

Voxento 2.0: A Prototype Voice-controlled Interactive Search Engine for Lifelogs

Ahmed Alateeq
ahmed.alateeq2@mail.dcu.ie
School of Computing
Dublin City University
Ireland

Mark Roantree
mark.roantree@dcu.ie
Insight Centre for Data Analytics
Dublin City University
Ireland

Cathal Gurrin
cathal.gurrin@dcu.ie
School of Computing
Dublin City University
Ireland

ABSTRACT

In this paper, we describe an extended version of Voxento which is an interactive voice-based retrieval system for lifelogs that has been developed to participate in the fourth Lifelog Search Challenge LSC'21, at ACM ICMR'21. Voxento provides a spoken interface to the lifelog dataset, which facilitates a novice user to interact with a personal lifelog using a range of vocal commands and interactions. For the version presented here, Voxento has been enhanced with new retrieval features and better user interaction support. In this paper, we introduce these new features, which include dynamic result filtering, predefined interactive responses and the development of a new retrieval API. Although Voxento was proposed for wearable technologies such as Google Glass or interactive devices like smart TVs, the version of Voxento presented here uses a desktop computer in order to participate in the LSC'21 competition. In the current Voxento iteration, the user has the option to enable voice interaction or use standard text-based retrieval.

CCS CONCEPTS

• **Human-centered computing** → **Sound-based input / output**; • **Information systems** → **Search interfaces**; • **Computing methodologies** → **Speech recognition**.

KEYWORDS

lifelog; interactive retrieval; voice interaction; speech recognition; speech synthesis

ACM Reference Format:

Ahmed Alateeq, Mark Roantree, and Cathal Gurrin. 2021. Voxento 2.0: A Prototype Voice-controlled Interactive Search Engine for Lifelogs. In *Proceedings of the 4th Annual Lifelog Search Challenge (LSC '21), August 21, 2021, Taipei, Taiwan*. ACM, New York, NY, USA, 6 pages. <https://doi.org/10.1145/3463948.3469071>

1 INTRODUCTION

Lifelogging refers to be the process of capturing a digital trace of an individual's life experiences using a suite of non-intrusive technologies such as wearable cameras, fitness trackers, smartphones

and various other applications [8]. The concept of Lifelogging has been predicted for many decades [3] and is now becoming possible through the development of low-cost sensors/storage and effective multimedia data analytics [8]. The captured data called lifelogs can include everything that can be digitised to reflect an individual's life experience [5]. Lifelog data typically contains images captured automatically using wearable cameras, in addition to other information related to physical actions and activities generated by sensors. Typical capturing of data includes location, biometric data such as heartbeats and physical activities such as walking [8]. MyLifeBits [5], the first interactive lifelog prototype, aimed to index all information and tasks related to an individual's life experience and Voxento has used features of MyLifeBits as a platform for the prototype presented in this paper.

The Lifelog Search Challenge (LSC) is an annual benchmark competition workshop aiming to tackle challenges regarding multimedia information retrieval in lifelogs data. LSC started in 2018 and was running for a period of three years 2019 and 2020 successfully [6]. This fourth workshop encourages researchers to develop an interactive lifelog retrieval and participate in a live competition to comparatively evaluate system performance in an open, metrics-based workshop. The process through which system performance is evaluated is described in detail in [6]. The benchmarking activity consists of all participating teams executing a large number (typically 18+) of information needs and submitting potentially relevant items to a real-time evaluation service, with each team receiving a score for every submission. The information needs are constructed in order to simulate how human remembers a past event [7] and the aim is to locate the relevant content as quickly and accurately as possible.

This paper presents the second version of Voxento, a prototype voice-controlled interactive retrieval system for lifelogs, designed to support novice users by providing a simple and effective voice-controlled interface and not relying on complex interactions that require experienced operators. Although there has been considerable research activity into off-line and virtual lifelog retrieval, to the best of our knowledge, there have been no voice-controlled interactive retrieval engines for lifelogs developed prior to Voxento. Based on previous participation in LSC'20 [2], we have enhanced the Voxento prototype by expanding the range of supported voice interactions, adding search filters, adding new interactive features and integrated to a new developed retrieval API provided by [1]. Each of these components will be discussed in the following sections. Voxento uses the Google Web Speech API to detect spoken words and recognise a wide range of commands. In turn, Voxento provides both a visual and a voice-based response for users that



This work is licensed under a Creative Commons Attribution International 4.0 License.

LSC '21, August 21, 2021, Taipei, Taiwan.

© 2021 Copyright held by the owner/author(s).

ACM ISBN 978-1-4503-8533-6/21/08.

<https://doi.org/10.1145/3463948.3469071>

behaves similar to a basic chat session. We kept the system as intuitive as possible and provided a user guide for novice users. The back-end retrieval engine uses the CLIP model [13] to generate semantic representations from the lifelog images and metadata has been enhanced, reorganised and restructured to support effective retrieval and, also, it was evaluated by running the query topics provided by LSC'20 [6].

2 RELATED WORK

Among other lifelog retrieval systems that participated in LSC, many different approaches and ideas have been proposed, such as the Virtual Reality retrieval system proposed by Duane et al [4], or prior to the LSC challenge, one notable prototype is the multi-modality, a cross-platform proposal by Qiu et al [12]. Voice interaction with lifelog archives has still not received much research attention and to the best of our knowledge. With the exception of our previous prototype [2], there are no voice-interaction systems for lifelogs. In the first version of Voxento [2] developed for the last year LSC challenge (in 2020), we created an initial prototype on a desktop computer, but with a vision that it could be ported to a wearable device, such as a Google Glass. In this paper, we report on the latest iteration of the Voxento prototype, which has been developed to participate in the 2021 edition of the challenge.

In order to place our research in the context of existing state-of-the-art, we now highlight the top three performing retrieval systems at the LSC'20 competition. The system that performed best was MySceal [16] which has multiple features: exploring query expansion, word embedding approaches to lifelog retrieval and a novel concept weighting methodology. The second best system was SOMHunter [11], based on the top performing system from Video Browser Showdown 2020, which was an adaptation of a video search engine combining temporal queries and a relevance feedback model for lifelog retrieval. Another system of note was Vitriivr [10], based on the top-performing system from LSC'19, which supports finding events based on using structured, semi-structured and unstructured data. All three systems put significant emphasis on supporting easy querying by users, with many systems implementing faceted filters explicitly, while Mysceal detected the possible filters directly from a free-text user query.

Voice-based interaction interfaces have a long history of using natural (two-way) vocal interactions to improve the user experience and to support users who may have difficulty using touch screens or keyboards. The authors in [15] created a voice-based search engine for visually impaired people, which supports voice commands to search the internet and speech synthesis to provide voice-based responses. Recently published research [17] discussed two possible speech-based selection techniques for voice-user interfaces: one for controlling interaction devices and the second for selecting target elements on the screen.

The Voxento system described in this paper is our first step of our research towards the provision of a truly pervasive and hands-free interface. The system is built on a state-of-the-art back-end search service and its contribution is in the voice interaction, which facilitates a hands-free desktop interface to support the user to quickly and effectively locate content of interest in the lifelog.

3 EXPERIENCES FROM VOXENTO AT LSC 2020

Voxento at the LSC'20 [2] highlighted the potential of speech interaction with lifelogs. The system did not perform as well as we hoped, but this was mainly due to a basic bag of words retrieval engine operating over limited metadata. Nonetheless, we can point to a number of potential improvements for this LSC event.

- **Search Filters:** Filters were not implemented in the previous version [2] and while we had an indexed date field, this was not sufficient to be competitive with other state-of-the-art systems and resulted in ranked lists being presented to the user that were too large for effective browsing when under the time constraints of a live competition. We solved this issue by adding multiple search filters discussed in section 5.4.
- **Search Engine:** In the previous version of Voxento [2], the search service was not effective. For LSC'21, we have integrated an improved back-end search service with enhanced metadata and effective retrieval [1].
- **User Interaction:** We have also improved the system interaction by expanding the range of voice interactions available to a user, as discussed in section 5.2.

We believe that these new features have enhanced the competitiveness of the Voxento prototype for the LSC'21 challenge.

4 LSC'21 DATASET

We briefly describe the LSC'21 dataset in this section with a more detailed description available in [9]. The LSC'21 dataset provided for the 4th Lifelog Search Challenge is similar to the previous LSC'20 with about 4% of images excluded for the current challenge. The dataset consists of lifelog data captured by one active lifelogger using wearable cameras for four months between 2015 and 2018, combined into a single lifelog. In addition, the final dataset included anonymised images, visual concepts and metadata. Each image's name represented a date and time in seconds, otherwise, the structure is quite different, possibly due to the fact that there were two different wearable devices. Visual concepts included text descriptions of detected scenes, concepts and objects for each image with the confidence score. Also, the metadata provided contains time, date, physical activities such as *steps*, biometrics such as *heart beats*, *calories*, locations such as *country name*, *geographic location* and other metadata information.

As we use a back-end API from [1], the authors have reprocessed the original dataset, resulting in enhanced metadata, which is summarised below:

- **Semantic Image Representation:** The authors utilise the CLIP model [13] to encode images into high-dimensional representations to detect rich semantics of the scene.
- **Metadata Enhancement:** The authors tried to improve and enrich the specific metadata information, for example location and time, to support information retrieval. The following steps are performed: (1) Imputing location name represented as a semantic name; (2) Identifying blurred images for removal; (3) Deriving specific fields from existing data, such as time of day, or weekday/weekend tagging.

- **Event Segmentation:** Segmentation of each day into events by identifying the current activity and the location name when the lifelogger performs some movement. Event segmentation can inform temporal sequencing of result documents using the interface.

5 OVERVIEW OF VOXENTO 2.0

In this section, we present an overview of the expanded Voxento architecture with a detailed description of the main components. Voxento 2.0 is based on the previous version with a new voice interaction feature and back-end search service, with enhancements to the interface discussed in section 5.4. There are three main components of Voxento as shown in Figure 1: the user’s voice and interface interaction, visual interface, and the back-end API [1]. The interface is web-based from the user’s perspective, but with a headset for voice interaction. On system loading, the voice interaction feature automatically engages and awaits the user’s command. The user can either utter a spoken command or alternatively, revert to using a mouse and keyboard to interact with the interface. The user can swap between these two interaction methodologies as described in the later section 5.2. The second component is the browser-based visual interface that handles the speech-to-text (and vice versa) speech API and displays results on-screen and communicates via spoken messages to the user. The final part is that the backend API will receive the search task requests and generate ranked lists of images, which is described in section 5.5.

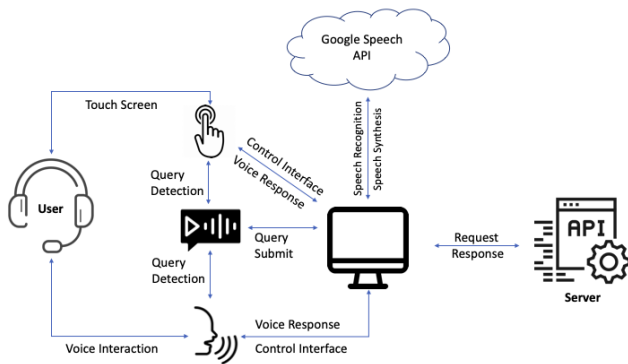


Figure 1: Overview System Architecture

5.1 Voice-based Implementation

The Google Web Speech API [14] is used to support voice interaction. This enables speech recognition and synthesis through the web browser, with the language set to English with a standard accent. This replicated the effective configuration from our previous work [2]. Using these settings permits the user to speak continuously and see the real-time query emerging from the system, with the provision of recommended additional/alternative terms for user selection.

In this work, we created two main speech recognition instances. The first handles the voice interaction which will be listening for defined commands when the system is running. The second is dedicated only to query detection and will react to specific command

utterances to formulate a query. We found that this approach was most effective while testing the system due to differences in how each type of speech needs to be handled. As a result, these instances may overlap and swap between each other without affecting user interaction. However, the user has a smooth interaction using only a single with no concern for the background configuration. See Figure 1 to illustrate how users can move between different interactions.

As described in the previous Voxento paper [2], a fully hands-free interface includes more than just speech recognition: a universal interface requires spoken output. In the updated version of Voxento, we maximise the spoken output to cover the all-important components and texts in the interface and seamlessly integrate spoken feedback to all user actions with a mouse. For example, a click-action from the mouse will have a response based on the specific action (e.g. selected image). This predefined voice response feature provides an interactive means to respond to the user’s commands or actions on the interface components. It is also possible for the user to hear the summary metadata for an image to be spoken as well as displayed on screen. Nevertheless, the system is not a full NLP interface that facilitates every possible verbal communications but is a more restrictive system that detects specific commands. We developed a number of features to enhance the user experience such as a range of expected commands or words for users in order to reduce the effort required to remember them, which are discussed in the next sections.

5.2 Voice Interaction Process

To explain the voice interaction steps, we design a lifecycle shown in Figure 3, which illustrates the steps a user experiences when using Voxento. Importantly, at each stage, the system maintains ‘current status’ which defines what voice commands are allowed at any point in time. For example, users cannot use the *select image 10* command if there are no images in the result panel. Therefore, this command should be used after submitting the query. Understanding that a user could make a mistake or forgot the exact command, we provide two main solutions: (1) we provide links to a Voxento guide, and (2) we provide a wider vocabulary of allowable commands to reduce the need for users to memorise certain keywords. For example, if the user wants to select an image to zoom in (explore an image in a larger size and metadata), the user can utter any of these options: *{select, open, choose, zoom} image + number* and also user can say just *number 10*. Lastly, we add an option for the user to turn the voice interaction on or off completely as a toggle switch on the top left of the screen, see Figure 2, with a note that turning off can be managed by a voice command. The user is also provided with a conventional text-based query and interaction mechanism to use if required.

We summarise the life cycle of voice interaction briefly in numbered points below:

- (1) When the page is loaded, the speech recognition of voice interaction will turn on and speak a greeting message. If a user responds, the system will respond with a follow-on message.
- (2) When running, the user has an option to say a command such as *start recording* or click on the record button.

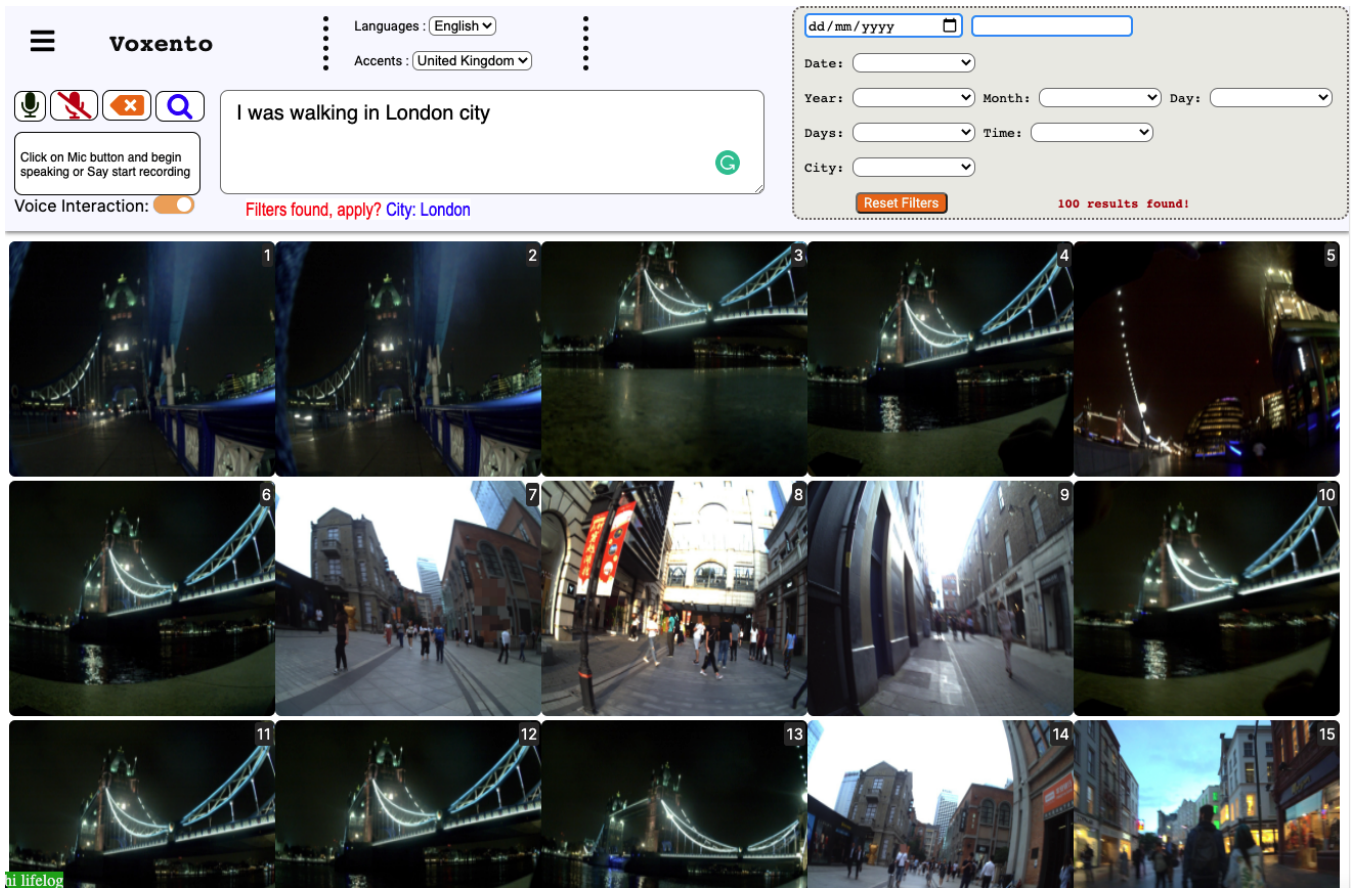


Figure 2: Voxento Main Interface

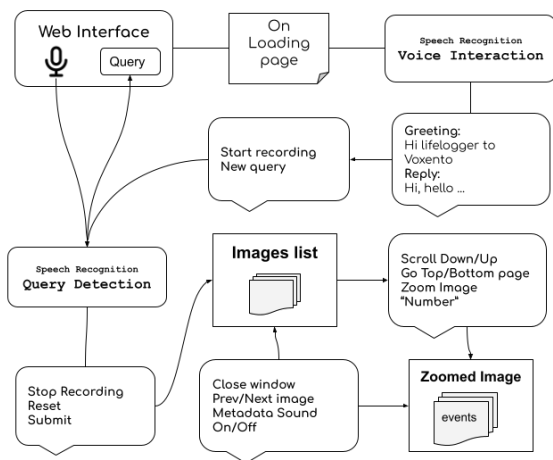


Figure 3: Voice Interaction Lifecycle

(3) The system is now in query detection mode, which displays the spoken query in real-time. At the end, the user has many

options to 'submit' the query to the retrieval engine. Submission is only performed through a user command and is not automatic.

- (4) When the image list is displayed on the screen, users can scroll the page down/up, top/bottom and select the image to zoom in, using voice commands. Moreover, users can resubmit a query and return to step 3, by saying *new query* or *start recording* commands, for example.
- (5) This step relates to the previous step, as when the user selects an image and zooms in, the user can move to the next and previous event related to the selected image. Moreover, the user can also say *sound* to hear the image's defined information like date, time, location.

There are a few more commands that are not mentioned in the lifecycle and these can be accessed through the top navigation bar "Available commands" link button. An example of the available commands is *Turn voice off* which will stop the voice interaction completely.

5.3 Query Speech Detection

In order to detect the speech for a query, we use a dedicated speech recognition engine as shown in Figure 3; an updated version of our previous work [2]. As with the previous implementation, Voxento

now supports system control via three basic commands that are summarised as below:

- **Stop recording:** When a user speaking the query decides to stop recording and modifies the query. For example *find flowers and stop recording*. The system will stop recording and the query will consist of 'find flowers'. Users then have an option to update the query also if needed.
- **Reset:** This command resets the query text, which may be needed when a user wants to start with a totally new query while simultaneously recording the query session.
- **Submit:** While speaking the query, the user can submit the current query directly to the search service.

5.4 User Interface

One of the goals of Voxento is to have a clear, easy to use and intuitive visual interface that is suitable for novice users, as shown in Figure 2. We position all control and filter functions in the top "query panel" to maximise space in the visual interface for the images. The query panel facilitates the selection of language and accent, with the default set to *English*. On the left, we have the recording control buttons: start and stop recording, reset and submit buttons which are inherited from the previous version [2]. The query text uses a large text box with a scroll feature if more than three lines of a query are needed. Underlying the query text box where the filters detected from within the query are located and are clickable for the user to apply these detected filters to the current result-set. The number of results, shown in the bottom right section, is updated whenever new results appeared or filters are selected. At the bottom of the visual interface, there is a green line showing speech whenever the user speaks so the user will know what's said and this will help when there is an incorrect detection for any of the commands. We highlight below the new features added to the interface:

- **Dynamic Results Filters:** This large bordered box on the right contains eight filters: calendar, date, year, month, day, day name, city and time. These filters are optimised to the result-set and facilitate further narrowing the result set. For example, the month option will be populated with only months that are represented in the result set. There is a reset button at the bottom that resets back to the original result-optimised configuration if needed. All filters are sorted in ascending order except day names which sorted based on weekdays order from Monday.
- **Calendar Filter:** This filter shown in Figure 4 is discussed separately because it is optimised to the query by (1) highlighting the dates that are found in the result-set, (2) restrict the selection to only these dates, and (3) adding a selected date (or date range) to the text box for easy modification later.
- **Detected Filters from the Query:** We have extracted (*years, month and day names, city*) from the lifelog and we compared them to the query text to automatically detect potential filters. If there is any match for any filter, the system will detect and display them to the user below the query text box as shown in Figure 2. Users will then have the option to either

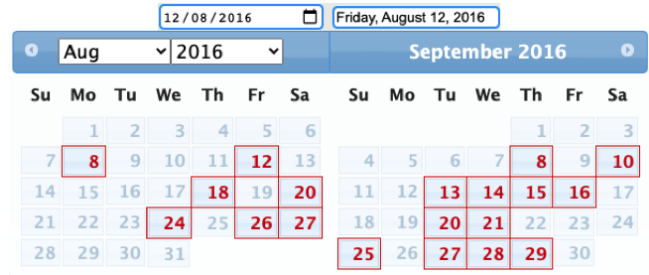


Figure 4: Calendar Filter

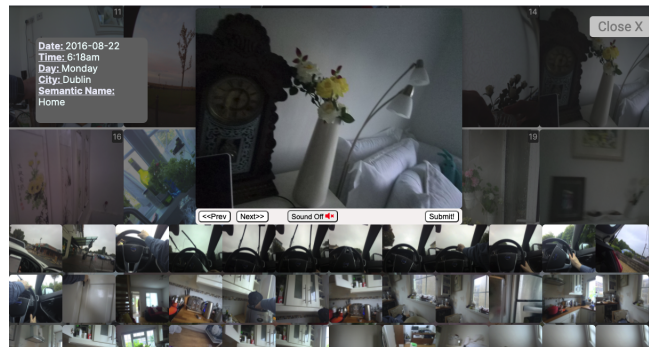


Figure 5: Zoomed image window with list of events and information

click on the detected filters or utter the *apply filters* command to apply them.

- **Message Centre:** As a result of using speech recognition and synthesis for voice interaction, it was important to create a message centre that adopts the following steps: (1) Show the stages: recording, submitting query etc.; (2) Show responses like Greeting and commands action; (3) Show Speech recognition errors when they occur; (4) Display information for example "Click mic and speak or say start recording". It is worth noting that any update to this box will result in the border being highlighted for a number of seconds.
- **Series of Events:** If the user clicks on an image, it will zoom to a large panel as shown in Figure 5 which contains the selected image in a larger size, with both previous and next images accompanied by spoken image metadata and a submit button if the user thinks that the image is relevant. At the bottom, there is a list of images based on the context of the event to which the image belongs, which provides temporal context and temporal navigation. The temporal navigation is restricted to the given event. For details see [1].

5.5 Search Engine

The previous Voxento system [2] was a standard retrieval system based on a MongoDB search engine where relevant columns were the indexed and free-text ranking engine, which applies stemming and stop word removal. However, this proved to be non-competitive, so we have employed the backend API provided by [1] with minor



Figure 6: A sample image based on query *UNIX book*

modification in call requests to support some of the interaction requirements for Voxento. In terms of major modifications, we enhanced the backend API by implementing stemming and stopword removal for the query. In addition, we now have the ability to exclude the filters words from the query to overcome the issue of having multiple filters in the query that affecting the result accuracy as explained in details in section 5.4. The back-end API is built using the Flask python framework using RESTful API functionality. The maximum number N , of results returned for a query, is set to 1,000. The provided backend API was evaluated in [1] and showed that the system was capable of detecting at least one target image in the top-50 results for 18 out of 24 LSC'20 topics, so we expect it to be very competitive.

The service translates a query into a vector representation using CLIP networks which then will be compared to image representation, already located in the server as a static file, using cosine similarity which results in the ranked list of image indices. The server then uses these images indices to fetch the image metadata which is also located in the server as a CSV file. Finally, it will send the ranked results as a JSON response to the user's interface. A sample image is displayed in Figure 6, based on the query: *UNIX book*.

6 CONCLUSIONS AND FUTURE WORK

In this paper, we presented the updated version of Voxento prototype interactive lifelog retrieval which has been significantly enhanced for participation at LSC'21. Voxento's voice interaction is more comprehensive and includes new interface elements to improve the user experience. We highlighted the system overview, described the voice interaction life cycle, described the dataset used, enhancements to the metadata and the new search engine. Regarding future work, this version of Voxento still requires a conventional desktop interface and mouse interaction. It is planned to move to a fully wearable platform or migrate from the desktop environment to a truly pervasive computing environment.

ACKNOWLEDGMENTS

We acknowledge the support of Science Foundation Ireland and the Insight Centre for Data Analytics through the grant number

SFI/12/RC/2289-P2 and the Ministry of Education in Saudi Arabia for sponsoring the PhD research of the primary author.

REFERENCES

- [1] Naushad Alam, Yvette Graham, and Cathal Gurrin. 2021. Memento: A Prototype Lifelog Search Engine for LSC'21. In *Proceedings of the Fourth Annual Workshop on the Lifelog Search Challenge (LSC'21)*. ACM, Taipei, Taiwan. <https://doi.org/10.1145/3463948.3469069>
- [2] Ahmed Alateeq, Mark Roantree, and Cathal Gurrin. 2020. Voxento : A Prototype Voice-controlled Interactive Search Engine for Lifelogs. In *Proceedings of the Third Annual Workshop on the Lifelog Search Challenge (LSC'20)*. ACM, Dublin, Ireland, 77–81. <https://doi.org/10.1145/3379172.3391728>
- [3] GC Bowker. 2007. The past and the Internet. *Structures of participation in digital culture* (2007), 20–36.
- [4] Aaron Duane, Björn Por Jónsson, and Cathal Gurrin. 2020. VRLE: Lifelog Interaction Prototype in Virtual Reality: Lifelog Search Challenge at ACM ICMR 2020. In *Proceedings of the 3rd Annual Workshop on the Lifelog Search Challenge*. ACM, Dublin, Ireland, 7–12. <https://doi.org/10.1145/3379172.3391716>
- [5] Jim Gemmell, Gordon Bell, and Roger Lueder. 2006. MyLifeBits: A personal database for everything. *Commun. ACM* 49, 1 (2006), 88–95. <https://doi.org/10.1145/1107458.1107460>
- [6] Cathal Gurrin, Tu Khiem Le, Van Tu Ninh, Duc Tien Dang-Nguyen, Björnór Jónsson, Jakub Lokoš, Wolfgang Hürst, Minh Triet Tran, and Klaus Schöffmann. 2020. Introduction to the third annual lifelog search challenge (LSC'20). In *ICMR 2020 - Proceedings of the 2020 International Conference on Multimedia Retrieval*. ACM, Dublin, Ireland, 584–585. <https://doi.org/10.1145/3372278.3388043>
- [7] Cathal Gurrin, Klaus Schoeffmann, Hideo Joho, Andreas Leibetseder, Liting Zhou, Aaron Duane, Duc-Tien Dang-Nguyen, Michael Riegler, Luca Piras, Minh-Triet Tran, Jakub Lokoč, and Wolfgang Hürst. 2019. Comparing Approaches to Interactive Lifelog Search at the Lifelog Search Challenge (LSC2018). *ITE Transactions on Media Technology and Applications* 7, 2 (2019), 46–59. <https://doi.org/10.3169/mta.7.46>
- [8] Cathal Gurrin, Alan F. Smeaton, and Aiden R. Doherty. 2014. *LifeLogging: Personal big data*. Vol. 8. 1–125 pages. <https://doi.org/10.1561/15000000033>
- [9] Cathal Gurrin, Björn Þór Jónsson, Klaus Schöffmann, Duc-Tien Dang-Nguyen, Jakub Lokoč, Minh-Triet Tran, Wolfgang Hürst, Luca Rossetto, and Graham Healy. 2021. Introduction to the Fourth Annual Lifelog Search Challenge, LSC'21. In *Proc. International Conference on Multimedia Retrieval (ICMR'21)*. ACM, Taipei, Taiwan.
- [10] Silvan Heller, Mahnaz Amiri Parian, Ralph Gasser, Loris Sauter, and Heiko Schuldt. 2020. Interactive Lifelog Retrieval with vitrivr. In *Proceedings of the 3rd Annual Workshop on the Lifelog Search Challenge*. ACM, Dublin, Ireland, 1–6. <https://doi.org/10.1145/3379172.3391715>
- [11] František Mejzlík, Patrik Veselý, Miroslav Kratochvíl, Tomáš Souček, and Jakub Lokoč. 2020. SOMHunter for Lifelog Search. In *Proceedings of the 3rd Annual Workshop on the Lifelog Search Challenge*. ACM, Dublin, Ireland, 73–75. <https://doi.org/10.1145/3379172.3391727>
- [12] Zhengwei Qiu, Cathal Gurrin, and Alan F. Smeaton. 2016. Evaluating Access Mechanisms for Multimodal Representations of Lifelogs. In *Proceedings of the 22nd International Conference on MultiMedia Modeling (MMM)*. Springer, Miami, FL, USA, 574–585. https://doi.org/10.1007/978-3-319-27671-7_48
- [13] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning Transferable Visual Models From Natural Language Supervision. arXiv:2103.00020 <http://arxiv.org/abs/2103.00020>
- [14] Glen Shires. 2013. *Voice Driven Web Apps: Introduction to the Web Speech API*. Retrieved April 24, 2021 from <https://developers.google.com/web/updates/2013/01/Voice-Driven-Web-Apps-Introduction-to-the-Web-Speech-API>
- [15] S. Sridharan, D. Abinaya, P. Swarna Prabaa, M. Jovitha, and K. Kayathri. 2019. Voice Based Search Engine for Visually Impairment Peoples. *International Journal of Advanced Scientific Research and Development (IJASRD)* 6, 3 (2019), 45. <https://doi.org/10.26836/ijasrd/2019/v6/i3/60307>
- [16] Ly Duyen Tran, Manh Duy Nguyen, Nguyen Thanh Binh, Hyowon Lee, and Cathal Gurrin. 2020. Myscéal: An Experimental Interactive Lifelog Retrieval System for LSC'20. In *Proceedings of the 3rd Annual Workshop on the Lifelog Search Challenge*. ACM, Dublin, Ireland, 23–28. <https://doi.org/10.1145/3379172.3391719>
- [17] Xiangliang Zhang, Chaoxi Lu, Jibin Yin, Hailang Xie, and Tao Liu. 2020. The Study of Two Novel Speech-Based Selection Techniques in Voice-User Interfaces. *IEEE Access* 8 (2020), 217024–217032. <https://doi.org/10.1109/ACCESS.2020.3041649>