/1526709.1526752

18. Seungwon Kim, Sasa Junuzovic, and Kori Inkpen. 2014. The Nomad and the Couch Potato: Enriching Mobile Shared Experiences with Contextual Information. In Proceedings of the 18th International Conference on Supporting Group Work (GROUP '14). ACM, New York, NY, USA, 167-177. DOI=http://dx.doi.org/10.1145/2660398.2660409
19. McLuhan, M. Understanding media: the extensions of man. McGraw-Hill, 1964.
20. Blair Nonnecke and Jenny Preece. 2000. Lurker demographics: counting the silent. In Proceedings of the SIGCHI conference on Human Factors in Computing Systems (CHI '00). ACM, New York, NY, USA, 73-80. DOI=http://dx.doi.org/10.1145/332040.332409
21. 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 (CHI '15). ACM, New York, NY, USA, 2573-2582. DOI=http://dx.doi.org/10.1145/2702123.2702257
22. Marco Sá, David A. Shamma, and Elizabeth F. Churchill. 2014. Live mobile collaboration for video production: design, guidelines, and requirements. Personal Ubiquitous Comput. 18, 3 (March 2014), 693-707. DOI=http://dx.doi.org/10.1007/s00779-013-0700-0
23. Guy Schofield, Tom Bartindale, and Peter Wright. 2015. Bootlegger: Turning Fans into Film Crew. In Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15). ACM, New York, NY, USA, 767-776. DOI=http://dx.doi.org/10.1145/2702123.2702229
24. John C. Tang, Gina Venolia, and Kori Inkpen. Meerkat and Periscope: I Stream, You Stream, Apps Stream for Live Streams. In Proceedings of the 34th Annual ACM Conference on Human Factors in Computing Systems (CHI '16). ACM, New York, NY, USA, in press.
25. Tazaki, A. InstantShareCam: Turning Users From Passive Media Consumers to Active Media Producers. Workshop Investigating new user experience challenges in iTV: mobility & sociability, 24th Annual ACM Conference on Human Factors in Computing Systems (CHI '06).
26. Raphael Velt, Steve Benford, Stuart Reeves, Michael Evans, Maxine Glancy, and Phil Stenton. 2015. Towards an Extended Festival Viewing Experience. In Proceedings of the ACM International Conference on Interactive Experiences for TV and Online Video (TVX '15). ACM, New York, NY, USA, 53-62. DOI=http://dx.doi.org/10.1145/2745197.2745206
27. Sami Vihavainen, Sujeet Mate, Lassi Liikkanen, and Igor Curcio. 2012. Video as memorabilia: user needs for collaborative automatic mobile video production. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12). ACM, New York, NY, USA, 651-654. DOI=http://dx.doi.org/10.1145/2207676.2207768
28. Andrew M. Webb, Chen Wang, Andruid Kerne, and Pablo Cesar. 2016. Distributed Liveness: Understanding How New Technologies Transform Performance Experiences. In Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing (CSCW '16). ACM, New York, NY, USA, 432-437. DOI=http://dx.doi.org/10.1145/2818048.2819974
29. Stefan Wilk, Stephan Kopf, and Wolfgang Effelsberg. 2015. Video composition by the crowd: a system to compose user-generated videos in near real-time. In Proceedings of the 6th ACM Multimedia Systems Conference (MMSys '15). ACM, New York, NY, USA, 13-24. DOI=http://dx.doi.org/10.1145/2713168.2713178
30. Takuro Yonezawa and Hideyuki Tokuda. 2012. Enhancing communication and dramatic impact of online live performance with cooperative audience control. In Proceedings of the 2012 ACM Conference on Ubiquitous Computing (UbiComp '12). ACM, New York, NY, USA, 103-112. DOI=http://dx.doi.org/10.1145/2370216.