Multi-Mode Clustering for Graph-Based Lifelog Retrieval

Luca Rossetto
Department of Informatics
University of Zurich
Switzerland
rossetto@ifi.uzh.ch

Oana Inel
Department of Informatics
University of Zurich
Switzerland
inel@ifi.uzh.ch

Svenja Lange
Department of Informatics
University of Zurich
Switzerland
lange@ifi.uzh.ch

Florian Ruosch
Department of Informatics
University of Zurich
Switzerland
ruosch@ifi.uzh.ch

Ruijie Wang
Department of Informatics
University Research Priority Program
"Dynamics of Healthy Aging"
University of Zurich
Switzerland
ruijie@ifi.uzh.ch

Abraham Bernstein
Department of Informatics
University of Zurich
Switzerland
bernstein@ifi.uzh.ch

ABSTRACT

As part of the 6th Lifelog Search Challenge, this paper presents an approach to arrange Lifelog data in a multi-modal knowledge graph based on cluster hierarchies. We use multiple sequence clustering approaches to address the multi-modal nature of Lifelogs in relation to temporal, spatial, and visual factors. The resulting clusters, along with semantic metadata captions and augmentations based on OpenCLIP, provide for the semantic structure of a graph including all Lifelogs as entries. Textual queries on this hierarchical graph can be expressed to retrieve individual Lifelogs, as well as clusters of Lifelogs.

CCS CONCEPTS

• **Information systems** → *Users and interactive retrieval*; *Specialized information retrieval*; **Multimedia and multimodal retrieval**.

KEYWORDS

Lifelogging, Lifelog Search Challenge, Knowledge Graphs, Graph-based Retrieval, Multi-modal Retrieval

ACM Reference Format:
Luca Rossetto, Oana Inel, Svenja Lange, Florian Ruosch, Ruijie Wang, and Abraham Bernstein. 2023. Multi-Mode Clustering for Graph-Based Lifelog Retrieval. In *6th Annual ACM Lifelog Search Challenge (LSC '23), June 12, 2023, Thessaloniki, Greece.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3592573.3593102

1 INTRODUCTION

Lifelogs are inherently multi-modal records of a person's everyday experience, capturing a wide range of information. While the primary component of the Lifelogs available in the context of this benchmark consists of first-person perspective images captured by a wearable camera, they are accompanied by substantial implicit and explicit contextual information. Such context can lead to different perspectives from which the data can be observed and along which it can be organized.

In this paper, we present our contribution to the 6th instance of the Lifelog Search Challenge [4]. Our approach focuses on sequentially clustering Lifelog entries using different aggregation semantics, structuring these resulting clusters hierarchically by semantics, interrelating them across aggregation schemes, and connecting them with other contextual information. We analyze images and their accompanying data to make sense of the complex structures underneath and try to couple them with an efficient way to search and browse.

The approach can be seen as a "spiritual" successor to previous instances of LifeGraph [13, 14], which participated in previous instances of LSC. We hence refer to it as LifeGraph 3. This time around, the focus lies on clusters that are inherently present in the data set. We identify different types of clusters and group the instances accordingly using various techniques. In particular, we infer temporal, spatial, and visual clusters that allow us to arrange sequences of the Lifelog entries into meaningful bins. The initial clusters are exclusively generated based on information contained in the challenge dataset, which is comprised of 18 months of Lifelog images and accompanying metadata generated by one person. The dataset is the same as in 2022 [7] and encompasses over 700k individual Lifelog entries. The pre-processed data is stored in a multi-modal knowledge graph and served to the browser-based frontend through an API. This allows users to construct different types of queries to filter the Lifelog entries based on the information available. Finally, the frontend also provides the functionality to traverse the hierarchy of the cluster of log entries, allowing us to refine the set of relevant Lifelog entries for a given query.

The remainder of this paper is structured as follows: Section 2 discusses related work, followed by a description of the graph construction in Section 3. Next, Section 4 outlines the query processes and Section 5 gives an overview of the system, before Section 6 concludes.

LSC '23 , June 12, 2023, Thessaloniki, Greece

ACM ISBN 979-8-4007-0188-7/23/06.
https://doi.org/10.1145/3592573.3593102

2 RELATED WORK

Over the last five years, several approaches and systems have been proposed in the Lifelog Search Challenge [5–7, 23]. Among the techniques used by the participating teams, we found concept-based search, multi-modal embeddings, and temporal queries to be the most common (see, for instance, a summary of approaches participating in the 2021 edition of the challenge [23]). Many participating systems are also long-standing participants in the challenge, which constantly augment their systems with novel relevant retrieval techniques. For example, the lifeXplore [8] system has been participating in LSC since the first edition, in 2018. It is based on the interactive video browsing and retrieval system called diveXplore [9] and is optimized for the efficient exploration and filtering of a large number of result images. During the last participation, lifeXplore had a new functionality that allowed combining several queries in a temporal view, further improving their calendar view's browsing capabilities.

The Myscéal system outperformed all other participants in the last three editions of the challenge [22, 24, 25]. The system is built around responsive retrieval of a multitude of semantic concepts extracted from the visual part of the data. While at its core, the system was primarily full-text based, in the last edition of the challenge, it was augmented with visual-text co-embeddings based on CLIP. In addition to visual-textual co-embeddings, in its fourth participation to the challenge, the LifeSeeker system [12] groups lifelog entries based on temporal and spatial information, emotions evoked by the music played while recording the lifelog entries, and lifelog entries clustering based on event information. More precisely, the event clustering groups consecutive images belonging to the same event and selects one representative image to simplify browsing.

The vitrivr system [15] is an open-source multimedia retrieval stack that supports the retrieval of a multitude of media types (i.e., image, audio, video, 3d models) and query modes suitable for these media types. It has participated in LSC since 2019 [15], when it also scored highest. In a nutshell, the vitrivr stack consists of three components: the Cottontail DB [3] database layer, the Cineast [17] query processing engine, and a user interface that allows for query refinement through various filtering techniques. Traditionally, this user interface is a browser-based application. Since 2021, however, a second version of the stack using a virtual reality-based user interface has joined the benchmark under the name vitrivr-VR [21].

Several systems chose to represent the lifelog entries as a graph structure [11, 13, 14] and then enhance the available information. The LifeGraph system participating in the 2020 [13] and 2021 [14] editions of LSC used a knowledge graph-based approach in order to facilitate semantic expansion and contextualization of concepts. By linking instances of detected concepts and objects visible in the Lifelog images with a large knowledge base, more abstract semantic concepts could be indirectly queried via graph traversal. Using a similar approach, Nguyen et al. [11] constructs scene graphs for individual Lifelog images. These scene graphs can then be compared to graphs constructed from textual queries.

In our approach, similarly to the methods proposed in [12, 22, 24, 25], we also take advantage of multi-modal embeddings, which are proving efficient in various applications such as image search, image captioning, or action recognition [2]. Furthermore, as the LifeGraph system, we also represent the semantics in the lifelog entries in a graph structure. While Nguyen et al. [12] use event clustering to select one representative lifelog entry and simplify browsing, we used several cluster types to group entries into semantically distinct categories, such as temporal, spatial, visual, and, to a limited extent, activity-based.

3 GRAPH CONSTRUCTION

The driving philosophy behind the structure of the graph is that of hierarchical sequence clustering of the Lifelog entries (i.e., the recorded images) using multiple different clustering criteria. Each cluster consists of a continuous series of log entries, while each log entry can be part of arbitrarily many clusters of different semantics but only one cluster of one type. Clusters are then aggregated along different levels of a semantic hierarchy. Figure 1 shows an example of a possible structure resulting from this process.

In order to construct the graph, some initial data pre-processing needs to be performed before the sequence clustering can be applied and further information can be extracted and related.

3.1 Pre-processing

Most of the clustering approaches used in our graph rely on the metadata provided with the dataset. This data comes in the form of a sparse table with one column per metadata dimension, encoded in a CSV file. For easier processing, we normalize and filter the table such that we have one row per image. Rows not associated with an image are discarded. In order to reduce the sparsity of the table, we identify sufficiently small gaps in each column which are bounded by identical values. In these cases, we back-fill these values throughout the gaps, assuming a constant value at this point in time. No interpolation across gaps bounded by different values is performed. For example, for a gap of size j, we assume the missing locations, if the images at times t_i and t_{i+j} share a location, supposing that the lifelogger has not moved.

3.2 Cluster types

The different types of clustering methods applied to the sequences of log entries can be grouped into several semantically distinct categories.

3.2.1 Temporal. The most fundamental clustering category is independent of any semantic content of the individual log entries and solely represents the time at which they were created. The smallest unit of time used here is the *day*, which are then further grouped into *month*s and *year*s. While this aggregation by itself is of limited use, it serves as an overarching structure for other aggregation mechanisms. The log entries in the dataset are not, in fact, completely continuous, as they do not contain the times the lifelogger was asleep. However, a night forms a natural boundary that no other aggregation scheme crosses. Each *day* cluster, therefore, forms a natural super-set of everything else being described in aggregation during that time.

3.2.2 Spatial. All spatial clustering schemes are concerned with the lifelogger's physical location. They all operate on the metadata provided with the dataset and do not perform any additional location estimation based on visual input. Specifically, the schemes

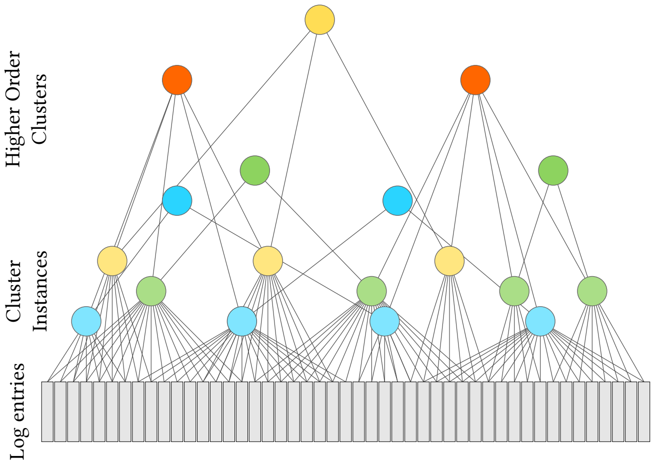

Figure 1: Example of cluster structure. Cluster instances on the lowest level consist of continuous sequences of log entries. The sequence of log entries does not need to be fully covered by all types of clusters. Higher order clusters aggregate multiple clusters of one or different types.

are based on the *latitude*, *longitude*, and *semantic name* columns of the metadata table. We use three different criteria to obtain spatial clusters.

Provided semantic location. The most straightforward of the spatial clustering approaches just uses the provided *semantic name* attribute and groups all log entries of a continuous sequence with the same attribute value.

Inferred semantic location. Since the provided semantic location labels are flat and offer no contextual information, we infer additional semantic labels based on the GPS coordinates. To do that, we query Wikidata[1] for the closest physical entity with a spatial position to any log entry. Continuous sequences of identical labels are clustered without taking any further information into account. Higher-order clusters can then be formed from adjacent clusters sharing a property, e.g., the country or city they are located in. We also consider properties that allow us to build hierarchies of clusters based on the data available in Wikidata, such as an identified locality is in district, district is in city, city is in county, county is in state, and state is in country.

GPS location. To cluster log entries based on GPS location, we quantize the location information to three significant digits, resulting in cells of roughly 100×100 meters. All log entries during which the lifelogger does not leave a cell are aggregated into a cluster. In addition, we use reverse geo-lookup[2] to identify the city and the country for every position, which serve as parent-clusters.

3.2.3 *Visual.*

The visual clustering schemes operate on information that can be extracted from the images directly. These mechanisms make no use of any of the provided metadata.

Shot segmentation. The original Lifelog entries are organized in time series and can be considered as frames of first-person video recordings with very low frame rates. For example, the 1,182 entries recorded on Jan 02, 2019, from 09:19 to 20:46 can be considered as a video with a frame rate of around 0.03 frames per second. Utilizing visual information, we segment lifelog entries into shots that depict distinct visual features and can be used for downstream searching and browsing on a shot-level granularity. A recent model called TransNetV2 [19] is employed for this task. It consists of several deep dilated convolutional neural networks and considers similarities between neighboring frames. Specifically, we convert consecutive entries of each date into an input video with the frame size 27×48, as required by the pre-trained TransNetV2.[3] Then, TransNetV2 computes a value for each entry that denotes the likelihood of the entry being a boundary between two shots. Based on empirical analyses, we define a threshold for this likelihood value, classify entries into boundaries and internals, and obtain final shot segmentations.

Scene classification. In order to cluster sequences by the type of environment the lifelogger finds themselves in, we apply a scene multi-class classifier trained on the Places [26] dataset to every image in the dataset. We keep all labels with a probability of at least 0.1. This leaves us with at least one label per image. For clustering, we use a greedy method that uses set intersection between the union of all labels of a sequence and a next sequence element as an inclusion criterion. If the intersection is non-empty, the next element is added to the sequence. Otherwise, a new sequence is started. Since there is a clear domain shift between the third-person perspective images in the training set of the classifier and the first-person perspective of the images associated with the life log entries, there are several instances where the scene classification produces unusable results. This, in turn, results in an uneven length of cluster sequences, since miss-classifications can break a sequence. To avoid meaningless sequences, we discard all clusters with a sequence length of less than 10.

3.2.4 *Activity.*

We aimed to use the heart rate column in the provided metadata as an indicator for time periods of increased physical activity or stress. However, preliminary experiments showed no discernible pattern to be visible in the images corresponding to such time periods. We, therefore, have to conclude that the provided biofeedback data is not of sufficient quality to be used for this purpose. Maybe this insight can inform the use of such data in future versions of the benchmark dataset.

3.3 Further information

In addition to the clustering information, we augment the log entries with further information that can be used for querying.

To capture the semantics contained within the individual images, we use a freely available instance of an OpenCLIP [2] model, which has been trained on the LAION-5B [18] dataset.

Each log entry is also associated with the raw text from the caption and OCR columns of the provided metadata table.

The Google Cloud Natural Language API[4] was used to extract common and named entities (i.e., a phrase that identifies or refers to

[1] https://www.wikidata.org
[2] https://docs.juliahub.com/ReverseGeocode/inQ9r/0.3.0/
[3] https://github.com/soCzech/TransNetV2
[4] https://cloud.google.com/natural-language

a real-world object or key information in the text, such as a person, a location, or a product, among others) from the captions and visual tags associated with the Lifelog entries, and link them, when possible, with Wikidata entries. Furthermore, when possible, each such concept was associated with a type (e.g., location, person, organization, or object, among others). To have a more comprehensive understanding of the semantics contained within the individual images, we also extracted synonyms of all visual and textual concepts of the Lifelog entries. For instance, the concept "car" can be referred to as "auto", "automobile", or "motor vehicle", among others. To extract the synonyms of all identified concepts, we used the English lexical database called WordNet[5] and the NLTK[6] python package.

We observed that some captions associated with the Lifelogs could potentially contain commonsense knowledge. Therefore, we map relevant concepts from captions with the semantic network represented in ConceptNet [10, 20], a knowledge graph that connects word phrases with labeled edges. For this, we used an off-the-shelf method proposed by Becker et al. [1]. As an example, a lifelog entry containing an image with tables and chairs could be related to dining.

4 QUERYING

Retrieval using the graph is achieved in a two-stage process. The first step consists of a query operation that selects one or several subgraphs with relevant properties or containing relevant information. The second step then uses these obtained results for interactive exploration, filtering, expansion, and browsing of the results, until the desired log entries are found.

4.1 Graph Querying

Queries in the graph are expressed exclusively using free-text and can be evaluated in a bottom-up or a top-down fashion, or an arbitrary combination of the two.

4.1.1 Bottom-up queries. Bottom-up queries target individual log entries directly. This is achieved either via full-text search using the provided captioning or OCR information or using the visual-text co-embedding provided by OpenCLIP. For each log entry that matches the query, its ID together with a similarity score and all the clusters it is contained in are returned.

4.1.2 Top-down queries. Rather than targeting individual log entries directly, it is also possible to retrieve them through their containing clusters. This can be done through top-down queries, where an arbitrary number of cluster values can be specified. Here, values for clusters of the same semantic type are aggregated using union whereas clusters of different types are intersected.

4.2 Graph Exploration

Once results have been retrieved, they can be explored along the cluster hierarchy. To make exploration more effective, each cluster is shown using one representative log entry by default. This enables a user to quickly discard irrelevant clusters. Clusters can also be expanded to show all retrieved entries belonging to it. In case of bottom-up queries, only directly retrieved log entries are shown upon expansion, except when the user explicitly requests to see all entries. Since all retrieved clusters and their types are known and shown along their hierarchy, results can be efficiently filtered along these categories, in case they turn out to be irrelevant after all. At any point in the cluster hierarchy, it is also possible to request all log entries that would be found underneath, analogously to a previously described top-down query. This enables a user to expand a retrieved result set with different levels of granularity without having to restart the querying process.

5 SYSTEM OVERVIEW

The system that stores the graph and implements the querying mechanisms described above is composed of two components. The *backend* component is responsible for persistently storing the graph, including the image-, vector-, and scalar-information and their relations, as well as providing all this data via an HTTP interface. This notion of a knowledge graph directly containing multi-modal information such as images, we call a MediaGraph. It goes beyond the other multi-modal knowledge graphs by not only treating multi-modal information as part of the graph on a semantic level but also consistently handling storage and data access jointly, independent of data type. The backend is also responsible for the querying mechanisms described in Section 4.1. It is built on top of the Cottontail [3] database management system and offers a RESTful API to communicate with the frontend.

The *frontend* is a browser-based application responsible for query formulation and for providing the graph exploration capabilities described in Section 4.2. It also communicates with the evaluation server [16] used during the benchmark, in order to submit relevant task results.

6 CONCLUSION AND OUTLOOK

In this paper, we presented our retrieval approach for the 2023 Lifelog Search Challenge, based on a graph structure constructed from a series of temporal, spatial, and visual clusters. Several different notions of similarity are used for clustering the sequence of log entries and they use different aspects of the information contained within the provided dataset. Clusters are organized into hierarchies and clusters of different types can overlap in time. Log entries are also directly associated with information directly available for similarity search, allowing for both a top-down (cluster-based) and a bottom-up (similarity-based) retrieval approach. While we tried to make use of as much information as the dataset would provide, some of it, especially as far as it described physiological information, turned out to be not suitable for our purposes. Maybe the way in which such data is represented could be reconsidered in future versions of the benchmark dataset.

ACKNOWLEDGMENTS

This work was partially funded by the Digital Society Initiative of the University of Zurich, the University Research Priority Program "Dynamics of Healthy Aging" at the University of Zurich, and the Swiss National Science Foundation through Project "MediaGraph" (Grant Number 202125), and Project "CrowdAlytics" (Grant Number 184994).

[5]https://wordnet.princeton.edu
[6]https://www.nltk.org

REFERENCES

[1] Maria Becker, Katharina Korfhage, and Anette Frank. 2021. COCO-EX: A tool for linking concepts from texts to ConceptNet. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*. 119–126.

[2] Mehdi Cherti, Romain Beaumont, Ross Wightman, Mitchell Wortsman, Gabriel Ilharco, Cade Gordon, Christoph Schuhmann, Ludwig Schmidt, and Jenia Jitsev. 2022. Reproducible scaling laws for contrastive language-image learning. *CoRR* abs/2212.07143 (2022). https://doi.org/10.48550/arXiv.2212.07143 arXiv:2212.07143

[3] Ralph Gasser, Luca Rossetto, Silvan Heller, and Heiko Schuldt. 2020. Cottontail DB: An Open Source Database System for Multimedia Retrieval and Analysis. In *MM '20: The 28th ACM International Conference on Multimedia, Virtual Event / Seattle, WA, USA, October 12-16, 2020*. ACM, 4465–4468. https://doi.org/10.1145/3394171.3414538

[4] Cathal Gurrin, Björn Þór Jónsson, Duc Tien Dang Nguyen, Graham Healy, Jakub Lokoc, 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 International Conference on Multimedia Retrieval* (Thessaloniki, Greece) *(ICMR '23)*. Association for Computing Machinery, New York, NY, USA.

[5] Cathal Gurrin, Tu-Khiem Le, Van-Tu Ninh, Duc-Tien Dang-Nguyen, Björn Þór Jónsson, Jakub Lokoč, Wolfgang Hürst, Minh-Triet Tran, and Klaus Schoeffmann. 2020. Introduction to the third annual lifelog search challenge (LSC'20). In *Proceedings of the 2020 International Conference on Multimedia Retrieval*. 584–585.

[6] Cathal Gurrin, Klaus Schoeffmann, Hideo Joho, Andreas Leibetseder, Liting Zhou, Aaron Duane, Duc-Tien Dang-Nguyen, Michael Riegler, Luca Piras, Minh-Triet Tran, et al. 2019. [invited papers] 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, Liting Zhou, Graham Healy, Björn Þór Jónsson, Duc-Tien Dang-Nguyen, Jakub Lokoc, Minh-Triet Tran, Wolfgang Hürst, Luca Rossetto, and Klaus Schöffmann. 2022. Introduction to the Fifth Annual Lifelog Search Challenge, LSC'22. In *ICMR '22: International Conference on Multimedia Retrieval, Newark, NJ, USA, June 27 - 30, 2022*. ACM, 685–687. https://doi.org/10.1145/3512527.3531439

[8] Andreas Leibetseder and Klaus Schoeffmann. 2021. LifeXplore at the Lifelog Search Challenge 2021. In *Proceedings of the 4th Annual on Lifelog Search Challenge* (Taipei, Taiwan) *(LSC '21)*. Association for Computing Machinery, New York, NY, USA, 23–28. https://doi.org/10.1145/3463948.3469060

[9] Andreas Leibetseder and Klaus Schoeffmann. 2022. diveXplore 6.0: ITEC's Interactive Video Exploration System at VBS 2022. In *MultiMedia Modeling*, Björn Þór Jónsson, Cathal Gurrin, Minh-Triet Tran, Duc-Tien Dang-Nguyen, Anita Min-Chun Hu, Binh Huynh Thi Thanh, and Benoit Huet (Eds.). Springer International Publishing, Cham, 569–574. https://doi.org/10.1007/978-3-030-98355-0_56

[10] Hugo Liu and Push Singh. 2004. ConceptNet—a practical commonsense reasoning tool-kit. *BT technology journal* 22, 4 (2004), 211–226.

[11] Manh-Duy Nguyen, Nguyen Thanh Binh, and Cathal Gurrin. 2021. Graph-Based Indexing and Retrieval of Lifelog Data. In *MultiMedia Modeling - 27th International Conference, MMM 2021, Prague, Czech Republic, June 22-24, 2021, Proceedings, Part II (Lecture Notes in Computer Science, Vol. 12573)*. Springer, 256–267. https://doi.org/10.1007/978-3-030-67835-7_22

[12] Thao-Nhu Nguyen, Tu-Khiem Le, Van-Tu Ninh, Minh-Triet Tran, Thanh Binh Nguyen, Graham Healy, Sinéad Smyth, Annalina Caputo, and Cathal Gurrin. 2022. LifeSeeker 4.0: An Interactive Lifelog Search Engine for LSC'22 *(LSC '22)*. Association for Computing Machinery, New York, NY, USA, 14–19. https://doi.org/10.1145/3512729.3533014

[13] Luca Rossetto, Matthias Baumgartner, Narges Ashena, Florian Ruosch, Romana Pernischová, and Abraham Bernstein. 2020. LifeGraph: A Knowledge Graph for Lifelogs. In *Proceedings of the Third ACM Workshop on Lifelog Search Challenge, LSC@ICMR 2020, Dublin, Ireland, June 8-11, 2020*. ACM, 13–17. https://doi.org/10.1145/3379172.3391717

[14] Luca Rossetto, Matthias Baumgartner, Ralph Gasser, Lucien Heitz, Ruijie Wang, and Abraham Bernstein. 2021. Exploring Graph-querying approaches in LifeGraph. In *Proceedings of the 4th Annual on Lifelog Search Challenge, LSC@ICMR 2021, Taipei, Taiwan, 21 August 2021*. ACM, 7–10. https://doi.org/10.1145/3463948.3469068

[15] Luca Rossetto, Ralph Gasser, Silvan Heller, Mahnaz Amiri Parian, and Heiko Schuldt. 2019. Retrieval of Structured and Unstructured Data with Vitrivr *(LSC '19)*. Association for Computing Machinery, New York, NY, USA, 27–31. https://doi.org/10.1145/3326460.3329160

[16] Luca Rossetto, Ralph Gasser, Loris Sauter, Abraham Bernstein, and Heiko Schuldt. 2021. A System for Interactive Multimedia Retrieval Evaluations. In *MultiMedia Modeling - 27th International Conference, MMM 2021, Prague, Czech Republic, June 22-24, 2021, Proceedings, Part II (Lecture Notes in Computer Science, Vol. 12573)*. Springer, 385–390. https://doi.org/10.1007/978-3-030-67835-7_33

[17] Luca Rossetto, Ivan Giangreco, and Heiko Schuldt. 2014. Cineast: A Multi-feature Sketch-Based Video Retrieval Engine. In *2014 IEEE International Symposium on Multimedia, ISM 2014, Taichung, Taiwan, December 10-12, 2014*. IEEE Computer Society, 18–23. https://doi.org/10.1109/ISM.2014.38

[18] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. 2022. LAION-5B: An open large-scale dataset for training next generation image-text models. *CoRR* abs/2210.08402 (2022). https://doi.org/10.48550/arXiv.2210.08402 arXiv:2210.08402

[19] Tomás Soucek and Jakub Lokoc. 2020. TransNet V2: An effective deep network architecture for fast shot transition detection. *CoRR* abs/2008.04838 (2020). arXiv:2008.04838 https://arxiv.org/abs/2008.04838

[20] Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 31.

[21] Florian Spiess, Ralph Gasser, Silvan Heller, Luca Rossetto, Loris Sauter, Milan van Zanten, and Heiko Schuldt. 2021. Exploring Intuitive Lifelog Retrieval and Interaction Modes in Virtual Reality with Vitrivr-VR. In *Proceedings of the 4th Annual on Lifelog Search Challenge* (Taipei, Taiwan) *(LSC '21)*. Association for Computing Machinery, New York, NY, USA, 17–22. https://doi.org/10.1145/3463948.3469061

[22] 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 *(LSC '20)*. Association for Computing Machinery, New York, NY, USA, 23–28. https://doi.org/10.1145/3379172.3391719

[23] 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* (2023), 1–1. https://doi.org/10.1109/ACCESS.2023.3248284

[24] 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

[25] Ly-Duyen Tran, Manh-Duy Nguyen, Nguyen Thanh Binh, Hyowon Lee, and Cathal Gurrin. 2021. Myscéal 2.0: A Revised Experimental Interactive Lifelog Retrieval System for LSC'21 *(LSC '21)*. Association for Computing Machinery, New York, NY, USA, 11–16. https://doi.org/10.1145/3463948.3469064

[26] Bolei Zhou, Àgata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. 2018. Places: A 10 Million Image Database for Scene Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* 40, 6 (2018), 1452–1464. https://doi.org/10.1109/TPAMI.2017.2723009