



Graph-Based Indexing and Retrieval of Lifelog Data

Manh-Duy Nguyen¹(✉), Binh T. Nguyen^{2,3,4}, and Cathal Gurrin¹

¹ Dublin City University, Dublin, Ireland
`manh.nguyen5@mail.dcu.ie`

² AISIA Research Lab, Ho Chi Minh City, Vietnam

³ University of Science, Ho Chi Minh City, Vietnam

⁴ Vietnam National University, Ho Chi Minh City, Vietnam

Abstract. Understanding the relationship between objects in an image is an important challenge because it can help to describe actions in the image. In this paper, a graphical data structure, named “Scene Graph”, is utilized to represent an encoded informative visual relationship graph for an image, which we suggest has a wide range of potential applications. This scene graph is applied and tested in the popular domain of lifelogs, and specifically in the challenge of known-item retrieval from lifelogs. In this work, every lifelog image is represented by a scene graph, and at retrieval time, this scene graph is compared with the semantic graph, parsed from a textual query. The result is combined with location or date information to determine the matching items. The experiment shows that this technique can outperform a conventional method.

Keywords: Lifelog · Scene graph · Information retrieval

1 Introduction

As explained in [15], a lifelog is a digital archive gathered by an individual reflecting their real-world life experiences. Lifelogs are typically media-rich, comprising digital images, documents, activities, biometrics, and many other data sources. Such lifelogs have been deployed for many use-cases, such as dietary monitoring [4], memory assistance [10], epidemiological studies [27] and marketing analytics [17]. Regardless of the application, a basic underlying technology is a retrieval mechanism to facilitate content-based access to lifelog items. Research into lifelogging has been gaining in popularity with many collaborative benchmarking workshops taking place recently - the NTCIR Lifelog task [12], the Lifelog Search Challenge (LSC) [13] and the ImageCLEFlifelog [23]. In all of these activities, the query process is similar; a textual query is provided, which acts as the information need for a new generation of retrieval engines that operate over multimodal lifelogs.

In this paper, we introduce a new approach to lifelog retrieval by utilizing a scene graph data structure [18] as the primary indexing mechanism, which

could represent both the objects visible in lifelogging images and the interactions between the objects. Textual user queries are mapped into the same graph space to be compared with the scene graph generated in the previous step to produce the ranked results. Non-visual lifelog data is integrated to support faceted filtering over the generated ranked list. In our experiments, the proposed system and a baseline were evaluated by eight volunteers in an interactive retrieval experiment. We highlight that this paper’s contribution is a first lifelog retrieval system to index the lifelog data in a graph-space and map a textual query to the graph space to facilitate similarity calculations. The original query dataset and the experiment design for the lifelog retrieval are also introduced. To facilitate repeatable science, we release our code for community use¹ and we evaluate using accessible datasets.

2 Related Work

Many interactive lifelog retrieval systems have been proposed in recent years, with MyLifeBits [11] being one of the pioneers, which considered the lifelog retrieval problem as a application of database inquiry [15]. Many novel retrieval approaches followed MyLifeBits, such as Doherty et al. [7], who build a linkage graph for lifelog events and presented a basic interactive browsing system. Lemoore [24] was an early interactive retrieval system which enriched lifelog images by incorporating object labels and facilitated retrieval via textual descriptive queries. Recently, many systems also followed this idea by annotating images with detected items and their semantic concepts in the corresponding metadata. Some of them used this textual information as a filter mechanism to enhance retrieval results produced with a visual-based input sketched by users [16, 21]. The number of matching concepts between a query and the annotation of images can be used as a ranking criterion in retrieval systems [8]. With different considerations, Myscéal [28] viewed this as a document retrieval problem by indexing textual annotations and matching with textual queries. Embedding techniques are also commonly based on the idea of encoding concepts from both queries and images tags into the same vector space to calculate the similarity between them [19, 20]. Regarding using graphs, LifeGraph [25] applied knowledge graph structure with the nodes representing detected things or scenes recognized in images. These entities can be linked with corresponding images and external sources to expand the information with frequent activities and relevant objects.

Generally, all the above systems do not focus on the interaction between objects in the lifelog data or the query. Some systems did make progress by encoding the entire textual input or generating captions for lifelogging images to describe activities appearing in them [30, 31]. However, these ideas did not focus on the association between objects in lifelog images. Some approaches have been proposed to describe visual relations within an image, such as [2, 5]. It was not until the scene graph structure [18] was introduced that there was

¹ To ensure repeatability, our code is publicly available on GitHub for references: <https://github.com/m2man/MMM2021-LifelogRetrieval>.

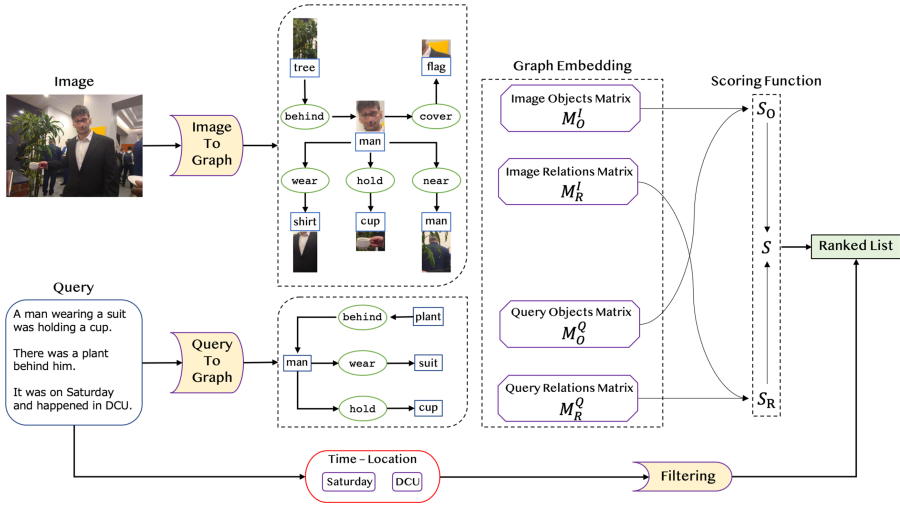


Fig. 1. The overview of our indexing approach for lifelog data and lifelog queries for retrieval purposes. Each lifelog image was converted to a graph, followed by an embedding stage to be stored as an object and relation matrices. A query was then parsed to a graph and encoded to matrices, which would be compared to those of each image by the scoring function. The result, combined with the filtering mechanism of time and location, was returned as a ranked list.

a clear and comprehensive solution to express object relationships in an image and initiate an interesting field for the research community [3]. The proposed graph structure can represent an image as a directed graph comprising nodes and edges where nodes describe objects appearing in the image, and edges indicate the relationship between objects. Many studies have tried applying scene graphs in image retrieval and achieved better results compared to using objects features only [18, 26].

In this paper, we address the lifelog retrieval challenge by indexing both images and textual queries as graphs, as depicted in Fig. 1. We then ranked the matching results based on the similarity between these graphs. Given that we work with multimodal lifelogs, the graph matching process’s outcome could be filtered by other information, such as geolocation or time, which are automatically extracted from the query. This approach facilitated the capture of the interactions between objects in images and the comparison of them with those described in the textual input. Eight users evaluated the proposed method in an experiment comparing the proposed graph-based approach with a recent baseline method using visual concept indexing. For this experiment, we used the LSC’18/19 dataset [14], and we created a new set of twenty semantic queries, including ten randomly chosen topics from the LSC’19 dataset (representing conventional lifelog queries) and ten manually created topics that focus on visually describing a known-item from a lifelog. It is noticeable that [6] also followed the

concept of using a scene graph for lifelogging visual data. However, this system used such a graph as a supplement to the retrieval process and did not consider a query as a graph like our proposed method.

3 Dataset

The lifelog data we used is the official data provided by the recent LSC'18 and 19 [1] comparative benchmarking challenges, which incorporated multimodal lifelog data from a single lifelogger who wore a small camera that passively captured images at the resolution of 1024×768 every 30s for 27 days, leading to the collection of more than 40,000 images. All identifiable information in the dataset was removed by blurring faces and readable textual content. The data also came with the biometric data (heart rate, galvanic skin response, etc.), physical activities (standing, walking, etc.), and GPS location along with its timestamps. We currently used the visual data with its location and date for this work, though future research will incorporate more aspects of the dataset.

Each of the twenty queries represents a lifelogger's textual description to recall a specific moment that happened during one particular time covered by the test collection. The result of a topic could be a single image or a sequence of images. An example of a topic, noted as LSC31, is “[LSC31] *Eating fishcakes, bread, and salad after preparing my presentation in PowerPoint. It must have been lunchtime. There was a guy in a blue sweater. I think there were phones on the table. After lunch, I made a coffee.*”. Additionally, we also built ten new topics that better describe the information need in terms of visual relationships, which we call Descriptive Interaction Topics (DITs). A sample DIT query, named DIT02, is “[DIT02] *I was eating a pizza. My hand was holding a pizza. There was a guy wearing a pink shirt talking to me. There was a black box on a table. It was on Friday morning. It happened at my workplace.*”. The answers to those topics can be illustrated in Fig. 2. In our experimental analysis, we report separately on the results using both types of queries.

4 Graph Generation

As our system aimed to solve the interaction between objects within an image and a semantic query by using a scene graph, it raised a challenge of how to represent these two distinct types of data into a standard graph structure.

4.1 Image to Graph

Although there are many methods for generating a scene graph for an given image, Neural Motifs [33] was chosen due to its accurate performance [3]. A predicted scene graph, noted as G , contains a set O indicating detected objects in the image, with its corresponding set of bounding box B , and the set of visual relations R where:



(a) Sample result for a LSC31



(b) Sample result for DIT02

Fig. 2. Sample results for the example queries for the two mentioned queries. A result for a single topic could contain more than one images.

- $O = \{o_1, \dots, o_m\}$: m labels of recognised objects in the image. Each object o_i was a single node in the graph G .
- $B = \{b_1, \dots, b_m\}$: m bounding boxes of O respectively in which

$$b_i = \{x_i, y_i, width, height\} \in \mathbb{R}^4$$

where (x_i, y_i) is the top-left coordinates of the object $o_i \in O$.

- $R = \{r_1, \dots, r_n\}$: n detected relationships between objects. Each r_k is a triplet association of a start object $o_i \in O$, a end object $o_j \in O$, and a predicate $p_{i \rightarrow j} \in \mathcal{P}$ where \mathcal{P} is a set including all labels of predicates in the Visual Genome. These relations could be considered as edges in G .

All elements in each set are assigned with their confidence score after running the Neural Motif model. We firstly remove inaccurate prediction by setting a threshold for object and relation. To expand the graph to obtain more interaction information in the image that could be not entirely captured by the model, we then create a fully connected graph of G , called G_{fc} , in which there was an edge connecting any two nodes. G_{fc} can be obtained by building missing edges in G with the procedure of visual dependency representations (VDR) [9] based on the bounding box set B . The starting node and the ending node of the constructed edge can be decided based on their predicted scores, whose higher score would be the subject and lower was the object. One drawback of this G_{fc} is that there are many noisy and unimportant relations since not all objects correlated to others. We apply Maximum Spanning Tree on the graph G_{fc} to remove the least meaningful edges with low weights to get G_{mst} . The weight of an edge is the sum of both nodes' scores and that of the predicate connecting them. The score of a predicate could be the score of r_j if this edge was in G or 0 if it was created by the VDR. It is worth noting that we only filter out edges from VDR and retain the original relations in G . In general, the expansion stages of getting the G_{mst} was to enlarge the set R and left two sets O and B intact. The entire process can be illustrated in Fig. 3.

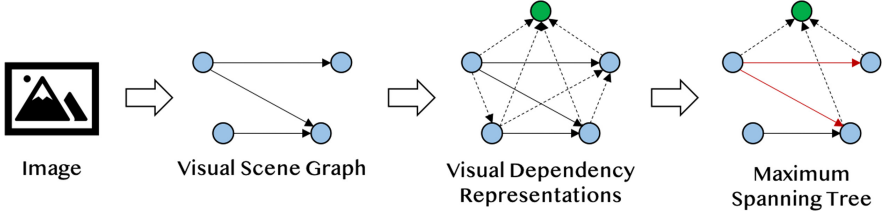


Fig. 3. Image To Graph Pipeline. A visual scene graph was firstly generated from an input image by Neural Motif [33] in which a blue node and an arrow represent an object and a relation respectively. The VDR [9] was applied to generate a fully connected graph with new relations illustrated by dotted arrows. A green node depicted a detected object in the image but not included in the scene graph. Finally, the maximum spanning tree process removed undesirable predicates. Red arrows show relations discarded by the tree but still kept as they were originally in the scene graph. (Color figure online)

4.2 Query to Graph

Besides images, a textual query may also contain several objects and the relations between them. For our proposed process, it is vital to have a graph describing the context of the topic. We applied the rule-based approach proposed in [26] to generate a semantic graph representing query items and their interactions, as described in the query text. Before this, any location and time query information can be extracted and archived for the later filtering mechanism by analyzing part-of-speech tagging from the topic, as utilized in Myscéal [28]. All words from the query are pre-processed to exclude stopwords and then lemmatized.

5 Image Retrieval

After building the graph structure for both images and a query, the retrieval problem became how to calculate the similarity between the semantic graph of the topic with the set of scene graphs of lifelogging photos. This section will describe how a graph is embedded and, based on that, find the similarity score.

5.1 Graph Embedding

Recall that a scene graph of an image G_{mst} has a set of detected objects $O = \{o_1, \dots, o_m\}$ and the expanded relation set $R = \{r_1, \dots, r_k\}$ gained after the spanning process. We now represent the graph in two embedded matrices: M_O^I and M_R^I describing the objects and relations information accordingly. $M_O^I \in \mathbb{R}^{m \times d}$ created with each row is the d -dimension feature vector of a label name of the corresponding object encoded by the Word2Vec [22] model ($d = 300$). Similarly, $M_R^I \in \mathbb{R}^{k \times 3d}$ can be obtained with each row, which is a concatenated embedded vector of a subject, predicate, and an object in the relation. With the same method, a description query can also be encoded into M_O^Q and M_R^Q .

5.2 Similarity Score

Our similarity function is adopted from [32] in which we match the object and relation matrices of a text to those of an image, respectively. Regarding the object matrices, we take the score of the most relevant object in the M_O^I with each of the objects in the M_O^Q . After that, we get the average for the object's similarity score S_O . Assuming there are n_O^I and n_O^Q objects in an image and a query, the S_O can be calculated as:

$$S_O = \frac{1}{n_O^Q} \sum_{t=1}^{n_O^Q} \text{MaxRow}[M_O^Q * \text{Transpose}(M_O^I)], \quad (1)$$

where $*$ is the normal matrix multiplication, $\text{MaxRow}(X)$ is the function to calculate the highest value of each row of a matrix X , and $\text{Transpose}(X)$ is the matrix operation to find X^T . Likewise, suppose that there are n_R^I and n_R^Q relations detected in an image and a query, the relation similarity scores S_R can be measured as follows:

$$S_R = \frac{1}{n_R^Q} \sum_{t=1}^{n_R^Q} \text{MaxRow}[M_R^Q * \text{Transpose}(M_R^I)] \quad (2)$$

Finally, the similarity score between two graphs, S , can be defined as $S = \alpha * S_O + \beta * S_R$ where α, β are obtained from empirical experimentation.

6 Experiments

Since this graph version only focused on indexing data as graphs and applying graph operations, we currently do not take the temporal retrieval issue into account. The baseline for the comparison was the modified version of the Myscéal system [29] that had been used in the ImageCLEF_{lifelog2020} benchmarking workshop and achieved third place out of six participants [23]. We chose this system as the baseline because the top two teams have not released their code at the time we were doing this research. The baseline utilised a standard design of a typical lifelog retrieval system that facilitates textual queries and generates a ranked list by utilizing a scoring function inspired by the TF-IDF. Both baseline and proposed methods were configured to return the top 100 images matching a given query. The users could revise their inquiries until they thought the answers were on the list. There was a time limit of five minutes for a volunteer to solve a single query in each system. The users were trained on each system using several sample queries before the official experiment.

As mentioned in Sect. 3, there were a total of twenty queries used in the experiment, which were divided into four smaller runs, namely A, B, C, and D, with each run containing five topics from either the LSC or DIT types. Eight volunteers were asked to perform two runs, one for each lifelog retrieval system. It means that a single volunteer would use a system to do five queries and

then use the other system to find the answers to another five queries. To avoid any potential learning bias between the first and second runs, we designed the experiment according to Table 1. With this configuration, we could ensure that each setting’s pair would be performed twice with different orders of the systems used to do the retrieval. For example, the couple of A-B experiment was done two times with User 1 and User 7 in a distinct context. While User 1 did run A with the baseline first followed by run B with the proposed system, User 7 used the proposed system for run A before doing run B with the baseline. This configuration allowed the entire query set to be executed twice on each system.

Table 1. The assignment of query subsets and systems for each user in our experiment in which A, B, C, D were our 4 runs (groups of five topics).

	Baseline	Proposed		Proposed	Baseline
User 1	A	B	User 5	C	D
User 2	B	C	User 6	D	A
User 3	C	D	User 7	A	B
User 4	D	A	User 8	B	C

7 Results and Discussion

To evaluate the retrieval system’s accuracy, we used the Mean Reciprocal Rank (MRR) on the top 100 images found by the systems’ users within the experimental timeframes. We chose this metric because it is sensitive to the ranking position, which was also the main criterion in our assessment. We illustrate the scores in Table 2. The graph-based method achieved a higher result (MRR of 0.28) compared to 0.15 for the concept-based system by considering all queries. By examining specific query types, the graph technique also obtained better scores. Due to a competitive MMR on DIT queries from both systems, the proposed method surpassed the baseline with the MMR of 0.41 and 0.2, respectively. The proposed system also got a higher score of 0.15 compared to that of the baseline with 0.1. It can be seen that both methods performed better on DIT topics than LSC topics. This might be because the DIT described the lifelog events in more detail than those in LSC as they had more objects and interactions in the queries. However, there was only a minuscule change in the baseline with an increase of 0.1 in the metric. In contrast, the graph-based retrieval engine witnessed an increase in the scores between two types of topics since this technique could capture the relationships between objects in the query and images, which was the critical point in the DIT set. The MMR of this system on DIT was nearly three times higher than LSC, which were 0.41 and 0.15 accordingly. Figure 4 illustrated the result of both systems for the DIT02 topic.

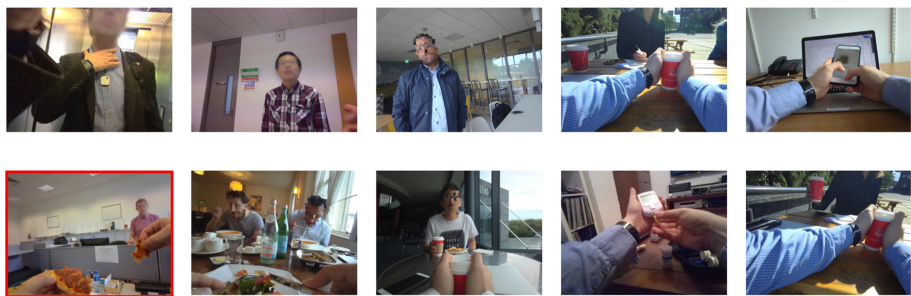


Fig. 4. Top 5 images of the baseline (above) and the proposed system (bottom) for the DIT02 query. The correct answer is marked with the red boundary. (Color figure online)

Table 2. Mean Reciprocal Rank scores of 2 systems on each type of queries and entire dataset.

	LSC	DIT	Entire
Baseline	0.1087	0.2027	0.1557
Proposed	0.1548	0.4166	0.2857

Figure 5 depicts the distribution of reciprocal rank on every query. It was interesting that the baseline system showed less variance than the proposed approach. The variance in the latter system became stronger for DIT topics. It might indicate that the new system was not easy to use as the baseline, making its scores fluctuate between users and queries. The parsing of a query into a graph stage could be the reason. As this step required users to input description in a certain format to fully catch the relations in a query, the volunteers needed to have more time to get familiar with using the graph-based system most efficiently.

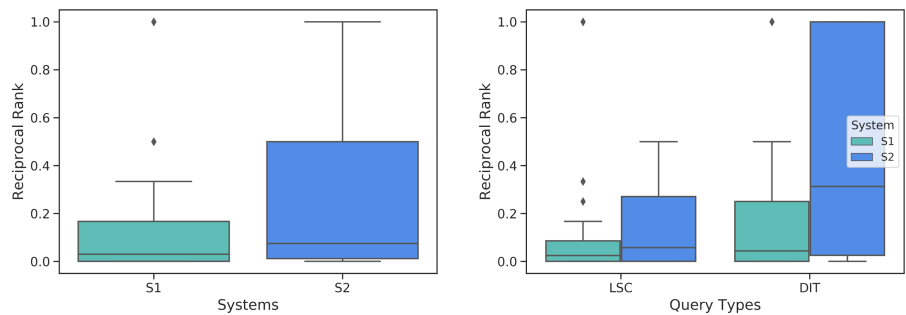


Fig. 5. The distribution of reciprocal rank of each query in overall (left) and on each query type (right) of two systems. S1 and S2 were the baseline and the proposed method accordingly.

8 Conclusion

In this paper, we employ a new perspective on the challenge of lifelog retrieval, where we transform into the graphs similarity matter. We applied graph indexing techniques in which lifelog images and queries are transformed into graphs, which were encoded into matrices in later stages, to capture visual relations between objects, hence improving the retrieved result's accuracy. We designed the experiments to evaluate our approach and compared it with an object-based baseline system with specific settings to reduce the bias of users' behaviors. The experimental results show that the graph-based retrieval approach outperformed the conventional method on both queries focusing on relationships and the ordinary topics used in the official lifelog search competition. Although there are some drawbacks, we believe that using relation graphs in the lifelog challenge is promising in this compelling field, especially when visual data of image contents are integrated into the graph structure. It poses an interesting avenue for future research on the topic of lifelog retrieval and related fields.

Acknowledgments. This publication has emanated from research supported in part by research grants from Science Foundation Ireland under grant numbers SFI/12/RC/2289, SFI/13/RC/2106, and 18/CRT/6223.

References

1. LSC 2019: Proceedings of the ACM Workshop on Lifelog Search Challenge. Association for Computing Machinery, New York (2019)
2. Aditya, S., Yang, Y., Baral, C., Fermuller, C., Aloimonos, Y.: From images to sentences through scene description graphs using commonsense reasoning and knowledge. arXiv preprint [arXiv:1511.03292](https://arxiv.org/abs/1511.03292) (2015)
3. Agarwal, A., Mangal, A., et al.: Visual relationship detection using scene graphs: a survey. arXiv preprint [arXiv:2005.08045](https://arxiv.org/abs/2005.08045) (2020)
4. Aizawa, K., Maruyama, Y., Li, H., Morikawa, C.: Food balance estimation by using personal dietary tendencies in a multimedia food log. *IEEE Trans. Multimed.* **15**(8), 2176–2185 (2013)
5. Aksoy, E.E., Abramov, A., Wörgötter, F., Dellen, B.: Categorizing object-action relations from semantic scene graphs. In: 2010 IEEE International Conference on Robotics and Automation, pp. 398–405. IEEE (2010)
6. Chu, T.T., Chang, C.C., Yen, A.Z., Huang, H.H., Chen, H.H.: Multimodal retrieval through relations between subjects and objects in lifelog images. In: Proceedings of the Third Annual Workshop on Lifelog Search Challenge, LSC 2020, pp. 51–55. Association for Computing Machinery, New York (2020)
7. Doherty, A.R., et al.: Experiences of aiding autobiographical memory using the SenseCam. *Hum.-Comput. Interact.* **27**(1–2), 151–174 (2012)
8. Duane, A., Gurrin, C., Huerst, W.: Virtual reality lifelog explorer: lifelog search challenge at ACM ICMR 2018. In: Proceedings of the 2018 ACM Workshop on the Lifelog Search Challenge, LSC 2018, pp. 20–23. Association for Computing Machinery, New York (2018)

9. Elliott, D., Keller, F.: Image description using visual dependency representations. In: *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pp. 1292–1302 (2013)
10. Gelonch, O., et al.: Acceptability of a lifelogging wearable camera in older adults with mild cognitive impairment: a mixed-method study. *BMC Geriatr.* **19**(1), 110 (2019)
11. Gemmell, J., Bell, G., Lueder, R.: MyLifeBits: fulfilling the Memex vision. In: *Proceedings of the Tenth ACM International Conference on Multimedia*, pp. 235–238 (2002)
12. Gurrin, C., Joho, H., Hopfgartner, F., Zhou, L., Albatal, R.: NTCIR lifelog: the first test collection for lifelog research. In: *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 705–708 (2016)
13. Gurrin, C., et al.: Comparing approaches to interactive lifelog search at the lifelog search challenge (LSC2018). *ITE Trans. Media Technol. Appl.* **7**(2), 46–59 (2019)
14. Gurrin, C., et al.: A test collection for interactive lifelog retrieval. In: Kompatsiaris, I., Huet, B., Mezaris, V., Gurrin, C., Cheng, W.-H., Vrochidis, S. (eds.) *MMM 2019*. LNCS, vol. 11295, pp. 312–324. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-05710-7_26
15. Gurrin, C., Smeaton, A.F., Doherty, A.R., et al.: Lifelogging: personal big data. *Found. Trends® Inf. Retr.* **8**(1), 1–125 (2014)
16. Heller, S., Amiri Parian, M., Gasser, R., Sauter, L., Schuldt, H.: Interactive lifelog retrieval with vitivr. In: *Proceedings of the Third Annual Workshop on Lifelog Search Challenge, LSC 2020*, pp. 1–6. Association for Computing Machinery, New York (2020)
17. Hughes, M., Newman, E., Smeaton, A.F., O'Connor, N.E.: A lifelogging approach to automated market research (2012)
18. Johnson, J., et al.: Image retrieval using scene graphs. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3668–3678 (2015)
19. Kovalčík, G., Škrhak, V., Souček, T., Lokoč, J.: Viret tool with advanced visual browsing and feedback. In: *Proceedings of the Third Annual Workshop on Lifelog Search Challenge, LSC 2020*, pp. 63–66. Association for Computing Machinery, New York (2020)
20. Le, T.K., et al.: Lifeseeker 2.0: interactive lifelog search engine at LSC 2020. In: *Proceedings of the Third Annual Workshop on Lifelog Search Challenge, LSC 2020*, pp. 57–62. Association for Computing Machinery, New York (2020)
21. Leibetseder, A., Schoeffmann, K.: Lifexplore at the lifelog search challenge 2020. In: *Proceedings of the Third Annual Workshop on Lifelog Search Challenge, LSC 2020*, pp. 37–42. Association for Computing Machinery, New York (2020)
22. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: *Advances in Neural Information Processing Systems*, pp. 3111–3119 (2013)
23. Ninh, V.T., et al.: Overview of ImageCLEF lifelog 2020: lifelog moment retrieval and sport performance lifelog. In: *CLEF2020 Working Notes*. CEUR Workshop Proceedings, CEUR-WS.org, Thessaloniki, 22–25 September 2020. <http://ceur-ws.org>
24. de Oliveira Barra, G., Cartas Ayala, A., Bolaños, M., Dimiccoli, M., Giró Nieto, X., Radeva, P.: LEMoRe: a lifelog engine for moments retrieval at the NTCIR-lifelog LSAT task. In: *Proceedings of the 12th NTCIR Conference on Evaluation of Information Access Technologies* (2016)

25. Rossetto, L., Baumgartner, M., Ashena, N., Ruosch, F., Pernischová, R., Bernstein, A.: Lifegraph: a knowledge graph for lifelogs. In: Proceedings of the Third Annual Workshop on Lifelog Search Challenge, LSC 2020, pp. 13–17. Association for Computing Machinery, New York (2020)
26. Schuster, S., Krishna, R., Chang, A., Fei-Fei, L., Manning, C.D.: Generating semantically precise scene graphs from textual descriptions for improved image retrieval. In: Workshop on Vision and Language (VL15). Association for Computational Linguistics, Lisbon (September 2015)
27. Signal, L., et al.: Kids’Cam: an objective methodology to study the world in which children live. *Am. J. Prev. Med.* **53**(3), e89–e95 (2017)
28. Tran, L.D., Nguyen, M.D., Binh, N.T., Lee, H., Gurrin, C.: Myscéal: an experimental interactive lifelog retrieval system for LSC’20. In: Proceedings of the Third Annual Workshop on Lifelog Search Challenge, pp. 23–28 (2020)
29. Tran, L.D., Nguyen, M.D., Nguyen, B.T., Gurrin, C.: An experiment in interactive retrieval for the lifelog moment retrieval task at imagecleflifelog2020. In: CLEF2020 Working Notes. CEUR Workshop Proceedings, CEUR-WS.org, Thessaloniki (2020). http://ceur-ws.org/Vol-2696/paper_103.pdf, <http://ceur-ws.org>
30. Tran, M.T., et al.: First - flexible interactive retrieval system for visual lifelog exploration at LSC 2020. In: Proceedings of the Third Annual Workshop on Lifelog Search Challenge, LSC 2020, pp. 67–72. Association for Computing Machinery, New York (2020)
31. Truong, T.D., Dinh-Duy, T., Nguyen, V.T., Tran, M.T.: Lifelogging retrieval based on semantic concepts fusion. In: Proceedings of the 2018 ACM Workshop on The Lifelog Search Challenge, LSC 2018, pp. 24–29. Association for Computing Machinery, New York (2018)
32. Wang, S., Wang, R., Yao, Z., Shan, S., Chen, X.: Cross-modal scene graph matching for relationship-aware image-text retrieval. In: The IEEE Winter Conference on Applications of Computer Vision, pp. 1508–1517 (2020)
33. Zellers, R., Yatskar, M., Thomson, S., Choi, Y.: Neural motifs: scene graph parsing with global context. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5831–5840 (2018)