MemoriEase at LSC: An Evolution from Lifelog Retrieval System to Conversational Q&A Assistant

Quang-Linh Tran and Xuan-Giang Phan

Abstract

Lifelogging, the practice of digitally recording daily life through wearable devices and online activity, has emerged as a powerful tool for memory augmentation, health monitoring, and lifestyle analysis. As lifelog data grows in volume and complexity, efficient and accurate retrieval systems become essential. This paper presents MemoriEase, a lifelog retrieval system designed to address key challenges in the field, including data redundancy, event ambiguity, and the need for reasoning over multimodal data. Participating in the Lifelog Search Challenge (LSC) from 2023 to 2025, MemoriEase evolved from a basic text-to-image retrieval system into a comprehensive platform featuring data cleaning, vector-based indexing using Elasticsearch, embedding-based search with BLIP2, and a conversational Retrieval-Augmented Generation (RAG) module for question answering. We describe the system's architecture, report its performance across three LSC editions, and reflect on the insights and limitations gained through real-world deployment. Our work highlights the growing capabilities and remaining challenges in building effective lifelog retrieval systems to support human memory and wellbeing.

Keywords: Lifelog Retrieval, Personal Archive, Chatbot Assistant, Multimedia

1 Introduction

Lifelogging refers to the process of logging our daily lives through digital devices [1]. It can be done by using wearable cameras to automatically capture or record point-of-view images/videos to log what we have seen and interacted with from our point of view. It can also be done by smartwatches to record our vital biometrics like heart rate, stress level to log our health condition. And last but not least, with the increasing use of digital devices and the internet, our footprint in these services also contributes an important picture of what we engage in online. From all the data collected, we

have a collection of lifelog data. The concept of lifelogging is not new, it was introduced in 1945 by Vannevar Bush in an article called "As we May Think" [2], in which he proposed an idea of the future of wearable mini-camera capture everything the wearer sees and an system called "Memex" saving all data and serving like a second brain. There are several clear opportunities for lifelogging in usage. However, due to the limitations of technology, the popularity of lifelogging has only increased in recent decades. Thanks to the feasibility of large storage and the development of personal digital devices and wearable gadgets like SenseCam [3] and Narrative Clip[1], the activity of lifelogging has become more popular. A milestone project is "MyLifeBits" [4, 5] in which Gordon Bell stores his articles, books, financial and legal records, memorabilia, photos, telephone calls, time-lapse photos, video, and web pages. From that data, it can be used to serve many useful applications. Lifelog data, which is extracted from smartwatches and PoV cameras, can be used to monitor the health of the lifelogger [6]. The biometrics from the smartwatches indicate the vital metrics related to the lifelogger's health, while the PoV images provide information on the exercise, food consumption, and water intake. Several researchers have further explored this topic with a wide range of sub-applications [7, 8]. Another important application of lifelogging is memory enhancement, in which the PoV images play a role in storing memorable events through automatically captured images [9]. This application can help lifeloggers reminisce about memories as well as things they may forget, like "Where is my key? I forgot to leave it somewhere.". Lifelog retrieval is the task for the purpose of this application.

Lifelog retrieval is a task of retrieving images from a lifelog data collection for a specific query [10]. This task mimics the need of users when they forget something and want to find it from the lifelog data. The example query above about the key shows the use case of this task in real-world applications. This task has a huge application in memory support and health & wellness monitoring [6, 10]. This task is a sub-task of text-to-image retrieval, but the retrieved images are PoV images from the lifelog data collection. Lifelog retrieval has been an active research area in recent years, with several benchmark challenges [11, 12] for several retrieval systems to showcase their performance. Lifelog Search Challenge (LSC) [11] is an active challenge for interactive lifelog search organized as a workshop in the ICMR conference. It has been run for 8 years with a significant development in both lifelog data volumes and the diversity of sub-tasks. In the latest LSC'25[13], a lifelog dataset of 725K PoV images, biometrics, and environmental data spanning 18 months is used to evaluate the performance of 9 lifelog retrieval systems. There are three sub-tasks in the challenge, including know-item search (KIS), Ad-hoc search (AD), and Question-Answering (Q&A). The difference between the three sub-tasks is the query used for retrieval. While the query for KIS requires finding the correct images for a single unique event in the lifelog data, the query for AD asks to retrieve as many correct images as possible for recurring events, such as eating at a restaurant or doing exercises. The Q&A task requires a textual answer for a question in lifelog data, such as "How many times did I fly with Emirates in 2020?". This question not only retrieves events of flying with Emirates but also counts the number of times and generates a textual answer for that. Each

[1] https://getnarrative.com/

of the sub-task provide useful information for lifeloggers in enhancing their memory, analysing their lifestyle, and providing insights into their life. In the live competition, the challenge has two sessions, one for expert users, who build their own system, and one for novice users, who are new to the system. This aims to foster user-friendly and enhance the user interface/user experience for the system. Each session has around 4 to 8 queries per sub-task. The scoring mechanism penalized the wrong submission and time, with more details described in the overview paper [14].

Along with the huge potential applications of lifelog retrieval and Q&A, there are also several significant challenges to solve this task accurately and efficiently. The first challenge of lifelog retrieval is the volume and growing velocity of lifelog data. With an image every 30 seconds in 16 active hours a day, the number of total images can be 2000. Adding additional data on biometrics, metadata of images, and expanding it to several years or decades can make the size of lifelog data collection enormous. The huge size of the data poses a challenge for storing and indexing efficiently for retrieval. The second challenge is the diversity and repetition of events in a lifelog data collection. The repeated activities, such as eating or watching TV, often appear visually similar, causing ambiguity, while diverse events require retrieval models to generalize across varied scenes and contexts. This challenge highlights the need for temporal information (time and order of events) to retrieve correct events. Last but not least, the newly emerging Q&A sub-task in lifelogdata requires not only retrieving correct data but also the capability to reason over the data to generate the answer.

This paper introduces a lifelog retrieval system called MemoriEase with several advancements to address the stated challenges. This system has participated in 3 LSC challenges, from 2023 [15] to 2025, with several improvements every year to address the challenge of lifelog retrieval. We proposed an efficient data cleaning process to discard redundant images and used a vector database, Elasticsearch[2], for indexing and retrieving data. The use of the BLIP2 embedding model [16] to retrieve images by an embedding-based approach makes the system robust for finding diverse events in lifelog data. Combining several filters in time and location helps to reduce the ambiguity of searching lifelog images. We also use a conversational search with an integrated Retrieval-Augmented Generation (RAG) [17] module to solve the Q&A sub-task in LSC. This paper describes the details of the system, from the first generation with a basic text-to-image retrieval system, to the latest version with conversational search and RAG. In addition, the performance of the system in three challenges is also presented, as well as the lessons learned from each challenge. We also provide a discussion on the system's strengths and drawbacks to improve the system for future development.

The structure of this paper is as follows: Section 2 presents the details on several parts of the MemoriEase system, from data processing and indexing to search, RAG, and the user interface. Section 3 provides information on the performance of MemoriEase in three LSC challenges. The discussion on the system is presented in section 4 before the conclusion in section 5.

[2]https://www.elastic.co/

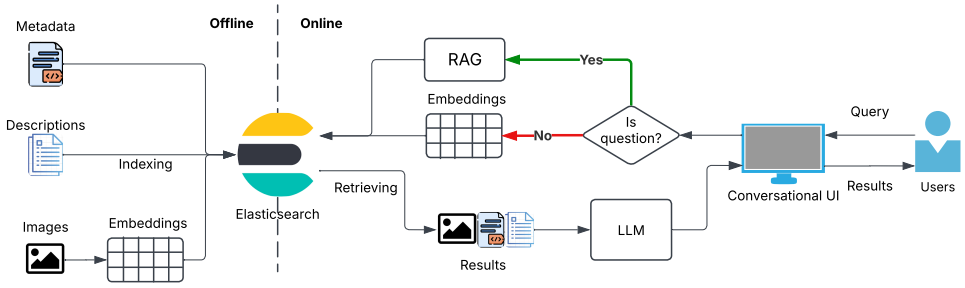


Fig. 1: MemoriEase overall architecture and flow

2 MemoriEase

In this section, we present the MemoriEase lifelog retrieval system, from the data processing and indexing step to the retrieval modules, Q&A modules, and user interface. Figure 1 provides an overall architecture of the system, from the flow of data processing in offline to retrieval by users in online. In the offline phase, the metadata, including time and location, descriptions/captions of images, and embeddings extracted from images, are indexed into Elasticsearch. In the online phase, when a user starts a chat with the conversational user interface, the system classifies whether it is a question or a query. If it is a query, it will be encoded into the embedding by the same embedding model that encodes images. If it is a question, it will go through an RAG module to be retrieved by description. The retrieved data then passes through an LLM to generate a summary or an answer to the question before returning to the user interface for users. More details about every step of the processing are described in the following subsections.

2.1 Data Processing and Indexing

2.1.1 LSC dataset

The lifelog dataset used for LSC from 2022 to 2025 is an 18-month lifelog dataset including 725K PoV images, metadata for images with time and locations, and visual concepts describing the visual content of images. The PoV images are captured from a Narrative Clip[3] wearable camera with a 1024x768 pixel resolution. All images are fully redacted and anonymised by removing the face and readable text, as well as certain scenes and activities manually filtered out to respect local privacy requirements. Some examples of the images can be seen in Figure 2.

2.1.2 Image processing

From the original lifelog images, we use the OpenCV package [4] in Python to calculate the edge weight of objects in the images. The edge weight is the numerical value

[3] https://getnarrative.com/
[4] https://opencv.org/

Fig. 2: Some examples of images from the lifelog dataset

assigned to an edge of objects in the image. The low edge weight means the image has no clear objects or is blurry. We chose the threshold of quantile 0.13 with the weight value of 350,000 to remove 94,450 images. In the first version of MemoriEase, we implemented an event segmentation algorithm to group images into an event, a sequence of images illustrating an activity like eating or watching TV. We use BLIP embedding of images to calculate the cosine similarity of images with the next or previous images in the timeline. If the similarity is higher than a predefined α (0.7 on the scale of 1), we group them in a single event. The result of this is 173,269 events. We then choose an image in the middle of the sequence of images of events to represent the event. However, in the competition, we observed that the representative image used to retrieve can lead to failed retrieval, due to the slight difference between the desired and representative image. In the latest version, we discard the event segmentation to search on images to avoid that problem. The images are then encoded into a 256-dimensional vector embedding by the BLIP2 model. This model is a state-of-the-art visual-language model with high performance on zero-shot text-to-image retrieval. We also perform a comparison of BLIP2 and CLIP models in retrieval.

2.1.3 Location data processing

From the latitude and longitude of the metadata for images, we obtain the information about the city, country. We also use the semantic name provided in the dataset, which is the name of the location the lifelogger visited, such as home, Dublin City University, Dublin Airport, etc. Three attributes, city, country, and semantic name, are used as filters for the retrieval module.

2.1.4 Time processing

From the UTC time provided in the dataset, we convert it to local time using the time zone of the city. It matches better on the query of LSC because most queries use the local timezone. We also extract several other time-related attributes for filtering, such as hour (in 24-hour format), date of week, is weekend, and time period (morning, afternoon, etc). These filters are significantly helpful for time-constrained queries like shopping on the afternoon of a weekend. All the extracted data is then formed into a table of 11 attributes to index into an Elasticsearch index. Table 1 provides the details of attributes in the Elasticsearch index for indexing.

Attribute	Description	Data type	Example
ImageID	Unique identifier for each lifelog image	text	20190101_103717_000
OCR	Text detected in the image	text	Cocacola
semantic name	Name of the current place	text	Home, Dublin airport
city	Name of the city and country	text	Dublin, Ireland
local time	Date and time at the location	datetime	2019-01-01 10:37:17
hour	Hour extracted from local time	integer	10
date of week	Day name extracted from local time	text	Tuesday
is weekend	1 if the day is Saturday or Sunday, else 0	binary	0
time period	Part of the day (e.g., Morning, Evening)	text	Morning
description	Text description of what the user sees	text	A room with curtains
BLIP2 vector	Vector representing the image from BLIP2	array	[0.2, 0.4, ..., 0,4]

Table 1: Properties of Elasticsearch index

2.2 Retrieval Modules

2.2.1 Query and filter processing

The retrieval modules play the role of retrieving lifelog images from both text and image inputs. We implement two retrieval modules in MemoriEase, including text-to-image retrieval and image-to-image retrieval. For text-to-image retrieval, when users input a textual query, the query is processed to extract relevant filters in the query. We use a heuristic approach to extract time-related filters such as date, hour, and time period. For location-related filters, we compare the semantic name and city dictionary in the Elasticsearch index, and if they match with the words in the query, we use them as filters. In addition, we also implement advanced filters for expert users, where they can use the @ symbol before some keywords such as weekend, start_hour, end_hour, ocr, and location to indicate the value for filters. These filters are applied before the cosine similarity calculation in Elasticsearch to reduce the number of candidate images.

2.2.2 Text-to-image retrieval

We removed all the filter-related words in the original query to get the processed query and encoded it into a vector embedding by the BLIP2 embedding model. Elasticsearch uses the K-Nearest Neighbor (KNN) algorithm for calculating the cosine similarity of K-candidate image embeddings and the query embedding. We use K around 1000 to balance the time of calculating similarity and the accuracy of retrieval. The result of this is a ranked list of 100 images that are relevant to the query. The text-to-image search is useful for the KIS sub-task in LSC, where the query requires several filters and a complicated semantic.

2.2.3 Temporal search

With the temporal nature of lifelog data, when an event happens before or after another event, such as “I am watching TV after having dinner.”, we construct a temporal search function to support this case. We provide three text-free search boxes for users to input the main event, the previous event, and the after event, and a time

gap constraint to indicate the maximum time difference between the main event and temporal events. The previous and after events are optional, in which users can input both or only one of them. The system will perform a text-to-image search for the main query first, and then use the datetime of each retrieved image and the time gap constraint as a filter to perform a search on temporal events. The final ranked list uses the weighted average score of the cosine similarity of the main event and the temporal events to rank the top 100 relevant images.

2.2.4 Conversational Search

Conversational search is proposed in the second version of MemoriEase for LSC'24 to support natural and engaging search for lifelog. In addition, with the multi-turn search in conversation, it mimics the process of remembering in memory, with new things popping up after every search and review of the images. Thanks to the development of large language models (LLM), they can aggregate information from multiple turns of conversation and produce human-like responses for users' queries or questions.

We use the GPT-4o-mini model as the conversational agent to aggregate the information from multi-turn conversations, and generate a summary for the query's results. It is prompted to generate the response with constructive information and suggest further direction of search for users.

2.2.5 Image-to-image retrieval

For simpler queries in the Adhoc subtask, we propose an image-to-image retrieval. This type of retrieval also starts with a query, and the same process as text-to-image is applied. However, after the returned results display on the interface for users, they can choose images from the results as the input for the next iteration of the search. They can iteratively search until no more matching images are found. This search is implemented thanks to the semantic matching of the embedding of images. We extract the image embeddings of the visual input and do the cosine similarity search in the Elasticsearch index before returning the ranked list of new results. This search proves effectiveness in finding specific objects accounting for a small fraction in the image, such as "stop before a red traffic light." or "shopping on an empty shelf.". The text-to-image retrieval only finds several relevant images "red traffic light" but when we use these images to search again, more similar images are returned. Combining two retrieval modules, we have a strong retrieval function for KIS and Ad-hoc sub-tasks and a foundation for the QA task, in which we propose a retrieval before generating answers.

2.3 RAG for QA Module

Retrieval-Augmented Generation (RAG) is proposed to inject external knowledge into Large Language Models (LLM) to avoid hallucination and provide information for generating answers. The use of LLM for lifelog questions has the potential to exploit the capability of LLM to generate answers from the provided lifelog description. We propose a RAG pipeline to retrieve information and generate answers to questions, which is depicted in Figure 3.

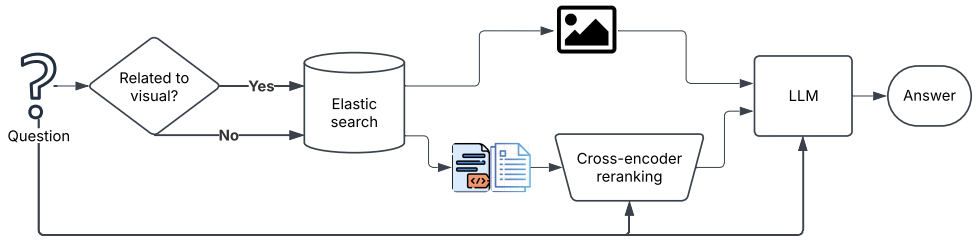


Fig. 3: RAG pipeline for lifelog questions

2.3.1 RAG for visual-related questions

Depending on the required information for the questions, such as questions related to the visual content of lifelog images (place leaving the key) or metadata of lifelog (city of traveling), we have different strategies to solve the questions. We use a rule-based approach based on question words to classify the type of question. And if it is a visual-related question, we use the question to encode an embedding and search for top K relevant images (K from 10 to 20). All the images and the question are prompts for an LLM model to generate an answer. We chose GPT-4o-mini as the LLM thanks to its capability to read images and reasonable cost.

2.3.2 RAG for metadata-related questions

For the metadata-related questions, it is different in the retrieval stage as we combine by weighted average the cosine similarity from the query-image embedding similarity and the query-description embedding similarity. The output is a ranked list of descriptions of images and corresponding metadata. We further rerank the list by using a cross-encoder reranking model pre-trained on the MS Marco Passage Ranking dataset to improve the rank of relevant descriptions. The reranked list is passed to LLM with the question to ask for an answer. The answer is shown on the user interface with corresponding images for further checking.

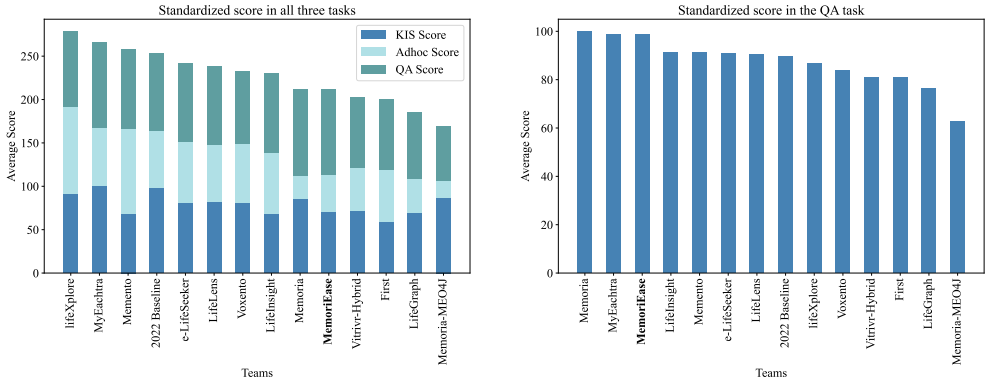
2.4 User Interface

2.4.1 Main page

We design a simpler but effective user interface to support both novice and expert users in LSC. Figure A1 illustrates the landing page when you enter the MemoriEase system. We have a security layer for users to log in to ensure that only authorized users can use the system to protect the lifelog data. Users can type a query or question directly into the search bar to start a conversational search session. Users can also choose to browse the lifelog data by time or by visual similarity of random images.

2.4.2 Conversational search page

When you input a query into the search bar, the result page will be displayed to show the user's chat on the left side and the result with a grid of images on the right side,



(a) MemoriEase performance in LSC23.

(b) MemoriEase performance in LSC23 QA sub-task.

Fig. 4: MemoriEase performance in LSC23 overall and QA sub-task.

as shown in Figure A2. The system will provide an answer to users' questions or a summary of results for queries. On the result side, accompanying each image is the semantic name indicating where the image was captured and the date and time in a human-readable format. The underline in the datetime implies that it can be clickable to show lifelog images and a highlighted clicked image, as depicted in Figure A3. In the image, three icons are shown to allow users to save the image (icon on the left), open similar images in the same event (icon in the middle), and submit the image for LSC. Users can also sort the retrieved results by time or semantic name, and choose to different interface for specific tasks.

2.4.3 Temporal search page

For advanced search with filters and temporal search for the KIS sub-task, users can input a query for the main event, the previous and after event, with a time gap constraint to search. The result will display on a triplet of 3 images, with the large image in the middle showing the main query, and the left and right images showing the previous and next event. An example of this interface is depicted in Figure A4.

2.4.4 Image-to-image search page

When users click on the Adhoc task, an image-to-image retrieval page will pop up with several random images for users to choose from. If you want to search for a specific query, the results will be displayed on the right side with a familiar grid format. Users can click on several images to choose as input and click search again. The new result will be shown with more similar visual content to the input images. An example of this is shown in Figure A5.

3 Performance at LSC

In this section, we provide information about the performance of the MemoriEase system in three LSC challenges. Analysis of the strengths and drawbacks is also discussed to draw the lessons learned from each challenge. The reported results are from the expert user only for a fair comparison, as the LSC'25 is only for expert users due to the restriction of traveling to ICMR 2025 in the United States of America.

3.1 MemoriEase at LSC'23

LSC'23 was the first time the MemoriEase system participated. There were 13 teams in LSC'23, except for the baseline from LSC'22, and the MemoriEase system ranked in 8th position, sharing with the Memoria team. MemoriEase found 7/10 correct images for KIS queries and provided 8/10 correct textual answers for the QA sub-task. However, it only scored 33/100 on the Ad-hoc sub-task.

Figure 4a illustrates the performance of systems in LSC'23. MemoriEase performed average on the KIS sub-task compared to other systems, but it poorly found many images in the Adhoc task. We found that the BLIP embedding model tends to find images with general information rather than specific characteristics of objects like blue balls. For example, a query like "stopping in front of a red traffic light" was found a lot of images before the green traffic light but not the red light. This was a big drawback that we aimed to improve in the next version of MemoriEase.

On the QA sub-task, the MemoriEase system performed very well compared to other systems, ranking in third position, as shown in Figure 4b. The system usually achieved the highest score for answering the question, except for failing to provide the answer for 2 questions. The question "I normally wear shirts, but what is the brand of the grey t-shirt that I wore at the start of COVID-19 time?" can be solved by finding the grey t-shirt from early 2020. However, the image was not found because it was segmented into an event. The representative image of the event is different from the t-shirt image, so we failed to find it. It is the reason why we remove the event segmentation in the following version of MemoriEase. This was the first time MemoriEase took part in the LSC challenge, so there were still a lot of things that were new and needed to be adapted to the system for that.

3.2 MemoriEase at LSC'24

LSC'24 attracted 21 systems and 35 participants in total (1 system can have multiple users), which led to a great competition between participants. Overall, the MemoriEase system only achieved the 21st position out of 35 participants and ranked 13th out of 21 systems. Our system worked best on the KIS task with a 65/100 score, followed by the QA task with a 55/100 score, and achieved the lowest score on the Adhoc task with only 53. This performance is on average compared to other systems. Figure 5 depicts the score of teams in LSC'24.

Although the Adhoc score was the lowest, it was an improvement from previous results in LSC'23. This is thanks to the image-to-image retrieval described in section 2.2.5. We used the textual query to find one correct image, and we continued to use that image to search for more. This approach proved effective for queries with less

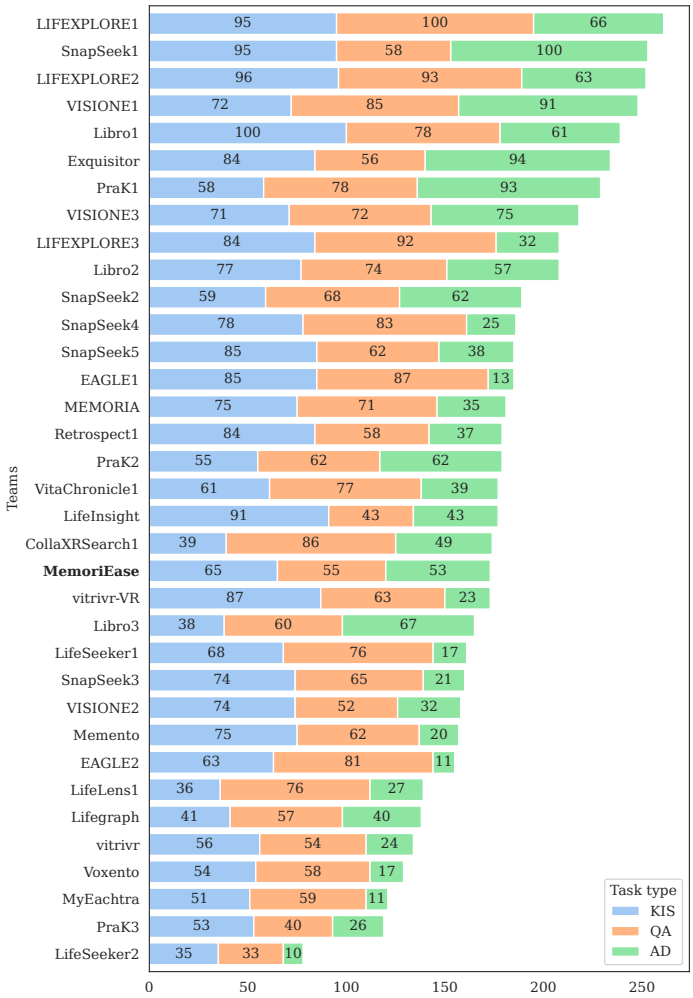


Fig. 5: MemoriEase performance in LSC24.

than 10 correct answers, like " Find examples of when I was waiting for a security guard to let me into my office building" and "Find examples of when I was shopping and the shelves were nearly empty of products". For other queries with more than 10 correct answers, the time and number of submitted answers played a big role in achieving a high score, and we are not as fast as other competitors. There is one query (I can not find my electronic violin. I left it with my cream coloured electric guitar (Telecaster style). Can you find images of them together?) that the system cannot find the correct answer because it mainly finds the images with the guitar instead of the violin, as the number of guitar images is outnumbered the only 2 violin images.

On the two other sub-tasks, the MemoriEase system found the correct answers for 4 out of 5 in KIS queries and 3 out of 5 QA queries. Although the system found the

answer in the top 100 retrieved images but the user still cannot submit the correct answer for queries like "It was as if the meeting took place on the Starship Enterprise. I remember the wall had yellow lighted shapes and designs". The event segmentation also caused a failure on the query "I remember she was wearing a hat and taking a photo of a lake.". In this case, the correct image was grouped into an event whose representative image differed from the target image, so we only identified it at the last moment, ultimately failing to submit it in time. Some QA queries were too challenging to find the answer, which required both reasoning and quick action, like "A,B,C,D,E,F, Y or Z? I can't recall which zone of the car park in the airport I parked in when I was going to France in 2019". This question needs to find the car park in the airport in the home country instead of in France, so most of the team fails to find the answer.

Although we did not achieve the high score in this LSC'24, we drew several lessons from the results of LSC'24. Firstly, the image-to-image retrieval worked well for the Adhoc sub-task, but the filter and information from the queries were also important, so we needed to enhance this type of search by integrating the automatic filter extraction and the incorporation of textual information in the image search. Secondly, the event segmentation can be efficient in grouping the images to reduce the search volume, but it can also reduce the accuracy if the event is wrongly segmented. We plan to discard this function in the next version of MemoriEase to enhance the accuracy of retrieval while the volume of 725K lifelog images is still acceptable for hardware. Finally, we implemented the preliminary version of RAG in MemoriEase for LSC'24, and it can solve some simple QA queries. It is promising to improve the RAG for next year's challenge.

3.3 MemoriEase at LSC'25

With 21 participants from 11 teams in LSC'25, it is a decrease compared to LSC'24. In addition, LSC'25 is also a hybrid competition, where most of the participants join online due to the travel restrictions to the USA for the ICMR 2025 conference. The MemoriEase system participated remotely in the expert session only. Remarkably, we achieved the third position in the leaderboard, following the Memoria and SnapSeek teams. We also achieved the second-highest position in the QA sub-task with a 96 score thanks to the RAG component. The score of KIS queries also increased by 10 points, but the performance on Adhoc sub-task remained the same compared to last year's competition. Figure 6 illustrates the leaderboard of LSC'25.

Specifically, we resolved 4 out of 6 QA queries, which is equal to top teams like MEMORIA, SnapSeek, and VitaChronicle, but thanks to the quick time of finding the answer, we ranked second place in the QA task, following the VitaChronicle team. There was a query that no other team found the correct answer except our system. This is "What is the name of the salesperson who sells me a Japanese car before June 2020?". We found the event of visiting a car showroom in May 2020, and submitted the name of the showroom called "Joe Duffy". However, the answer was wrong, and the feedback was the name of the salesperson instead of the name of the showroom. We used the browsing by time feature in our system to scroll to the image of talking to a salesperson and found his name tag in the table. We submitted the result at the last second and achieved the highest score for that query. This is an interesting insight

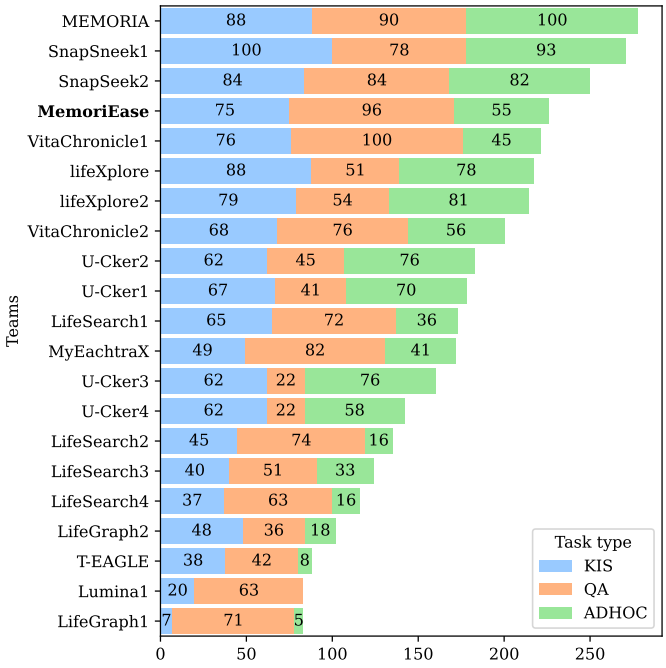


Fig. 6: MemoriEase performance in LSC25.

that to find the correct answer, one needs an excellent combination of the good use of system functions and logical, reasoning thinking on the way to find the answer. Some answers for queries are not explicitly found on the first time of search but through refinement, the answer can be found.

On the KIS task, we also found 4 out of 6 correct answers, but it is far below the 6 out of 6 correct answers from top teams like SnapSeek or lifeXplore. We found out that if we can not find the correct answers in the first 3 hints, it is difficult to find the correct answers even when the time and location of queries are revealed in the last hint. The two wrong queries are "I was in the kitchen of a restaurant" and "I was shopping for a rat trap with my brother". The system failed to find the answer because it only found a place like a kitchen, but in a shop with a lot of products, the correct answer is nearly an empty kitchen with a lot of pipes. The temporal information did not help to find the answer. In the second query, the system found the mousetrap with the label "Mouse trap" instead of the rat trap. The correct answer showed the small rat trap, which may lead to the embedding model failing to encode it.

The Adhoc task in LSC'25 this year is challenging with 6 queries, but 5 of them have fewer than 10 correct images. Only the query "Drinking beer other than Guinness with other people" has more than 10 correct images. Due to this scenario, the relevance feedback for visual similarity search in our system did not work well to find as many visually similar images. We relied mostly on text-to-image search to find the answers and failed to find the answer for the query "Menu in a picture frame in a restaurant".

The object of the menu in a picture frame is unique and strange to the embedding model, so it only found the image of a normal menu or a big menu hanging on the wall. If we could find one single correct image of the picture frame menu, we could find more similar images, but it failed in the first steps, so we cannot submit any correct images for this query. We achieved an average score of 55 in this task due to the slow submission time and wrong submissions.

Overall, we performed quite well in the LSC'25 with the third position. Thanks to the RAG for QA, we can find the answer quickly, and a little luck in the car salesperson's name helps us to achieve the second position in the QA task. Queries in LSC are becoming more and more difficult, which require not only good systems but also quick action and logical thinking. Some components in our system can be improved to deal with the weakness during competition and will be introduced in later versions.

4 Discussion

MemoriEase has undergone substantial evolution across its three-year participation in the Lifelog Search Challenge (LSC), transforming from a straightforward retrieval system into a sophisticated platform integrating multimodal retrieval, conversational search, and Retrieval-Augmented Generation (RAG). This section discusses the contributions, strengths, limitations, and insights drawn from deploying MemoriEase in real-world competitive settings.

4.1 System Components and Contributions

MemoriEase comprises several integrated modules that collectively address the multifaceted challenges of lifelog retrieval and question answering:

- **Data Cleaning and Indexing:** The use of image filtering based on edge weights and the elimination of low-quality images has improved retrieval efficiency without sacrificing coverage. The decision to discard event segmentation in later versions helped mitigate retrieval errors linked to mismatched representative images.
- **Vector-Based Retrieval:** Leveraging BLIP2 embeddings for text-to-image and image-to-image retrieval has enhanced semantic matching between user queries and images. The addition of temporal search functions enables the system to capture contextual sequences, which is critical for lifelog scenarios.
- **Conversational Interface:** Incorporating GPT-4o-mini for multi-turn dialogue supports natural interactions and iterative query refinement, mimicking human memory retrieval processes. This has proven valuable for engaging both novice and expert users.
- **RAG for QA:** The RAG pipeline significantly improved performance in the QA sub-task, allowing the system to combine visual data with metadata and generate natural language answers. This integration helps bridge the gap between retrieval and reasoning.

4.2 Strengths

Several strengths distinguish MemoriEase:

- **Flexibility Across Tasks:** The system handles diverse LSC sub-tasks, from Know-Item Search (KIS) and Ad-hoc retrieval to complex QA, adapting to different query types and user needs.
- **Improved QA Performance:** The RAG module demonstrated excellent capability in solving knowledge-intensive queries, contributing to MemoriEase's high rankings in the QA sub-task, particularly in LSC'25, where it ranked second.
- **User Interaction Design:** A well-designed user interface facilitates both simple and advanced queries, supporting exploratory search and fine-grained filtering, which is crucial for lifelog analysis.
- **Iterative Improvements:** The system shows a clear trajectory of learning from prior competitions, adjusting components like event segmentation and retrieval strategies to address observed shortcomings.

4.3 Weaknesses and Challenges

Despite its progress, MemoriEase faces persistent challenges:

- **Ad-hoc Retrieval Limitations:** The system consistently underperformed in the Ad-hoc sub-task, largely due to difficulties in retrieving highly specific visual concepts or rare objects, especially when no initial matching image can be found for iterative refinement.
- **Visual Ambiguity:** BLIP2, while effective for general semantic matching, struggles with distinguishing fine-grained visual differences (e.g., distinguishing a red traffic light from a green one), leading to retrieval errors in detailed queries.
- **Event Segmentation Trade-offs:** Early reliance on event segmentation introduced errors when representative images differed from target images, prompting its removal. However, removing segmentation increases the search space, potentially impacting retrieval speed as data volumes grow.
- **Processing Speed and Competition Pressure:** Although the system's technical capabilities have improved, response time remains critical in live competitions. Delays in refining searches or generating answers can result in lower scores even when the correct result is ultimately found.

4.4 Performance Reflection

Over three LSC competitions, MemoriEase has demonstrated significant advancements. From an eighth-place finish in LSC'23, with notable gaps in Ad-hoc retrieval and event-based errors, the system evolved to achieve third place overall in LSC'25. Particularly in QA, MemoriEase has become highly competitive, successfully addressing complex queries that combine visual and textual reasoning. However, bridging the performance gap in the Ad-hoc sub-task remains a priority for future development.

Collectively, the experience across the LSC challenges underlines the increasing complexity of lifelog search tasks, which demand not only technical robustness in retrieval and reasoning but also agile user interactions and fast response times. Future

improvements will focus on enhancing fine-grained visual discrimination, integrating smarter filtering in image-to-image retrieval, and further optimizing the RAG module for speed and precision.

5 Conclusion

In this paper, we introduce the MemoriEase system at the LSC challenge. This system has developed significantly from LSC'23 to LSC'25, with a transformation from a basic lifelog retrieval system to a conversational search with RAG enhanced for the QA sub-task. We described the details of the system from data processing, retrieval, and Q&A module to the user interface. The performance of MemoriEase at the three LSC challenges is also discussed to draw on the experience and drawbacks of the system. In the future, we aim to further improve the user interface to enhance the user experience.

References

[1] Gurrin, C., Smeaton, A.F., Doherty, A.R., *et al.*: Lifelogging: Personal big data. Foundations and Trends® in information retrieval **8**(1), 1–125 (2014)

[2] Bush, V., *et al.*: As we may think. The atlantic monthly **176**(1), 101–108 (1945)

[3] Hodges, S., Williams, L., Berry, E., Izadi, S., Srinivasan, J., Butler, A., Smyth, G., Kapur, N., Wood, K.: Sensecam: A retrospective memory aid. In: UbiComp 2006: Ubiquitous Computing: 8th International Conference, UbiComp 2006 Orange County, CA, USA, September 17-21, 2006 Proceedings 8, pp. 177–193 (2006). Springer

[4] Gemmell, J., Bell, G., Lueder, R., Drucker, S., Wong, C.: Mylifebits: fulfilling the memex vision. In: Proceedings of the Tenth ACM International Conference on Multimedia, pp. 235–238 (2002)

[5] Gemmell, J., Bell, G., Lueder, R.: Mylifebits: a personal database for everything. Communications of the ACM **49**(1), 88–95 (2006)

[6] Kim, S., Yeom, S., Kwon, O.-J., Shin, D., Shin, D.: Ubiquitous healthcare system for analysis of chronic patients' biological and lifelog data. IEEE Access **6**, 8909–8915 (2018) https://doi.org/10.1109/ACCESS.2018.2805304

[7] Choi, J., Choi, C., Ko, H., Kim, P.: Intelligent healthcare service using health lifelog analysis. Journal of medical systems **40**, 1–10 (2016)

[8] Kumar, G., Jerbi, H., Gurrin, C., O'Mahony, M.P.: Towards activity recommendation from lifelogs. In: Proceedings of the 16th International Conference on Information Integration and Web-based Applications & Services, pp. 87–96 (2014)

[9] Kikhia, B., Hallberg, J., Bengtsson, J.E., Savenstedt, S., Synnes, K.: Building digital life stories for memory support. International journal of Computers in Healthcare **1**(2), 161–176 (2010)

[10] Ribeiro, R., Trifan, A., Neves, A.J.: Lifelog retrieval from daily digital data: narrative review. JMIR mHealth and uHealth **10**(5), 30517 (2022)

[11] Gurrin, C., Zhou, L., Healy, G., Bailer, W., Dang Nguyen, D.-T., Hodges, S., Jónsson, B.Þ., Lokoč, J., Rossetto, L., Tran, M.-T., *et al.*: Introduction to the seventh annual lifelog search challenge, lsc'24. In: Proceedings of the 2024 International Conference on Multimedia Retrieval, pp. 1334–1335 (2024)

[12] Zhou, L., Gurrin, C., Dang-Nguyen, D.-T., Healy, G., Lyu, C., Ji, T., Wang, L., Hideo, J., Tran, L.-D., Alam, N.: Overview of the ntcir-17 lifelog-5 task. In: Proceedings of the 17th NTCIR Conference on Evaluation of Information Access Technologies. Https://doi. Org/10.20736/0002001329 (2023)

[13] Gurrin, C., Zhou, L., Healy, G., Tran, A., Rossetto, L., Bailer, W., Dang-Nguyen, D.-T., Hodges, S., Þór Jónsson, B., Tran, M.-T., Schöffmann, K.: Introduction to the 8th annual lifelog search challenge, lsc'25. In: Proceedings of the 2025 International Conference on Multimedia Retrieval. ICMR '25, pp. 2143–2144. Association for Computing Machinery, New York, NY, USA (2025). https://doi.org/10.1145/3731715.3734579 . https://doi.org/10.1145/3731715.3734579

[14] Tran, A., Bailer, W., Dang-Nguyen, D.-T., Healy, G., Hodges, S., Jónsson, B., Rossetto, L., Schoeffmann, K., Tran, M.-T., Vadicamo, L., Gurrin, C.: The State-of-the-Art in Lifelog Retrieval: A Review of Progress at the ACM Lifelog Search Challenge Workshop 2022-24 (2025). https://arxiv.org/abs/2506.06743

[15] Gurrin, C., Jónsson, B.Þ., Nguyen, D.T.D., Healy, G., Lokoc, J., Zhou, L., Rossetto, L., Tran, M.-T., Hürst, W., Bailer, W., *et al.*: Introduction to the sixth annual lifelog search challenge, lsc'23. In: Proceedings of the 2023 ACM International Conference on Multimedia Retrieval, pp. 678–679 (2023)

[16] Li, J., Li, D., Savarese, S., Hoi, S.: Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In: International Conference on Machine Learning, pp. 19730–19742 (2023). PMLR

[17] Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Rocktäschel, T., *et al.*: Retrieval-augmented generation for knowledge-intensive nlp tasks. Advances in neural information processing systems **33**, 9459–9474 (2020)

Appendix A User interface examples

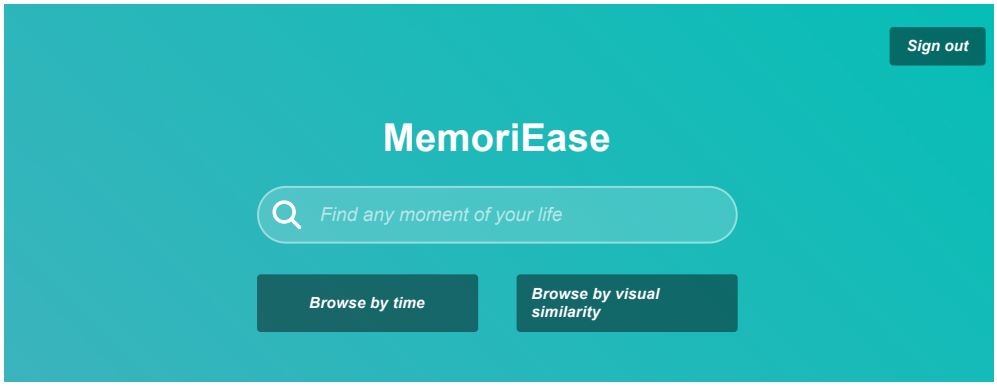


Fig. A1: MemoriEase landing page

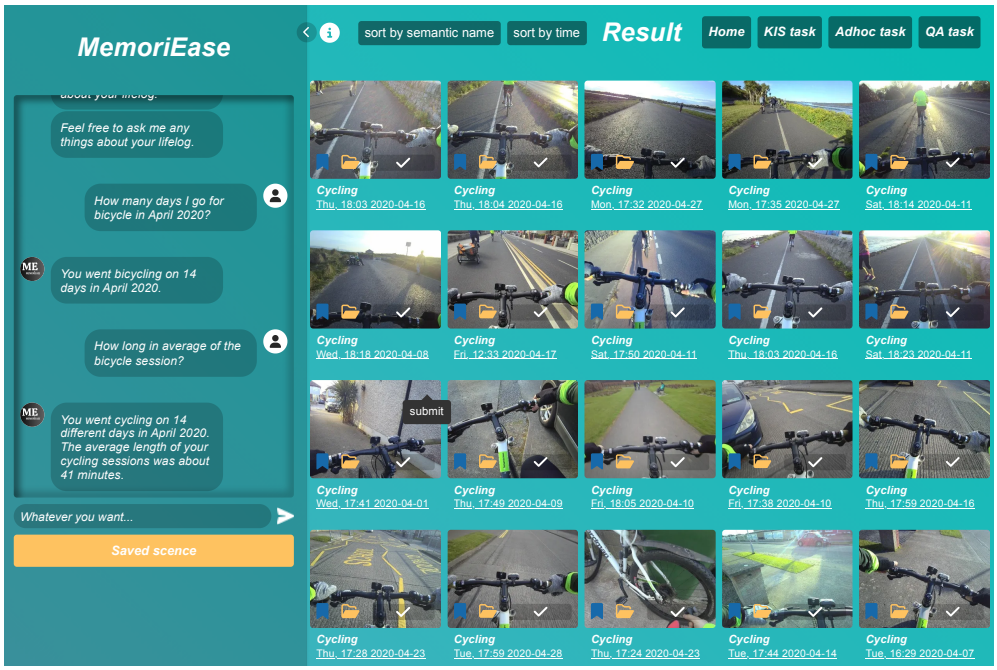


Fig. A2: Conversational search user interface

Fig. A3: Time browsing interface for the clicked image

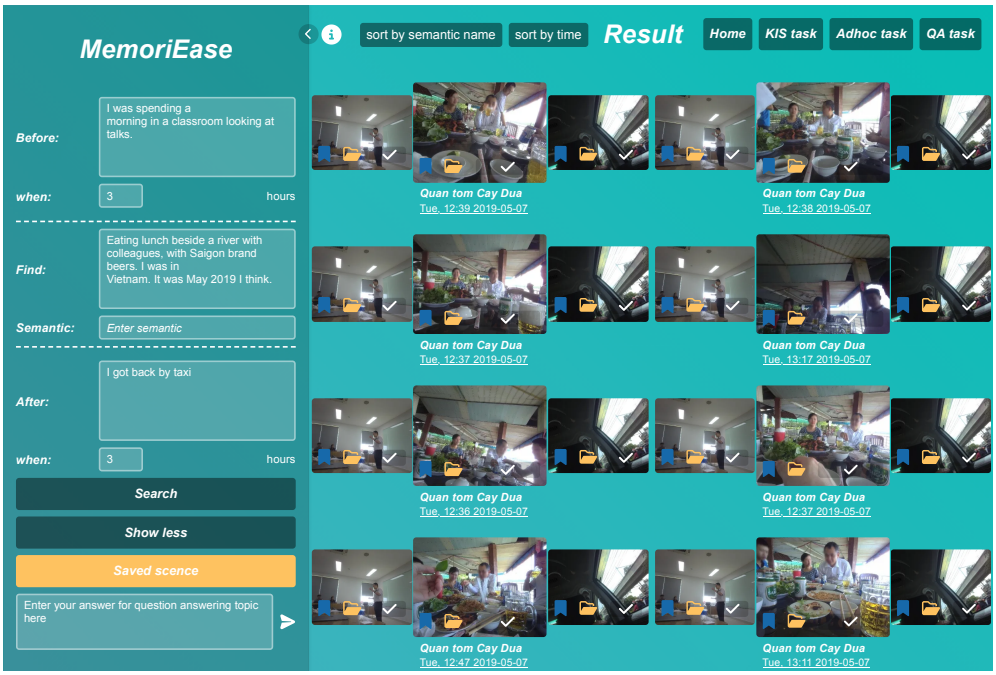


Fig. A4: Advanced temporal search interface for KIS sub-task

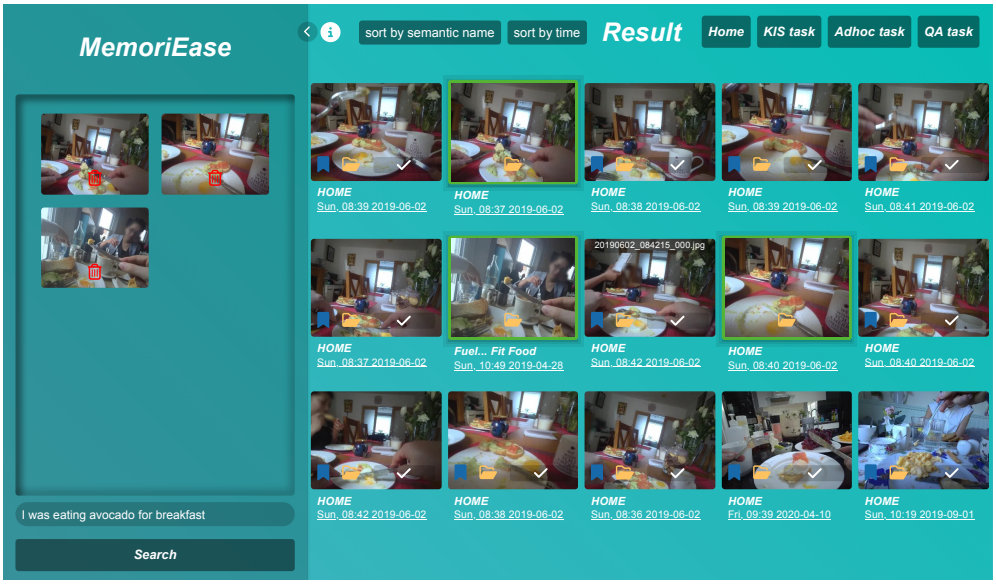


Fig. A5: Image-to-image retrieval interface