Searching Temporally Distant Activities in Lifelog Data With PraK Tool V2

Zuzana Vopálková
Department of SW Engineering
Faculty of Mathematics and Physics
Charles University
Prague, Czech Republic

Jakub Yaghob
Department of SW Engineering
Faculty of Mathematics and Physics
Charles University
Prague, Czech Republic

Michael Stroh
Department of Visual Computing
Faculty of Sciences
Konstanz University
Konstanz, Germany

Udo Schlegel
Department of Data Analysis
Faculty of Sciences
Konstanz University
Konstanz, Germany

Jakub Lokoč
Department of SW Engineering
Faculty of Mathematics and Physics
Charles University
Prague, Czech Republic

ABSTRACT

We present a new version of our interactive cross-modal search system PraK that was updated for the needs of lifelog search challenge data and tasks. The architecture of PraK allows to hide lifelog data specific details behind already existing interface of video data services. Hence, the already existing application service designed for the Video Browser Showdown could be tested for the visual part of the lifelog dataset. Nevertheless, we add more features to address search challenges specific for lifelog data. In the new version, the interface of the services is slightly updated to incorporate meta-data based filtering as well as to support temporally distant activity queries. Another major update of the system is shift of state management from the web frontend to the application backend, which leads to more centralized architecture but allows to deploy PraK components in a more efficient way.

CCS CONCEPTS

• **Human-centered computing** → *Empirical studies in interaction design*; • **Information systems** → **Search interfaces**.

KEYWORDS

Lifelog data, Interactive Retrieval Systems, Similarity Search

ACM Reference Format:

Zuzana Vopálková, Jakub Yaghob, Michael Stroh, Udo Schlegel, and Jakub Lokoč. 2024. Searching Temporally Distant Activities in Lifelog Data With PraK Tool V2. In *The 7th Annual ACM Lifelog Search Challenge (LSC '24), June 10, 2024, Phuket, Thailand*. ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3643489.3661131

LSC '24, June 10, 2024, Phuket, Thailand

ACM ISBN 979-8-4007-0550-2/24/06
https://doi.org/10.1145/3643489.3661131

1 INTRODUCTION

Lifelogging is a phenomenon that inspires various research activities, projects and competitions already for several decades (e.g., MyLifeBits [3], NTCIR-lifelog task [4]). A significant part of the challenge is analysis of large personal datasets comprising visual and structured multi-modal data collected for a long time period. The dataset provides valuable information allowing to build useful applications [7], ranging from lifestyle/health monitors to search systems for personal memories.

Interactive search systems for lifelog databases are stimulated, among other factors, by the Lifelog Search Challenge [5, 6, 8, 28] (LSC) that provides a challenging multi-modal dataset of one lifelogger. The dataset contains more than half-million of images taken by a wearable camera. The images are accompanied by the time of creation, while GPS and bio-metric meta-data are also associated to some of the images. The challenge consists of three different types of tasks – known-item search, ad-hoc search, and question answering. While the dataset is available months ago the LSC event, the tasks are revealed on-site during the competition to participating teams. Note that thanks to the DRES evaluation server [23], teams can participate at the challenge remotely as well. The competition format is similar to the Video Browser Showdown [15], where teams try to solve a given task simultaneously within a fixed time interval. The time limit depends on a task type and is usually around five minutes.

In this paper, we present a new version of the PraK system that was introduced at the Video Browser Showdown 2024 organized at the Multimedia Modeling conference in Amsterdam. The system emphasizes clear separation of various data sources and more advanced interactive retrieval mechanisms. The overall architecture is detailed in Figure 1, showing stateless video data services (label video substitutes a distinct time interval in the lifelog domain) providing basic data access methods. The more advanced logic is designed on top of partial results returned by the stateless services. The extensions of the system for LSC 2024 comprise functionality to search for temporally distant activities as well as meta-data filters and smarter application backend allowing more network-efficient implementation of the search engine.

2 RELATED WORK

In 2023, twelve teams participated at LSC@ICMR [5] in Thessaloniki. Many of the systems [1, 2, 10, 19, 22, 24, 25, 27] were extensions of their previous versions, while several new systems [11, 20, 26, 30] were presented as well. In this section, we summarize the key approaches used in the top three LSC'23 tools. We note that the dominant text-image search approach was based on joint embedding networks (CLIP based [21]), similar observation to the one from the Video Browser Showdown [15].

We start with MyEachtra [27] as the system's predecessor [29] was a champion of several previous LSC events. Compared to its previous versions, MyEachtra shifted from images to events as units for ranking and visualization. The overall event score is computed as a weighted sum of used queries, supporting visual features, location info, and time. According to the presented study, using mean pooled embedding of event images turned out to be the most effective event representation for a set of LSC'22 benchmark queries.

The overall winner of LSC'23 was the new version of lifeXplore [25]. The system relied mostly on text-image joint embedding combined with filters for time information. Although the system supported also additional retrieval models (e.g., EfficientNet B2, YOLO v7), they were just rarely used according to the report provided by the authors. We also note that the designed user interface was very simple (lifeXplore was successful in the novice session) and without temporal query functionality. Nevertheless, the system won the whole competition and also the study in the paper shows impressive performance of OpenCLIP based models for ranking of lifelog dataset images just based on the initial text descriptions.

The third system at LSC'23 was Memento [1], a system allowing users to switch between various joint embedding models and implementing efficient retrieval approaches to speed up the search. Specifically, the user interface allows to switch between different UI layouts (and selected models) depending on a currently evaluated type of task. In other words, task specific interfaces are supported. According to the evaluations presented in the paper, different joint embedding approaches can be effectively used for different search scenarios. The authors also report significant improvement of search efficiency when FAISS library is used (specifically, inverted file indexing).

3 PRAK SYSTEM ARCHITECTURE AND DESCRIPTION

The PraK system is a framework composed of three interconnected components, each serving a pivotal role in the system's overall functionality. At its core, the Video Data Service functions as a stateless in-memory database, offering immediate access to data and embeddings derived from the corresponding model upon query. This layer is seamlessly integrated with a stateful Application Service Backend, which not only bridges the connection between the Frontend and the Video Data Service but also executes essential transformation steps for the image/video data. These transformations include necessary post-processing steps for a seamless display of statistics or video items, or following stateful user-specified transformations like Bayesian Relevance Feedback to manipulate the display ordering of presented video items. The Web Frontend completes the System by providing an easily accessible endpoint for different users to use the PraK system independently from one another to complete retrieval tasks. It is designed to present the results of the users' queries in an intuitive and user-friendly manner. These components harmonize to provide a robust and efficient infrastructure for handling and displaying video or image data.

3.1 PraKs Stateless Video Data Service

PraK is designed to work with sequences of representative images, either extracted from video files or obtained directly from a wearable camera. The individual image analysis is preceded by deriving embeddings for the extracted keyframes or original images as the core of image-image and text-image similarity. The system is engineered to flexibly load any deep learning embedding model, including but not limited to models like CLIP [21] and OpenCLIP [12], allowing for seamless model interchangeability based on the specific requirements of the task at hand. Moreover, it offers a stateless REST API, enabling other components to access the processed data along with the model embeddings effortlessly. To further extend the basic set of features, the layer also incorporates functionality that facilitates the nearest neighbor search for either an example image or text query using the model embeddings. This capability ensures that other components can retrieve the most relevant data directly just by an API call, streamlining the process of information retrieval and data analysis.

3.2 PraKs Application Service Backend

We utilize a minimal Python web server, Uvicorn, coupled with Fast-API as the Application Service Backend, which is responsible for serving the Web Frontend and handling user-session-specific data. This setup facilitates multi-user application memory independent of the Video Data Service. Our design rationale stems from the principle of maintaining a stateless Video Data Service that focuses exclusively on serving static video data to various user Web Frontend application instances. We may handle, process, and visualize the data in these Web Frontend applications differently, depending on the user's requests or retrieval requirements. Based on active connections to our Application Service Backend via browser endpoints, the server dynamically generates a temporary cache to store client data. For each user, this Application Service Backend retains specific client Frontend information, including:

- Top-k items displayed to the user, with associated feature vectors and item scores. These scores dictate the display order of items on the Frontend.
- A history of the user's l-previous displays, based on their past queries and r-next displays, should the user undo multiple actions.

This history cache empowers users to swiftly undo unsuccessful queries, restoring the Web Frontend display to a more favorable state. Furthermore, by preserving subsequent queries in the cache post-undo operations, users can revisit actions without needing to recall the specific query that led to a particular result. As users execute different actions in a reverted state, these subsequent query displays are replaced, ensuring a seamless and consistent user experience with both undo and redo operations.

Leveraging the image CLIP features within the Application Service Backend, we can compute diverse item re-ranking mechanisms

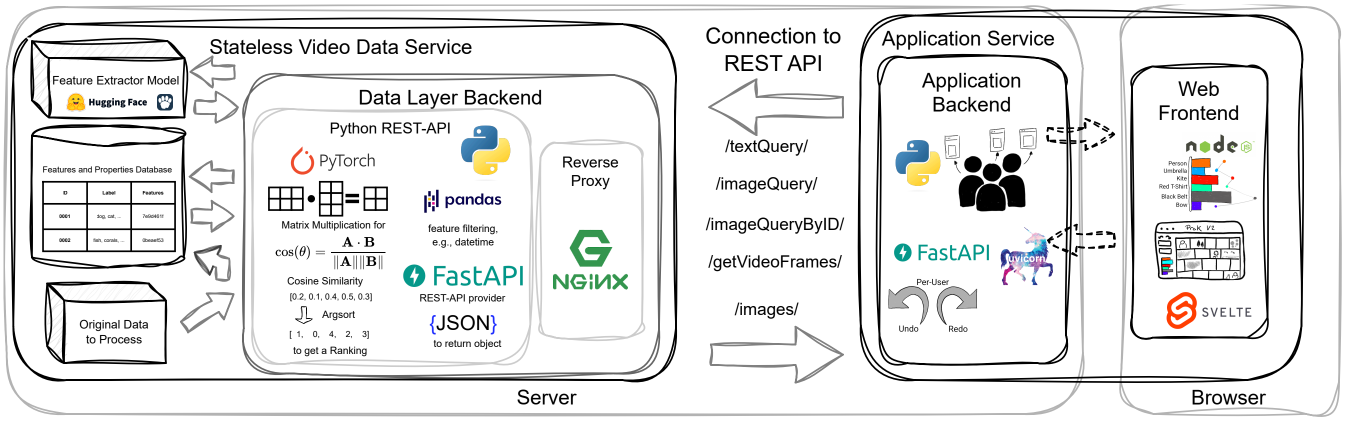

Figure 1: Architecture of the presented video search system based on stateless Video Data Services. The Application Backend handles Bayesian re-ranking and state for various users, while Web Frontend allows interaction with the system.

based on user feedback, such as Bayesian relevance feedback. Importantly, this is achieved without additional load on the Video Data Service. The separation between the Video Data Service and the Application Service Backend also allows us to scale the necessary hardware independently. This approach optimizes load balancing, catering to the number of concurrent users the system may have.

3.3 PraKs Web Frontend

The PraK system's Web Frontend component offers a platform for the interactive refinement of search results obtained from our data services. It incorporates a mix of tried-and-true functionalities alongside innovative features to enhance user engagement and search efficiency. Central to its design is a robust suite of search options that draw upon the successes of previous Video Browser Showdown (VBS) implementations [15]. These include various visualization techniques, allowing users to view ranked images or frames in a sorted grid format or in a line-by-line manner for frames of the same video. Using this modality integrates temporal video context for a more intuitive browsing experience. The system also facilitates dynamic query reformulation, supporting both text and example image queries, the latter of which can include images sourced externally, thereby broadening the scope of search inputs.

A notable aspect of the system is its use of presentation filters, which streamline search results by displaying only the most relevant items from each video based on user-defined criteria. For these active filters, we assume that each data item has a list of detected semantic classes (e.g., CLIP classification) or that the classes correspond to centroids obtained by an unsupervised clustering technique like k-means or k-medoids. With these classes and corresponding class confidence scores, interactive search features like informative statistic charts and label-selection-based filters to the displayed items can be added. The previous iteration of the PraK System [18] first introduced this feature.

Such features not only aid in the visualization of search data but also empower users to refine their search results through a more informed decision-making process, blending automated content analysis with user insights for a highly targeted search experience. This content-based refinement can become one of the most valuable interactions in datasets where formulating queries is difficult for non-domain experts due to a lack of specialized vocabulary. We purposefully display the most relevant keywords present in a given search, thus enabling users to re-formulate more content-driven queries even with scarce domain knowledge. Some example scenarios from previous VBSs are medical datasets or scuba diving video data. This is complemented by the implementation of Bayesian relevance feedback successfully used in SOMHunter [13] and CVHunter [14], utilizing joint-embedding features to refine the search results further based on user interactions, thus making the search process more tailored to the user's needs.

The design of the front-end interface reflects a commitment to usability, presenting users with an array of options for dataset selection, feature model specification, and custom query formulation. It fosters an interactive search environment where users can easily navigate between text-based queries and image-based searches, with the added functionality of Bayesian updates and class label filters to enhance the relevance and precision of search outcomes.

4 ADDED FEATURES FOR LSC 2024

This section describes a set of new features that were added to the PraK version [18] since the Video Browser Showdown competition in 2024 in Amsterdam.

First, we focus on temporally distant activities that represent a part of lifelog search scenarios. When remembering events from the past, it is possible that people remember a sequence of actions that happened during some day. This can be illustrated on a task from a previous LSC event where the sequence consisted of three actions - arriving at an airport with a plane, having a burger in a restaurant, and leaving the airport in a plane. Every single action is not so unique in the lifelog dataset of a researcher who travels a lot and likes burgers. However, the sequence as a whole starts to be more unique for a given time interval (e.g., one day). Indeed, temporal ordering of actions was used by many interactive search systems during previous competitions [9]. In tools designed by our team in previous years [13, 16, 17], the temporal sequence was determined in a window limiting the duration period between two actions. In the new version of Prak, we consider another approach allowing the detection of temporally distant activities within one day, with or without a predefined ordering of actions. Specifically, we consider two algorithms based on a joint embedding approach (OpenCLIP or more recent models) to rank days based on temporally distant activities query $Q_{TDA} = \{q_1, \cdots, q_n\}$. Note that vectors of all images are pre-computed in advance, while $q_i = f_{text}(Q_i)$ are computed once the query is available using the corresponding text embedding function f_{text} for a text Q_i describing one action. Both algorithms assume that the system uses the cosine distance $\delta_{cos}(q_i, o_j)$ for each vector representation of an image O_j captured in a day, where D will be used as a label for the set of these images.

The first algorithm in its basic form computes the score of a day D without any ordering expectations as:

$$score_D = \sum_{q_i \in Q_{TDA}} \max_{j=1...|D|} (\delta_{cos}(q_i, o_j))$$

This approach allows to find a day, where users specify a list of actions like "riding a horse", "driving a car", and "swimming in a pool". These actions could happen in an arbitrary order. The second algorithm considers also time ordered sequences of images with respect to the ordering of text queries. Following the example, the user could search for a sequence "riding a horse THEN driving a car THEN swimming in a pool" that happened in this particular order some day. Finding the optimal set of time-ordered frames corresponding to ordered k text queries leads to an expensive algorithm using multiple nested loops. To speed up the computation, we significantly limit the size of input for the price of more fuzzy specification of boundaries between actions. Specifically, we aggregate all images to clusters representing hours of the day. For each cluster, the algorithm computes $\max_{j=1...|H|}(\delta_{cos}(q_i, o_j))$ for each $q_i \in Q_{TDA}$ and images from a given hour H. Provided that the algorithm computes a vector of action scores $s_{q_1}, \cdots, s_{q_n}$ for each hour of a day, the algorithm then finds the overall sub-optimal score of the day as illustrated in Table 1. Please note that in the example, proper time ordering of action queries q_2 and q_3 is not guaranteed. On the other hand, the algorithm is way more faster and the effect might be just marginal assuming also an option that users can confuse the order of temporally close activities.

hour of the day	10	11	12	13	14	15	16
action q_1 score	0.5	0.3	**0.1**	0.4	0.6	0.4	0.0
action q_2 score	0.4	0.0	0.3	0.2	**0.1**	0.3	0.4
action q_3 score	0.1	0.2	0.3	0.5	**0.1**	0.2	0.2

Table 1: Finding optimal scores with the softer time ordering constraint for actions. Hour based aggregation of images is used to compute the overall score of one selected day.

Ranking of days based on the description of temporally distant activities requires appropriate visualization. We plan to experiment with various row-based visualizations of a matching day. In the basic setting, the method will show the top matching images that led to detected minimal scores $s_{q_1}, \cdots, s_{q_n}$. The images will be surrounded by temporally close images from the same day.

The second extension of the PraK system is straightforward and rather technical, as the LSC dataset and tasks are designed to be multi-modal. Hence, filtering using datetime or other metadata is necessary for most successful LSC systems. To support metadata-based filtering, we extend video data services (video at VBS is analogous to a day at LSC) by a new parameter, allowing the specification of a filter to be applied before the similarity search. The data layer backend extension comprises a component allowing SQL-like queries (we use pandas) that keeps a table of meta-data and allows memory-based processing of filters provided by the application service.

Last, we updated the PraK architecture by moving the control of the current state of relevance scores to the application backend. The primary motivation is to keep data-intensive communication on a fast backbone (larger sets of high-dimensional vectors) while only small lists of image links are sent to the web frontend that relies on currently available (potentially very slow) internet connection. Nevertheless, we keep the option to send all the data to the web frontend for potentially expensive active learning models that could burden the application backend server.

5 COMMUNICATION PERFORMANCE STUDY

Since the PraK tool is not one monolith component with methods accessing shared memory, we also present a simple communication performance study providing insights into data transfer limits. Due to the different environments involved, first, we analyze whether data transfers over the network could represent a performance bottleneck. We note that this issue was more prevalent in the previous version of PraK where all feature vectors were transferred to the web frontend via a network without guaranteed bandwidth. Second, we performed a preliminary analysis of data serialization methods.

5.1 Network and platform configurations

In the first step, we focused on network communication limits between two components, where the highest volumes of data transfers are expected – stateless Video Data Service and the Application Service Backend.

The services run on a cluster consisting of several servers with Intel Xeon 6348 processors with 1 TB RAM, running VMware vSphere version 7.0U3p on top of them. The servers in the cluster are interconnected by 10 GbE (Dell MXL 10/40GbE switch, Intel X710 SFP+ network cards). Table 2 shows measured results for all tested deployment configurations. All rows show the measured performance (bandwidth and latency, average of 10 runs) of the communication between two components on our cluster. The OS in the Virtual Machines (VMs) is always Ubuntu 22.04.

The first measurement, VM-VM-same, shows the communication performance between two VMs running on the same server. In this case, it is a memory-bound task because the vSphere hypervisor does not use external communication over the network card when two VMs on the same node communicate but only move packets in memory. The second measurement, VM-VM-sep, shows the communication performance between two VMs running on different

Test name	Bandwidth (GB/s)	Latency (μs)
VM-VM-same	3.39	25.3
VM-VM-sep	1.17	35.1
VM-NAT-VM-sep	1.14	85.1
CLI-NAT-VM-sep	0.12	113.5
DOCK-DOCK-same	4.39	12.1
home-NAT-VM-sep	0.007	5121

Table 2: Measured network performance.

servers. The measurements show that we are at the performance limit of a 10 GbE network and that virtualization does not affect the performance of network services. The third measurement, VM-NAT-VM-sep, represents the performance when one of the servers is behind NAT. This corresponds to a real-world situation where NAT is used to access servers due to a lack of free IPv4 addresses. We can see that the performance has measurably decreased. In particular, the latency is almost triple that of the previous measurement due to the additional embedded communication. However, the bandwidth is still very close to the limit of 10 GbE technology. The next measurement, CLI-NAT-VM-sep, is a measurement between a regular office PC connected via 1 GbE to the network infrastructure in the same building where our cluster is located. The measurement shows that we are again at the limit of 1GbE technology in this case, and there is no major delay or performance loss. The DOCK-DOCK-same measurement is the performance of network communication between two containers created with Docker inside a single VM. Again, this is a real-world situation (currently employed by PraK) where both of our services are placed in different containers in a single VM to optimize performance. The results show that we achieved the best results in both measured parameters.

All measurements show that the system performance is not hampered by the different technologies used to run our services and communicate between them and is at the expected upper limit of the technologies used.

As an exceptional measurement of home-NAT-VM-sep, we measured the network performance when connected to a laptop from home via WiFi. Although the home connection is at a high level compared to the national average (WiFi 6 router and laptop, 1000/60 Mbps Vodafone cable connection), the resulting performance is several orders of magnitude worse than the same CLI-NAT-VM-sep test, which was performed in the same building as our cluster. Several factors come into play here, but probably the most significant is the aggregation of the connection on both the client and the building side with the cluster, where the entire IT part of the faculty is connected by 1x10 Gb/s fiber optic cable to the outside world.

5.2 Data serialization techniques

In the second step, we focused on the actual data transfer between services implemented in JSON format, specifically on serialization, network transfer, and deserialization. In response to a query, video data services send the top k results (e.g., $k = 2000$) with a high-dimensional vector for each item in the result set, where the individual components of the vector are float numbers. To measure and analyze the performance limits of this data transfer, we developed a simple test framework in C++.

In the first test, we created the simplest JSON representation for the high-dimensional data, assuming 2000 vectors with 1000 dimensions. The total size of the JSON string was about 36.6 MB, float numbers were represented here as a full float format, and the length of the number notation was typically 16 characters. If, for simplicity, we generally assume a situation where our services are running in different VMs (corresponding to the VM-VM-sep measurement), then purely transferring such a JSON string would take around 31 ms. We still have to take into account serialization and deserialization time, which can be highly non-trivial. In the case of our simple testing framework, it was 20.3 ms/77.4 ms for serialization/deserialization. However, different JSON libraries may have quite different values for this serialization and deserialization time, the transfer time remains.

If we want to reduce the transfer time, it is necessary to reduce the generated string. One option is to stick with the JSON format and try to round the numbers to some fixed number of decimal places. For the analysis, we tested a length of 5 decimal places, so the usual length of the number entry is 7 characters. This resulted in a reduction of the JSON string size to 18.4 MB, resulting in a transfer time of around 16 ms. On the other hand, this change affected serialization and deserialization times, where serialization took slightly longer at 23.1 ms (calculating rounding to the required number of digits), while deserialization ran slightly faster at 58.5 ms (reading and parsing fewer characters).

Another simple option is to change the format to some binary format. Here, the BSON format, which is just the binary equivalent of the JSON format, is the first option to go. When encoding the same amount of data, we get a BSON format output of 24.6 MB, resulting in a transfer time of around 21 ms. However, the serialization and deserialization times changed significantly and were significantly faster, taking 15.8 ms and 25.2 ms for serialization and deserialization, respectively.

The appropriate choice of libraries and communication format for the data seems crucial to achieve higher performance for the backend and data services. Improving communication and data serialization can help speed up the whole PraK tool. We will focus on this in future work on the PraK tool and other limitations we found during the competition.

6 CONCLUSIONS

In this paper, we present an extended version of the PraK tool, an interactive video search tool, that was updated to meet the needs of the lifelog search challenge. The overall architecture, components, features, and updates were presented in detail. As a new important feature, temporally distant activities were introduced as a search model to find a sequence of actions that happened during a day. We believe that combining new features and state-of-the-art joint embedding approaches will lead to a competitive performance at the lifelog search challenge.

ACKNOWLEDGMENTS

This publication has been funded by SVV project number 260 698 and by Czech Science Foundation (GAČR) project 22-21696S.

REFERENCES

[1] Naushad Alam, Yvette Graham, and Cathal Gurrin. 2023. Memento 3.0: An Enhanced Lifelog Search Engine for LSC'23. In *Proceedings of the 6th Annual on Lifelog Search Challenge* (Thessaloniki, Greece) *(LSC '23)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3592573.3593103

[2] Ahmed Alateeq, Mark Roantree, and Cathal Gurrin. 2023. Voxento 4.0: A More Flexible Visualisation and Control for Lifelogs. In *Proceedings of the 6th Annual on Lifelog Search Challenge* (Thessaloniki, Greece) *(LSC '23)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3592573.3593097

[3] Jim Gemmell, Gordon Bell, and Roger Lueder. 2006. MyLifeBits: A Personal Database for Everything. *Commun. ACM* 49, 1 (Jan 2006), 88–95. https://doi.org/10.1145/1107458.1107460

[4] Cathal Gurrin, Hideo Joho, Frank Hopfgartner, Liting Zhou, Van-Tu Ninh, Tu-Khiem Le, Rami Albatal, Duc-Tien Dang-Nguyen, and Graham Healy. 2019. Advances in Lifelog Data Organisation and Retrieval at the NTCIR-14 Lifelog-3 Task. In *NII Testbeds and Community for Information Access Research*, Makoto P Kato, Yiqun Liu, Noriko Kando, and Charles L A Clarke (Eds.). Springer International Publishing, Cham, 16–28.

[5] Cathal Gurrin, Björn Þór Jónsson, Duc-Tien Dang-Nguyen, Graham Healy, Jakub Lokoč, Liting Zhou, Luca Rossetto, Minh-Triet Tran, Wolfgang Hürst, Werner Bailer, and Klaus Schoeffmann. 2023. Introduction to the Sixth Annual Lifelog Search Challenge, LSC'23. In *Proceedings of the 2023 ACM International Conference on Multimedia Retrieval, ICMR 2023, Thessaloniki, Greece, June 12-15, 2023*. ACM, 678–679. https://doi.org/10.1145/3591106.3592304

[6] Cathal Gurrin, Klaus Schoeffmann, Hideo Joho, Andreas Leibetseder, Liting Zhou, Aaron Duane, Duc-Tien Dang-Nguyen, Michael Riegler, Luca Piras, et al. 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.

[7] Cathal Gurrin, Alan F. Smeaton, and Aiden R. Doherty. 2014. LifeLogging: Personal Big Data. *Foundations and Trends® in Information Retrieval* 8, 1 (2014), 1–125. https://doi.org/10.1561/1500000033

[8] Cathal Gurrin, Liting Zhou, Graham Healy, Bailer. Werner, Duc-Tien Dang-Nguyen, Steve Hodges, Björn Þór Jónsson, Jakub Lokoč, Luca Rossetto, Minh-Triet Tran, and Klaus Schöffmann. 2024. Introduction to the Seventh Annual Lifelog Search Challenge, LSC'24. *International Conference on Multimedia Retrieval (ICMR'24)*. https://doi.org/10.1145/3652583.3658891

[9] Silvan Heller. 2023. *Temporal Multimodal Video and Lifelog Retrieval.* PhD Thesis. https://dbis.dmi.unibas.ch/publications/2023/PhD-SilvanHeller-23/.

[10] Nhat Hoang-Xuan, Thang-Long Nguyen-Ho, Cathal Gurrin, and Minh-Triet Tran. 2023. Lifelog Discovery Assistant: Suggesting Prompts and Indexing Event Sequences for FIRST at LSC 2023. In *Proceedings of the 6th Annual on Lifelog Search Challenge* (Thessaloniki, Greece) *(LSC '23)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3592573.3593104

[11] Maria Tysse Hordvik, Julie Sophie Teilstad Østby, Tu-Khiem Le Manoj Kesavulu, Thao-Nhu Nguyen, and Duc-Tien Dang-Nguyen. 2023. LifeLens: Transforming Lifelog Search with Innovative UX/UI Design. In *Proceedings of the 6th Annual on Lifelog Search Challenge* (Thessaloniki, Greece) *(LSC '23)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3592573.3593096

[12] Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. 2021. *OpenCLIP*. https://doi.org/10.5281/zenodo.5143773

[13] Miroslav Kratochvíl, František Mejzlík, Patrik Veselý, Tomáš Souček, and Jakub Lokoč. 2020. SOMHunter: Lightweight Video Search System with SOM-Guided Relevance Feedback. In *MM '20: The 28th ACM International Conference on Multimedia, Virtual Event / Seattle, WA, USA, October 12-16, 2020*, Chang Wen Chen, Rita Cucchiara, Xian-Sheng Hua, Guo-Jun Qi, Elisa Ricci, Zhengyou Zhang, and Roger Zimmermann (Eds.). ACM, 4481–4484. https://doi.org/10.1145/3394171.3414542

[14] Jakub Lokoč, František Mejzlík, Tomáš Souček, Patrik Dokoupil, and Ladislav Peška. 2022. Video Search with Context-Aware Ranker and Relevance Feedback. In *MultiMedia Modeling*. Springer International Publishing, Cham, 505–510.

[15] Jakub Lokoč, Stelios Andreadis, Werner Bailer, Aaron Duane, Cathal Gurrin, Zhixin Ma, Nicola Messina, Thao-Nhu Nguyen, Ladislav Peška, Luca Rossetto, Loris Sauter, Konstantin Schall, Klaus Schoeffmann, Omar Shahbaz Khan, Florian Spiess, Lucia Vadicamo, and Stefanos Vrochidis. 2023. Interactive video retrieval in the age of effective joint embedding deep models: lessons from the 11th VBS. *Multim. Syst.* 29, 6 (2023), 3481–3504. https://doi.org/10.1007/S00530-023-01143-5

[16] Jakub Lokoč, Gregor Kovalčík, Tomáš Souček, Jaroslav Moravec, and Přemysl Čech. 2019. A Framework for Effective Known-item Search in Video. In *Proceedings of the 27th ACM International Conference on Multimedia, MM 2019, Nice, France, October 21-25, 2019*, Laurent Amsaleg, Benoit Huet, Martha A. Larson, Guillaume Gravier, Hayley Hung, Chong-Wah Ngo, and Wei Tsang Ooi (Eds.). ACM, 1777–1785. https://doi.org/10.1145/3343031.3351046

[17] Jakub Lokoč and Ladislav Peška. 2023. A Study of a Cross-modal Interactive Search Tool Using CLIP and Temporal Fusion. In *MultiMedia Modeling - 29th International Conference, MMM 2023, Bergen, Norway, January 9-12, 2023, Proceedings, Part I (Lecture Notes in Computer Science, Vol. 13833)*, Duc-Tien Dang-Nguyen, Cathal Gurrin, Martha A. Larson, Alan F. Smeaton, Stevan Rudinac, Minh-Son Dao, Christoph Trattner, and Phoebe Chen (Eds.). Springer, 397–408. https://doi.org/10.1007/978-3-031-27077-2_31

[18] Jakub Lokoč, Zuzana Vopálková, Michael Stroh, Raphael Buchmueller, and Udo Schlegel. 2024. PraK Tool: An Interactive Search Tool Based on Video Data Services. In *MultiMedia Modeling - 30th International Conference, MMM 2024, Amsterdam, The Netherlands, January 29 - February 2, 2024, Proceedings, Part IV (Lecture Notes in Computer Science, Vol. 14557)*, Stevan Rudinac, Alan Hanjalic, Cynthia C. S. Liem, Marcel Worring, Björn Þór Jónsson, Bei Liu, and Yoko Yamakata (Eds.). Springer, 340–346. https://doi.org/10.1007/978-3-031-53302-0_30

[19] Thao-Nhu Nguyen, Tu-Khiem Le, Van-Tu Ninh, Cathal Gurrin, Minh-Triet Tran, Thanh Binh Nguyen, Graham Healy, Annalina Caputo, and Sinead Smyth. 2023. E-LifeSeeker: An Interactive Lifelog Search Engine for LSC'23. In *Proceedings of the 6th Annual on Lifelog Search Challenge* (Thessaloniki, Greece) *(LSC '23)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3592573.3593098

[20] Tien-Thanh Nguyen-Dang, Xuan-Dang Thai, Gia-Huy Vuong, Van-Son Ho, Minh-Triet Tran, Van-Tu Ninh, Minh-Khoi Pham, Tu-Khiem Le, and Graham Healy. 2023. LifeInsight: An Interactive Lifelog Retrieval System with Comprehensive Spatial Insights and Query Assistance. In *Proceedings of the 6th Annual on Lifelog Search Challenge* (Thessaloniki, Greece) *(LSC '23)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3592573.3593106

[21] 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. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event (Proceedings of Machine Learning Research, Vol. 139)*, Marina Meila and Tong Zhang (Eds.). PMLR, 8748–8763. http://proceedings.mlr.press/v139/radford21a.html

[22] Ricardo Ribeiro, Luísa Amaral, Wei Ye, Alina Trifan, and António J. R. Neves. 2023. MEMORIA: A Memory Enhancement and MOment RetrIeval Application for LSC 2023. In *Proceedings of the 6th Annual on Lifelog Search Challenge* (Thessaloniki, Greece) *(LSC '23)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3592573.3593099

[23] Luca Rossetto, Ralph Gasser, Loris Sauter, Abraham Bernstein, and Heiko Schuldt. 2021. A System for Interactive Multimedia Retrieval Evaluations. In *International Conference on Multimedia Modeling*. Springer, 385–390.

[24] Luca Rossetto, Oana Inel, Svenja Lange, Florian Ruosch, Ruijie Wang, and Abraham Bernstein. 2023. Multi-Mode Clustering for Graph-Based Lifelog Retrieval. In *Proceedings of the 6th Annual on Lifelog Search Challenge* (Thessaloniki, Greece) *(LSC '23)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3592573.3593102

[25] Klaus Schoeffmann. 2023. lifeXplore at the Lifelog Search Challenge 2023. In *Proceedings of the 6th Annual on Lifelog Search Challenge* (Thessaloniki, Greece) *(LSC '23)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3592573.3593105

[26] Florian Spiess, Ralph Gasser, Heiko Schuldt, and Luca Rossetto. 2023. The Best of Both Worlds: Lifelog Retrieval with a Desktop-Virtual Reality Hybrid System. In *Proceedings of the 6th Annual on Lifelog Search Challenge* (Thessaloniki, Greece) *(LSC '23)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3592573.3593107

[27] Ly Duyen Tran, Binh Nguyen, Liting Zhou, and Cathal Gurrin. 2023. MyEachtra: Event-based Interactive Lifelog Retrieval System for LSC'23. In *Proceedings of the 6th Annual on Lifelog Search Challenge* (Thessaloniki, Greece) *(LSC '23)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3592573.3593100

[28] Ly-Duyen Tran, Manh-Duy Nguyen, Duc-Tien Dang-Nguyen, Silvan Heller, Florian Spiess, Jakub Lokoč, Ladislav Peška, Thao-Nhu Nguyen, Omar Shahbaz Khan, Aaron Duane, Björn þór Jónsson, Luca Rossetto, An-Zi Yen, Ahmed Alateeq, Naushad Alam, Minh-Triet Tran, Graham Healy, Klaus Schoeffmann, and Cathal Gurrin. 2023. Comparing Interactive Retrieval Approaches at the Lifelog Search Challenge 2021. *IEEE Access* 11 (2023), 30982–30995. https://doi.org/10.1109/ACCESS.2023.3248284

[29] Ly-Duyen Tran, Manh-Duy Nguyen, Binh Nguyen, Hyowon Lee, Liting Zhou, and Cathal Gurrin. 2022. E-Myscéal: Embedding-Based Interactive Lifelog Retrieval System for LSC'22. In *Proceedings of the 5th Annual on Lifelog Search Challenge* (Newark, NJ, USA) *(LSC '22)*. Association for Computing Machinery, New York, NY, USA, 32–37. https://doi.org/10.1145/3512729.3533012

[30] Quang-Linh Tran, Ly-Duyen Tran, Binh Nguyen, and Cathal Gurrin. 2023. MemoriEase: An Interactive Lifelog Retrieval System for LSC'23. In *Proceedings of the 6th Annual on Lifelog Search Challenge* (Thessaloniki, Greece) *(LSC '23)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3592573.3593101