Automated Video Description for Blind and Low Vision Users

Aditya Bodi
San Francisco State University
San Francisco, CA, United States
abodi@sfsu.edu

Pooyan Fazli
San Francisco State University
San Francisco, CA, United States
pooyan@sfsu.edu

Shasta Ihorn
San Francisco State University
San Francisco, CA, United States
sihorn@sfsu.edu

Yue-Ting Siu
San Francisco State University
San Francisco, CA, United States
ysiu@sfsu.edu

Andrew Scott
San Francisco State University
San Francisco, CA, United States
ats@sfsu.edu

Lothar Narins
San Francisco State University
San Francisco, CA, United States
lnarins@sfsu.edu

Yash Kant
Georgia Tech
Atlanta, GA, United States
ykant6@gatech.edu

Abhishek Das
Facebook AI Research
Menlo Park, CA, United States
abhshkdz@fb.com

Ilmi Yoon
San Francisco State University
San Francisco, CA, United States
ilmi@sfsu.edu

ABSTRACT

Video accessibility is crucial for blind and low vision users for equitable engagements in education, employment, and entertainment. Despite the availability of professional description services and tools for amateur description, most human-generated descriptions are expensive and time consuming, and the rate of human-generated descriptions simply cannot match the speed of video production. To overcome the increasing gaps in video accessibility, we developed a system to automatically generate descriptions for videos and answer blind and low vision users' queries on the videos. Results from a pilot study with eight blind video aficionados indicate the promise of this system for meeting needs for immediate access to videos and validate our efforts in developing tools in partnership with the individuals we aim to benefit. Though the results must be interpreted with caution due to the small sample size, participants overall reported high levels of satisfaction with the system, and all preferred use of the system over no support at all.

CCS CONCEPTS

• **Human-centered computing → Accessibility technologies**; **Accessibility systems and tools**.

KEYWORDS

Video Accessibility; Video Description; Blind and Low Vision Users; Artificial Intelligence

ACM Reference Format:
Aditya Bodi, Pooyan Fazli, Shasta Ihorn, Yue-Ting Siu, Andrew Scott, Lothar Narins, Yash Kant, Abhishek Das, and Ilmi Yoon. 2021. Automated Video Description for Blind and Low Vision Users. In *CHI Conference on Human Factors in Computing Systems Extended Abstracts (CHI '21 Extended Abstracts), May 8–13, 2021, Yokohama, Japan.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3411763.3451810

1 INTRODUCTION

In an era of commonly-adopted virtual environments, online videos are a common medium for engagement. Nevertheless, the vast majority of videos remain inaccessible to blind and low vision people. These individuals face unique challenges in our modern environment, where much critical information related to education, employment, entertainment, and community is presented in the form of digital videos. Inaccessible information can result in social exclusion or become life threatening if individuals require access to it in order to make decisions related to their health and safety. A longstanding way to bridge this gap is by adding description to videos. Video description, also referred to as audio description or simply description, facilitates non-visual access to visual components of a video by providing a narrative audio track that is synchronized with the video. Narrations relay information about settings, actions, on-screen text, and any other visual information that would otherwise be missed.

Professional video descriptions in live and recorded formats have been promoted since the 1920s; however, similar to the advent of novice-created videos for social media, such as YouTube, video description has evolved to warrant contributions from non-professionals who can fill in more immediate needs for video accessibility. This shift from professional to amateur description became possible in 2013, when the Smith-Kettlewell Eye Research Institute (SKI) launched a web-based tool called YouDescribe [35]. YouDescribe enables anyone to record and upload descriptions of YouTube videos. The platform has resulted in a community of more than 3000 volunteer describers and an active wish list where blind and low vision viewers can request descriptions for videos. However, 92.5% of videos on the YouDescribe user wish list remain undescribed. The time, training, and confidence needed to create high quality descriptions were hypothesized to limit potential describers from completing more descriptions. To address these barriers, we

CHI '21 Extended Abstracts, May 8–13, 2021, Yokohama, Japan

ACM ISBN 978-1-4503-8095-9/21/05…$15.00
https://doi.org/10.1145/3411763.3451810

designed and developed a system to automatically generate video descriptions and answer blind and low vision users' queries on the videos without sighted assistance. We hypothesize that greater effectiveness and usability of the developed system will promote broader equity in video and information accessibility by reducing dependence of blind viewers on sighted volunteers.

1.1 Contributions

Video descriptions typically offer information in two formats: 1) *inline descriptions* that play concurrently with a video and embed information within natural gaps in a video's audio; and 2) *extended descriptions* that play while a video is paused, so that more information can be conveyed without imposing time limits. This paper investigates the application of a new video accessibility system that maintain the two formats for description with innovations regarding the mode and delivery. The system consists of two main parts: 1) an artificial intelligence (AI)-based tool, called NarrationBot, that generates inline or extended *baseline descriptions* focusing on describing scenes in the video; and 2) an AI-based tool, called InfoBot, that delivers extended *on-demand descriptions* by pausing the video and providing additional information as dictated by a viewer's queries. InfoBot facilitates blind and low vision viewers' engagement with videos through asking questions and requesting descriptions. We evaluated the system with eight blind and low vision users. Though the results must be interpreted with caution due to the small sample size, participants overall reported high levels of satisfaction with the system under different conditions, and all preferred use of the system over no support at all.

2 BACKGROUND AND STATE OF THE ART

There has been a wide range of work performed in the field of human-computer interaction (HCI) and accessibility for blind and low vision users. Advances have occurred in visual graphics [18], data visualizations [33], maps [22, 23], programming [26], and video games [3, 13, 28]. For video accessibility, tools have been developed to facilitate authoring descriptions by sighted describers. LiveDescribe [4] was the first known computer-assisted video description technology, followed by various prototype softwares that used computer vision techniques, such as face and text recognition, to extract visual content from videos [11, 12] and support describers' ability to create and refine descriptions [25]. These tools have become the foundation for systems that can automatically detect silent periods within a video in which descriptions can be inserted, pause a video when extended descriptions are warranted, and decrease the human labor needed to produce descriptions. However, none of these tools can generate partial or full descriptions automatically.

The machine learning approaches to video description, dominated by deep learning, are usually divided into two sequential stages: 1) visual content extraction or the encoding stage and 2) text generation or the decoding stage. For encoding, convolutional neural networks (CNNs) [31] are used to learn visual features, and for decoding, different variations of recurrent neural networks (RNNs), such as long short-term memory (LSTM) [15] and gated recurrent units (GRU) [6] are used [17]. Recent state-of-the-art methods [21, 30] have replaced the RNNs with BERT [9] due to the success of Transformers [32]. The output caption can be a single sentence or multiple sentences. Our work also uses CNNs and RNNs and focuses on generating descriptions from representative keyframes of the input video. In a recent work, we developed a human-in-the-loop machine learning (HILML) approach to facilitate novice describers' ease of authoring video descriptions [36, 37]. The developed system generates baseline video descriptions that are then revised by human volunteers. This human-AI collaboration produces high-quality video descriptions while requiring reduced effort from volunteer describers. In contrast, this work takes a step towards removing humans from the loop.

3 SYSTEM

We built a system for automatically generating baseline and on-demand descriptions for videos. There are several unique features to each and common features for both. Figure 1 shows the overall architecture of the developed system.

3.1 Baseline Description

Baseline descriptions inform blind and low vision users about the visual components of a video, such as the main objects; actors, the spatial relations and interactions between them, and their actions; on-screen text; and the setting, costumes, and lighting. We developed a tool, called NarrationBot, consisting of various modules to automatically generate baseline descriptions. A benefit of our modular approach is that as technology matures in the future, modules can be replaced by their improved counterparts. Below we explain different modules of the developed tool.

3.1.1 Keyframe Selection. YouTube videos normally play at 30 or 60 frames per second, too many frames to efficiently work with and not much visually changing between them. Many of these frames capture a blurry, unclear snapshot of objects in the scene. Hence, this module selects a subset of frames, or *keyframes*, which represent the main events in the video clearly. To select the keyframes, we sample ten frames per second of video and run a pre-trained YOLOv3 [27] object detection model on each of the extracted frames. Each frame is given a score based on the confidences of the detected objects. Since the object detection model often returns many overlapping detections of the same object, we choose only the highest confidence detection of each distinct object in the frame. If we let $c_{o,i,f}$ denote the confidence of the ith instance of the detection of object o in the frame at timestamp f, the frame score is given by

$$\text{score}(f) = \sum_{o} \max_{i} c_{o,i,f}^2. \tag{1}$$

After manual tuning the granularity of our keyframe selection algorithm, we found that extracting one keyframe every six seconds on average led to best results while also keeping the compute costs manageable. Concretely, given the previous keyframe timestamp F_n and a target interval $T = 6$ between keyframes, we choose the next keyframe timestamp F_{n+1} to be

$$F_{n+1} = \underset{f}{\text{argmax}}\, \text{score}(f)(f - F_n)(F_n + 2T - f). \tag{2}$$

3.1.2 Image Caption Generation. This module uses the Pythia [17] caption generation model to create a description for every selected

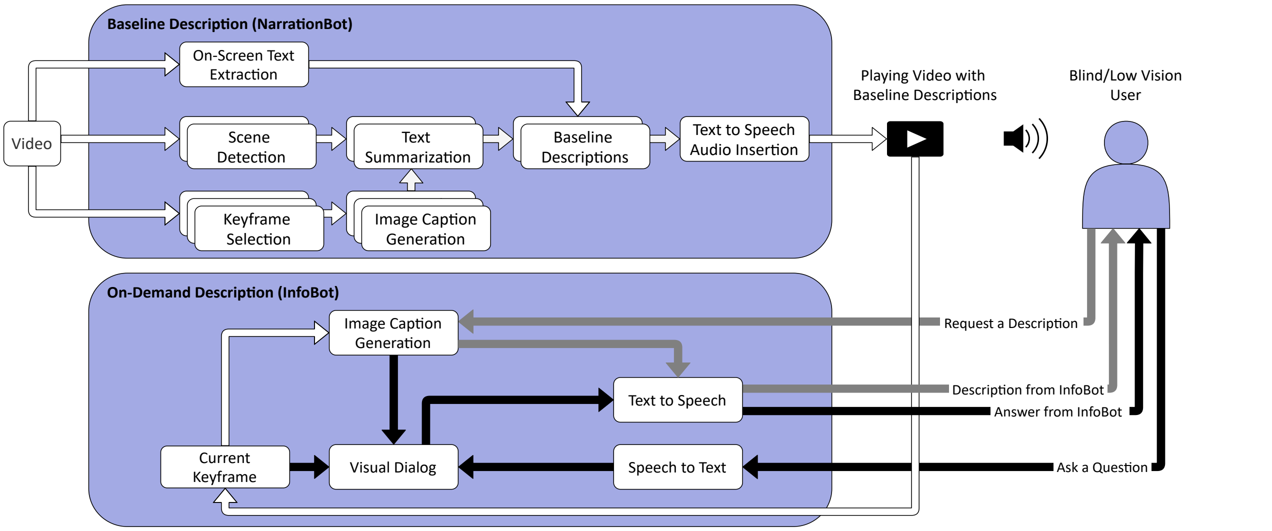


Figure 1: The system architecture for generating baseline (NarrationBot) and on-demand (InfoBot) descriptions for videos.

keyframe. The Pythia model uses a Bottom-up-Top-Down (BuTD) [1] attention network for visually-grounded language modeling. The bottom-up part is composed of two CNNs to identify objects and annotate bounding boxes in the images. The top-down part, composed of two LSTM networks, conditions on these detected objects through soft attention to generate text. Object detections from the bottom-up network are passed to the top-down language model for next word prediction. The bottom-up network was trained on ImageNet [8], and the top-down network was trained on the COCO dataset [20].

3.1.3 Scene Detection. This module partitions the video into a sequence of scenes of varying time spans using Microsoft Azure Video Indexer [16]. Scenes are defined as a set of consecutive frames that are semantically related and temporally adjacent, depicting a high-level concept or story. The scene is our base unit for generating and embedding baseline descriptions within a video. We use the information from the dialog timestamps, extracted by ListenByCode API [5], to merge the scenes having a continuous dialog element.

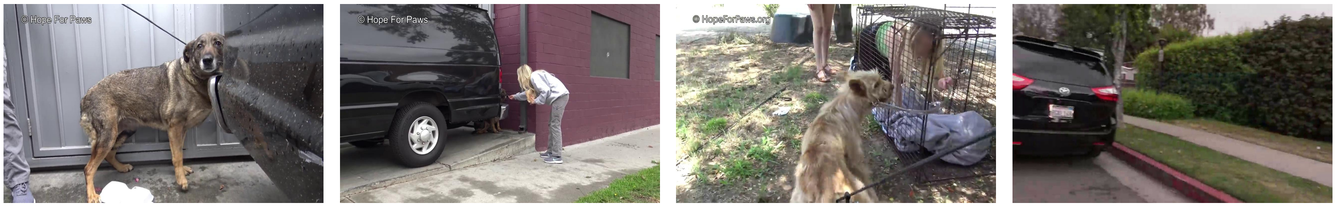
3.1.4 On-Screen Text Extraction. Text is a rich source of information in videos. This module extracts captions and scene text (e.g., license plates, building and street signs, handwritten documents) that convey information that may not be present in the audio. Using the selected keyframes to extract the on-screen text proved to be less than ideal because the text would occasionally be in the process of fading in or out since it is not of interest to the object detection model we use for selecting keyframes. Thus, we run an optical character recognition (OCR) API [2] on every extracted frame to ensure we do not miss any text. This produces a large collection of texts often repeated from frame to frame, which we must then filter and select the best representatives out of. We split the detected texts into clusters based on similarity across consecutive frames using a Levenshtein distance metric [19]. After choosing one text from each cluster in which the text remains very similar for at least five frames, we do a further pass in which we remove non-ASCII characters and remove any text that has appeared at least three times previously in the chosen texts. This is to prevent watermark logos from being read every time the text on screen is read.

3.1.5 Text Summarization. This module generates a baseline description for each scene by summarizing the captions generated for the selected keyframes. We pick the three most distinct but also most repeated captions in the scene by calculating their pairwise similarity scores and concatenate them. We use the BLEU score [24] to obtain the similarity between two captions. The captions with a BLEU score of above 40% are put in the same cluster. The clusters are then sorted by the number of captions with the sum of BLEU scores of all individual captions in each cluster breaking ties. After sorting, one caption with the highest BLEU score is chosen from each of the top three clusters. The selected three captions are summarized into a baseline description for the scene.

3.1.6 Text to Speech and Audio Insertion. As the final part of the process, this module converts descriptions from text to speech and finds appropriate positions in the video to play them. A scene's description is played between the scene's start and end times. The system finds empty gaps for each scene in the video, which are sections with no background music, speech, or dialog. The audio description is played inline if the length of an empty gap is sufficient to insert it. Otherwise, the video is paused and an extended description is played.

3.2 On-Demand Description

On-demand descriptions, generated by InfoBot, provide more information in real time when requested or asked by a user. InfoBot works by first identifying a frame from the video at the point where the viewer decides to pause it. Occasionally, the point where the viewer pauses contains blurred objects (due to motion, for example) or lacks vital visual information, in which case InfoBot automatically refines the choice of keyframe to be close to but not exactly the point at which the video is paused. The two main functionalities of InfoBot are described below.



Question: What kind of dog is it?
InfoBot Answer: German Shepherd

Question: What is the woman wearing?
InfoBot Answer: A long sleeve shirt and pants

Question: How big is the dog?
InfoBot Answer: Medium sized

Question: Is there a car in this video?
InfoBot Answer: Yes

Figure 2: Blind and low vision participants' questions and answers with InfoBot.

3.2.1 Requesting a Description. Users can request a description at any instant of the video. The selected keyframe will have the description as a caption. This description is obtained and read to the user. This kind of functionality will give the user an option to know the information about the current instant of the video. The users can use this functionality by pressing the 'D' key on the keyboard. The video will be auto-paused while the additional description is being read out. The video will be auto-played once the complete description is read out.

3.2.2 Asking a Question. Users can pose natural language questions about visual elements of a video and get natural language responses back in real time. For automatically answering queries without humans in the loop, we make use of the visual dialog model [7]. This model is trained on 120k examples of question-answer dialogs paired with images from the COCO dataset [20] collected via a 2-person chat interface where one side cannot see the image and is tasked with asking questions to understand the image better. Specifically, given an image, dialog history consisting of a sequence of question-answer pairs, and a follow-up question, this model predicts a free-form natural language answer to the question. This model extracts attention-based [1] object detection features based on Mask-RCNN [14] for each image and combines these features with the dialog history in an LSTM network to predict the answer. Users can press the 'Q' key to ask a question on the video, which is recorded via a microphone. A pause at the end of the user's speech terminates the recording. We convert the question to the corresponding text. Next, we feed the relevant keyframe, the user's question, and the previous dialog history (if available) as inputs to the visual dialog model to generate an answer. The answer received from the visual dialog model is converted back to audio via a text-to-speech API and read aloud to the user. The visual dialog model is capable of handling multiple questions allowing the user to ask/clarify their questions better. Hence, users can ask as many questions as they wish at any particular instance of the video. Figure 2 shows sample questions and answers our blind and low vision participants had with InfoBot.

4 EXPERIMENTS AND USER STUDY

4.1 Experimental Design

We used a mixed methods design to evaluate satisfaction with the system for blind and low vision users. For the qualitative component, we solicited written feedback to open-ended questions regarding participants' opinions about the system and its utility. For the quantitative assessment, we used a randomized, multivariate repeated measures design with six system conditions outlined below. Six videos[1] with similar content and length were chosen from the YouDescribe wish list to be used in combination with the system. Each participant was administered all six conditions across all six videos in random order and with random assignment of system conditions to videos. The videos we used in the experiment were related to dog rescues. All videos were about five minutes and shared the same narrative components to maintain consistency between videos: background information, specific setting of the rescue, treatments for injuries, and an adoption. Before beginning the study, each participant also completed an interactive tutorial session to learn how to use NarrationBot and InfoBot while watching a video. All videos and the tutorial were presented in English and all InfoBot and NarrationBot support was provided in English as well. After successfully completing the tutorial, each participant was administered videos under each of the following six conditions:

Condition 1 – without NarrationBot and without InfoBot. This condition does not allow access to any tools of the system or provide any extra support.

Condition 2 – without NarrationBot and with InfoBot. This condition does not allow access to NarrationBot, but users can request descriptions or ask questions on the videos through the interactive InfoBot.

Condition 3 – with NarrationBot and without InfoBot. This condition has no InfoBot functionality but provides baseline descriptions via NarrationBot.

Condition 4 – with HILML and without InfoBot. This condition is similar to the third condition. In condition 4, however, baseline descriptions are provided by NarrationBot and then revised by sighted volunteers (HILML). The revised descriptions are converted to audio and played inline or extended. The audio voice is identical to that of the system. Users are not informed whether it is condition 3 or 4.

Condition 5 – with NarrationBot and with InfoBot. This condition allows users access to both tools of the system: NarrationBot and InfoBot. The baseline descriptions are provided by NarrationBot.

Condition 6 – with HILML and with InfoBot. This condition is similar to condition 5. The only difference is that the baseline descriptions are provided by NarrationBot and then revised by

[1] Note that this research was carried out at SFSU, and consistent with academic standards, these six YouTube videos were used only for the purpose of validating the idea(s) proposed in the paper. Figure 2 reproduced with permission from the video authors.

Table 1: InfoBot usage statistics. Mean number of uses per user with an 80% confidence interval.

	NarrationBot + InfoBot	HILML + InfoBot	InfoBot only	Total for all conditions
Number of questions asked	14.63 [7.49, 21.77]	3.00 [1.29, 4.71]	8.38 [3.55, 13.21]	8.67 [3.16, 14.18]
Number of descriptions requested	19.50 [15.64, 23.36]	8.00 [5.14, 10.86]	15.88 [7.79, 23.97]	14.46 [8.61, 20.31]
Total number of InfoBot uses	34.13 [23.54, 44.72]	11.00 [6.71, 15.29]	24.25 [11.72, 36.78]	23.13 [12.44, 33.82]

Table 2: User understanding, enjoyability, and satisfaction of videos under six different system conditions.

	Ability to understand video		Enjoyability of video		Overall satisfaction with video	
	Mean	Std. Dev.	Mean	Std. Dev.	Mean	Std. Dev.
Condition 1 – No support	2.38	1.60	2.75	1.75	2.63	1.30
Condition 2 – InfoBot only	4.25	1.58	4.13	1.25	4.50	1.31
Condition 3 – NarrationBot only	5.00	1.07	4.50	1.51	4.75	1.16
Condition 4 – HILML only	5.00	0.76	4.88	0.83	4.75	0.89
Condition 5 – NarrationBot + InfoBot	5.00	1.07	4.75	1.16	4.88	1.13
Condition 6 – HILML + InfoBot	5.50	0.53	4.75	0.71	4.88	0.83

sighted volunteers (HILML). The revised descriptions are converted to audio and played inline or extended. The audio voice is identical to that of the system. Users are not informed whether it is condition 5 or 6.

After each video, participants completed a brief survey to provide feedback on ease of use, comprehension, enjoyment, and overall satisfaction for the tools available during that video. All questions use a 6-point Likert-type scale, with higher scores indicating a more favorable rating. After completing all six videos, participants completed a final survey to provide demographic information and overall impressions of the system, including written qualitative responses. 8 participants with an average age of 38 and all fluent in English completed the study. One of these participants is low vision (large print for reading, requires audio description for videos), and seven are blind. Five participants are male and three are female; one is Asian-American, two are Latinx, three are White, and one participant chose not to disclose race/ethnicity. All participants have attended at least some college, and two have earned master's degrees. Participants indicated that they watch YouTube and videos more generally on a daily / weekly basis. Below, we present descriptive statistics on the use of the system and some qualitative responses from these participants.

4.2 Results

4.2.1 InfoBot Usage. InfoBot was used by each participant ∼23 times per five minute video, with roughly 9 of those uses in the service of asking a question ('Q' key), and 14 to request a description ('D' key). Table 1 shows the mean number of times the InfoBot description ('D' key) and question-answering ('Q' key) functions were used by participants. Though results must be interpreted with caution due to the small sample size, participants tended to use the description function more frequently than the question function across all conditions. We believe the relative ease of requesting descriptions instead of asking questions could have led to greater frequency of use for the description functionality.

These results also show that participants used InfoBot more frequently when the information received via other means was less accurate. In the InfoBot-only condition (which had no other information provided) and the NarrationBot + InfoBot condition (which had some other information provided automatically, but this information is not always accurate), they used InfoBot much more often than they did in the HILML + InfoBot condition (which has verified information provided automatically). It is interesting that participants used InfoBot more frequently in the NarrationBot + InfoBot condition than in the InfoBot-only condition, even though they had no other information in the latter.

4.2.2 User Satisfaction. Though the results must be interpreted with caution due to the small sample size, participants gave the highest scores for understandability and satisfaction to the HILML + InfoBot condition followed closely by both the NarrationBot + InfoBot condition and the HILML-only conditions as well as the NarrationBot-only condition. Participants also gave relatively positive scores to the InfoBot-only condition but gave the lowest overall scores to the no-support condition. The mean scores and standard deviations for all conditions are presented in Table 2.

After participants had viewed all of the videos, they took a final survey to provide qualitative feedback about their overall impressions of the system. These responses provide context and insight that would be difficult to assess via quantitative methods. Participants responded enthusiastically about the idea of this type of technology when providing this written feedback. For example, one participant wrote, *"These tools would be an excellent way to get descriptive information on certain video content that would make videos more enjoyable. The ability to apply AI to any video out there to get visual content described is a game changer for entertainment purposes."* Another said, *"Using these tools helped me understand the smaller details in a video, things I did not even realize I was missing. These tools have a lot of potential."*

5 DISCUSSION

Due to the small sample size associated with this pilot data, any interpretations must be made with caution; however, the data do point to interesting trends that may be emerging from the data. First, participants' lower scores for the no-support condition appear to indicate that they have a preference for any available tool that can help with video description. This was further supported in their qualitative responses; for example, one participant noted, *"I would definitely use them if they [were] available"* and another said, *"All forms of audio description should be welcomed, not just for us visually impaired people, but [also for] sighted people who don't want to look at a screen after a long day of work, or a long commute to and from work."* These results align with the evolution of ubiquitous computing [10][34] and ensuing emphasis on the role of technology in facilitating blind and low vision individuals' access to information [29].

Further, participants gave similar scores to the four baseline-only and baseline + InfoBot conditions (conditions 3–6). A larger sample size will allow for more variability as well as the ability to determine whether these four conditions are functionally the same for blind and low vision users or whether there are small but meaningful differences between them. A larger sample size and expanded qualitative analysis may also provide insight into the data regarding the InfoBot usage, which unexpectedly showed that participants used the InfoBot more often in the NarrationBot + InfoBot condition than they did in the InfoBot-only condition.

6 CONCLUSION AND FUTURE WORK

Ultimately, greater effectiveness and usability of the developed system will promote broader equity in video and information accessibility. By maintaining how and what information is generated dependent on human factors, AI-driven tools can potentially overcome gaps in video accessibility that human-driven tools could not accomplish alone. In order for AI-driven accessibility systems to truly be of service, tools and datasets must be developed in partnership with end users to ensure that they serve the intended purpose. For example, although our initial pilot was focused on improving access to YouTube videos, a participant identified a future implication to *"see [how] these tools [could be] implemented as a browser extension or be built directly into the YouTube experience, but also be available for other video services on the web and in apps."* Finally, our shift away from dependency on sighted human assistance better empowers blind and low vision individuals to dictate their own points of access.

REFERENCES

[1] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2018. Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. pp. 6077–6086.

[2] Google Cloud Vision API. Accessed Date 2021-01-10. https://cloud.google.com/vision/docs/ocr

[3] David Bar-El, Thomas Large, Lydia Davison, and Marcelo Worsley. 2018. Tangicraft: A Multimodal Interface for Minecraft. In *International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS)*. pp. 456–458.

[4] Carmen J Branje and Deborah I Fels. 2012. Livedescribe: Can Amateur Describers create High-Quality Audio Description? *Journal of Visual Impairment & Blindness* 106, 3 (2012), pp. 154–165.

[5] Listen by Code Speech-to Text API. Accessed Date 2021-01-10. https://www.listenbycode.com/

[6] Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*. pp. 1724–1734.

[7] Abhishek Das, Satwik Kottur, Khushi Gupta, Avi Singh, Deshraj Yadav, José MF Moura, Devi Parikh, and Dhruv Batra. 2017. Visual Dialog. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. pp. 326–335.

[8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A Large-Scale Hierarchical Image Database. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. pp. 248–255.

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding. *arXiv:1810.04805* (2018).

[10] Michael Friedewald and Oliver Raabe. 2011. Ubiquitous Computing: An Overview of Technology Impacts. *Telematics and Informatics* 28, 2 (2011), pp. 55–65.

[11] Langis Gagnon, Claude Chapdelaine, David Byrns, Samuel Foucher, Maguelonne Heritier, and Vishwa Gupta. 2010. A Computer-Vision-Assisted System for Videodescription Scripting. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR Workshops)*. pp. 41–48.

[12] Langis Gagnon, Samuel Foucher, Maguelonne Heritier, Marc Lalonde, David Byrns, Claude Chapdelaine, James Turner, Suzanne Mathieu, Denis Laurendeau, Nath Tan Nguyen, et al. 2009. Towards Computer-Vision Software Tools to Increase Production and Accessibility of Video Description for People with Vision Loss. *Universal Access in the Information Society* 8, 3 (2009), pp. 199–218.

[13] Eitan Glinert and Lonce Wyse. 2007. AudiOdyssey: An Accessible Video Game for both Sighted and Non-sighted Gamers. In *International Academic Conference on the Future of Game Design and Technology (Future Play)*. pp. 251–252.

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. pp. 770–778.

[15] Sepp Hochreiter and Jurgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation* 9, 8 (1997), pp. 1735–1780.

[16] Microsoft Azure Video Indexer. Accessed Date 2021-01-10. https://azure.microsoft.com/en-us/services/media-services/video-indexer/

[17] Yu Jiang, Vivek Natarajan, Xinlei Chen, Marcus Rohrbach, Dhruv Batra, and Devi Parikh. 2018. Pythia v0.1: The Winning Entry to the VQA Challenge 2018. *arXiv:1807.09956* (2018).

[18] Richard E Ladner, Melody Y Ivory, Rajesh Rao, Sheryl Burgstahler, Dan Comden, Sangyun Hahn, Matthew Renzelmann, Satria Krisnandi, Mahalakshmi Ramasamy, Beverly Slabosky, et al. 2005. Automating Tactile Graphics Translation. In *Proceedings of the International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS)*. pp. 150–157.

[19] Vladimir Levenshtein. 1966. Binary Codes Capable of Correcting Deletions, Insertions and Reversals. *Soviet Physics Doklady* 10 (1966), pp. 707–710.

[20] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft COCO: Common Objects in Context. In *European Conference on Computer Vision (ECCV)*. pp. 740–755.

[21] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. ViLBERT: Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language tasks. In *Neural Information Processing Systems (NeurIPS)*. pp. 13–23.

[22] Joshua A Miele, Steven Landau, and Deborah Gilden. 2006. Talking TMAP: Automated Generation of Audio-Tactile Maps using Smith-Kettlewell's TMAP Software. *British Journal of Visual Impairment* 24, 2 (2006), pp. 93–100.

[23] Devi Archana Paladugu, Zheshen Wang, and Baoxin Li. 2010. On Presenting Audio-tactile Maps to Visually Impaired Users for Getting Directions. In *ACM SIGCHI Conference Extended Abstracts on Human Factors in Computing Systems (CHI)*. pp. 3955–3960.

[24] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A Method for Automatic Evaluation of Machine Translation. In *Annual Meetings of the Association for Computational Linguistics (ACL)*. pp. 311–318.

[25] Amy Pavel, Gabriel Reyes, and Jeffrey P. Bigham. 2020. Rescribe: Authoring and Automatically Editing Audio Descriptions. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology (UIST)*. pp. 747–759.

[26] Venkatesh Potluri, Priyan Vaithilingam, Suresh Iyengar, Y Vidya, Manohar Swaminathan, and Gopal Srinivasa. 2018. CodeTalk: Improving Programming Environment Accessibility for Visually Impaired Developers. In *Proceedings of the Conference on Human Factors in Computing Systems (CHI)*. pp. 1–11.

[27] Joseph Redmon and Ali Farhadi. 2018. YOLOv3: An Incremental Improvement. arXiv:1804.02767 [cs.CV]

[28] Jaime Sánchez and Matías Espinoza. 2011. Audio Haptic Videogaming for Navigation Skills in Learners Who are Blind. In *International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS)*. pp. 227–228.

[29] Yue-Ting Siu and Ike Presley. 2020. Access Technology for Blind and Low Vision Accessibility. APH Press, Louisville, KY.

[30] Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. 2019. Vl-BERT: Pre-Training of Generic Visual-Linguistic Representations.

arXiv:1908.08530 (2019).

[31] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. 2015. Going Deeper with Convolutions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. pp. 1–9.

[32] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is All You Need. In *Advances in Neural Information Processing Systems (NeurIPS)*. pp. 5998–6008.

[33] Steven Wall and Stephen Brewster. 2006. Feeling what you hear: tactile feedback for navigation of audio graphs. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI)*. pp. 1123–1132.

[34] Mark Weiser. 1991. The Computer for the 21st Century. *Scientific American* 265, 3 (1991), pp. 94–105.

[35] YouDescribe. Accessed Date 2021-01-10. https://www.youdescribe.org/

[36] Beste Yuksel, Pooyan Fazli, Umang Mathur, Vaishali Bisht, Soo Jung Kim, Joshua Junhee Lee, Seung Jung Jin, Yue-Ting Siu, Joshua A Miele, and Ilmi Yoon. 2020. Human-in-the-Loop Machine Learning to Increase Video Accessibility for Visually Impaired and Blind Users. In *ACM Designing Interactive Systems (DIS)*. pp. 47–60.

[37] Beste Yuksel, Pooyan Fazli, Umang Mathur, Vaishali Bisht, Soo Jung Kim, Joshua Junhee Lee, Seung Jung Jin, Yue-Ting Siu, Joshua A Miele, and Ilmi Yoon. 2020. Increasing Video Accessibility for Visually Impaired Users with Human-in-the-Loop Machine Learning. In *ACM SIGCHI Conference Extended Abstracts on Human Factors in Computing Systems (CHI)*. pp. 1–9.