Moments in Focus: A Transformer-based Approach for Moment-centric Lifelog Retrieval

Thao-Nhu Nguyen, B.Sc.

A Dissertation submitted in fulfillment of the

requirements for the award of

Doctor of Philosophy (Ph.D.)

to the

Dublin City University

School of Computing

Supervisors

Prof. Cathal Gurrin

Dr. Liting Zhou

Dr. Tai Tan Mai

Dr. Binh Nguyen

April 2025

Declaration

I hereby certify that this material, which I now submit for assessment on the programme of study leading to the award of Doctor of Philosophy is entirely my own work, and that I have exercised reasonable care to ensure that the work is original, and does not to the best of my knowledge breach any law of copyright, and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of my work.

Sign: Student No.: 20214185 Date: 03/04/2025

(Thao-Nhu Nguyen)

Acknowledgements

I would like to express my deepest gratitude to my supervisor, Prof. Cathal Gurrin, for his continuous support and guidance throughout my Ph.D. journey. His expertise, patience, and encouragement have been invaluable in shaping my research and academic growth. I am also grateful to my co-supervisors, Dr. Annalina Caputo, Dr. Liting Zhou, Dr. Tai Tan Mai, and Dr. Binh Nguyen, for their insightful feedback and support during my PhD research and thesis writing.

I would like to thank to my internal and external examiners, Dr. Margaret Farren and Prof. Paolo Rota, for their valuable feedback and suggestions. Their thorough review and critical evaluation have significantly strengthened this thesis and provided important insights for future research directions.

Very special thanks to Dr. Jianquan Liu, Mr. Yamazaki Satoshi, and Mr. Zongyao Li at NEC Corporation, for the opportunity to collaborate on the research project. Their expertise and insights have been invaluable in shaping the direction of this research and have greatly enriched my understanding of the field.

I would like to extend my appreciation to all the members of the CRT-AI, Insight Lab, and especially my colleagues in the Human Modeling Group in DCU, for their collaboration and ongoing support. I am grateful for the shared experiences, late-night discussions, and the sense of community that made this journey truly memorable. Not to mention my friends and fellow Ph.D. students, who have been a source of motivation and support throughout this journey.

There is no words to express my gratitude to my family, my parents, and my brothers and sisters, who have always been my biggest supporters. I am deeply thankful for their unwavering belief in me and their constant support throughout my academic journey.

Last but not least, this work would not be impossible without the support and funding received from Science Foundation Ireland. This dissertation has emanated

from research supported in part by research grants from Science Foundation Ireland Centre for Research Training in Artificial Intelligence under grant number 18/CRT/6223 and ADAPT - Centre for Digital Content Technology under grant number SFI/13/RC/2106_P2.

To everyone who has contributed to this journey in ways both large and small thank you for making this research possible and for believing in me.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

NTCIR	NII Test Collection for IR Systems
GPS	Global Positioning System
TREC	Text Retrieval Conference (TREC)
LSC	Lifelog Search Challenge
ACM	Association for Computing Machinery
ICMR	The Annual ACM International Conference on Multimedia Retrieval
VR	Virtual Reality
NLP	Natural Language Processing
CNN	Convolutional Neural Network
HOI	Human-Object Interactions
TF	Term Frequency
IDF	Inverse Document Frequency
LIRT	Lifelog Image Retrieval Task
LMRT	Lifelog Moment Retrieval Task
LLQA	Lifelog Question Answering
LSTM	Long Short-Term Memory
CV	Computer Vision

Abstract

Moments in Focus: A Transformer-based Approach for Moment-centric Lifelog Retrieval

Thao-Nhu Nguyen

Lifelogging, the process of creating a personal digital record of daily activities, is gaining popularity as a form of digital diary. With this rise, the increasing demand for effective methods to extract specific activities from large, multimodal datasets becomes more crucial. This research explores novel approaches to improve information retrieval from personal lifelog data, focusing on moment-based retrieval to approximate human memory recall. This thesis begins with the development of LifeSeeker, a baseline image-based lifelog search engine utilising image concept-based indexing and retrieval techniques. Building on this foundation, the research leverages recent advancements in multimodal embedding models, particularly the Contrastive Language-Image Pre-training (CLIP) from OpenAI, to enhance search capabilities. Through interactive user studies, the embedding-based enhanced retrieval model demonstrates significant improvements over the conventional concept-based approach across various evaluation metrics. Recognising that humans encode and retrieve their experiences as sequences of events, we propose a shift from single-image-based to moment-based retrieval units in lifelog data, more closely aligning with human memory processes. For this purpose, I investigate the application of video moment retrieval techniques to lifelog data, exploiting similarities between video and lifelog data as continuous frame sequences. This led to the development of a novel Parallel Transformer Framework (PaTF) for moment-level retrieval from continuous visual streams. In particular, the PaTF combines the strengths of the transformer architecture with enriched multimodal embeddings (visual and semantic features) to capture the temporal context of lifelog moments. The integration of the PaTF into a

comprehensive moment-based lifelog retrieval system demonstrates significant improvements in retrieval effectiveness compared to the baseline LifeSeeker. In summary, the primary contribution of this thesis is the development and validation of a moment-based lifelog search engine utilising a transformer-based framework. This approach is expected to advance the field of lifelog retrieval and benefit personal information management, memory augmentation, and retrospective life analysis.

Chapter 1

Introduction

1.1 Overview

Have you ever wondered "*What did you do last Christmas eve?*" or "*When was the last time you went to the beach?*". Most of us have asked ourselves these types of personal-event-related questions at some point in our lives. The answers to these questions often appear straightforward, but in many cases, many individuals struggle to provide precise answers or might even misremember the events entirely [12]. How often do we find ourselves scrolling through our phone photo gallery, searching for particular events, or trying to piece together the fragments of our memory to recall a past event? The fact that human memory is not perfect and memory fade, which makes it increasingly challenging to recall past events accurately. This struggle stems from the nature of human memory, which psychologists have suggested in three stages: encoding (the initial perception and learning of information), storage (the preservation of memory traces over time), and retrieval (the accessibility of memory on demand) [13]. Failing in any of the three stages leads to forgetting or incorrect remembrances in memory. However, research has shown that strengthening these processes may result in stronger memory for an individual [14]. Indeed, proper encoding strategies are critical for improving one's memory, which could involve integrating new information with existing knowledge and creating an association of information. Meanwhile, the key to better retrieval is to provide relevant cues that match the desired information according to the principle of encoding specificity [15]. This understanding of memory systems has led to extensive studies in cognitive

psychology, demonstrating that documenting one's life experiences and providing them with those memory cues can help people enhance their retrieval of memory [16, 14, 17, 18]. Additionally, the use of external aids such as photographs, reminders, or notes can serve as powerful triggers and often benefit the recollection process of past events [19, 20, 21].

The concept of recording our everyday experiences is not new. It has been around since our ancestors documented their daily lives and hunting scenes via hand-painted cave interiors. In the modern age, people have more tools to help them document their lives, from writing diaries, blogging, taking photos, or recording videos to support the mind in consolidating their life experiences. These techniques, however, are only able to capture a subset of one's activities due to the requirement of manual efforts. This time-consuming and labour-consuming manual documentation creates substantial challenges in managing and reviewing the vast amounts of collected data at later stages. As a result, the emergence of lifelog technology, the process of totally passively capturing personal daily events [22], has opened up a new approach to the problem of managing personal collections. It represents a transformation from the traditional manual methods of recoding life events to a passive and effortless process. Using multiple wearable devices, it ultimately creates a visual diary encoding every aspect of one's life without human intervention, including a wide range of multimodal data such as egocentric images, audio, videos, text data, GPS coordinates, biometric data, or human-computer interaction data. Not to mention, these technological advancements have brought new challenges in managing and making the vast amount of collected data accessible and meaningful to users. Despite numerous lifelogging retrieval systems introduced to retrieve standalone images, a huge gap remains in the research on how one can effectively retrieve past moments from the lifelog collection. Traditional image-based search tools often fail to capture the temporal and contextual relationships within the lifelog events. This creates a disconnect between how humans naturally remember and how the system supports the

recollection of past events. With that being said, the motivation behind this research is to develop a moment-based lifelog retrieval system that supports users in managing and recalling their past events, using a video-moment-retrieval-based method as the core engine. The remainder of this chapter provides the introduction to lifelogging, lifelog retrieval, and video moment retrieval; the research motivation and challenges; the hypothesis and research questions; and the research contributions.

1.1.1 Lifelogging

Loftus et al. stated, "*Human memory is far from perfect or permanent, and forgetfulness is a fact of life*" [23], an important aspect of human memory that has led researchers to explore solutions to improve memory recall. A key component of this process is comprehending our experiences and being able to gain worthwhile insights from them. With the huge amount of life events accumulated over time, it is not surprising that a dedicated field of research has emerged around the subject of methods to store and manage life experiences more efficiently. The conceptual foundation of this field traces back to 1945 when Vannevar Bush initially proposed a desk-based system called Memex [24], where an individual stores "*all his books, records, and communications*". Recognising the vision of "*enlarged intimate supplement to one's memory*", Memex would be able to extend human memory by creating links between fragments of related information in different documents since memory works by associating items together. These links are called *data logs*, and the process of gathering them is known as *lifelogging* [25], which are used as the record of knowledge of an individual to retrieve for personal use or share with others.

Within the scope of this research, we use the definition of *lifelogging* suggested by Dodge and Kitchin [26], where lifelogging is referred to as "*a form of pervasive computing consisting of a unified digital record of the totality of an individual's experiences captured multimodally through digital sensors and stored permanently*

as a personal multimedia archive". The person who engages in lifelogging is called a *lifelogger* and the outcomes collected from the lifelogging process are called *lifelogs* or *lifelog data*. This process continuously and passively gathers the daily activities of an individual without the need for human involvement, creating a rich source of a personal digital diary via several wearable devices such as cameras, smartphones, or other sensors. As a result, lifelog data can be collected in various formats, including images, videos, audio, text, and other types of data.

With the development of technology, several research projects have been conducted to convert theoretical concepts from Bush's vision into practical implementations. One of the early research studies on lifelogging was conducted by Steve Mann, also known as the "*father of wearable computing*", who introduced new types of wearable sensing devices in 1997 [27]. He recognised the importance of "*Point of View*" (PoV) in capturing the first-person view of the wearer, which is crucial to recording personal experiences. Later in 2004, Aizawa et al. [28] attempted to incorporate the "*context*" of the lifelog data (such as GPS location) in conjunction with the visual data, enriching the captured experiences with additional metadata. Lifelogging has received more attention since the introduction of the SenseCam [1] in 2004, a wearable camera that captures images automatically based on the user's movement and environment. This innovation marked two paradigm shifts in personal data collection: first, the transition from the traditional camera to wearable cameras, specifically designed for first-person perspective recording, and second, the shift from active to passive data capture without any human involvement required. The SenseCam's versatile design allows users to wear the camera flexibly in different positions (wear it around the head, dangle it around the neck, clip it on the pocket, or attach it to the shirt), making it adaptable to user preferences while maintaining the egocentric viewpoint. Through this passive data collection, the SenseCam, together with other wearable devices, has enabled capturing the nature of every moment in one's life. Reviewing the lifelog data, users are triggered to recall their autobiographical memory with associated

emotions and feelings from the initial moments. Building upon these advancements, the MyLifeBits system [2, 3, 29], developed by Gemmell et al., emerged as a comprehensive implementation of Memex's vision. Going beyond the simple data storage tool, it is introduced as a tool to "*total capture*" everything in one's life with the help of SenseCam [1]. Specifically, MyLifeBits provides an efficient way to organise, generate annotations, create hyperlinks, visualise, and retrieve the data, which benefits personal collection management. Inspired by those works, Gurrin's work [30] introduced a practical application of turning lifelog experiences into memory cues, which is beneficial for the recollection process. The field is further explored with the use of smartphones as a lifelogging tool, which is more convenient and accessible. This approach can be an alternative to normal wearable cameras [31]. Beyond visual lifelog data, systems like Loggerman[1] [32] expanded the scope of lifelogging to gather the data logs from the user's computer activities to understand the user's behaviour and provide insights into the user's daily computer usage.

Following those early efforts in gathering personal lifelog data, there are more wearable devices developed for lifelogging such as cameras (OMG Autographer[2], Narrative Clip[3], or Google Glass[4], MeCam[5], etc.), physical activity trackers (Fitbit[6], Apple Watch[7], etc.), human computer interaction systems (Loggerman), smartphone apps (Moves, Saga, etc.), and various biometric sensors that can gather personal data. Capturing images from the first-person point of view, in conjunction with associated information such as location and time, may give valuable insight to the users as it records personal experiences similar to how the brain encodes the original ones. Consequently, it has shown promising applications not only in patients with Alzheimer's disease [33], dementia [34], aphasia [35], memory impairment [36], but

[1] `http://loggerman.org/`
[2] `https://oxfordmetrics.com/`
[3] `http://getnarrative.com/`
[4] `https://www.google.com/glass/photography/`
[5] `hhttps://mecam.me/products/mecam-hd.`
[6] `https://www.fitbit.com/`
[7] `https://www.apple.com/watch/`

also in healthy people [37, 38, 39].

1.1.2 Lifelog Retrieval

As Klein [21] defined memory, "*according to the "received view" is any state or process that results from the sequential stages of encoding, storage, and retrieval*". Human memory is a complex cognitive system that is composed of multiple memory systems, including sensory memory, short-term memory, and long-term memory, as shown in Figure 1.1. Sensory memory refers to the information of the five senses: hearing, vision, touch, smell, and taste. At the short-term level, working memory is responsible for reserving essential information for a brief period of time, whereas iconic memory briefly stores information received from our senses. In long-term memory, semantic memory maintains our general knowledge about the world, while episodic memory helps us recall our personal experiences and autobiographical events. Particularly, episodic memory refers to the ability to remember specific events, situations, and experiences, which are encoded in a specific context and time [15]. It is responsible for how we store our autobiographical memories [40]. The process of "*total capture*" has been highlighted with five potential benefits (the 5Rs) of memory access via lifelog systems, including (1) Recollecting (tracing back details of the past events), (2) Reminiscing (reliving the past events for emotional reasons), (3) Retrieving (locating specific digital information), (4) Reflecting (looking back to the past events from the perspective of other people), and (5) Remembering intentions (remembering prospective events or reminding events at some future points) [41].

The challenge of accessing episodic memory becomes apparent in the scenarios when people demand to recall their past experiences, such as answering questions like "*What did you do last Christmas Eve?*" or "*When was the last time you went to the beach?*". Several preliminary types of research used lifelog data, particularly visual lifelogs, as contextual cues that have the potential for supporting the memory system, specifically human episodic memory [42]. Lifelog retrieval refers to

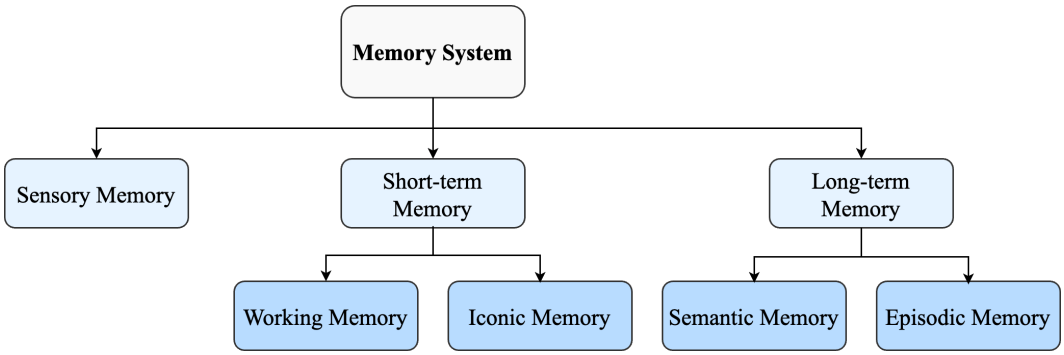
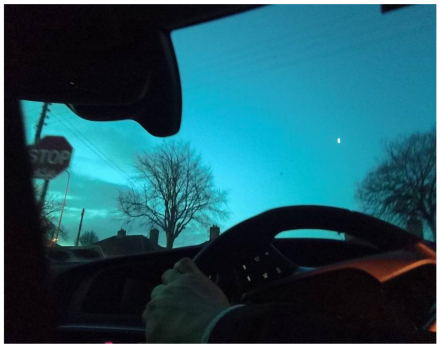


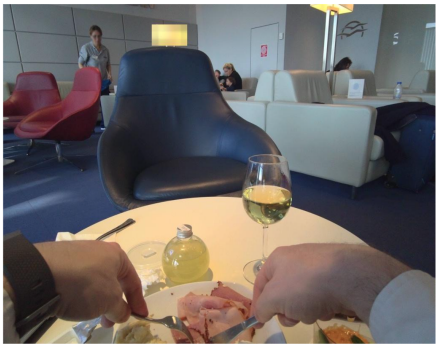
Figure 1.1: The structure of human memory

the process of searching for particular retrieval units (e.g., images, moments, events) from personal lifelog data. With large-scale lifelog data collection, the manual retrieval of past moments becomes impossible, necessitating the development of lifelog retrieval mechanisms. Lifelog retrieval tools not only keep a digital record of an individual's experiences but also provide an efficient way to organise and access an enormous collection of events during their lifespan.

The evolution of lifelog retrieval systems can be traced back to the early 2000s, when Gemmell et al. made an early effort to introduce MyLifeBits [2], an SQL-based lifelog browsing tool. This system gathered various types of data, such as images, videos, emails, and documents. Following this, Lee et al. [4] created the My Visual Diary system, which focused on a standalone retrieval system for effective structuring and retrieving from the lifelog image collections. They displayed such a visual diary on an intuitive interactive browser interface that enables users to access the multimodal data by navigating the UI. Later that year, Doherty et al. [12] proposed a computer-driven solution, inspired by MyLifeBits, to help individuals structure and manage their digital collection as a sequence of events. These early systems have proven the potential of lifelog data as a form of digital assistance, which could support people recollecting their past moments in life [36, 43]. Moreover, the data collected in this way illustrates how the modality of information was formatted when displayed to the user, thus allowing the user to form more detailed queries than other conventional personal information management (PIM) tools.



"objects": "sky, dark, day"
"ocr": "STOP"
"caption": "a person driving a car"
"date": "2019-01-27"
"local_time": "07:52:05"
"gps": [53.39001397, -6.14585188]



"objects": "indoor"
"ocr": ""
"caption": "a person holding a glass of wine"
"date": "2019-01-12"
"local_time": "14:07:39"
"gps": [40.5242182, 22.9757013]

Figure 1.2: Example photos captured by wearable devices with the provided corresponding information (collected from the LSC'22 dataset).

With the increasing popularity of lifelogging, especially the development of lifelog retrieval systems, several benchmarking challenges have been established to evaluate the performance of lifelog retrieval systems. One of the most well-known challenges that have been organised since 2015 is the NTCIR Lifelog Task [44], followed by the ImageCLEF Lifelog Task [45] in 2016, and the Lifelog Search Challenge (LSC) [46] in 2018. These competitions aim to evaluate the effectiveness of lifelog search tools in either interactive or non-interactive settings, specifically focusing on their ability to retrieve past lifelog images that descriptively match the given textual queries. Research teams from around the world have participated in these challenges, showcasing their state-of-the-art lifelog retrieval systems and competing against each other to achieve the best retrieval performance. First extracting the meaningful information from the lifelog data, then measuring the similarity between the query and the indexed data, and finally displaying the results of the ranked list for the users to interact with. Depending on the competition design, the system's performance is evaluated based on various metrics, including Precision and Recall in the NTCIR Lifelog Automatic task, Normalise Discounted Cumulative Gain (NDCG) in the ImageCLEF Lifelog Task, and LSC Score (as described in Section 3.3.1) for all

tasks in the LSC competitions. Examples of lifelog images and their corresponding metadata from the LSC'22 [47] dataset are shown in Figure 1.2.

1.1.3 Video Moment Retrieval (VMR)

While lifelog retrieval tasks focus on the retrieval of images from one's life, Video Moment Retrieval (VMR) encounters a broader field of research that aims to localise a specific moment within a video. In particular, VMR focuses on localising specific temporal segments (moments with start and end marks) within an untrimmed video according to a textual query. The key connection between the two tasks lies in their data format as continuous and chronological data streams that capture temporal events in sequence. As videos represent sequences of frame-capturing temporal events, lifelog data is similarly temporal-ordered images documenting personal experiences. The difference between the two tasks is the retrieval unit, where the former targets temporal segments or moments and the latter focuses on individual images, with the illustration shown in Figure 1.3.

This insight leads to an interesting observation that the approaches for VMR tasks could be transferred to lifelog retrieval tasks, where we consider videos as sequences of images. Exploring the approaches for VMR tasks could open up new opportunities for enhancing lifelog retrieval, especially in the context of personal recollection.

Great efforts have been made to resolve this task with the development of two main approaches, the proposal-based and proposal-free methods. Most previous work [48, 49, 50, 51] operates the VMR task by initially generating predefined candidates or proposals within the video, followed by matching techniques to rank them based on the learned representations (proposal-based methods). Although effective to some extent, such methods require significant efforts to annotate moment boundaries, leading to challenges in annotation and scalability. In the meantime, other frameworks [52, 53, 54, 55] implemented the proposal-free methods that learn the cross-modal interactions and attempt to regress the

(a) Lifelog Retrieval task. The query result is highlighted with three red boxes, users can submit any of the three frames.

(b) Video Moment Retrieval task. The query moment is highlighted with a red box starting at 1.9s and ending at 17.4s.

Figure 1.3: Illustration of lifelog retrieval task and video moment retrieval task.

probabilities of all frames, then choose the peaks as the start and end of the event's segments. Recently, transformer architecture [10], a powerful tool aiming to capture long-range dependencies in sequential data with attention mechanisms to model the entire sequence simultaneously, has been applied to the VMR task [56, 57, 58]. These transformer-based methods have shown promising results in predicting the temporal boundaries of the target moments without the need for predefined proposals.

Building upon these developments, we propose a novel framework that makes use of proposal-free techniques to reduce the required level of human involvement. To better represent the visual content, we further leverage a two-stream transformer-based architecture, which captures both visual and semantic feature representations. This dual-branch approach is expected to enhance the retrieval performance of the VMR task by leveraging both visual and semantic cues. This also benefits the lifelog retrieval task for the same reason.

1.2 Research Motivation and Challenges

The number of lifelogging retrieval systems developed in recent years has significantly increased. Yet, most of them still take into account the standalone images as basic units of retrieval, which basically solve the tasks in the lifelog competitions but not always the real-life situations of finding the past. Instead, an "*event*" or "*moment*" is defined as "*a segment of time at a given location that is conceived by an observer to have a beginning and an end*", according to Zacks and Tversky [59], which is closely related to the targets of the task. There are two reasons why we should consider the moment as the retrieval unit. First, standalone lifelog images miss the temporal and continuous nature of lifelog data and cannot capture the whole context of the event [22]. Second, following the event segmentation theory, the human cognitive system encodes experiences by dividing continuously experienced moments into small chunks called events [60]. Subsequently, people tend to remember and trace back their episodic memory on an event basis [15]. Therefore, to provide a better understanding of how past moments occurred and are encoded in one's memory, designing tools that have the ability to closely simulate human memory and make past moments more accessible when users ask to look back is becoming important.

To achieve that, I first explore the approaches for the moment retrieval task in video data, which is similar to lifelog data in continuous format. Adopting those techniques, I will further construct a novel retrieval system that focuses on the moment as the unit of retrieval which refers to specific activities or moments of the lifelogger. Moreover, with the increasing popularity of transformers, a mechanism that has shown promising results in modeling sequential data without losing the temporal information, I will explore the potential benefits of leveraging transformer-based techniques. In doing so, it is able to enrich the understanding of lifelog data helping to bridge the gap between visual and textual content. Not to mention, providing an intuitive user interface might facilitate the recollecting process.

This work was motivated by the idea of how human memory works, where visual lifelogs are segmented into events. The associations between them are calculated, and finally, their importance is identified. The system was designed to support the episodic memory of the lifelogger by providing a more intuitive way to access past events via visual lifelogs and additional hints from associated metadata (e.g. location, time, and physical state).

For the remainder of this dissertation, the terms *event* and *moment* will be used interchangeably to refer to a combination of activities that occurred in a short period of time. In the context of lifelogging, it is represented by a sequence of contiguous first-person view images, whereas it is a short video with start and end marks in the context of video moment retrieval. Additionally, *personal recollection* relates to the process of recollecting past events of an individual. It is also worth mentioning that I will refer to both "*I*" and "*we*" as I am working as a part of the research team. When I mention "*we*", this refers to research activities in which I collaborated with my colleagues, but I was the primary researcher or developer. Otherwise, when I write "*I*", this specifies the work that I have done by myself.

1.3 Hypothesis and Research Questions

With that being stated, I consider the following hypothesis and research questions to be addressed in this dissertation:

Hypothesis

The integration of transformer-based techniques into a moment-based lifelog retrieval approach enhances the performance of retrieving one's past experiences when compared to conventional image-based lifelog retrieval systems.

Associated with this hypothesis, we have formulated three main research

questions in order to effectively investigate this hypothesis, as follows:

Research Question 1 (RQ1): How can a state-of-the-art baseline lifelog interactive retrieval system be developed to effectively support users in retrieving their past experiences?

Given the large-scale lifelog data collection, existing conventional lifelog retrieval systems, which mainly focus on matching text queries with the visual concept keywords, have shown limitations in capturing the temporal and contextual relationships within the lifelog events. To address this limitation, we have identified the need to develop a state-of-the-art lifelog interactive retrieval system that can effectively retrieve past moments from the lifelog data. We concentrate on developing a conventional lifelog interactive retrieval system whose function is to support an individual in retrieving past moments from their life. In particular, our system, named LifeSeeker, is composed of two main components: a core search engine that employs advanced embedding models for both text-to-image matching and semantic understanding, coupled with a straightforward user interface for user interaction. This base system will be used as a benchmark for comparison with the novel retrieval system developed in the future.

Research Question 2 (RQ2): How can a state-of-the-art transformer-based Video Moment Retrieval technique be designed to effectively localise target moments in video sequences?

As most of the current lifelog retrieval systems focus on standalone images, there is a gap in the research on how to effectively retrieve past moments from the lifelog collection. In addressing this research question, we will investigate current state-of-the-art techniques for video moment retrieval. Our objective is to propose a novel approach that better localises the target onset and offset of the moment in the video. To achieve this, we develop an attention-based Parallel Transformer Framework (PaTF) that enriches the feature representations by exploring both low-level visual cues and high-level relational contexts of video-query pairs. Our

framework consists of two parallel transformers: one for the visual-textual stream and the other for the semantic-textual stream. The visual-textual stream extracts the links between global visual features and textual information, while the semantic-textual stream emphasises the relations between objects via scene graph representations. Furthermore, we conduct a comprehensive evaluation experiment on the Charades-STA benchmark dataset to demonstrate its promising performance in localising target video moments in comparison to the state-of-the-art methods. This can be used as a foundation for the future integration of the proposed technique into the lifelog retrieval system to better retrieve moments from lifelog data.

Research Question 3 (RQ3): To what extent can the proposed transformer-based moment retrieval approaches, when applied to lifelog retrieval, improve the lifelog retrieval performance of past moments when compared to existing SOTA interactive retrieval systems?

After developing the novel video moment retrieval technique, I investigate the potential benefits of incorporating transformer-based VMR techniques into the lifelog retrieval system to answer this research question. As for the similarity in the continuous and chronological format between lifelog data and video data, we will explore the transferability of these techniques onto the lifelog dataset. The system shifts the retrieval unit from a single image to a moment, which is a sequence of images. Extensive studies will be conducted to evaluate the performance of the proposed system in a specialised lifelog moment dataset. The proposed system will be evaluated against the baseline lifelog retrieval system to demonstrate its effectiveness in recalling past experiences.

1.4 Research Contributions

This dissertation makes several contributions to the field of lifelog retrieval, which are highlighted as follows:

- We developed and implemented a robust baseline lifelog retrieval system, named LifeSeeker, which incorporates current best practices in lifelog retrieval systems. The system features an efficient indexing mechanism, a powerful matching engine, and an intuitive user interface. I reported the results of our lifelog retrieval system in multiple lifelog search competitions, including the NTCIR Lifelog Task, and Lifelog Search Challenge (LSC). The system has been evaluated against other state-of-the-art systems to demonstrate its effectiveness.

- I proposed a novel approach for the video moment retrieval task, named Attention-based Parallel Transformer Framework (PaTF). This approach states as the state-of-the-art technique for localising target moments in videos with extensive evaluations of public datasets.

- I identified the challenges in structuring the lifelog data into events and proposed a novel lifelog retrieval system that focuses on the moment as the retrieval unit.

- I integrated the PaTF model into a lifelog retrieval system, name PaTFLifelog, to support users in recalling their past experiences. The system is evaluated against the baseline system to demonstrate its effectiveness in retrieving past moments from lifelog data.

1.5 Dissertation Outline

This chapter provides an overview, motivations, and challenges of the research, as well as the hypothesis and research questions. The remainder of the dissertation is organised as follows:

- **Chapter 2** provides a comprehensive review of personal lifelogging technologies, with particular emphasis on the benchmarking challenges and lifelog search systems within the domain. Moreover, the latest techniques for

video moment retrieval, as well as transformer-based retrieval approaches, are also explored to identify the research gap and potential research directions.

- **Chapter 3** outlines the research methodology and evaluation methods to address the proposed research questions and validate the hypothesis.

- **Chapter 4** presents the construction and development of the baseline lifelog retrieval system, named LifeSeeker over research iterations. The system's evolution is deeply analysed and evaluated through multiple lifelog retrieval challenges. In particular, key components of the system are detailed, including the data indexing, retrieval engine, and user interface. Not only that, the system's performance is also presented, discussing its capabilities and limitations.

- **Chapter 5** introduces a novel approach to video moment retrieval through the development of the Attention-based Parallel Transformer Framework (PaTF). An in-depth analysis of the model's architecture and training process is provided, followed by the evaluation of the model's performance, demonstrating its effectiveness in localising target moments in videos.

- **Chapter 6** explores the application of the video moment retrieval techniques built in Chapter 5 to lifelog data, demonstrating how a transformer-based moment-centric lifelog retrieval system can support users in recalling their past experiences. It presents the adaptation process, implementation details, and evaluation results of the proposed system against the baseline system.

- **Chapter 7** synthesises the key findings and contributions of this dissertation. In addition, the limitations of the research are also discussed, followed by suggestions for future work. Finally, a publication list is highlighted.

Chapter 2

Related Work and Background

In this chapter, I will give an overview of the material that is relevant to this dissertation. In particular, I first introduce the concept of lifelogging and lifelog retrieval in Section 2.1. Several lifelog retrieval benchmarking challenges and various state-of-the-art techniques in the existing lifelog retrieval systems are discussed. These insights will help us to understand the key components of one lifelog retrieval system and the challenges that they are facing. Then, to develop a moment-based retrieval system for lifelog data, I present the concept of video moment retrieval and the applications of the vision transformer architecture to solve the task in Section 2.2. Lastly, I discuss the applications of event-based approaches in lifelog retrieval context and the potential to enhance the retrieval performance in Section 2.3.

2.1 Lifelogging and Lifelog Retrieval

2.1.1 Personal Lifelogging

As aforementioned, lifelogging refers to the process of gathering every slice of one's life to keep a personal record of their daily experiences. This field has been presented for over 30 years and has evolved with the advancements of technology [61, 42]. The idea of managing personal information had not gained much attention from the community up until the emergence of MyLifeBits, a personal digital archive system developed by Gordon Bell and Jim Gemmell at Microsoft Research Labs [2]. Inspired by Memex vision [24], MyLifeBits benefits from the SenseCam wearable camera [1]

to "*enlarge intimate supplements to one's memory*". The authors used this system as a lifetime storage facility for Gordon's data, including textual documents (articles, books, CDs, letters, memos, and papers), visual documents (Sensecam-captured photographs, pictures, and videos), and audio documents (voice recordings). The advancement of digital technologies blends into our lives in various ways, from the way we communicate, work, and entertain to the way we record and share our daily activities. Wearable devices have become more popular recently, which has posed an opportunity for approaches to lifelog data capture and management. From wearable cameras, wearable sensors, and mobile devices, lifelogging has become more and more accessible and affordable for the general public [22].

As a demonstration, in 2008, Lee et al. [4] created the first SenseCam image management system, called My Visual Diary, in which lifelog images were structured and indexed. This tool allows users to navigate and retrieve a digital record of past activities through a web-based interface. Notably, lifelogging technologies have also demonstrated their potential in supporting memory systems, especially in assisting episodic memory for individuals with memory impairment [62, 33, 34, 36, 43, 35] while also enhancing memory capabilities for people with normal cognitive function [12, 37, 4]. Beyond memory support, the adaptation of lifelog data has expanded across various domains, with applications ranging from commercial marketing, healthcare, wellness, vehicular logging, and computer usage logging to quality of life. In the context of personal documentation, the lifelog collection serves as a passive personal digital diary [4], enabling individuals to capture and share their personal experiences for self-reflection, sharing between family members, or even for public sharing. The commercial sector has leveraged lifelogging technology to measure the effectiveness of advertising campaigns by tracking consumer exposure via an automatic market system [63]. In healthcare, the applications have been diverse and impactful as Reddy et al. described DietSense [64] as a diet monitor tool that utilises lifelog meal images to track the user's eating habits. Aizawa et al. [65] created a food log application that analyses

food images and the balance of the diet of users. Metsis et al. [66] proposd a sleep monitor system for the detection and treatment of sleep disorders. In comprehensive wellness monitoring, Zini et al. [67] proposed a system designed to monitor one's quality of life through activities, sleep quality, level of fatigue, and mood, representing users as gauge charts on smartphones. The monitoring of physical activity across sedentary travel behaviour has been enhanced by Kelly et al. [68], contributing to enhancing personal wellness. A vehicular lifelogging system used in-car sensors to keep track of the ongoing vehicle discoveries, social context, and driving environment [69]. In the workplace context, Loggerman[1] [32], developed by Hinbarji et al., represented an innovative lifelogging application collecting human-computer interaction data, including keyboard, mouse, and screen activities, to support the user in understanding their computer usage patterns.

Lifelog Collections

The wide-range multimodal lifelog dataset can be collected from various sources, such as wearable cameras, wearable sensors, and mobile devices, which can be divided into different categories [70] as below:

- Passive visual capture: images, videos, and corresponding temporal information captured by always-on wearable cameras. These images are usually captured from the first-person perspective, providing a detailed record of the lifelogger's daily activities. These devices
- Personal biometrics: wearable sensors that capture physiological data, such as heart rate, galvanic skin response, body motion, and sleep patterns.
- Mobile activity and context: call logs, SMSs, or user activities monitored on the mobile and user's context (location, movement, or acceleration) captured by continuously using the mobile devices.

[1] `http://loggerman.org/`

- Desktop/laptop computer activity: user's interaction with the computer, such as keyboard, mouse, and screen activities, emails, and web browsing.
- Active capture: user's active input, either direct (blogging) or indirect (posting on social media, sharing photos, updating status), initiated by the user.

While non-visual metadata is useful for other purposes, such as health monitoring or activity recognition, passive visual data is one of the most common types of lifelog data used in lifelog retrieval systems thanks to its rich content and detailed context [22]. As my focus for this dissertation is on developing an efficient lifelog search tool, I will not discuss those non-visual categories in detail. Instead, I put my attention on the passive visual data and the corresponding metadata to support the retrieval process.

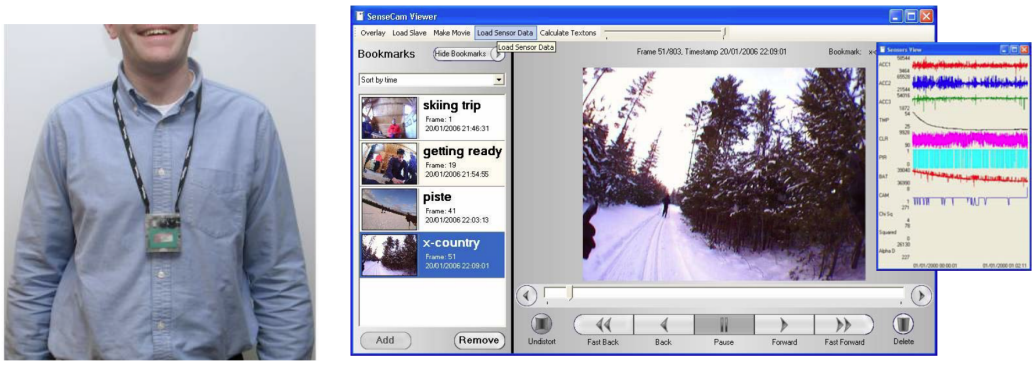
2.1.2 Early stage Lifelog Retrieval Systems

With a huge amount of lifelog data captured daily, browsing and looking for a specific occasion throughout the lifelog collection poses a significant challenge. Subsequently, the demand for developing lifelog retrieval systems to support users in the retrieval process has been raised. To meet this demand, several early efforts have been made to design and implement lifelog retrieval systems, which are discussed in this section. While SenseCam Photo Viewer is the first widely adapted passive capture device and viewer, MyLifeBits [2] serves as the first personal digital memory platform. My Visual Diary [4] contributes to automated personal event segmentation and organisation. Together, these systems introduced the fundamental concepts and interaction factors, serving as a strong foundation for the development of more advanced lifelog retrieval systems in the following years.

SenseCam Photo Viewer

The SenseCam [1] was designed as a fish-eye camera to capture images from the first-person perspective automatically. Different from other cameras, the fish-eye

lens allows the camera to capture a wide-angle view of the scene, providing valuable information about the context of the lifelogger's activities and surroundings. There are several positions in which users can choose to wear the SenseCam, most commonly around the neck (as shown in Figure 2.1a), clipped on the shirt, or on the belt, making it adaptable to various usage scenarios. Without human intervention, the SenseCam could passively capture up to approximately 2000 images on a daily basis, accumulating more than half a million images annually.



(a) SenseCam Wearer (b) SenseCam Viewer

Figure 2.1: Microsoft's SenseCam Camera Wearer and Photo Viewer Interface, reported in [1]

Notably, the SenseCam design prioritises the ease of use and the lightweight nature of the device, users can wear it throughout the day without feeling uncomfortable. Therefore, SenseCam has no built-in display or viewfinder, which means that the lifelogger cannot see the frame view while capturing or reviewing the images after they are taken. Instead, the lifelogger can manage and review the taken images on a PC-based viewer application, as shown in Figure 2.1b. The interface is quite simple and straightforward, presenting users with an image slideshow and sequential navigation in the main area, complemented by a playback control bar at the bottom. Users"*Bookmarks*" option is available for the user to mark certain images as important and make further annotations, which are displayed in the "*Bookmarks*" tab on the left side of the UI. While the SenseCam Photo Viewer is clearly useful for basic image-viewing tasks, it does not support

any form of search or retrieval functionality. Despite these limitations, passive and continuous life document applications have been used in many lifelogging research studies in various research fields, as discussed ealier in Section 1.1.2.

MyLifeBits

In 2002, Gemmell et al. introduced MyLifeBits [2], a platform structure using SQL databases of resources and links. The system can store and connect various types of data, including contacts, documents, emails, events, photos, and videos. By implementing an annotation system, MyLifeBits enabled users to create annotations, particularly creating links between data and connecting images to the calendar or the place where the image was taken.

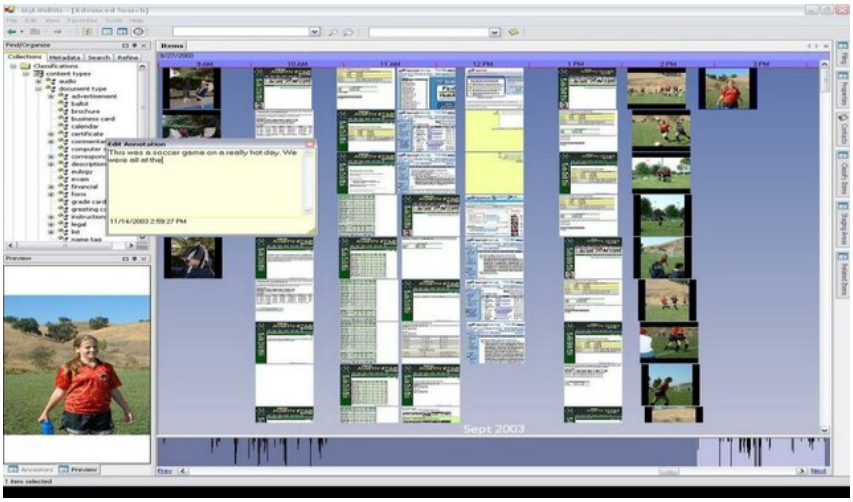
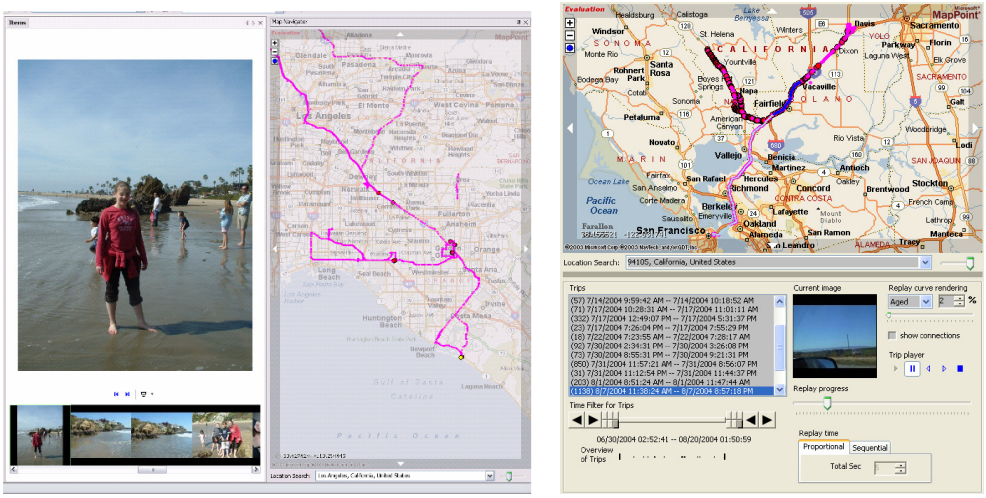


Figure 2.2: The User Interface of MyLifeBits system, as reported in [2]

The tool offers options for data visualisation via four views: (1) Detail view, (2) Thumbnail view, (3) Timeline view, and (4) Clustered-time view. The Timeline result interface, as shown in Figure 2.2, displayed the results in a timeline with the ability to zoom in and out and adjust the time range. The corresponding histogram below the timeline showed the distribution over time. Aris et al. [3] subsequently enhanced the MyLifeBits system with advanced location and time search capabilities. Their Map UI, illustrated in Figure 2.3a, featured a dual-pane interface, one side displays the lifelog images, and the other side shows the map with the location of

that image highlighted by a red dot. Those surrounding images are also displayed on the map with smaller purple dots. In the meantime, users can replay their trips by choosing a particular trip or adjusting the time filter on the timeline, as shown in Figure 2.3b.



(a) Map UI (b) Full trip replay UI

Figure 2.3: MyLifeBits Map UI and Full trip UI, reported in [3]

In summary, the MyLifeBits system is a pioneer in the field of lifelogging, providing a platform for users to store and manage their data. This first attempt to digitalise personal data collection has inspired many researchers to develop their own lifelogging systems to manage and retrieve their experiences.

My Visual Diary

My Visual Diary, developed by Lee et al. in 2008 [4], represents a significant step forward in lifelog data management, introducing event segmentation and visualisation techniques to transform raw lifelog data into the personal visual diary format. The management tool offers users the ability to annotate and browse throughout the lifelog images in a more intuitive way. In order to organise the lifelog images captured from the SenseCam camera, the authors employed an event segmentation technique that combined context-based and content-based methods [71] to segment lifelog images into a sequence of an average of 22 distinct events on

a daily basis (such as eating, driving, or working, etc.). The context-based method used motion sensors to detect the event boundaries, which was widely used in the lifelogging community at the time. Additionally, lifelog images can also be segmented by the content-based method, which implemented Hearst's text-tiling technique [72]. For each segmented event, users can annotate the event with a title, description, and location, and then the system selected the "*landmark photo*", the most meaningful frame to represent the event content [73]. Secondary to the landmark keyframe selection, the system also introduced the concept of an "*event novelty*" score. Specifically, this score was calculated based on the visual uniqueness of that event compared to the other events on the same day, coupled with automatic detection of face-to-face conversation[74].

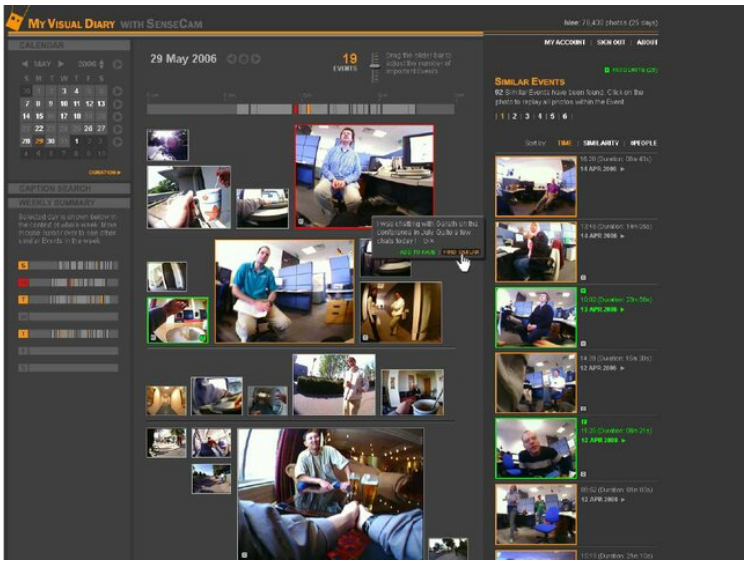


Figure 2.4: The User Interface of My Visual Diary, as reported [4]

The system's web-based interface, visualised in Figure 2.4, features a multi-modal browsing capability that allows users to choose the date on the calendar or input the keyword-based query to match with their annotated caption if available. Following this, an exhibition of sequences of result events in the form of a timeline is placed in the main area of the UI. The higher the event importance score, the larger the frame size is displayed. The smaller photos mean the lower novelty of that event, which is

either frequent activities or mundane events. While the left side shows the calendar to choose the date to browse, the ranked list of events sorted by either time or relevance is displayed on the right side. My Visual Diary's significance extended beyond its technical innovations, serving as both a memory aid for triggering and consolidating episodic memories and an inter-disciplinary platform for accessing personal digital memories.

Although those early lifelogging systems focus on managing the lifelog images as a photo album for browsing, they still lack the ability to assist users in finding the desired moment in the lifelog data on demand. To go beyond simple browsing, an appropriate mechanism is needed to better support user interaction with lifelog data, allowing users to navigate and explore the lifelog collection more efficiently. This is the motivation for the development of lifelog retrieval systems and the introduction of lifelog retrieval challenges in the following years.

2.1.3 Lifelog Retrieval Benchmarking Challenges

Passively capturing thousands of egocentric images—images from first-person vision with time-aligned sensor data per day, the collection of one lifelogger over the years would be too enormous to organise and retrieve manually. As a result, designing mechanisms for managing and retrieving purposes using multimedia data presents a considerable challenge for researchers. This task, also called the Lifelog Moment Retrieval task (LMRT), has been investigated in several benchmarking challenges, such as NTCIR Lifelog [75, 76], ImageCLEF Lifelog [45, 77, 78, 79], and the Lifelog Search Challenge (LSC) [80, 81, 82] with the aim of developing lifelog search engines and being judged by appropriate evaluation metrics.

2.1.3.1 NTCIR Lifelog Challenge

The first collaborative benchmarking exercises[2] for indexing and retrieving multimodal lifelog data were organised at NTCIR-12 in 2016 [44]. This task was

[2] http://lifelogsearch.org/ntcir-lifelog/NTCIR12/

carried out at NTCIR in 2015, 2017, and 2018, together with the release of a novel personal life archive introduced every year of the organisation. Apart from the collection of first-person point-of-view images captured from the OMG Autographer wearable camera, the dataset was also provided with XML descriptions of semantic location (e.g., home, work, airport, etc.) and physical activities (e.g., walking, transport, cycling, etc.). There were two main tasks in this challenge: the Lifelog Semantic Access Task (LSAT) asked to explore and retrieve the lifelogs, and the Lifelog Insight Task (LIT) which asked to analyse and visualise the lifelog data. Since our emphasis is on the retrieval task, I will not discuss the LIT task in detail.

In the 2016 competition, five systems were evaluated in terms of precision and recall on 48 different topics in either an interactive or non-interactive manner. They all make efforts to enhance the visual concepts. As a baseline search engine, Zhou et al. [83] indexed and hierarchically organised multimodal data into basic units of minute, which were matched with input queries to locate the desired moments. The best-performing team, LIG-MRIM [84], enriched the concepts provided with sentiment features and improved the retrieval process with a query expansion function. Meanwhile, Lin et al. [85] integrated an image recognition model to gain more meaningful content for their automatic lifelog search engine. LEMoRe team [86] leveraged both low-level and high-level features, which are provided by the organisers and extracted by the CNN model, respectively. Apart from implementing a similar approach to the baseline search engine, the last team, QUT from Australia [87], annotated clusters of images based on visual similarity. The initial results of this pilot task proposed a potential domain for the lifelogging research area in the future.

2.1.3.2 ImageCLEF-Lifelog Challenge

The first edition of Lifelog task was introduced in ImageCLEF 2017[3]. There were two sub-tasks in this competition: Activities of Daily Living Understanding Task (ADLT) and Lifelog Moment Retrieval task (LMRT). This challenge leveraged the dataset from NTCIR12 [44] which is comprised of more than 88,000 lifelog images gathered from 3 different lifeloggers during 87 days.

Early systems in ImageCLEF-17 [45], such as I2R [88] and UPB, employed multi-step approaches combining textual and visual information with basic clustering techniques using either k-means or a hierarchical tree structure. By ImageCLEF-18 [77], teams like CAMPUS-UPB [89], AILab-GTI [90], and HCMUS [91] introduced more advanced methods incorporating convolutional neural networks, weakly supervised learning, and multimodal fusion techniques to solve LMRT tasks. Later in ImageCLEF-19 [79], participant teams (REGIM-Lab [92], UAPT [93], HCMUS [94], BIDAL [95], ZJUTCVR [96], and ATS [97])implemented advanced approaches such as LSTM classifiers for query enhancement, blur detection systems, multiple concept extraction models, and modern clustering algorithms. The evolution of these systems reflected a clear progression in gaining more attention from the research community to the lifelog retrieval task, with the adoption of deep learning methods and semantic concept extraction in later iterations.

2.1.3.3 Lifelog Search Challenge (LSC)

As part of the ACM International Conference on Multimedia Retrieval (ICMR), Lifelog Search Challenge (LSC) is an annual lifelog retrieval competition first launched in 2018, with the idea inspired by the Video Browser Showdown (VBS)[4]. The LSC challenge, regconised as one of the most popular benchmarking challenges to evaluate the performance of interactive real-time lifelog retrieval systems, has

[3] `https://www.imageclef.org/2017/lifelog`
[4] `https://videobrowsershowdown.org/`

attracted a large number of researchers to showcase their innovative ideas and solutions in lifelogging retrieval.

The challenge's objective is to develop an interactive lifelog retrieval system that has the ability to store and retrieve lifelog data from a lifelogger's collection. Precisely, given a text description, participants handle the scenario of identifying one or more particular images related to the lifelogger's activities within a given time constraint. Three main types of LMRT tasks used in those challenges (with the query examples from the LSC'23 [98]) are described as follows:

- Known-Item Search task (**KIS**): Find any single image from a pre-selected list that answers a textual information need. The information needed is composed of several hints, with details increasingly displayed at every fixed interval of time. KIS task example query is shown in Table 2.1.

- Ad-Hoc Search task (**ADS**): Find as many images of a described item or activity (the query) as possible. For example, the query is "*Find examples of when I was trying to repair an electric shower*, users can submit as many images as possible within the time limit.

- Question Answering Search task (**QAS**): Find a textual answer from the images that answers the question in the information needed (e.g., car registration number, house number, the colour of dogs, etc.). "*I can't find my hand drill/electric screwdriver. Assuming that today is the 1st of July 2020, when was I last using it?* is an example question. The answer is the date on which the images of the lifelogger last used the screwdriver are found.

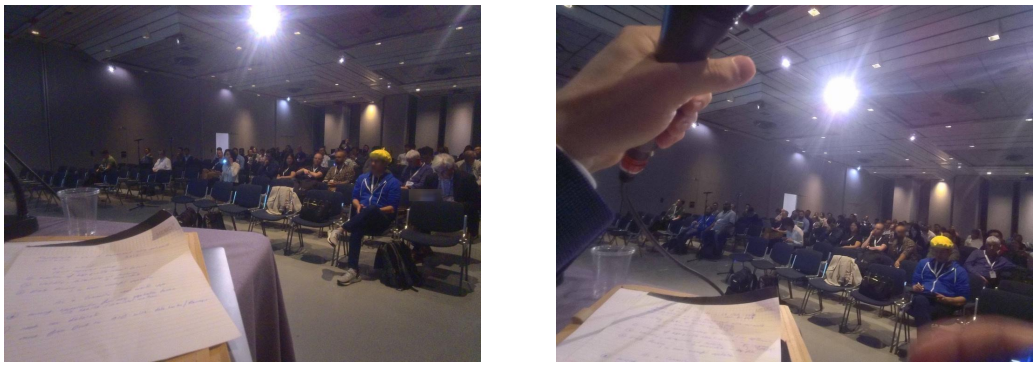
The time constraint is set specifically for each task, either 180 or 300 seconds. The KIS tasks are allocated 300 seconds, and the ADS and QA tasks are given 180 seconds. Precisely for the KIS task, each piece of hints is displayed every 30 seconds via a countdown timer, with an example query from the LSC'23 shown in Table 2.1. The participants can submit their potential answers at any time during the task. The ground-truths of this task are pre-selected by the organisers. Those

judgment images from the lifelogger's log are considered relevant if they satisfy all clues from the given query, otherwise, they are deemed incorrect answers. For this specific task, there is more than one correct answer, and participants can submit any of them. Two corresponding correct answers that match the example input from Table 2.1 are shown in Figure 2.5.

On the other hand, the ADS and QA submissions are judged in real-time by the organisers as there is no ground-truth provided in advance. While the ADS answers are judged image by image based on their relevance to the query, the QA submission is considered relevant if the text submission matches the ground-truth answer, with some flexibility applied as they can be described in different ways.

Table 2.1: An example of KIS task from LSC'23. Task 1 with its temporally advancing descriptors, which were revealed at 30-second intervals. After 150 seconds, the full description is shown for another 150 seconds until the end of the task.

Time	**Text**
0s	There was a man in the front row with a yellow hat on...
30s	There was a man in the front row with a yellow hat on. I was on a stage in a room with a lot of people watching...
60s	There was a man in the front row with a yellow hat on. I was on a stage in a room with a lot of people watching. I was on some sort of panel...
90s	There was a man in the front row with a yellow hat on. I was on a stage in a room with a lot of people watching. I was on some sort of panel and writing notes on paper...
120s	There was a man in the front row with a yellow hat on. I was on a stage in a room with a lot of people watching. I was on some sort of panel and writing notes on paper. I remember the man had a blue sweater/top on also...
150s	There was a man in the front row with a yellow hat on. I was on a stage in a room with a lot of people watching. I was on some sort of panel and writing notes on paper. I remember the man had a blue sweater/top on also. It was in France at ACM MM2019.



(a) 20191023_133612_000 (b) 20191023_140634_000

Figure 2.5: Examples of search target images of queries from Table 2.1. The ID shows its corresponding time and date with `YYYYmmdd_HHMMSS_000` format, where `Y, m, d, H, M, S` are year, month, day, hour, minute, and second, respectively.

2.1.4 Lifelog Retrieval Systems at LSC

In the first year of the LSC challenge, the competition was held with a total of six teams participating. Over the past years of organisation, the challenge has gained the attention of many research teams, with the number of participants increasing to 13 teams coming from different countries in 2023. A concise overview of the advanced techniques employed by participating systems in LSC'22 and LSC'23 is highlighted in Table 2.3. Through those challenges, the latest advancements in image retrieval and exploration are showcased to the research community, along with the progress made in improving image exploration and analysis. The key components of lifelog retrieval systems are Data Processing, Concept Search, Joint text-visual embedding methods, Query by Example, Relevance Feedback, and User Interfaceand Interaction, which are discussed in the following sub-sections.

2.1.4.1 Data Preprocessing

Prior to extracting visual features, lifelog images are preprocessed to remove noise and irrelevant information, followed by the image stabilisation [118, 5] to enhance the quality of egocentric images captured from wearable devices. Time is also provided corresponding to the image capture time. It is worth mentioning that the given

Table 2.3: List of participating systems and selected approaches used by them in LSC'22 and LSC'23. The symbol ✓ indicates that a method is used.

System info		Processing		Search					User Interface and Interaction							
	Country	Metadata Processing	Additional metadata	Concepts	Joint Embedding	Query-By-Example	Relevance Feedback	Other	Time Filter	Location Filter	Day Preview	Scene Clustering	Temporal Navigation	Map	Other
LifeSeeker [99, 100]	IE	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓		
FIRST [101, 102]	VN		✓	✓	✓	✓		✓	✓	✓	✓	✓	✓	✓	
LifeGraph [103]	CH	✓	✓	✓	✓				✓	✓		✓			
LifeInsight [104]	VN	✓		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	
Memento [105, 106]	IE			✓	✓	✓			✓	✓	✓	✓	✓		✓
MEMORIA [107, 108]	PT	✓	✓	✓	✓				✓	✓	✓	✓	✓		✓
vitrivr [109]	CH		✓	✓	✓	✓			✓	✓	✓				✓
Myscéal [6]	IE	✓		✓	✓	✓			✓	✓	✓	✓	✓	✓	
MyEachtra [110]	IE		✓	✓	✓	✓		✓	✓	✓	✓	✓	✓	✓	✓
vitrivr-VR [111, 112]	CH			✓	✓				✓	✓	✓		✓	✓	✓
Voxento [113, 114]	IE			✓	✓				✓	✓					✓
lifeXplore [115, 116]	AT		✓	✓	✓	✓			✓	✓	✓		✓		
MemoriEase [117]	IE	✓		✓	✓	✓		✓	✓	✓	✓	✓	✓		
LifeLens [7]	NO			✓	✓	✓			✓	✓	✓	✓	✓	✓	✓

time is fixed to the setup timezone of the camera, which is not always the same as the timezone of the lifelogger at that moment. Cases are even worse when the camera loses the connection to the GPS signal, which leads to the wrong timestamp or even missing the timestamp. Hence, a crucial step in the preprocessing stage is the process of data alignment, where the lifelog images are synchronised with the corresponding sensor data, such as GPS, accelerometer, and gyroscope, to provide more context for the retrieval process [119, 6, 105].

Different systems handle the data representation by transforming the given GPS coordinates into human-readable addresses. For instance, MEMORIA [108] utilised the Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN) [120] clustering algorithm to cluster the location data into meaningful places, while E-Myscéal [6] integrates Kikhia et al.'s [121] specialised location clustering technique. Taking a different approach, LifeSeeker [100] exploited the Google Map Geocoding API[5] to refine GPS data and augments it with pre-defined location categories, while LifeGraph [103] handled missing locations through a proximity-based algorithm, assigning unknown locations to the nearest known location within a time window. Building upon existing frameworks, MyEachtra inherited the semantic location names from VAISL [122], and LifeGraph queried Wikidata[6] to get the closest physical location to GPS coordinates. While other metadata such as heart rate, sleep patterns, and music preferences are also collected, their utility in general retrieval tasks has proven limited, though they might be valuable for personal record-keeping purposes.

2.1.4.2 Data Search Methods

To address the problem of locating the desired life event in the LSC competition, various research teams attempted to build real-time interactive systems based on visual concepts, while others exploited embedding models to bridge the semantic gap between text and image. Once the query is input, the system starts to match

[5] `http://developers.google.com/maps/documentation/geocoding`
[6] `https://www.wikidata.org`

the query with the aforementioned indexed concepts or embeddings. Then, a ranked list of relevant images is displayed to the user in descending order, which is calculated on the basis of every image-text pair similarity.

Visual Concepts Search

The most common technique to make lifelog data searchable and accessible in the early stages of lifelog retrieval is the concept-based approach. This technique's key idea is to transform lifelogging images into a collection of visual annotations, such as objects, colour, and text, prior to matching them with the input keywords. For the LSC challenge, the organisers provide two types of visual concepts, objects, and scenes. The objects[7] are detected using an object detection model trained on the COCO dataset [123] dataset and Microsoft's Vision API[8]. The scenes[9], including 102 labels (e.g., waiting in line, working, open area, etc.), are detected using a scene recognition model trained on the Places365 dataset [124]. Each detected concept includes bounding boxes and confidence scores to indicate detection reliability.

To enrich the lifelog content in the LSC dataset, data augmentation techniques are also applied to the images to extract additional metadata. Early works like LEMoRe [86] and BiDAL [95] focused on extracting low-level features such as histograms of oriented gradients (HOG) and color histograms. Modern approaches expanded content analysis with object detection and recognition capabilities. For instance, LifeXplore [116] further employed EfficientNet B2 [125] for semantic concepts extraction alongside the YOLOv7 [126] for object detection. In the meantime, FIRST implemented a comprehensive object detection approach using the combination of Faster RCNN [127], EfficientDet [128], and their own object detectors for items that appear in daily life activities. LifeSeeker combineed Bottom-up Attention model [129], YOLOv4 [130], and Faster R-CNN [127] with ResNet-101 [131].

[7] full list of object categories can be found at `https://github.com/amikelive/coco-labels/blob/master/coco-labels-2014_2017.txt`

[8] `https://azure.microsoft.com/en-us/products/ai-services/ai-vision/`

[9] full list of scene categories can be found at`https://github.com/CSAILVision/places365/blob/master/labels_sunattribute.txt`

Scene understanding represents another important aspect of visual content thanks to its ability to provide more valuable context for the image. Scene recognition was enhanced by both LifeSeeker [119] and MEMORIA [107] through their implementation of Places365 [124], which enabled the recognition and classification of 365 scene categories. Furthermore, text detection and recognition benefit users in extracting textual information from the images. FIRST [101] enriched the given concepts with Conceptual Captions [132] tags, while text detection and recognition were performed using CRAFT [133] in LifeXplore, Google Vision API[10] in LifeSeeker, and Hytext [134] in vitrivr. Moreover, vitrivr [5] also generated semantic captions for images through the attention-based model [135] trained on the COCO-2014 dataset [123]. LifeGraph [103] extracted common and named entities from the images using the Google Cloud Natural Language API[11], and also provided the synonyms of all concepts using an English lexical database called WordNet [136] and the NLTK Python package [137].

With the indexed concepts, the next step is to effectively match these concepts to the input queries. One of the most common ranking methods in information retrieval, particularly concept search, is the term weighting approach. TF-IDF (Term Frequency-Inverse Document Frequency) [138, 139] is used to represent the importance of terms indocuments based on their frequency. By doing so, rare concepts are given higher weights than common ones, highlighting the uniqueness of the content.

- *Term Frequency (TF)* counts the frequency of one term appearing in the document.
- *Inverse Document Frequency (IDF)* is the logarithm of the total number of documents divided by the number of documents containing the term.

[10] `http://cloud.google.com/vision/docs/ocr`
[11] `https://cloud.google.com/natural-language`

- *TF-IDF* is the production of *TF* and *IDF*, calculated as below:

$$\text{TF-IDF} = \text{TF} \times \text{IDF} = tf_{ij} \times \log\left(\frac{N}{df_i}\right) \tag{2.1}$$

  where tf_{ij} is the frequency of the term i in document j, df_i is the number of documents containing the term i at least once, and N is the total number of documents.

Apart from the term weighting method, which dominated early LSC challenge implementations, other concept-matching methods were also applied to rank the results. LifeGraph [103] took a different stance by looking at the internal relations of the data collected from multiple modalities, which were then connected into the large static knowledge databases, "Classification of Everyday Living" (COEL) and Wikidata, to provide more context for understanding the query. On top of the keyword-based search option, SomHunter [140] used the weighted self-organising maps (SOM) to offer a wide exploration of the result. Their new version for LSC'21 was enhanced by integrating an embedding model as an extra search engine in order to enrich the contextual understanding.

For better indexing and searching those concepts, research teams exploited databases to store and index the lifelog concepts. For instance, several top performing teams, including LifeSeeker [99, 119], Myscéal [6], LifeInsight [104], MemoriEase [117] have adopted Elasticsearch[12] as their primary indexing platform, leveraging its efficient concept matching capability. In the meantime, vitrivr [109] and vitrivr-VR shared the Cottontail DB [141] and Cineast [142] for specialised multimedia data management. LifeGraph [103] also used Cottontail DB [141] for its concept-managing database. MEMORIA [107] switched from PostgreSQL[13] in 2022 to Neo4j[14] in 2023 to better handle complex data relationships, while LifeXplore [115, 116] opted for MongoDB[15] database.

[12] `https://www.elastic.co/`
[13] `https://www.postgresql.org/`
[14] `https://neo4j.com/`
[15] `https://www.mongodb.com/`

Most concept-based searching tools, including our system, rely on the analysis of both visual and non-visual content. Concerning visual content, we try to extract the utmost valuable features from the images by leveraging techniques in object detection, text recognition, and scene recognition, while non-visual content is augmented with additional information from external sources. The embedding model would be a considerable choice to expand the image understanding by giving more insights into the relationship between visual and contextual concepts. For this specific search approach, we regard non-visual content as textual features, which are then input to approaches from the document retrieval area.

Joint Embedding Search

Though some teams have explored the embedding models in the early stages of the LSC challenge [143, 144] to embed image and text separately, the embedding models have not become recognised useful tools for lifelog retrieval until the introduction of joint visual-and-textual embedding models. These embedding models have posed a new direction in information retrieval and lifelog retrieval specifically. Contrastive Language-Image Pre-training (CLIP) [145], developed by the OpenAI team, is a multimodal embedding model that learns the relation between visual and semantic concepts of the scene. Particularly, the model is trained on a large-scale dataset of images and text pairs to learn visual representations associated with the text descriptions. With the zero-shot transferability, CLIP has been widely used for different tasks ranging from self-supervised learning [146], action recognition [147] to image captioning [148]. In the lifelog area, there have been several teams [149, 140] who applied CLIP as part of their search mechanism, yielding good results at LSC'21. Instead of using keywords associated with the content, both lifelog images and text queries are embedded into the same high-dimensional latent space with the use of such models. Then, the similarity between those feature vectors is measured and ranked in descending order. Proving its effectiveness throughout the competition by reducing the semantic gap of the visual-and-textual relation, other teams have also deployed this model as their underlying search engine, and ours is no exception [6,

113, 7, 105, 119, 108]. As the field progressed, more variants of the joint-embedding models, such as OpenCLIP [150], and BLIP [151] have also been integrated into some teams' systems to enhance the search process. There are three main versions of the OpenCLIP with different vision transformer [152] have been utilised across teams, including ViT H/14 trained on the LAION-2B dataset [153], ViT L/14@336 trained on the LAION-5B dataset [154], and ViT-B/32 xlm roberta trained on LAION-5B [155]. In the LSC'22 and LSC'23, almost participating teams exploited these models as their underlying search engine to enhance the retrieval performance, the specific model versions used by each team are summarised in Table 2.4.

Table 2.4: Employed joint text-visual embedding models by the participant teams in LSC'22 and LSC'23.

Model	**System**
CLIP [156, 145]	EMyscéal [6] Voxento 2.0 [113] LifeLens [7]
CLIP ViT-B/32 [156, 145]	Memento 2.0 [105] LifeSeeker 4.0 [119] MEMORIA [108]
CLIP ViT-L/14 [156, 145]	Memento 2.0 [105] Voxento 3.0 [114]
OpenCLIP ViT-H/14 trained on LAION-2B [153, 150]	MyEachtra [110] LifeXplore [116] LifeGraph [103] vitrivr [109]
OpenCLIP ViT-L/14@336 trained on LAION-5B [154, 150]	ELifeSeeker [100]
OpenCLIP ViT-B/32 xlm roberta trained on LAION-5B [155, 150]	vitrivr-VR [112]
BLIP [157, 151]	LifeInsight [104] MemoriEase [117]
ensembled model	Memento 2.0 [105] Memento 3.0 [106]
custom model	FIRST [102] vitrivr [109] vitrivr-VR [112]

Apart from inheriting those CLIP-based models, teams make efforts to improve the retrieval performance by either introducing additional modules or customising the existing models. For example, Memento 3.0 [106] adopted an approach in which the model can be switched between 9 different versions of embedding models, including CLIP [145], OpenCLIP [150], and their weighted ensembled models to enhance the retrieval performance. vitrivr and vitrivr-VR utilised the approach [158] similar to W2VV++ [159] to embed lifelog data, while FIRST [101] introduced a module to extract more subtle and local visual features, which decomposed the image into multiple sub-regions and generated an adaptive CLIP-based embedding set corresponding to that image.

Semantic similarity matching is another approach to measuring the similarity between the input query and visual content. Specifically, it calculates the similarity between two feature vectors in the same latent space using the Cosine similarity. Basically, the similarity score is measured for each pair-wise combination of the query feature and the image feature, and the results in descending order. The higher the similarity score, the more relevant the image is to the query. To address the computational challenges inherent in processing high-dimensional embedding spaces, some teams leverage FAISS (Facebook AI Similarity Search) [160], an open-source vector library, to perform the similarity search in the high-dimensional space [105, 116, 101, 114]. Other used Milvus [161] vector database, an enhanced vector database built upon FAISS's core technology, to store and index the embeddings [104, 100].

Query-By-Examples

Query-By-Example (or visual similarity) functionality allows users to use a frame as a query to discover visually or semantically similar content, which is useful for the scenario where users either want to search for visually similar images or have one frame in mind but struggle to describe it in words. These systems [105, 104, 7] leveraged the same feature extraction mechanisms employed in text-to-image search, typically utilising visual transformer components from multimodal models like CLIP

to compute image-to-image similarity metrics. The score between pairs of images was calculated and ranked similarly to the text-to-image search.

Relevance Feedback

Relevance feedback refers to the process of refining the search results based on the user's feedback, enabling iterative refinement of the search results based on user input. Depending on the user's demand, the system adjusts the search results by either expanding the search space to include more results or narrowing down it to focus on more specific matches. LifeInsight [104] used the Rocchio algorithm [162] to dynamically generate and refine query vectors, either creating initial search parameters or modifying existing query vectors for enhanced search and result re-ranking operations. Sharing a similar idea, LifeSeeker [100] collected the user feedback via automatically generated yes/no questions, building upon the visual concepts. This binary feedback mechanism expects to quickly eliminate irrelevant results and support users to focus on content that matches their search intent.

Other search methods

Beyond the aforementioned search methods, research teams also implemented innovative additional search modules to enhance the retrieval performance. FIRST [101] integrated a Visual Examples module, shared with LifeInsight [104], using external sources to extend the search space. This module first queried the Internet sources for relevant input concepts, then used these images as references to search through the lifelog data. LifeInsight [104] further innovated by implementing an AI-assisted crowd-sourcing mode that runs multiple parallel retrieval processes with different description variants, followed by a voting scheme to re-rank candidate moments based on collective results. MyEachtra [110] took a specialised approach by integrating an additional module for the QA task using the FrozenBiLM [163]. This model took the question directly as a prompt query and then returned the potential textual answers.

2.1.4.3 User interface and interaction

Offering an interactive user interface is undeniably crucial for the success of a lifelog retrieval system, as it acts as a bridge between the user and the system. The more intuitive and user-friendly the interface is, the easier the user can interact with the system to find the results. Many teams have put a lot of effort into designing a web-based interface that has a variety of functionalities to support the retrieval process. The foundation of these interfaces includes essential components such as a search bar to input the query, an organised display of search results, and fundamental filtering options for metadata refinement. Developing over the years, participant teams have come up with innovative ideas to enhance their user interface and interaction. They equip users with various functionalities to interact with the lifelog data, as shown in Table 2.3.

Time filters and location filters are the two most common functions that are provided by all systems to help users narrow down the search scope based on the time and location. Users can either input keywords or type text queries, and then the system matches those keywords to the concept collection to eliminate irrelevant frames. LifeSeeker and E-Myscéal implemented natural language processing techniques to automatically extract temporal and geographical patterns from natural language queries. The result display is also an important part of the UI, where the images are shown in a list or grid view, with the option to view the image in full screen. Teams also provided various options to view the results, such as the sequence view, event view, or in clusters [6, 7, 117, 100]. The sequence view displays the images in chronological order, while the event view groups the images based on the event they belong to, and the cluster view groups the images based on the similarity of the visual content. They can present all images or choose some representative images to display for each event or cluster. Temporal navigation offers users the ability to explore the lifelog results based on the time range, especially useful for the situation of multiple events happening consecutively, and

users might have partial knowledge of a timeline and need to explore temporally surrounding moments.

Other functionalities

Conventionally, the search modality in lifelog search engines is *text-based search*, where users input queries in the form of either keywords or natural language. Additionally, the search modality has been extended to other forms, such as sketch-based, map-based, or calendar-based search, to enhance the user experience.

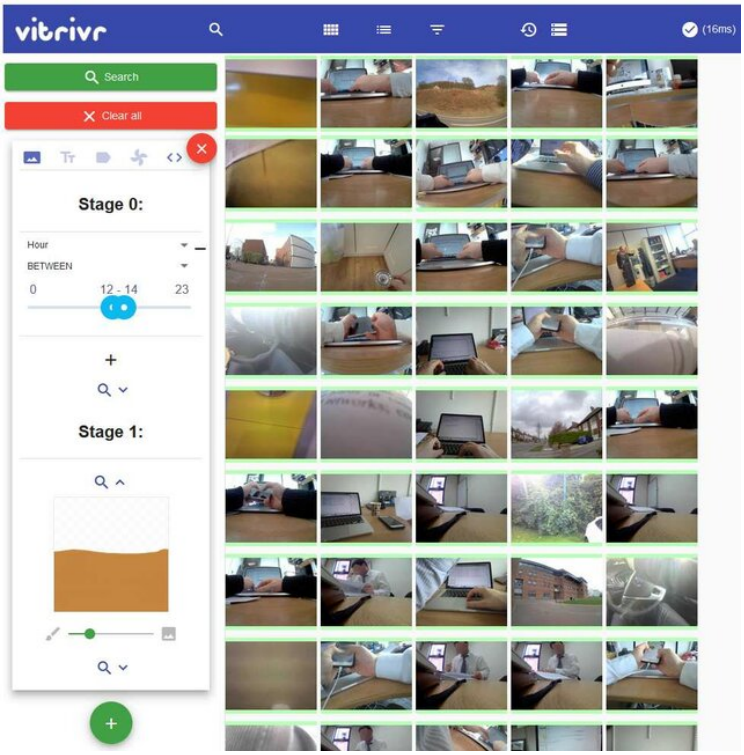
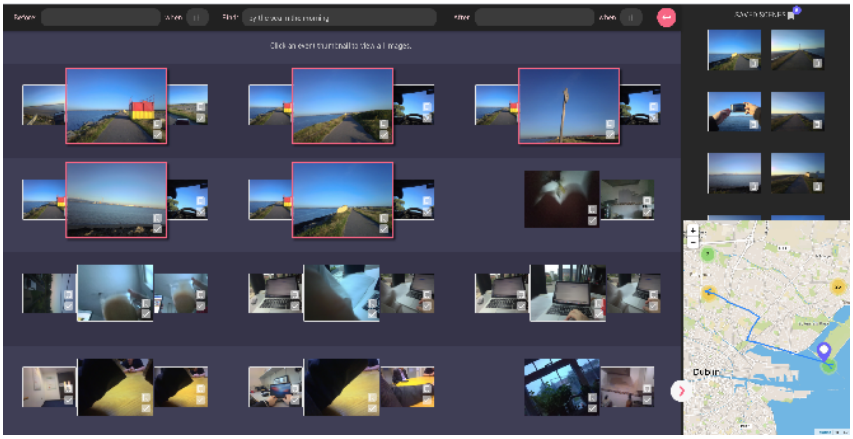
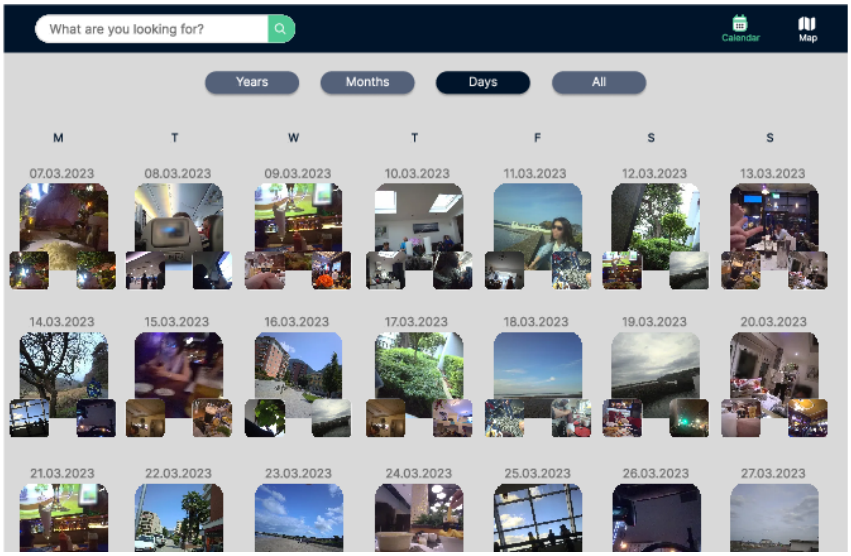


Figure 2.6: The User Interface of vitrivr [5] system when sketching a query.

For situations where users are unable to describe the scene in words, or they might have the scene in their mind but cannot express it in text, the *sketch-based query* is a good alternative. The system allows users to draw a sketch of the scene they are looking for, and the system then return the most relevant images based on the sketch. Inspired by the works [164, 165, 166] from Video Browser Showdown, Vitrivr [109] and vitrivr-VR [112] facilitated the browsing process by providing multiple search modalities via sketches and audio, then retrieved by the Cineast [142] engine. An



(a) E-Myscéal system's User Interface with map view [6]



(b) LifeLens system's User Interface with calendar view [7]

Figure 2.7: The User Interface of Myscéal [6] and LifeLens [7].

example sketch-based query is shown in Figure 2.6 when users draw a brown region in the bottom half of the canvas to find images with similar ground or surface coloring. While this function is a promising approach to enhancing the user experience by matching exactly with their demand, it is not widely used because the real-world images sometimes are not related to sketches as imagined by the users.

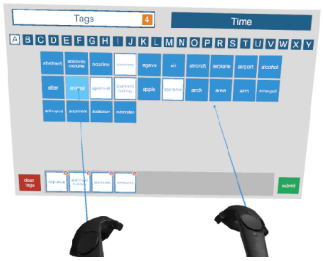
To avoid the usual display problem of duplicate images displayed in the search results, *scene clustering* is implemented by some teams to group similar images into clusters and then display some representation frames for each cluster. While some teams [102, 6, 103, 110, 108, 117] applied the event segmentation in advance, others

[100, 7] processed the clustering as a post-processing step. For instance, LifeGraph constructed the graph using multiple different clustering criteria (temporal, spatial, visual, or activity), FIRST clustered the images through heuristics on CLIP features and geographical proximity, and LifeSeeker [100] implemented a clustering algorithm to group images within part of the day into clusters, then averaged the relevant score for each cluster.

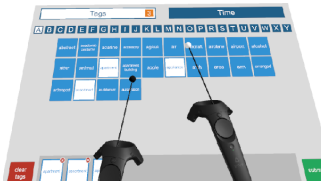
Subsequently, the *map-based view* and *calendar view* are also developed as alternative ways to review and filter out the results based on the location and time, examples shown in Figure 2.7. The system displays the map with the lifelog images pinned on the map, allowing users to opt for some specific area on the map to explore the images in that area [6, 110, 109, 7, 102, 104]. In the meantime, the calendar view helps users navigate the lifelog data based on the time range. The system displays the calendar with the dates that the lifelogger captured the images, and users can select the date to view the images captured on that day [106, 7]. vitrivr [109] also equipped users with the *color-based search* functionality, where users can select the color they want to search for, and the system returns the images with the most dominant colors.

Duane et al. [167] brought their system into the Virtual Reality (VR) space, where users can emerge themselves in the 360-degree view of the lifelogging world, with the UI design shown in Figure 2.8. Vitrivr-VR [9, 112] designed their system's user interface in the VR space. The interactive retrieval process, with illustration in Figure 2.9, was facilitated with several VR-related functionalities, such as an interactive map for spatial query formulation, a sequence image view for browsing neighboring images (Figure 2.9b), and a cylindrical results view for result exploration (Figure 2.9a). Other teams [8, 168, 169, 170] also exploited the VR technology to offer a more immersive experience for users to interact with the lifelog data.

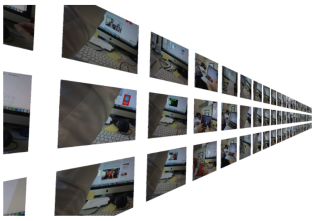
Coupled with using the traditional text-based search, many teams varied their means of inputting the query. Voxento [113] utilised voice control as their main modality to navigate the tool, which supports retrieval by providing users with a



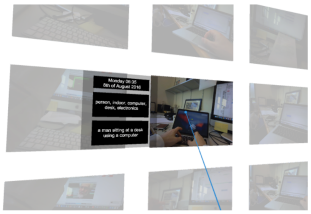
(a) Distance-based User Interaction



(b) Contact-based User Interaction



(c) A VR Ranked List



(d) Image Metadata

Figure 2.8: The User Interface of the VRLE system, as reported in [8]

(a) Cylindrical view of the results

(b) Sequence view of the results

Figure 2.9: The User Interface of the Vitrivr-VR system, as reported in [9]

list of voice commands and interactions with the help of the Google Web Speech API [171]. This API facilitated speech-to-text conversation using the Chrome browser. The authors made an improvement in an automatic speech recognition, called Whisper API [172], to enhance the voice control experience.

2.1.4.4 Discussion

As previously mentioned, our emphasis is on the lifelog retrieval task, which refers to the process of locating the desired events in the lifelog collection based on the user's query. Building a lifelog retrieval system requires a combination of various components, including visual processing, data indexing, retrieval methods, and a user interface. Through the development of the lifelog retrieval systems in the LSC challenges, we have observed a significantly improvement in both their architecture and functionalities to enhance the retrieval performance and user experience over time, as summarised in Table 2.3. We learned that the embedding models have become a more popular option for the core search engine, as they can bridge the semantic gap between visual and textual content. Meanwhile, the concept-based search is still a reliable method to filter out the irrelevant images based on the visual concepts. Alternative search modalities, such as sketch-based, map-based, and calendar-based search, are also used across teams to provide users with more options to interact with the lifelog data. The user interface and interaction are also crucial for the success of the system, as they provide users with a more intuitive and interactive way to interact with the lifelog data. Drawing from these insights and our experiences in LSC challenges over the years, we developed our system to incorporate visual processing with multiple object detection models, an embedding-based semantic model for improving cross-modal understanding, and a user-friendly interface with various functionalities to enhance the user experience.

2.2 Video Moment Retrieval

As compared to the lifelog retrieval task, the Video Moment Retrieval task is described more closely as the real situation of finding a slice of time in a video yet still challenging. The task aims to identify an event time interval semantically matching a given text query, which requires a deep understanding of semantic relations between visual and textual content. Throughout years of development,

there are two main types of VMR approaches: proposal-based VMR [48, 50, 54, 173, 49, 174, 175] and proposal-free VMR [176, 52, 177, 178, 56, 179, 58, 57, 180]. While the former measures the similarity between the pre-segmented clip proposals and the text query, the latter predicts the probabilities of each frame being the event boundaries based on their high dimensional representation.

2.2.1 Proposal-based VMR techniques

To get a closer look at the proposal-based VMR, I discuss the existing methods that have been proposed in the literature. The dominant paradigm in proposal-based VMR involves a two-stage architecture: moment segmentation followed by relevance ranking. These methods first convert the input long video into small clips as proposal candidates and then sort them based on their relevance to the given text description. Implementing a multi-scale temporal sliding window strategy, the existing methods [48, 50, 173, 175] converted input videos into a set of candidate segments at different temporal scales. The sliding window parameters, including window sizes and stride lengths, strongly influence the selection process. Gao et al. [48] proposed Cross-modal Temporal Regression Localiser (CTRL) to jointly model visual features extracted from pre-trained C3D models and text features, while VLG-Net [173] adopted Syntactic Graph Convolution Networks (SyntacGCN) to encode video and sentence embeddings, followed by a graph matching layer. MMRG [50] captured multi-modal relational graphs of the visual and textual content from the proposals. Meanwhile, FVMR [54] generated moment proposals utilising 2-dimensional maps [175], indicating the moment's start and end times. Using handcrafted heuristics, TCMN [181] generated the proposals based on temporal dependencies between events in the query and the video.

However, despite these advances, proposal-based methods still face challenges in choosing the target candidates. The models rely heavily on the moment candidates' boundary accuracy, meaning that errors in the proposal generation stage can lead

to incorrect retrieval results. Additionally, the annotation process for training such models may require substantial human effort and may introduce subjective bias, as different annotators may have different opinions on the event boundaries. This also makes it more challenging to scale the models to long videos or large-scale datasets due to the high computational cost of generating proposals.

2.2.2 Proposal-free VMR techniques

On the other hand, the proposal-free VMR incorporates a moment generation stage and a moment localisation stage into a single, end-to-end trainable module that directly predicts the start and end times of a moment without the need to pre-segment the proposals. UVCOM [179] addressed the VMR task with a Comprehensive Integration Module (CIM) designed to achieve intra- and inter-modality interaction across multi-granularity. As a result, the model improved the video's understanding by recognising both local relationships and global knowledge accumulation throughout the entire video. Furthermore, LGI [177] took extracted video and text embeddings as input and applied their local-global video-text interaction models in three levels (segment-level fusion, local context modeling, and global context modeling). By doing so, the authors were able to capture an in-depth relationship between video and query and output the moment predictions.

Vision Transformer

Transformer architecture, first introduced by Vaswani et al. [10], presents a powerful tool for modeling long sequences and capturing the dependencies between elements. Unlike the traditional Recurrent Neural Networks (RNNs) and Long-Short Term Memory (LSTM) networks, transformers are able to process the entire long sequence simultaneously through the attention mechanisms, making a significant advancement in Natural Language Processing domain. This lead to the development of the Vision Transformer (ViT) [152] which successfully adapted the

transformer architecture to the Computer Vision (CV) field with promising results in various tasks, ranging from image recognition, object detection, semantic segmentation, and video understanding. The ViT implements an distinctive approach to processing images by first dividing the input image into fixed-size patches and then flattening these patches into a sequence of vectors. These vectors are then fed into a transformer encoder to capture the global dependencies between the patches.

With the success of transformers in the CV domain, recent works have applied this architecture to the VMR task. Based on the concept proposed by DETR [182] for object detection, recent works [56, 58, 57] resolved the VMR task by leveraging an encoder-decoder transformer architecture that considers the VMR task as a direct set of prediction problems. Concretely, the authors leverage a transformer module together with three different heads (including saliency, fore/background, and moment coordinate heads) to predict the moments. UnLoc [178] introduced a unified framework that exploits the large-scale pre-trained model, CLIP [145] for video understanding. Particularly, the authors leveraged the combination of a two-tower architecture (vision and text encoders) and a mid-level video-text fusion module to regress the moment boundaries. Inspired by those works, our framework is constructed as a proposal-free VMR framework that leverages the power of transformer architecture [10] in capturing long sequences to identify moment boundaries.

Despite the undeniable role of encoding low-level visual features to represent visual content, little effort has been made to explore the impact of high-level relational semantic cues for the VMR task. For that reason, we propose to jointly learn from both aforementioned aspects by using a parallel transformer architecture including a global stream (with visual embedding representations) and a relational stream (with scene graph representations). To this end, we hypothesise that this approach can better exploit not only the visual meaning but also the semantic relations between the query and the video. Particularly, the global stream

encodes the overall visual content of each video frame with the pre-trained embedding models such as CLIP [145] or I3D [183]. Meanwhile, the relational stream models the fine-grained details and relationships among different objects in the frame via the scene graph generation model. Additionally, each stream is fed into a transformer module to learn their representations before ensembling the features of both streams for prediction.

2.3 Event-based Approaches

As discussed in Section 1.2, a common definition of an event is stated as "*a segment of time at a given location that is conceived by an observer to have a beginning and an end*" [60]. Structuring a continuous egocentric data stream into manageable units, called events, could help eliminate redundant images and highlight the important ones. There are many prior works concerning solving this problem by either the use of audio, temporal, sensor, or visual data.

Ellis and Lee [184] leveraged the Bayesian information criteria model to segment data captured from low-cost commercial portable audio equipment into distinct events that aim to create an "*automatic diary*" of daily life activities. Similarly, other studies addressed the temporal segmentation problem by using CNN approaches to extract both contextual and semantic information [185, 186]. Moreover, the time-constrained K-means clustering approaches are integrated to deal with personal video recordings [187] [188]. Doherty and Smeaton's work in 2008 treated sensor data (movement, light intensity from a wearable device) as a means of activity change in detecting event boundaries. Event-based approaches have also been implemented in lifelog retrieval tools since the early days of lifelogging. A study proposed a prototype of Hearst's text-tiling technique on descriptions obtained from lifelog images, MPEG-7 (colour layout, colour structure, scalable colour, and edge histogram) [71]. Precisely, they determined a threshold value based on the dissimilarity between feature vectors with the aim of

distinguishing a sequence of images or an event from others. On the other hand, the use of high-level sources of features, such as visual concepts and image categorisation (objects and activities), proved to be better in solving the event segmentation problem for lifelog data rather than using the low-level ones [189, 190, 191]. Recently, Tran et al. [110] proposed an approach to define event boundaries based on the combination of the difference between semantic location, the cosine similarity of visual features, and the time differences between images.

Once distinct events are segmented from the first-person image sequences, selecting the appropriate keyframe to represent the entire event becomes crucial. Conventionally, the keyframe is chosen from a fixed position (first, middle, or last) for all events in order to simplify and process information quickly [192]. Doherty et al. [73] proposed a keyframe selection technique in which the representative keyframes were chosen based on the quality of the frame in terms of contrast, colour variance, global sharpness, noise, saliency, and some external sensor data. Alongside the image quality, Photchara et al. [193] extracted users' excitement by exploiting physiological responses in skin conductance captured from wearable bio-sensors. Those egocentric images that meet the requirements for both image quality and emotional impacts are selected as keyframes.

Although there are a small number of research projects focusing on event-based approaches for lifelog retrieval, as discussed in Section 2.3, most of them are proposal-based approaches. As mentioned above, this requires a lot of effort to pre-segment and annotate the appropriate moments. In this dissertation, I aim to explore the proposal-free VMR technique due to its advantages over the current lifelog retrieval systems. With this attempt, I aim to investigate the usability of the event-based lifelog search tools in assisting human recollection. To the best of my knowledge, this is the first research that explores the video-moment-retrieval-based approaches in lifelog retrieval to closely align the way humans access and retrieve their memory.

2.4 Conclusion

This chapter presented a comprehensive literature review of lifelogging, lifelog retrieval benchmarks, and lifelog retrieval systems, event-based approaches in lifelog retrieval, as well as the state-of-the-art techniques in video moment retrieval. Through this analysis, a significant gap in current research approaches was identified: existing lifelog retrieval systems are mostly developed on the concept of standalone images as the basic unit of retrieval, which is not in line with the human cognitive system. This observation has motivated our research direction toward exploring moment-based approaches in lifelog retrieval, with the goal of better mimicking the way humans access and retrieve their memory. The exploration of video moment retrieval techniques offers valuable insights for this transition, as these techniques are designed to identify a segment of time in a video that semantically matches a given text query, which is closely related to the natural recall patterns of human memory. This provides a promising foundation for adapting similar principles to lifelog data. The research methodology, operating constraints, and evaluation methods used in this dissertation are described in the following chapter.

Chapter 3

Research Methodology and Evaluation Methods

This chapter outlines the research methodology employed to address the proposed research questions and test our hypothesis. They are structured as follows: Section 3.1 describes the research methodology, Section 3.2 outlines the operating constraints, and Section 3.3 presents the evaluation metrics.

3.1 Research Methodology

The research methodology is defined as systematically developing, implementing, and evaluating a novel approach to solve a research problem. In order to answer the research questions, we have adopted the research methodology framework [194] which is depicted in Figure 3.1.

The research methodology framework begins with the problem identification and formulation in the research field, which is lifelog retrieval. Following this initial phase, we conduct an extensive literature review to understand the state-of-the-art methods and techniques that have been used to address the problem. This comprehensive review revealed significant limitations of existing approaches in lifelogging domain, particularly their reliance on concept-based retrieval methods and the use of single images as the primary unit of retrieval. These limitations become particularly apparent when considering the complexity of human memory retrieval processes, which naturally operate at the level of moments or episodes

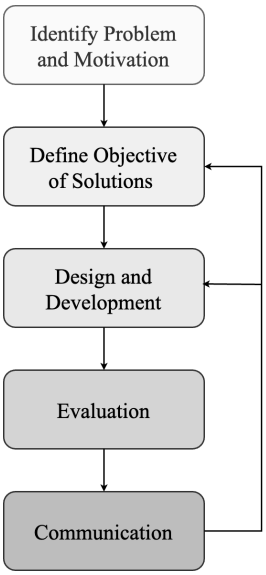


Figure 3.1: The research methodology framework.

rather than discrete images. Furthermore, we observed that lifelog data inherently exists as a continuous stream of visual information, sharing a similar format with video data rather than isolated images. Adopting advanced video moment retrieval techniques for lifelog data could potentially improve the performance of retrieving past experiences. Therefore, I formulate the research hypothesis as *The integration of transformer-based techniques into a moment-based lifelog retrieval approach enhances the performance of retrieving one's past experiences when compared to the conventional image-based lifelog retrieval systems.* Three research questions are then formulated to investigate this hypothesis. In order to answer these research questions, we propose to develop a novel lifelog retrieval system based on the video moment retrieval techniques to the lifelog data to improve the retrieval performance. Additionally, to simulate the way humans retrieve their memory, we propose to consider moments as a unit of retrieval instead of a single image. The final step is to evaluate the proposed methods and compare them with the state-of-the-art methods.

Following the research methodology in Figure 3.1, we conducted several experiments to address the research questions formulated in Section 1.3 as follows:

- **RQ1**: Our approach to RQ1 follows an iterative development methodology to develop and validate an interactive lifelog retrieval system through multiple evaluation cycles. This methodology employs a process of design, implementation, real-world testing, and refinement, where each iteration involves selecting suitable technologies and search functionalities, and designing the user interface to participate in the Lifelog Search Challenge (LSC) and NTCIR Lifelog Challenge. Each development cycle follows a structured approach: (1) analysis of user feedback and previous performance limitations identified through challenge participation, (2) identification of specific improvement areas based on competitive results (3) targeted implementation of technological and interface enhancements, (4) evaluation through user experiment, previous challenge, and new competition. The detailed development of the system, including architecture, user interface design, and retrieval mechanisms, along with the performance results of each participation in the challenges, is presented in Chapter 4, which helps me answer RQ1.

- **RQ2**: The second research question builds upon the memory retrieval theory to address the current limitations of lifelog retrieval systems. We identified the need to consider moments as a unit of retrieval instead of a single image because humans tend to recall their past experiences in moments rather than individual images. To address this, we propose a novel framework that integrates transformer-based techniques to retrieve moments from video data. The methodology follows a structured three-phase approach: (1) comprehensive analysis of existing VMR techniques to identify the research gap, (2) systematic development of the novel framework to solve the VMR task, and (3) evaluation using video moment dataset (Charades-STA) to validate the framework's performance. With this novel framework development and its comprehensive evaluation of the Charades-STA dataset, as described in Chapter 5, we determined the potential of the proposed

approach to improve the performance of retrieving moments from video data, which helps to answer RQ2.

- **RQ3**: The third research question covered the development of a novel moment-based lifelog retrieval system, addressing a significant gap in existing approaches where only a few lifelog retrieval systems had considered moments as the primary unit of retrieval. Building upon the insights gained from the previous research questions, we adapted the advanced transformer-based techniques to lifelog data to improve the performance of searching past experiences. The framework addresses the key challenge of scaling moment retrieval from single videos to large-scale lifelog collections through a structured two-stage approach: (1) analysis of the adaption requirements for VMR techniques to lifelog data, (2) implementation of the adapted approach, developing PaTFLifelog system as a hierarchical two-stage system (period retrieval followed by moment boundary detection), (3) validation and comparable evaluation against the baseline system. The development process, integration methodology, and comparative analysis results are detailed in Chapter 6, demonstrating how this novel approach improves upon conventional image-based retrieval methods by providing contextually rich moments that align with natural human memory processes. In this way, we fulfilled the RQ3 as well as the hypothesis.

3.2 Operating Constraints

In this dissertation, we clarify the operating constraints that may affect the research outcomes. The constraints are as follows:

1. This research was constrained by the time limit of the PhD study, which was estimated to be four years to complete the research cycle. Additionally, the time allocation had to account for participation in annual Lifelog Search Challenge events, which served as key evaluation milestones for the developed

systems.

2. The research utilised lifelog data collected from a single lifelogger over a specific time period, as provided for the LSC challenges.
3. The queries at the challenges are generated by the lifelogger, or those who are authorised by the lifelogger to access the data, not subject to influence by traditional information retrieval expectations.
4. The comparative evaluation of lifelog data must respect the laws and data governance laws of the jurisdiction in which the research is performed and ethical approval must be sought from the research institution. All research procedures must be reviewed and approved by the research institution's ethics committee.
5. Since no new lifelog data was collected during the research, the performance of the systems was constrained by the quality of the existing data collections. This included the limitations in image quality, completeness of metadata, accuracy of timestamps, and any gaps in the lifelog data.
6. The baseline lifelog retrieval system constructed from Research Question 1 is reasonable for the SOTA system as evaluated by its performance in several lifelog retrieval challenges. The performance of this baseline system in competitive challenges validated its status as a SOTA reference point for comparison.
7. The performances of the baseline system (in RQ1) and the proposed system (in RQ3) are compared and evaluated based on the same set of queries and lifelog data in terms of Hit Rate at k and Mean Average Precision.

These constraints are maintained for this thesis and act as limiting factors to focus the research efforts on the proposed research questions.

3.3 Evaluation Metrics

The evaluation metrics are used to measure the performance of the proposed system, each of them is adapted to the appropriate task. In this work, we consider several evaluation metrics in information retrieval and lifelog retrieval (including LSC Interactive Scoring Metrics, Precision and Recall, Recall at k, and Hit Rate at k), and video moment retrieval (including Recall at k with IoU). The evaluation criteria are described in detail in the following sections.

3.3.1 LSC Interactive Retrieval Metrics

For the specific LSC challenge, the ultimate goal is to retrieve the relevant lifelog image that matches a given query as fast and accurately as possible. Hence, the LSC organisers have employed different evaluation metrics tailored to different task types, following the scoring scheme in Video Browser Showdown (VBS) [195]. They are adopted and modified to better suit the lifelog retrieval task, which measures both the speed and accuracy of interactive lifelog retrieval systems. The challenge was conducted in an interactive manner, which means that one user used the system to perform the search and submit the image that they thought best illustrated the query.

For KIS and QA tasks, the score [80] of one LSC participant retrieving the correct answer at the time t is officially calculated as follows:

$$S_i = \max\left(0, M + \frac{D-t}{D} \times (100 - M) - W \times 10\right) \tag{3.1}$$

where M refers to the minimum score earned, D denotes the query's duration and W represents the number of wrong submissions for each query. For each task, these numbers can be different. Specific to this case, M is set to 50, while D is set to 180 for QA tasks and 300 for KIS tasks. As can be seen from the formula above, within the time constraints, the score is linearly decreased until the minimum score (50).

Then the final score is taken by subtracting each negative submission by 10 points. The evaluation process differs between the KIS and QA tasks, as the KIS tasks are evaluated automatically by the system since they already have a clear correct answer set, while the QA tasks require human judges due to the nature of text-based answers that are subjective and can be flexibly interpreted in some cases.

Regarding non-ground-truth tasks like Ad-hoc tasks, the evaluation metric follows a different approach based on the performance of the team across all the teams. With human judges evaluating the relevance of the submitted images to the query in real-time, the ADS task score is calculated using the following formula:

$$S_i = 100 \times \frac{C}{C + W/2} \times \frac{C}{N} \tag{3.2}$$

where C and W represent the number of correct submissions and wrong submissions, respectively, and N is the total number of relevant images in the ground-truth pool across all teams. The number of wrong submissions is divided by 2 to reduce the penalty for wrong submissions (as compared to KIS/QA tasks) because ADS tasks allow unlimited submissions. By using this formula, users are encouraged to explore more while still being penalised for incorrect submissions.

If the participant fails to find any correct answers within the time limit, they receive a zero score. The score of each task will then be normalised to the range of $[0, 100]$ to facilitate the comparison between different participants. The final score for each task category (KIS, QA, and ADS) is calculated by first averaging the scores of all queries within that category, then normalising this average score to the range of $[0, 100]$.

3.3.2 Precision and Recall

Alongside the LSC score, we also consider the Precision and Recall metrics to evaluate the performance of the proposed system. Precision and Recall are standard measurements in many applications, including information retrieval,

machine learning, and computer vision. They measure the relevance of the retrieved items and the completeness of the retrieval, respectively. The Precision and Recall are calculated based on the number of True Positives (TP), False Positives (FP), and False Negatives (FN) as shown in Table 3.1.

- True Positives (TP): the number of relevant images that are correctly retrieved.
- False Positives (FP): the number of irrelevant images that are incorrectly retrieved.
- False Negatives (FN): the number of relevant images that are not retrieved

Table 3.1: Precision and Recall

	Relevant	Non-relevant
Retrieved	True Positive (TP)	False Positive (FP)
Not Retrieved	False Negative (FN)	True Negative (TN)

While Precision measures the proportion of relevant images among the retrieved images, Recall measures the proportion of relevant images that are successfully retrieved across all relevant images. With the definitions above, Precision and Recall are calculated as below:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{3.3}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{3.4}$$

Although Precision and Recall are useful metrics for evaluating the performance of the retrieval system, they do not consider the ranking of the retrieved items, considering all items equally relevant regardless of their ranking. Moreover, both metrics are sensitive to the selection of two parameter: the cut-off rank level k and the total number of relevant items in the dataset N_q. In particular, the $P@k$ cannot achieve perfect precision ($P@k = 1$) when the k is larger than the total number of relevant items in the dataset. Similarly, the perfect $R@k$ can not be reached if the total number of relevant items is larger than k.

Therefore, we also consider Mean Average Precision (MAP), which addresses these limitations by incorporating both precision and ranking position into a single metric. MAP calculates the mean of the Average Precision (AP) scores over all queries, where AP for each query is the average of $p(r)$ in precision-recall curve over the interval from $r = 0$ to $r = 1$. By doing so, it puts the ranking order of retrieved items into consideration. If we generate a list of precision-recall values based on different chosen k and draw a curve to represent every precision-recall dots for top k retrieved items, this curve is called precision-recall curve. This curve plots precision $p(r)$ as a function of recall r. Average precision is the average value of all precision values. In Equation 3.5, $P(k)$ is the precision of the I_k in the retrieved item list, and $rel(k)$ is 1 if I_k is relevant, otherwise, it is 0.

$$\text{AP} = \frac{1}{|I_{rel}|} \cdot \sum_{k=1}^{n} P(k) * rel(k) \tag{3.5}$$

$$rel(k) = \begin{cases} 1 & \text{if } I_k \text{ is relevant} \\ 0 & \text{if } I_k \text{ is not relevant} \end{cases} \tag{3.6}$$

$$\text{MAP(Q)} = \frac{1}{Q} \cdot \sum_{q} AP(q) \tag{3.7}$$

3.3.3 Automatic Retrieval Metrics

In order to validate the performance of the backend core of the retrieval system, the Hit Rate at k ($H@k$) was considered as one of the evaluation metrics. It measures the possibility of the top-k results containing at least one relevant result. This metric aligns with the scoring scheme in the LSC, where the participants need to identify and submit one correct image to solve the task. The high $H@k$ indicates that users have a high chance of finding at least one relevant image in the top-k results.

$$H@k = \frac{1}{N_q} \sum_{i=1}^{N_q} h(k, q_i) \tag{3.8}$$

where N_q is the total number of testing queries, and $h(k, q_i)$ is a binary function that returns 1 if at least one of the top-k retrieved images is relevant to the query q_i, and 0 otherwise.

3.3.4 Recall at k with IoU for VMR task

The ("$R@k - IoU = v$") metric, proposed in TALL [48], is adopted as an evaluation criterion for specifically measuring the Video Moment Retrieval (VMR) task. Particularly, it determines whether at least one of top-k selected moments for the query q whose IoU is larger than v [48]. As this metric is obtained on a query level, the overall performance will be the average value across all queries, denoted as follows:

$$R(k, v) = \frac{1}{N_q} \sum_{i=1}^{N_q} r(k, v, q_i), \tag{3.9}$$

where the $r(k, v, q_i)$ is defined as a binary function that returns 1 if at least one of the top-k selected moments of the query q_i has $IoU > v$, and 0 otherwise. N_q is the total number of testing queries. It is worth noting that the Recall at k does not consider the ranking of the retrieved items but only the presence of one relevant item existing in the top-k results.

Intersection over Union (IoU), also referred to as the Jaccard Index, is an evaluation metric that measures the overlap between the ground-truths and the predictions. The IoU is calculated as the ratio of the intersection over the union of the ground-truth (gt_s, gt_e) and the predicted moment (p_s, p_e).

$$\text{IoU} = \frac{\text{Intersection}}{\text{Union}} = \frac{\min(gt_e, p_e) - \max(gt_s, p_s)}{\max(gt_e, p_e) - \min(gt_s, p_s)} \tag{3.10}$$

For instance, we consider a video query *"Person putting the picture of his daughter on his desk"* where the ground-truth moment $(gt_s, gt_e) = (1.9, 17.4)$, shown in Figure 1.3b. If our VMR system retrieves a moment from $(p_s, p_e) = (5.3, 18.1)$, the IoU

would be calculated as:

$$\text{IoU} = \frac{\min(17.4, 18.1) - \max(1.9, 5.3)}{\max(17.4, 18.1) - \min(1.9, 5.3)} = \frac{17.4 - 5.3}{18.1 - 1.9} = \frac{12.1}{16.2} \approx 0.747$$

With a threshold v of 0.7, this retrieved moment would be considered successful ($r(k, v, qi) = 1$) since $0.8186 > 0.7$, contributing positively to the overall $R(k, v)$ metric.

3.4 Chapter Summary

In summary, this chapter outlines the research methodology used to address the research questions and test the hypothesis. Not to mention, the operating constraints, such as data limitation, query generation, and the baseline system are acknowledged. The evaluation criteria for the proposed system and techniques are also presented, including the LSC Interactive Scoring Metrics, Precision and Recall, Recall at k, Recall at k with IoU, and Mean Average Precision. Following this, the development of the baseline lifelog retrieval system will be described in the next chapter.

Chapter 4

Baseline Lifelog Retrieval System

4.1 Introduction

This chapter presents a comprehensive approach to construct and evaluate a state-of-the-art lifelog retrieval system, named LifeSeeker, addressing the first research question (**RQ1**): *How can a state-of-the-art baseline lifelog interactive retrieval system be developed to effectively support users in retrieving their past experiences?* Specifically, LifeSeeker aims to tackle the retrieval challenges by providing an efficient and user-friendly platform for searching and exploring the vast personal lifelog archives.

As the primary developer of LifeSeeker, my main contributions include the design and implementation of the system's architecture, the improvement of advanced indexing and retrieval techniques across four versions, and active participation in the NTCIR-16 Lifelog task in 2021 and the Lifelog Search Challenges (LSC) for the years 2021, 2022, and 2023. Additionally, we also conducted several user studies to evaluate the system's performance and usability. It is worth noting that this work is a collaboration with other colleagues, building upon the foundations laid by previous researchers in the field of lifelog retrieval and approaching the state-of-the-art performance in the lifelog retrieval task.

In the subsequent sections, I will explain the data preprocessing process in Section 4.2, while the key components of the E-LifeSeeker search engine, such as the indexing and retrieval process, will be described in detail in Section 4.4. Subsequently, a deep dive into the user interface and interaction will be presented in Section 4.5 to provide

a comprehensive understanding of the system's functionality. Finally, Section 4.7 discusses the system performance throughout the participated competitions.

4.2 Dataset

LSC'23 dataset[1], the latest version of the LSC dataset, is a lifelog data collection of a single lifelogger obtained over a period of 18 months, from January 2019 to June 2020 (527 days), with details as follows:

- Core Image dataset: continuous streams of *725,000* images from the first-person point of view. The images were captured every 30 seconds with the use of the OMG Autographer and Narrative Clip[2] wearable cameras in a resolution of 1024×768. To protect the privacy of both the lifelogger and people's surroundings, these egocentric images are fully anonymised, i.e., human faces are blurred and sensitive texts are censored.

- Metadata: Associated with the images taken, a rich set of metadata in CSV format capturing real-world activities was also provided, including spatial information (GPS coordinates), temporal information (date and time), and biometric data (heart rate, skin response, sleep duration). In addition, annotations related to objects, textual information (OCRs), location, visual attributes, and categories are also provided to enrich the indexing stage.

- Additional Semantic Locations: semantic name corresponding to the lifelogger's location at the time of image capture, provided by Myscéal team's developers [122].

- Additional flight data: flight data, including flight numbers, departure airports, and arrival airports, provided by the developers of the Voxento team [114].

As discussed, our emphasis is on the lifelog images and their associated metadata, which are the primary sources of information for the lifelog retrieval

[1] `http://www.lifelogsearch.org/lsc/2023/lsc_data/`
[2] `http://getnarrative.com/`

system. Therefore, the biometric data is not considered in this work. Figure 4.2 and Listing 4.1, respectively, show an example of the lifelog image and its corresponding metadata structure in the LSC'23 dataset.

4.3 LifeSeeker Overview

4.3.1 System Development History

In the Lifelog Search Challenge (LSC) competitions, given one user and one query at a time, the system's objective is to help users handle the scenario of locating desired life moments given a piece of information as a query within a time constraint. Each image must satisfy all the clues from the input query to be considered relevant. Otherwise, they are looking for better moments than this one. LifeSeeker has evolved in multiple research iterations, beginning with the first version (a concept-based retrieval approach) released by Le et al. [196] in the Lifelog Search Challenge 2019 (LSC'19), an annual competition for interactive lifelog retrieval tools. This was followed by an enhanced generation in LSC'20 [197]. Following this, I, a primary developer, and my colleagues attempted to improve the LifeSeeker to resolve the lifelog tasks in four different benchmarking challenges, including the NTCIR-16 Lifelog, LSC'21, LSC'22, and LSC'23 [198, 99, 119, 100]. The experiences gained from the work of other teams and from our own would be used to optimise our system over each research iteration, both the core search engine and the User Interface (UI), intending to enhance the system performance for the next version and ensure that it will be the state-of-the-art search engine. All major differences between the three versions of the system in the LSC challenge are summarised in Table 4.1.

LifeSeeker 3.0 [99] in LSC'21 inherited the main ideas from the original version that built upon the concept-based retrieval approach relying on the analysis of both visual and non-visual content. The system was designed to process the input query and match it with the pre-defined concepts extracted from the lifelog images.

Table 4.1: Comparision of different versions of LifeSeeker from LSC'21 to LSC'23.

Component		**LifeSeeker 3.0** [99]	**LifeSeeker 4.0** [119]	**E-LifeSeeker** [100]
Data Processing	Visual concepts	SINPER [199] pre-trained on COCO [123], PlacesCNN [124] pre-trained on Places365 [124], Bottom-up Attention model pre-trained [129] on Visual Genome dataset [200]		
	Visual features	–	CLIP ViT-B/32	CLIP ViT-L/14@336px
	OCR	Microsoft Vision API	Google Cloud Vision API	
	Location data	–	Location semantic names	
	Time data	Time alignment: part of day, day, month, year, etc.		
Retrieval	Primary Ranking Algorithms	TF-IDF	Cosine Distance	
	Visual Similarity	Bag-of-words using SIFT features		Cosine distance using CLIP features
	Filtering	–	Elasticsearch filtering	
User Interface	Search result display	Ranked list of images		Image clusters (by part of day)
		Single page with lazy loading function		
	Query suggestion	Concepts suggestion	–	–
	User feedback	–	–	Automatic yes/no questions
	Search history	–	–	Search timeline with all previous actions
	Temporal display	Before - Current - After with adjustable time step		All frames in the same part of day
	Navigation	Image zooming and other shortcuts		

The key enhancements included the integration of the Elasticsearch engine to facilitate the search process of sophisticated queries. Sharing the similar core search engine, LifeSeeker 3.0 was further optimised to adapt to the automatic evaluation process in the NTCIR-16 Lifelog task, where no user interaction was involved [198]. Particularly, query pre-processing was integrated to handle temporal and spatial information, while ranking post-processing algorithms were implemented to maximise the result's possibility of being relevant to the query.

The concept-based retrieval approach, however, has its huge limitations, especially when the system needs to handle complex queries or queries with desired concepts that are not explicitly defined or annotated in the lifelog data. In LSC'22, LifeSeeker 4.0 [119] marked a significant shift in the system's functionalities, transitioning from the concept-based retrieval approach to the embedding-based retrieval system. The system was equipped with the Contrastive Language-Image Pre-training (CLIP) model [145] to embed all images and queries into one high-dimensional space for similarity matching. As this model was trained on a large-scale dataset, it enabled the system to capture more semantic information between the visual and textual content.

E-LifeSeeker [100] participated in LSC'23 and continued to leverage the advances of the CLIP model, but with the latest version of the model, CLIP ViT-L/14@336px. The system was further improved with the integration of the text-and-image joint embedding models to bridge the gap between textual and visual semantic meanings. This version also introduced a syntax-based filter mechanism to narrow down the search space based on the result from the embedding model. To provide a more intuitive and user-friendly experience, the user interface also underwent a significant redesign with better navigation and interaction experiences. With these improvements, E-LifeSeeker achieved a competitive performance in LSC'23 competition, ranked third overall, demonstrating the system's potential to be a state-of-the-art lifelog retrieval system. It is worth noting that I will mention LifeSeeker as the general term for

the latest version of the system (E-LifeSeeker).

4.3.2 System Design

The latest iteration, E-LifeSeeker, presents a significant advancement of the system by incorporating the text-and-image joint embedding models as the system's underlying search engine. This integration allows the system to not only explore the visual representations more efficiently but also bridge the gap between textual and visual semantic meanings. The system architecture, depicted in Figure 4.1, consists of two main stages: (1) an *Offline Stage* is responsible for storing metadata and indexing embedded features of the lifelog data that forms the foundation of the system, and (2) an *Online Stage* facilitates the real-time retrieval process and connects users to the system via the user interface. Specifically, a searchable database is created in the offline stage by initially indexing all provided metadata (including spatial, temporal, and other information) into inverted files. Meanwhile, the egocentric images were processed through an Image Encoder to generate high-dimensional vectors for further matching in the next stage. The online stage, on the other hand, handles user queries and performs similarity matching on the pre-processed database in real time, returning the most relevant results measured by the Cosine similarity score. The higher the score, the more similar the content is likely to be. Furthermore, users have options for filters adding more information related to time, location, objects, and text visible in the images (such as part of the day, at work, etc.). The system also incorporates a relevance feedback mechanism, allowing users to refine the search space.

The LifeSeeker's retrieval server is developed using the Django framework[3], which plays the role of a middleware supporting the communication between the client-side requests (user interface and interaction) and different retrieval modules.

E-LifeSeeker employs a web-based user interface developed using the ReactJS framework[4], comprising four main components: a free-text search and filter box, a

[3] `http://www.djangoproject.com`
[4] `http://reactjs.org`

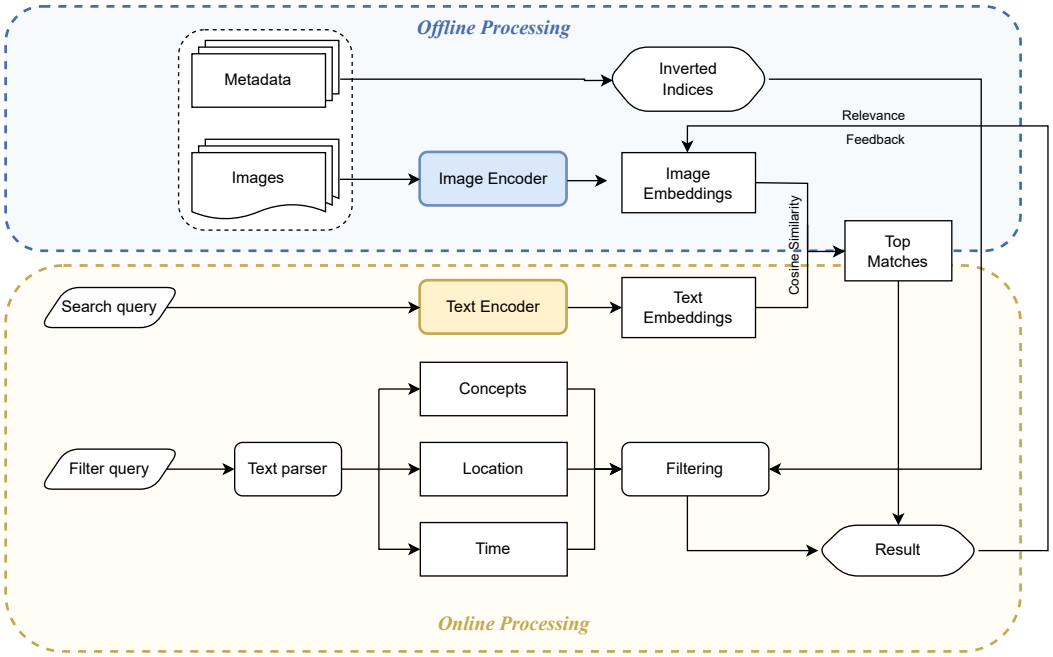


Figure 4.1: ELifeSeeker system architecture.

relevance feedback area, a progress timeline bar, and a vertically-scrollable panel displaying the retrieval results of preceding moments and successive ones. Users provide any query string to the search box describing the desired life moment, followed by filter terms following a pre-defined syntax (described in Section 4.4.2.2) if needed. Upon query submission, those most relevant lifelog images are then displayed on the vertically scrollable panel for further browsing or scanning interaction. Furthermore, users can also explore the image details, provide relevance feedback, or navigate through the search history using the progress timeline bar.

4.4 Search Engine

We suppose that D is defined as the lifelog dataset collection, where each lifelog image $d_i \in D$ is associated with a set of metadata L_i including the location, time, and other information (as in Listing 4.1). Given the user's input query q, the search engine is designed to retrieve the most relevant lifelog images based on the visual

representations. To this end, LifeSeeker is equipped with two main modules: an offline stage for indexing and processing all lifelog data and an online stage for real-time interactive search and retrieval.

4.4.1 Offline Process - Indexing

In order to facilitate the search process, the lifelog data is structured and indexed into a searchable database. The indexing process aims to gain the most insights from the provided data, which involves the extraction of visual concepts from the images, the refinement of temporal and spatial metadata, and the embedding of visual features. The extracted concepts and metadata are then stored in inverted indices for additional filtering options. The visual features are embedded into a high-dimensional space using the CLIP model, which enables the system to capture complex relationships between the visual and textual content.

Figure 4.2: The corresponding image as described by the concepts in Listing 4.1

Listing 4.1: A sample metadata for a lifelog moment generated by the Indexing module


```
"_id": "20160927_140817_000",
"minute_id": "20160927_1408",
"image_path": "LSC/2016-09-27/20160927_140817_000.jpg",
"date": "2016-09-27",
"local_time": "15:08",
"day_of_week": "tuesday",
"month": "september",
"year": 2016,
"part_of_day": "afternoon",
"gps": [53.38571962, -6.258157063],
"activity_type": "walking",
"lat": 53.38572,
"lon": -6.258157,
"location_name": "work",
"location_type": "dcu, university",
"city": "Dublin",
"country": "Ireland",
"location_address": ["wad", "whitehall a ed", "dublin 9",
  "dublin", "county dublin", "leinster", "ireland"
],
"place_category": ["elevator/door", "elevator lobby"],
"microsoft_tag": ["text", "wall", "door", "indoor", "floor"],
"yolo_concept": ["tv"],
"visual_genome": ["white sign", "tiled floor",
  "black television", "wooden door", "wooden wall",
  "white table"
],
"ocr": "cademic offices first floor school office/reception
  faculty of engineering & computing dcu first floor faculty
  administration offices cngl 1sim"
```


4.4.1.1 Temporal Data

The time data is crucial for the filtering process, allowing users to narrow down the search space based on the time of the day or the date of the image. When referring to time, we have different ways to describe it. For instance, "*September 27, 2016, at 15:08*" can be referred to as "*2016/09/27 at 15:08*", "*Tuesday afternoon in September 2016*", or "*September 2016, after 3 pm*". Therefore, to handle input queries containing variable time formats, these different variations need to be indexed in the search engine in advance. We note that the local time gives a more intuitive view into a day in the lifelogger's data compared to the standard UTC

time collected from wearable devices, especially when the lifelogger was traveling to another country in another hemisphere. For example, when the lifelogger is in Thailand (with timezone GMT+7), the camera keeps showing the time at the base timezone. Therefore, we aligned the current time into the local timezone at the location where the lifelogger was at that time to ensure the consistency of the time data as what the lifelogger remembers. The time data is then further refined into different parts of the day (morning, afternoon, evening, night) and other time-related information (day, month, year, etc.), with details shown in Listing 4.1. Each image has a `minute_id` that we can process as follows:

- `date`: The date of the image, in the `YYYY-MM-DD` format;
- `month`: Name of the month (e.g. January, September, December, ...);
- `year`: The year in the `YYYY` format;
- `local_time`: The time in the lifelogger's local timezone in 24-hour format `HH:MM`;
- `day_of_week`: One of the seven days of the week expressed in the lifelogger's local time (e.g. Monday, Tuesday, Sunday, ...);
- `part_of_day`: The part of the day, whether it is *early morning* (04:00 to 07:59), *morning* (08:00 to 11:59), *afternoon* (12:00 to 16:59), *evening* (17:00 to 20:59), or *night* (21:00 to 03:59), based on the local time.

4.4.1.2 Spatial Data

Another crucial attribute in every lifelogger's life moments is the locations they have been, as it adds more context to the query generation process to find more relevant images. To expedite the search process, we have concentrated on revising location information, particularly addressing the incorrectly located points, which consequently led to the identification of distant places having no connection with the correct answer. From the geographic coordinates collected from wearable

devices, we identify the detailed address of the image using the Geocoding API from Google Map Platform[5]. Apart from the address, we enrich the location metadata by tagging cities and countries, which is especially valuable for locations outside Ireland, the lifelogger's home base, while using elevation with a threshold could indicate whether the lifelogger is on a plane or not. To further structure the location data, we categorised the locations into 32 pre-defined place categories, as listed in Table 4.2. For those locations without specific semantic names, we assigned them to the nearest known location within a fixed radius. On top of that, the Microsoft Vision API is used as an automatic feature extractor for lifelog images in order to extend the variety of visual concept annotations. Each image has information related to the location of the lifelogger at that moment as follows:

- `latitude`: Angular coordinate specifies the north-south position of the image on the surface of the earth;
- `longitude`: Angular coordinate specifies the east-west position of the image on the surface of the earth;
- `location_name`: Semantic name of the location (i.e. Dublin Airport, DCU, ...);
- `location_type`: One of the 32 predefined categories in Table 4.2;
- `location_address`: Detailed address associated with the lifelogger's location;
- `city`: Name of the city associated with the lifelogger's location;
- `country`: Name of the country associated with the lifelogger's location.

4.4.1.3 Visual data

Images captured from the wearable camera are information-rich, as moments are illustrated in detail (i.e. how the surroundings look, who appears in that moment, and which objects are seen). The process of indexing visual data comprises two

[5] `http://developers.google.com/maps/documentation/geocoding`

Table 4.2: Location categories

ID	Name	ID	Name
1	Airport	17	Home
2	Antique store	18	Hotel
3	Apartment	19	Howth
4	Bank	20	Office
5	Bar, pub	21	Park
6	Bus stop	22	Pharmacy
7	Car	23	Plane
8	Castle	24	Restaurant
9	Church	25	Shop
10	Coffee shop	26	Shopping Center
11	Convenience store	27	Sister home
12	DCU	28	Station
13	Dental clinic	29	Store
14	Department store	30	Street
15	Embassy	31	University
16	Hall	32	Unknown

main components: extracting visual concepts from the images and embedding the images into visual features. While the former identifies various elements within the lifelog images (such as objects, scenes, and text), the latter captures the semantic information between the textual and visual content. Together, the two components provide informative representations of the lifelog images.

Visual concepts

- *Object detection*: Object tagging is an essential component for most concept-based retrieval systems. Thus, visual concepts of lifelog images, obtained from object detection models, are always provided as part of the lifelog dataset in all collaborative research tasks and challenges in the lifelogging domain [81]. Besides the visual concepts shared by the lifelogger/task organisers, which were generated using Microsoft Vision API, we further considered other object detection models with the aim of tagging more objects from lifelog images. The first model, YOLOv4 [130] which was pre-trained on the COCO dataset [123], can detect 80 different categories of common objects in daily life. Meanwhile, the Bottom-up Attention model [129] is able to detect 1600 object classes along with 400 associating attribute types (i.e. black pillar, wooden floor, red car,

etc.) by using multi-GPU pre-training of Faster R-CNN [127] with ResNet-101 [131]. This model not only increases the number of concepts by a significant amount but also enables the retrieval of concepts at a finer level of detail using their corresponding attributes. The fields `microsoft_tag`, `yolo_concept`, and `visual_genome` in Listing 4.1 illustrate a sample result of the visual concepts generated by Microsoft Vision API, YOLOv4 and Bottom-up attention model, respectively.

- *Scene recognition*: While object detection focuses on individual objects within the image, scene recognition aims to understand the surroundings, giving more insight into where the lifelogger was (i.e., waiting in a lobby, exercising outdoors, working in an office). To achieve this, we utilised the PlacesCNN [124] model pre-trained on the Places365 dataset, which classifies images into 365 place categories. For example, the lifelog moment displayed in Figure 4.2 was recognised as "`elevator/door`" and "`elevator lobby`" as shown in the field `place_category` in Listing 4.1.

- *Optical character recognition (OCR)*: Text presented in lifelog images could provide valuable information not only about the lifelogger's activities but also about interaction. To convert text into visual concepts, we leverage the OCR tool from Google Vision API[6] to detect and recognise text content. The extracted texts were then aggregated into a single string (as shown in Listing 4.1 in the `ocr` field) that can be indexed by the search engine in the latter stage.

Visual embedding features

Although the visual concepts provide a general context of the lifelog images, the semantic information between the textual and visual content has not been fully exploited. To address these shortcomings, LifeSeeker leverages the cutting-edge Contrastive Language-Image Pre-training (CLIP) model [145], developed by the

[6] `http://cloud.google.com/vision/docs/ocr`

OpenAI team. Our implementation uses a Vision Transformer [152] pre-trained at 336-by-336 pixel image resolution (ViT-L/14@336px) as the image encoder, which generates 768-d image embedding. The latest version of this model, OpenCLIP [150], was trained on an enormous dataset called the LAION-5B dataset [201], consisting of 5.85 billion CLIP-filtered image-text pairs. Consequently, this model could have the potential to capture more complex and abstract relationships between the visual and textual content, thus improving the retrieval performance. A comprehensive user experiment (detailed in Appendix A) was conducted that demonstrated the advantages of visual features over conventional visual concepts, showing improved retrieval speed and performance. These capabilities led to our decision to choose the visual features as the primary search engine for LifeSeeker.

4.4.1.4 Summary

E-LifeSeeker indexing module integrates multiple advanced techniques to process and manage the lifelog data efficiently. As its core, Milvus [161], a high-performance vector database system, is used to store and manage the large-scale 768-dimensional embedding vector database. This enables optimised query processing beyond simple vector similarity search, which helps shorten search time, improve retrieval performance, and ensure high scalability. Meanwhile, Elasticsearch serves as a filter tool for other visual concepts, textual content, and other metadata, aiming to facilitate filtering and faceted search. The retrieval server, built using the Django framework[7], plays the role of a middleware supporting the communication between the client-side requests (user interface and interaction) and different retrieval modules. Each lifelog image is associated with a rich metadata structure, as shown in Listing 4.1.

[7] `http://www.djangoproject.com`

4.4.2 Online Process - Retrieval

4.4.2.1 Free-text Search

The free-text search function represents the core search engine of the system. When users input a string query, the system initially processes through a pre-trained CLIP Text Encoder [145] in real-time, which transforms the natural language description into a high-dimensional vector. This vector is in the same semantic space as the aforementioned pre-computed image embedding features. Next, the relevance score between every image-query-vector pair is measured using the Cosine Similarity metric. The higher the score, the more similar the content is likely to be. The top-k images with the highest similarity scores are then returned to users for further exploration.

4.4.2.2 Elasticsearch Filter

Elasticsearch[8] has been deployed to the system in the second version as the concept-matching backbone [197]. Starting from the third version, the system underwent a significant improvement with the introduction of the language-image pretraining model (CLIP) [145] to embed all images into a high-dimensional space. So, the role of Elasticsearch has been shifted to a filter to narrow down the search space based on the result from the embedding model. This filter functionality is an essential part of the system, as there are many cases where users want to search for images based on specific concepts, locations, or times, which cannot be handled by the embedding model alone. In order to reduce the query analysis time and allow flexibility in controlling how each keyword should behave when retrieving lifelog images, we have implemented a syntax-based filter mechanism in the system as follows:

```
<CONCEPTS> ; <LOCATION> ; <TIME>
```

where each query part (`<CONCEPTS>`, `<LOCATION>` and `<TIME>`) corresponds to a category described in Section 4.4.1. A syntax-based query can be formed by

[8] `http://www.elastic.co`

specifying keywords in each part of the syntax above. For instance, the following query is a valid input toLifeSeeker:

```
flower teddy bear ; bedroom home ; after 7pm on Monday
```

The Searching process in Elastic Search mode was carried out by employing the *query string query*[9] to match `<CONCEPTS>` and `<LOCATION>` keywords, while the *term query*[10] and *range query* mechanisms were used to filter images using the given `<TIME>` keywords.

We leverage the SUtime library [202], a state-of-the-art temporal tagger tool, to get accurate temporal expressions such as year, month, season, or even date. For instance, the `after 7 pm on Monday` will be parsed into the time field so that `Monday from 19:00:00 to 23:59:59` can match with the temporal filter.

4.4.2.3 Visual Similarity Search

The visual similarity search function offers an alternative search option, which is particularly useful on occasions when the input query is difficult to describe textually or when users intend to explore more images visually similar to the current image. Moreover, this function especially benefits users for ad hoc retrieval tasks, where they are required to find as many visually similar images as possible to the query image. Nevertheless, this function is also useful for recalling moments based on their visual memory cues or for discovering other images that may not be apparent through the text-based search alone. The process involves image-to-image matching, where the feature vector of the selected image is compared with all other image feature vectors in the database. For computing the similarities between image-image pairs, the cosine similarity is employed as the distance metric.

[9] `http://www.elastic.co/guide/en/elasticsearch/reference/current/query-dsl-query-string-query.html`

[10] `http://www.elastic.co/guide/en/elasticsearch/reference/current/query-dsl-term-query.html`

4.4.2.4 Temporal Browsing

With the temporal browsing function, users are provided with a dynamic way to explore the lifelog data within its chronological context. Users can navigate through the selected moment and its temporally related images by adjusting the temporal range between them, getting a better understanding of the activities and context surrounding that event or during a day. This temporal exploration capability is useful to indicate the specific moment that matched the input query among several similar events that occurred in one day or one period of time.

4.4.2.5 Relevance Feedback

To better support user interaction with the system, we have incorporated a relevance feedback mechanism that allows users to refine search results dynamically. In particular, after users enter a query, the system will automatically generate a question related to the visual content of their desired images, and users have the option to choose to include or eliminate those. The process of generating questions begins with aggregating all visual concepts present in the retrieved images and then calculating the frequency of each concept presented or absent across the result sets. By doing so, the system can split the concepts into two groups: the positive group (concepts present in the images) and the negative group (concepts absent in the images). The system then generates a question based on the ratio of each concept between two groups (the one closest to 1), asking users whether they want to include or exclude those concepts in the next search. The top-ranked concept is selected to form the next yes/no question, such as "*Is there a ceiling?*" or "*Are there people present?*". This ensures that each question generated by the system regardless of the user's answer, will help to refine the search results and improve the ranking algorithm.

4.4.2.6 Ranked List Clustering

This function is a post-processing technique for the display of results, addressing the issue of duplicate images in the search results. Similarly to almost all conventional lifelog retrieval systems, our previous UI displayed results as all keyframes individually ranked by their relevance score. This approach, however, usually led to difficulties in navigation, especially when the desired moments were found within a sequence of duplicated images consecutively in one event. For example, when querying for "the lifelogger was watching the TV", it would return hundreds of images of this activity, many of which are temporally sequential. To overcome this problem, we introduce a post-processing technique to display the results by clustering them based on temporal features. They are clustered together and displayed in the top 3 with the highest score, with the option to view all images in the group if desired. The re-ranking algorithm has the execution steps below:

1. Initial ranking: The system generates the ranked list of all matching images after entering a new query.
2. Temporal grouping: This ranked list is then segmented into sub-lists by grouping images from the same part of the day (each day is divided into 4 parts: morning, afternoon, evening, and night), creating natural clusters that align with the lifelogger's daily routine.
3. Cluster scoring: For each temporal sub-list, the system calculates the average confidence score of the top 3 highest-ranked images in the group.
4. Final re-ranking: The sub-lists are then re-ranked based on their new average confidence scores, with the highest-scoring group displayed first.

The reasons behind this approach are to reduce the number of individual images displayed to users, to avoid showing duplicate images, and to provide a more intuitive way to navigate through the search results. Moreover, the division of the results into

four parts of the day helps distribute the number of images evenly across the day, ensuring a balanced representation of the lifelogger's daily activities.

4.5 User Interface and User Interaction

4.5.1 User Interface

The interactive user interface of E-LifeSeeker, illustrated in Figure 4.3, is designed as an intuitive web-based application that allows users to interact with the system through a web browser. The user interface consists of four main components, including the free-text search and filter box (**A**), the relevance feedback display (**C**), the search progress bar (**D**), and vertically-scrollable panel displaying the results retrieved in groups (**E**). The details of each component are as follows:

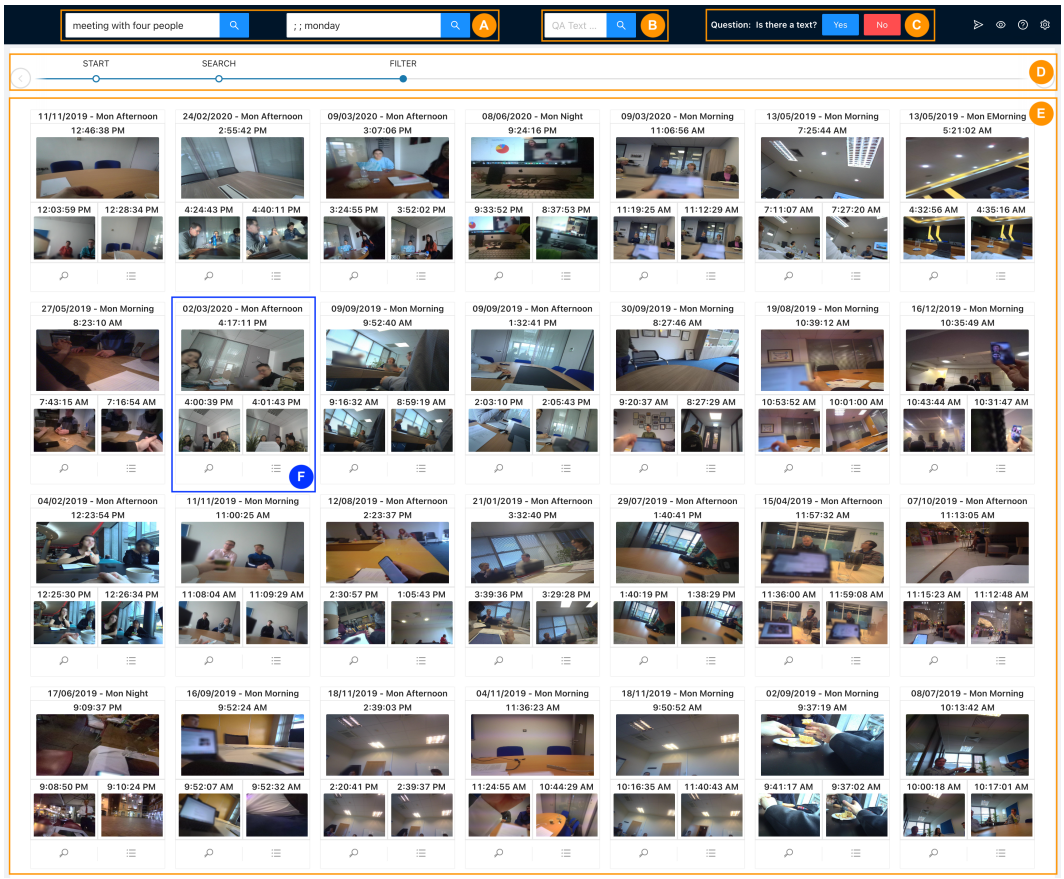


Figure 4.3: The User Interface of E-LifeSeeker system. The screen displayed the results of an example query "*meeting with four people*".

- **Free-text search and filter box (A)**: The first box positioned at the top left of the UI is where users input their query string and apply filters describing the desired life moment.

- **Question Answering submission box (B)**: This box allows users to submit their answers to the Question Answering task, which requires submitting a text-based answer.

- **Relevance feedback display (C)**: Located at the top right corner, this feature presents a yes/no question to help users filter out possible filters based on their input query. This feature also optimises the use of screen real estate and avoids showing duplicate images while maintaining access to other nearby moments via temporal features in the user interface.

- **Search progress bar (D)**: A horizontal bar that visually represents the progress of the search process, showing the current position and enabling users to navigate between search stages. This feature is handy for complex search queries that involve multiple steps, allowing users to go back to any previous step to correct the queries or modify their search criteria if needed.

- **Vertically-scrollable panel (E)**: The largest area of the UI, displaying the ranked list of relevant image groups obtained from the input search query and filter box (**A**). Each group is represented by the top 3 relevant keyframes, sorted by relevance and time period obtained from Section 4.4.2.6. This panel implements a lazy loading mechanism, where additional images are loaded as users scroll, reducing initial loading time and improving overall user experience, especially for large result sets. The vertically scrollable panel and its items are designed for optimal moment-scanning and browsing on a 27-inch monitor ($367.69mm \times 612.49mm$). Specifically, on a 27-inch monitor, there are at most four rows of images in normal-screen mode and six rows in full-screen mode. Each row of the panel consists of at most 7 groups (each represented by 3 highest-relevant score images). It is worth noting that this

optimisation is specifically designed for the Lifelog Search Challenge to reduce the overhead time to find the correct moments to submit by scrolling up and down. According to our experience in previous Lifelog Search Challenges, viewing as many top results as possible without scrolling can result in a higher chance of finding the correct images to submit.

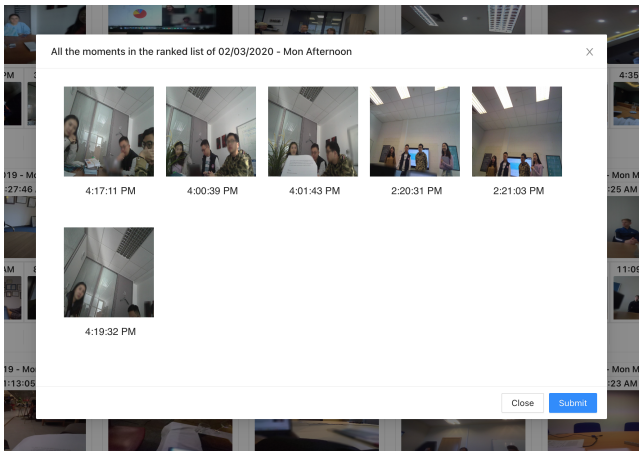
- **Expandable group (F)**: Groups of images in a specific time period, sorted by relevance and represented by the top 3 relevant keyframes. Additionally, the group and image are displayed with the time format `ddmmYYYY-P` and `HH:MM:SS` respectively, where `Y, m, d, H, M, S` are year, month, day, hour, minute, and second, and `P` is the period of the day (morning, afternoon, evening, or night) correspondingly. Each group is expandable to either a popup box that highlights the relevant frames of that group in descending order (the search symbol on the bottom left, Figure 4.4a), or a popup box that shows all the keyframes from the day (the list symbol on the bottom right, Figure 4.4b).

The very top right of the UI exhibits additional buttons for support functionalities including (1) *Submission* button for submitting the chosen frames, (2) *Find similar* button for showing all visually similar frames to a specific keyframe, (3) *Helper* button for displaying all the shortcuts and tips for the system, and (4) *Settings* button for logging in and out and adjusting server configurations.

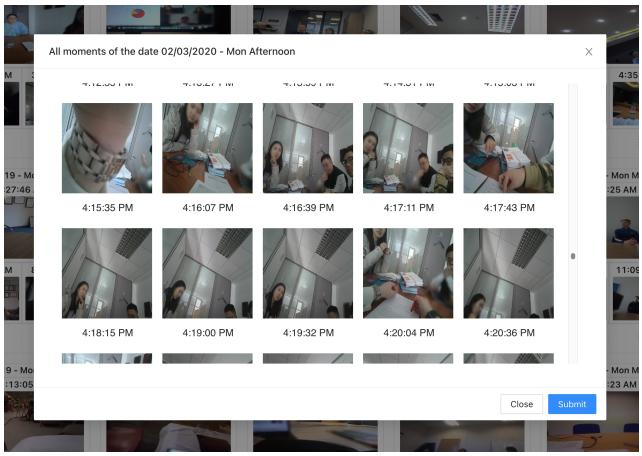
4.5.2 User Interaction

The workflow of user interactions is designed to be intuitive and efficient, guiding users through a multi-step process to find their desired lifelog moments as below:

1. Query input: Users begin by entering a query string into the free-text search box (**A**), the first box located on the top left of the UI. The query can be in the form of a full sentence or any sequence of keywords. For example, users can input a query as *"meeting with four people"*.



(a) Relevant-frame view of the group



(b) All-frame view of the group

Figure 4.4: The details view of **Expandable Group (F)** in the UI showed in Table 4.3.

2. Search Refinement: After inputting the query, users can further enhance their search using the second box (**A**), filtering the current query with three different filter types including time, object, and location. For instance, *";; Monday morning'* can be the filter for the above example query.

3. Result Exploration: Once the input query is submitted, results are displayed on the vertically scrollable result panel (**E**) that occupies the majority of the UI. Users can either scan or browse the ranked list of relevant image groups, which are clustered by time period, sorted by relevance, and represented by the top-3 most relevant frames.

4. Detailed View and Similarity Search: When users identify a potentially relevant group, the system provides options for detailed exploration, either expanding a full timeline day view or accessing just the relevant frames of that group. Furthermore, an option to view all visually similar frames to the current chosen frame from a single group is also available.

5. Relevance Feedback: The relevance feedback display (**C**), located at the top right corner, will automatically generate a yes/no question to help users filter out possible filters based on their input query. The displayed results will be updated according to the chosen answer. For the example in Figure 4.3, after users enter a query, the system will automatically generate a question related to the visual content of the desired images as *"Is there a ceiling?"*.

Throughout this process, users are able to track and go back at any stage to either change the filter or redo the current action if needed using the search progress bar (**D**), which provides a clear view of the search journey. The entire search process is iterative, allowing users to refine their search strategy based on the intermediate results until they find the target image. Ultimately, users can either submit the chosen potential images by clicking the "submit" button for the KIS and ADS tasks or submit the text-based answer to the Question Answering task in the submission box (**B**).

4.6 System Modifications for the NTCIR Challenges

Sharing the same dataset with LSC'21, NTCIR16 - Lifelog4[11] [203] aims to advance the state-of-the-art lifelog retrieval system in solving two types of tasks: the Known-Item Search task (**KIS**) and the Ad-Hoc Search task (**ADS**), with 24 topics prepared in each task. Moreover, there is no time restriction on submissions made for tasks that automatically submit without user interaction during the search process, but there is a limit of 100 images submitted for each query. Since

[11] `http://www.lifelogsearch.org/ntcir-lifelog/NTCIR16/`

LifeSeeker is constructed as an interactive retrieval system that requires human involvement to solve the retrieval tasks, we attempted to equip our current system with two new functionalities to operate in an automatic manner for this challenge [198]. They include query pre-processing and ranking post-processing, which are described as follows:

Query Pre-processing We integrate a text parser to process the input query prior to matching in the Elasticsearch engine. Its ultimate goal is to automatically separate the query, manually entered by the users, into keywords belonging to the three aforementioned fields: concept, location, and time. In particular, specific places and proper nouns are split by matching with the syntax location dictionary. In the meantime, we leverage the SUtime library [202], a state-of-the-art temporal tagger tool, to obtain temporal expressions such as year, month, season, or even date. On the other hand, those terms about objects, humans, or not belonging to the two above fields are considered concepts. For instance, the query "*I was drinking coffee while waiting in a car repair/sales store in May 2018*" will be parsed into "*repair/sales store*", "*May 2018*" and "*drinking coffee*" corresponding to location, time, and concept fields, respectively. The three fields are then concatenated into one single input string to the search engine.

Ranking Post-processing Since there is no human interaction included in these automatic tasks, we try to maximise the resulting possibility by offering three different ranking approaches. While the traditional method (**A1**) focuses on the images with the highest scores, the other two expandable post-processing techniques (**A2** and **A3**) consider the top-ranked images and their temporally close images.

- (**A1**) Get the top-ranked 100 images (if available);
- (**A2**) Combine the first ten images with the eight temporally close neighbouring images (4 forwards and 4 backward);
- (**A3**) Combine the first 20 images with the four temporally close neighbouring

images (2 forwards and 2 backward).

4.7 System Performance

4.7.1 Results in the NTCIR Lifelog Search Task

With the aim of examining the variance in the query generation process, we got efforts from two different users, denoted as "**U1**" and "**U2**". For each topic, they attempted to generate a single string as the input to the search engine. Then, the ranked list results were achieved by the three aforementioned post-processing approaches (**A1**, **A2**, and **A3**), meaning that they generated six official runs for the competition.

Table 4.3: Results of LifeSeeker runs in NTCIR16 competition. "**U**x - **A**y" stands for "User x using approach y". (There are 6 runs corresponding to 2 users with 3 post-processing ranking methods). The best values are highlighted in bold.

Run	# tasks solved	# KIS tasks solved	# AD tasks solved	# images correct/submitted	MAP	P@5	P@10
U1-A1	29	9	**20**	320/4629	**0.0299**	**0.0833**	**0.0750**
U2-A1	**31**	**11**	**20**	334/4677	0.0211	0.0583	0.0583
U1-A2	16	5	11	238/4530	0.0236	0.0500	0.0583
U2-A2	22	7	15	**365**/4708	0.0237	0.0792	0.0729
U1-A3	19	4	15	229/3714	0.0286	0.0792	0.0667
U2-A3	22	7	15	275/3846	0.0168	0.0583	0.0625

Table 4.3 indicates the result summary of all runs of LifeSeeker in the NTCIR16-Lifelog4 challenge [198] (full details can be found in Table B.1 in the Appendix B), which highlights the number of tasks solved, the number of Known-item search tasks solved, the number of Ad-hoc search tasks solved, the number of images correct, the mean average precision (MAP), top-5 precision, and top-10 precision. Generally, people relying on the traditional post-processing approach (**A1**) have surpassed those who use the other two time-inclusion methods with a higher number of tasks achieved, more than eight tasks. Precisely, A2 and A3 were not the appropriate options since they can solve 22 tasks for **U2** and less than 20 tasks for **U1**. Nevertheless, **U2** has an undoubted ability to form an information need from the provided topics by solving the most tasks across all runs.

4.7.2 Results in the Lifelog Search Challenge

LifeSeeker has been evaluated in the annual real-time Lifelog Search Challenge (from 2021 to 2023), against other interactive lifelog retrieval systems. In this specific section, we will highlight and discuss the performance of the two most recent versions of the system, LifeSeeker 4.0 and E-LifeSeeker, in LSC'22 and LSC'23, respectively. The scores in this competition considered the search speed (the faster, the better) and the number of incorrect attempts (the less, the better), which are evaluated via the Distributed Retrieval Evaluation Server (DRES) [204, 205]. In particular, they are calculated following the equations illustrated in Equation 3.1 and Equation 3.2. It is worth noting that all scores are normalised to 100 points for each task, where the highest score is the best performance.

Table 4.4: Statistics of the top-5 teams in the LSC'22 competition. The best values are highlighted in bold.

Team name	*Task*			**Total Score**
	$Ad-hoc$	KIS	QA	
Myscéal [6]	98	**100**	**100**	**298**
LifeSeeker [99]	**100**	88	96	284
Memento [105]	66	92	79	237
FIRST [101]	51	95	75	221
Voxento [113]	49	87	56	192

As can be seen from Table 4.4, we achieved second place out of 9 teams, with a total score of 284, in LSC'22. This proved our system's overall performance across different tasks. Notably, we excelled over all other teams in Ad-hoc tasks with the highest score, showing our system's ability to locate and retrieve multiple relevant images at once. In the Question Answering (QA) task, LifeSeeker also exhibited a notable performance, scoring 96 points, just 4 points behind the top team. The KIS task, however, was somewhat more challenging than other tasks, resulting in 88 points. One of the possible reasons for this could be unique scoring mechanisms of the task (illustrated in Equation 3.1), which impose a penalty for each incorrect answer. This means that the faster the answer is submitted, the higher the rewards,

but also the higher the risk of submitting a wrong submission. This creates a trade-off between the time to submit and the accuracy of the answer, either submitting the answer rapidly to gain more points or taking more time to ensure accuracy.

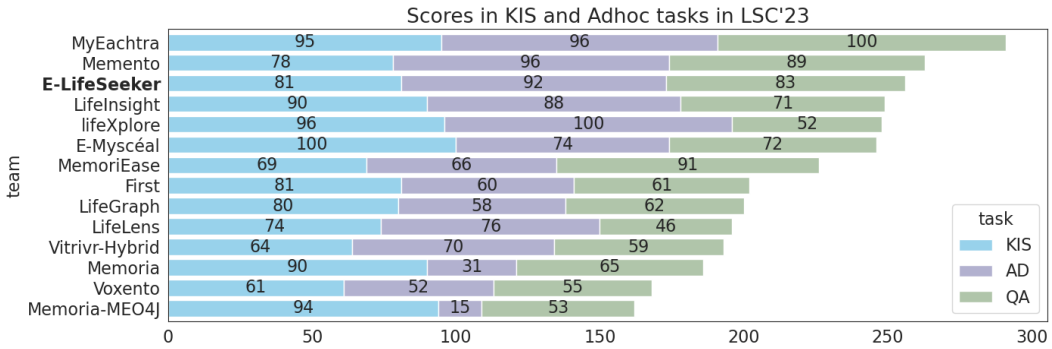
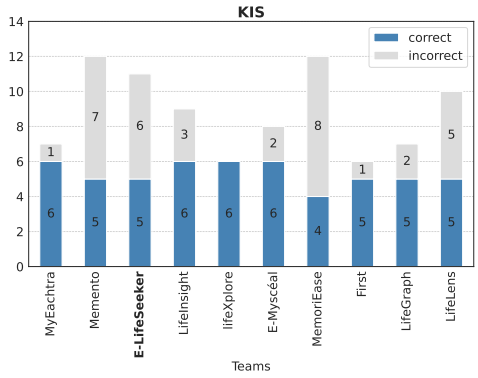


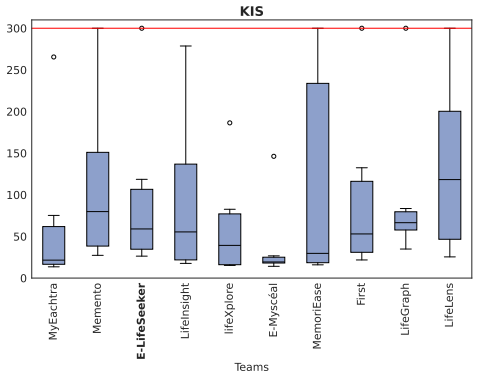
Figure 4.5: The performance of the top-performing teams in the LSC'23 competition (best viewed in colors).

From LSC'23 results, reported in Table 4.5, E-LifeSeeker achieved a total score of 256, securing the third place among the 14 participating teams. In general, our system demonstrated remarkable performance across all three task types, with normalised scores of 81 for Known-Item Search (KIS), 92 for Adhoc Search (AD), and 83 for Question Answering (QA). MyEachtra [110], an event-based approach, implemented a separate module for the QA task, which helped them achieve the highest score in this task. Meanwhile, EMyscéal [6] and lifeXplore [116] ranked first in the KIS and AD tasks, respectively. A detailed analysis of the results reveals varying strengths and weaknesses between different tasks. For this specific challenge, we will analyse the expert score only, as I was the one who participated in the challenge, while the newcomer score will be excluded.

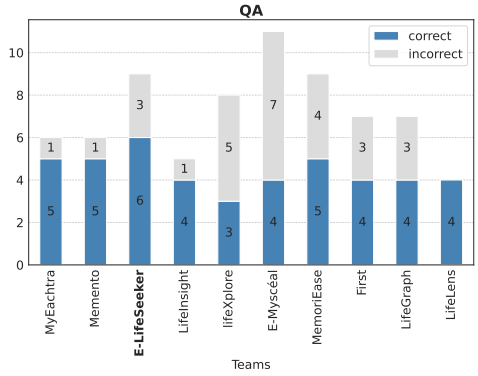
We further analyse the performance of E-LifeSeeker in LSC'23 by comparing the elapsed time until the first correct image and the number of correct/incorrect submission distributions for each task, as shown in Figure 4.6. The distribution of solving time reveals how efficiently each user performed, as it directly affects the scoring through a time-dependent calculation metric (Equation 3.1). Not to mention that in the context of the LSC competition, any incorrect answer will be penalised



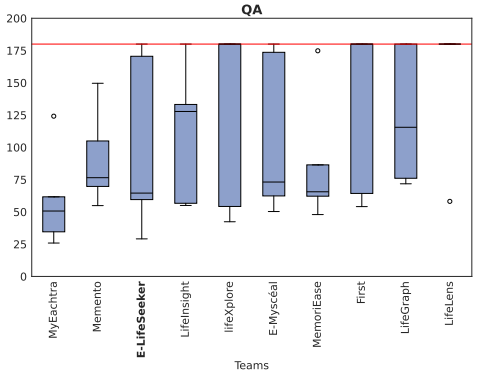
(a) Correct and incorrect submissions



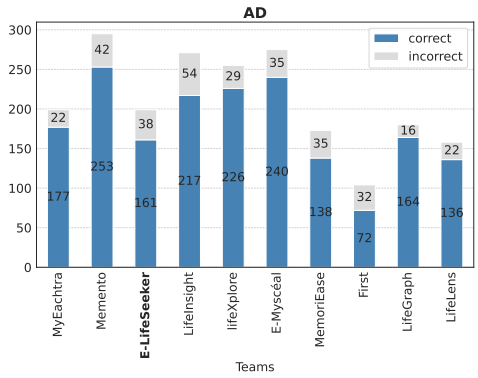
(b) Submission time



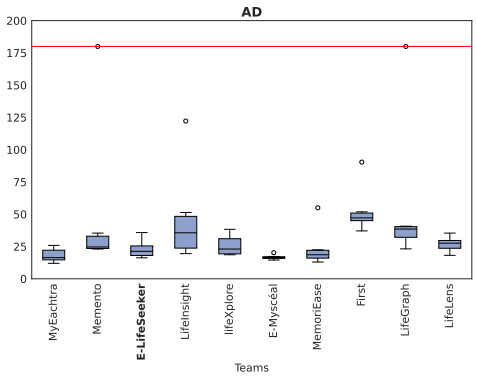
(c) Correct and incorrect submissions



(d) Submission time



(e) Correct and incorrect submissions



(f) Submission time

Figure 4.6: Number of correct and incorrect submissions and submission time for each task type in the LSC'23 competition (best viewed in colors).

by an amount of 10 points (out of 100 points for that task), which is important to highlight the number of both relevant and irrelevant answers. This penalty means that even if the system can submit the answer quickly, the score will be deducted if

they make multiple wrong submissions.

Figure 4.6c shows that for QA tasks, E-LifeSeeker submitted 6 correct answers and only 1 incorrect answer, which is indeed the most queries solved among all teams in this category. However, it is important to note that there were 3 failures, and the submission times were not the fastest among the teams, as shown in Figure 4.6d. This resulted in the third-rank score for the QA task. The trade-off between accuracy and speed is apparent when compared to EMyscéal [6] performance, which only solved 5 QA queries, but they submitted the correct answers faster and maintained a lower error rate, which led to the highest score in this task. In the meantime, the KIS task and AD task are a bit more challenging. Although the solving time was competitive, we failed to submit the correct answer 6 times for the KIS tasks, impacting the score due to the wrong submission penalties. For the AD task, we failed 38 out of the 199 queries submitted. This indicates the reason for the lower score in the KIS and AD tasks compared to the QA tasks.

4.8 Discussion and Conclusion

LifeSeeker has improved over time through four different iterations of the research cycle, each contributing to its current capabilities in lifelog search. After constructing the baseline system, we identified several key takeaways from the system development process, including (1) the underlying search engine, (2) the filtering mechanism, (3) the user interface, and (4) additional refinements to support user interaction. In regards to the search engine, it is responsible for indexing the metadata of lifelog images into databases and analysing the free-text queries by the users to match the query terms with the metadata of the previously indexed images to return relevant results. The feasibility of the underlying search engine in extracting valuable features and matching them with the input query has been evaluated in the benchmarking results. Particularly, we captured the multimodal information from the lifelog images, including visual concepts, and

spatial, temporal, and textual information, to enhance the search process. The semantic meaning between textual and visual content was fully exploited by leveraging the CLIP model. All of this created a comprehensive understanding of the lifelog images, which is crucial for the retrieval process. Filter mechanisms maintain a vital role in the search process as they have the ability to narrow down the search scope.

Additionally, the intuitive UI with advanced functionalities might benefit the retrieval process in terms of user experience and interaction. The interface enhances lifelog moment browsing through various means of interaction with retrieval results, such as moments metadata details, adaptive temporal display, and visually similar moments exploration. The interface design choice and the implementation details of the search engine were explicitly outlined with clear details in Section 4.5. With all these features, LifeSeeker demonstrates its adequate performance in real-time challenges against others. In particular, the system ranked in top positions in both the NTCIR Lifelog Search Task and the Lifelog Search Challenge, proving its competitive capabilities in the lifelog retrieval domain. With that being said, the first research question **RQ1**, *How can a state-of-the-art baseline lifelog interactive retrieval system be developed to effectively support users in retrieving their past experiences?* , has been addressed by the development of the LifeSeeker system as a state-of-the-art lifelog retrieval system. This system was considered as the foundation for the later research and development of the moment-based lifelog retrieval system of our hypothesis.

While LifeSeeker has shown its abilities in lifelog search, our experiences, especially in LSC'22 and LSC'23, have also revealed some limitations and areas for improvement. Speaking of the limitations, the performance of LifeSeeker system was affected by the network speed, server location, and user search speed. This is apparent in LSC'22, where the geographical distance between the server (based in Ireland) and the competition (in New Jersey) caused slow response times. This impacted the system's response time, particularly in loading photos and calculating

similarities using GPU resources, which resulted in a slower response time. The same issue was observed in LSC'23, where the competition was held in Greece. However, there is still room for improvement, such as exploring advanced embedding models and matching techniques. The system currently uses a pre-trained model that is trained on a normal dataset with third-person perspective data. We can fine-tune the model more on the first-person perspective data to improve the system's performance. Specifically, reducing the number of incorrect submissions in those tasks by taking a bit of time to check the neighbor images before submitting while maintaining the fast solving times could enhance the overall performance. The system could be further enhanced by incorporating more advanced temporal browsing techniques to better support users in exploring lifelog data within its chronological context.

4.9 Chapter Summary

In general, we develop an interactive lifelog retrieval system that is competitive in the lifelog retrieval domain, which has been evaluated in real-time challenges against others and is considered a state-of-the-art lifelog retrieval system. Through the research and try, we equipped the system with advanced techniques to make lifelog data manageable and searchable. Not to mention, the intuitive user interface is designed as a bridge to connect users with the system, providing a seamless experience for users while using the system to look for their desired moments. LifeSeeker exploits the visual concepts and the spatial, temporal, and textual information to enhance the search process. Apart from that, the semantic meaning between textual and visual content has been fully exploited by leveraging the CLIP model. The evaluation through the benchmarking challenges gives more insights into system performance, proving its ability to solve the task of locating desired moments in one's life.

Chapter 5

Video Moment Retrieval

5.1 Introduction

After constructing a baseline approach for the lifelog retrieval system, we recognise a gap between the search tool and the way humans access their memories as humans memorise their experiences as chunks of events/moments [60]. Instead of focusing on individual images as we did in conventional lifelog retrieval techniques, we lean our attention toward moments or events, which is closer to the objective of the Video Moment Retrieval (VMR) task. In addition, the similarity between the video and lifelog data format, structured in continuously chronological order, presents an opportunity to adapt advanced techniques in the VMR tasks for lifelogging applications. There are more research efforts in video understanding as video is one of the most popular media types with tons of available sources in public. Taking a more natural moment-based approach to the lifelog retrieval system, it is required to understand the SOTA in moment retrieval techniques in a more general context of video understanding and then adapt them to the lifelog retrieval system.

Previous research in VMR has primarily followed two main approaches: proposal-based methods and proposal-free methods. Studies [48, 49, 50, 51] tackled the VMR task with predefined candidates or proposals within videos, then utilising matching techniques to rank them based on the learned representations (proposal-based methods). While effective to some extent, such methods face significant practical and scalable challenges as they demand extensive human

efforts for moment boundary annotations. In the meantime, other frameworks [52, 53, 54, 55] adopt the proposal-free methods that learn the cross-modal interactions and attempt to regress the probabilities of all frames, then choose the peaks as the start and end of the event's segments (proposal-free methods). Recognising the advantages of reducing human involvement and resources, our framework will make use of proposal-free techniques.

Despite the undeniable role of encoding low-level visual features to represent visual content, little effort has been made to explore the impact of high-level relational semantic cues for the VMR task. For that reason, we propose a Parallel Transformer framework (PaTF) to jointly learn from both aforementioned aspects by using a parallel transformer architecture including a global stream (with visual embedding representations) and a relational stream (with scene graph representations). To this end, we hypothesise that this approach can better exploit the visual meaning but also the semantic relations between the query and the video. Particularly, the global stream encodes the overall visual content of each video frame with the pre-trained embedding models such as OpenCLIP [150] or I3D [183]. Simultaneously, the relational stream models the fine-grained details and relationships among different objects in the frame via the scene graph generation model. Additionally, each stream is fed into a transformer module to learn their representations before ensembling the features of both streams for prediction.

For the context of this dissertation and this particular chapter, a "*moment*" or "*event*" is referred to as a combination of activities that happened in a short period of time (characterised by a start and an endpoint). This definition is closely aligned with how individuals tend to structure and retrieve their personal histories. This chapter will explore in depth our novel approach to the task of Video Moment Retrieval and present the answer to Research Question 2 (**RQ2**): *How can a state-of-the-art transformer-based Video Moment Retrieval technique be designed to effectively localise target moments in video sequences?*

The remainder of this chapter is organised as follows: a detailed explanation

of the Attention-based Parallel Transformer Framework (PaTF) in Section 5.2, the experimental setup and results in Section 5.3, and finally the summary of the chapter in Section 5.4.

5.2 An Attention-based Parallel Transformer Framework

In this section, a detailed description of the proposed Attention-based Parallel Transformer Framework (PaTF) is presented, answering the research question 2.1 The PaTF leverages both global and local visual features, combining them with rich textual information to achieve accurate video moment retrieval. We suppose that a single video $\mathbf{X}$ could be represented as a sequence of frames $\mathbf{X} = \{x_1, x_2, \ldots, x_T\}$, where $\mathbf{T}$ is the total number of frames in the video. Within the scope of this research, $\mathbf{T}$ can be either fixed to a specific number or varied across different videos. The ultimate goal of the VMR task is to localise the time frame $\mathbf{Y} = (s_i, e_i)$ of a moment within the untrimmed video $\mathbf{X}$ that best aligns with the input query $\mathbf{Q}$, where the moment $\mathbf{Y}$ is represented by the start time s_i and the end time e_i of the moment ($s_i < e_i$). In other words, to make it more approachable, we consider the task into a regression problem for distances as below:

$$\mathbf{X} = \{x_1, x_2, \ldots, x_T\} \rightarrow \hat{\mathbf{Y}} = (\hat{d}_i^s, \hat{d}_i^e) = (i - d_i^s, d_i^e - i) \tag{5.1}$$

where d_i^s and d_i^e are the ground-truth start and end time of the moment $\mathbf{Y}$, respectively, and i is the current position of the frame in the video $\mathbf{X}$.

5.2.1 Overview

To have a closer look at the PaTF framework, we illustrate the three stages in Figure 5.1, including a feature extraction stage, a transformer stage, and a regression stage. The framework takes each video $\mathbf{X}$ and a corresponding textual query $\mathbf{Q}$ as input and outputs a ranked list of time intervals in decreasing order of relevance. The feature extraction stage is responsible for extracting the representations of video keyframe

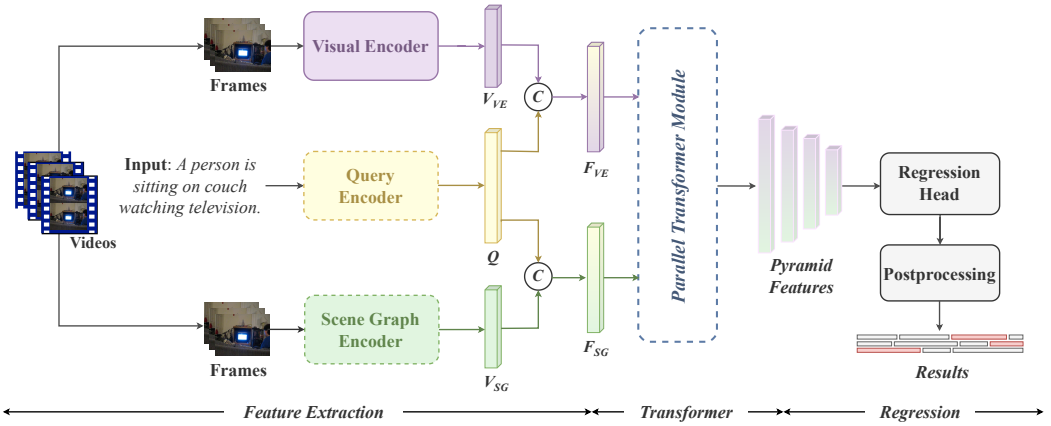


Figure 5.1: An overview of PaTF, which consists of three main stages, including a feature extraction stage, a transformer stage, and a regression stage. The feature extraction stage begins with the input video frames are processed by a Visual Encoder and a Scene Graph Encoder separetely, while the input query is processed by a Query Encoder. These features are then concatenated and passed through a Parallel Transformer Module, followed by a Regression Head to predict the target boundaries.

sequences and input text query tokens individually. In particular, the two-stream visual features are obtained in a local and global manner using the pre-trained Scene Graph Encoder and Visual Encoder, respectively. In the meantime, the text features are embedded by the pre-trained Query Encoder, transforming input text tokens into rich feature representations. Following this, the interaction between the two visual-textual-joint features, each constructed by the concatenation of one visual stream (either from the visual encoder or the scene graph encoder) with the textual features (obtained from the query encoder), is captured by the Parallel Transformer Module in the transformer stage. The Parallel Transformer Module is designed with three variants: Dual Self-Attention block (SA), Dual Cross-Attention block (CA), and Combined-Attention block (SA & CA). The output of this module is a set of Pyramid Features, representing multi-scale feature representations, capturing information at different levels of abstraction and temporal scales within the video. Ultimately, in the regression stage, a Regression Head is employed to process the results from the previous stage, generating a ranked list of potential candidates. These candidates then undergo a postprocessing step, where the precise onsets and offsets of the target

moments are extracted.

5.2.2 Feature Extraction

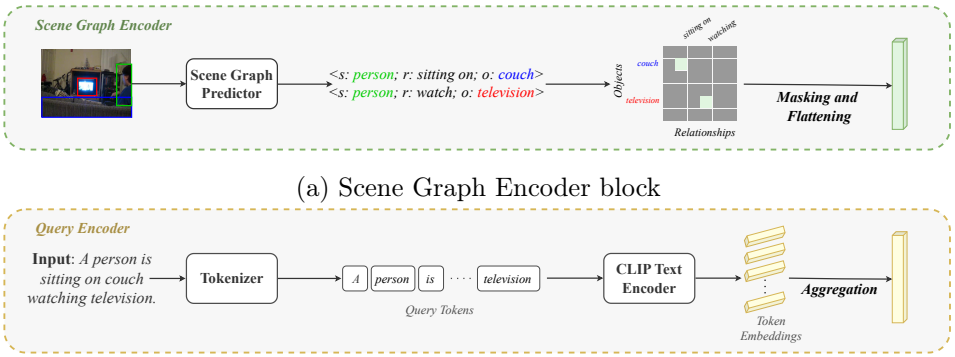
The feature extraction stage plays a crucial role in the framework as it is in charge of extracting the informative visual and textual features from the input video and query. Visual features are obtained in a dual context: visual representation via the Visual Encoder and high-level semantic cues via the Scene Graph Encoder. Simultaneously, high-dimensional textual features are extracted from query tokens by the Query Encoder. The visual-textual joint features are then constructed by concatenating the visual and textual features before being fed into the Parallel Transformer Module.

5.2.2.1 Visual Encoder

With the aim of capturing the global content of an image or video frame, the visual embedding model transforms the video frame sequence into a high-dimensional feature vector $\mathbf{V}_{VE} \in \mathbb{R}^{\mathbf{T}\times\mathbf{D}}$, where $\mathbf{T}$ and $\mathbf{D}$ denote the total number of frames for the video $\mathbf{X}$ and the dimension of each feature vector (768). Specifically, we used the OpenCLIP image embedding model [150] with a Vision Transformer [152] pre-trained at 336-pixel resolution ($ViT - L/14@336$) on the LAION-5B dataset [201] to extract the visual features. In addition to the OpenCLIP model, we also use pre-extracted temporal features I3D [183] provided by MIGCN [174]. The I3D features are extracted from the video frames using the pre-trained I3D model (Two Stream Inflated 3D ConvNets) [183], which is a widely used model for action recognition tasks.

5.2.2.2 Scene Graph Encoder

To capture the interaction between objects in an image, we use scene graphs, graph-based representations of the objects and their relationships, to represent the structural layout of the image. Particularly, a scene graph $\mathbf{G}$ of a single frame consists of a set of triplets $\{s_i, r_i, o_i\}_{i=1}^{|\mathbf{G}|}$, where s_i, o_i, and r_i are the subject,



(a) Scene Graph Encoder block

(b) Query Encoder block

Figure 5.2: Query Encoder and Scene Graph Encoder block for Feature Extraction.

object, and relationship between s_i and o_i of the i^{th} triplet, respectively.

In the stage of scene graph generation in Figure 5.2a, we utilised the pre-trained version of the Neural Motifs model [206], which is developed based on the Faster R-CNN [127] object detector, as the scene graph predictor. For scene graph generation, we use the Action Genome dataset [207], built upon the Charades dataset, providing both action labels and spatio-temporal scene graph labels. Once we have the predicted scene graph, we generate a confidence matrix $\mathbf{C} \in \mathbb{R}^{|\mathbf{O}|\times|\mathbf{R}|\times 3}$ in which $|\mathbf{O}|$ and $|\mathbf{R}|$ are the number of classes of objects and relationships, and the third dimension is about the confidence scores. In the Action Genome dataset, $|\mathbf{O}| = 35$ and $|\mathbf{R}| = 25$. Then, a mask is applied to the confidence score matrix to filter out all irrelevant information before being flattened to return the corresponding scene graph features $\mathbf{V}_{SG} \in \mathbb{R}^{|\mathbf{O}||\mathbf{R}|}$. Specifically, in the mask $\mathbf{M} \in \mathbb{R}^{|\mathbf{O}|\times|\mathbf{R}|}$, the rows corresponding to classes appearing in the text query are filled with ones, otherwise zeros.

5.2.2.3 Query Encoder

The query encoder, as illustrated in Figure 5.2b, represents a fundamental part of our framework, which embeds the input textual query $\mathbf{Q}$ into a fixed-dimensional vector representation for further visual-textual matching. The process begins by decomposing the input query into a list of distinct query tokens $q = \{q_1, q_2, ..., q_K\}$,

with each token representing a semantic unit of the query. Following this, a pre-trained OpenCLIP model [150] is adopted to convert tokens into numerical feature vectors $\tilde{q} = \{\tilde{q_1}, \tilde{q_2}, ..., \tilde{q_K}\}, \tilde{q} \in \mathbb{R}^{\mathbf{K}\times\mathbf{D}}$, where $\mathbf{K}$ and $\mathbf{D}$ denote the number of total tokens and the dimension of each token-embedded vector, respectively. The final step of the query encoding process employs an aggregation layer (a learnable linear layer) to unite all token features into one final feature $\mathbf{Q} \in \mathbb{R}^{1\times\mathbf{D}}$, which represents the overall query. Particularly, this aggregation layer employs learnable parameters $w \in \mathbb{R}^{\mathbf{D}\times 1}$ and averages $\tilde{q}$ with a series of weights $\alpha \in \mathbb{R}^{\mathbf{K}\times 1}$ as below:

$$\mathbf{Q} = \alpha^T \cdot \tilde{q}, \text{ where } \alpha = \text{Softmax}(\tilde{q} \cdot w) \tag{5.2}$$

5.2.2.4 Visual-textual joint features

Once the features are extracted as described above, each pair of visual-textual features is concatenated as the input to two streams of the parallel transformer module. The first joint feature, $\mathbf{F}_{VE}$, is constructed by the combination of features of the visual encoder V_{VE} and the query encoder $\mathbf{Q}$, capturing the direct relationship between video frames and text. Simultaneously, we create the second joint feature, $\mathbf{F}_{SG}$, by concatenating the output from the graph encoder V_{SG} and query encoder $\mathbf{Q}$. In doing so, we highlight the object interaction within the image frames and the text query. These features are constructed as follows:

$$\mathbf{F}_{VE} = Concat(V_{VE}, \mathbf{Q}) \tag{5.3}$$

$$\mathbf{F}_{SG} = Concat(V_{SG}, \mathbf{Q}) \tag{5.4}$$

5.2.3 Parallel Transformer Module

The transformer architecture, introduced by Vaswani et al. [10], is a powerful model that can model long data sequences, such as sentences or video frames, without losing their context with the use of attention mechanisms. Unlike traditional recurrent neural networks (RNNs) [208] or long short-term memory

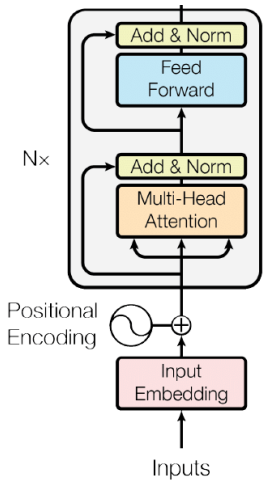
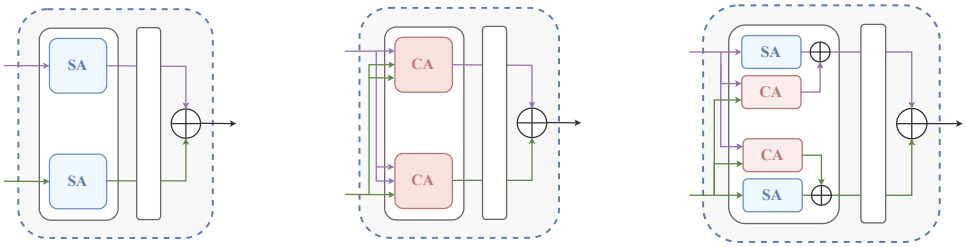


Figure 5.3: The structure of Transformer Encoder, reported in [10]. The encoder begins with an embedding layer, a positional encoding layer, followed by a stack of N identical layers, each consisting of a multi-head attention sub-layer and a position-wise feed-forward layer.

networks (LSTMs) [209], which process sequences sequentially, transformer models, as illustrated in Figure 5.3 and Figure 5.6, enable the model to capture long-range dependencies and context by concentrating on different parts of the input sequences. With these advancements, transformers have been widely used in various tasks, such as their original use in machine translation [10, 210], language understanding [211, 212, 213], text generation [214, 215], and even in computer vision tasks such as image recognition [152, 216], and video understanding [217, 218]. In the context of video moment retrieval, transformer-based models have demonstrated promising results in capturing the temporal dependencies and interactions between video frames and textual queries [53, 56, 58, 57]. Building upon the success of transformer models, we propose a Parallel Transformer Framework (**PaTF**) to jointly learn from both visual and textual features, addressing the lack of relational semantic cues between objects in the current VMR frameworks. Specifically, the parallel transformer module is comprised of two parallel branches: one for processing the visual features in the global context of images using a pre-trained model and the other one for encoding relationship cues in a structured and explicit way by the scene graph representations that capture

the semantic and spatial relationships between objects using a graph neural network. The two branches are then processed to the parallel transformer module to learn the representations of the visual-textual joint features simultaneously. As depicted in Figure 5.4, we explore three implementation variants for parallel transformers: Dual Self-Attention block (SA), Dual Cross-Attention block (CA), and Combined-Attention block (SA & CA). Self-Attention is inherited from the standard transformer architecture [10], the others are slightly modified from the Self-Attention block to adopt the Cross-Attention mechanism. The detailed implementation of each block is described in the following sections.



(a) (b) (c)

Figure 5.4: Parallel Transformer module: a Dual Self-Attention block (SA), b Dual Cross-Attention block (CA), and c Combined Attention block (SA & CA).

5.2.3.1 Positional Encoding

Prior to feeding the visual-textual joint features into the transformer module, we incorporate positional encoding into the input features to provide the model with information about the position of each token in the sequence. Following the original transformer architecture [10], sine and cosine functions are used to encode these positions as follows:

$$PE_{(pos,2i)} = \sin\left(\frac{pos}{10000^{2i/\mathbf{D}}}\right) \tag{5.5}$$

$$PE_{(pos,2i+1)} = \cos\left(\frac{pos}{10000^{2i/\mathbf{D}}}\right) \tag{5.6}$$

where *pos* is the position of the token in the sequence, i is the dimension of the positional encoding, and D is the dimension of the feature vector. The positional encoding is then added to the input features before feeding them into the transformer encoder module, shown in Figure 5.3.

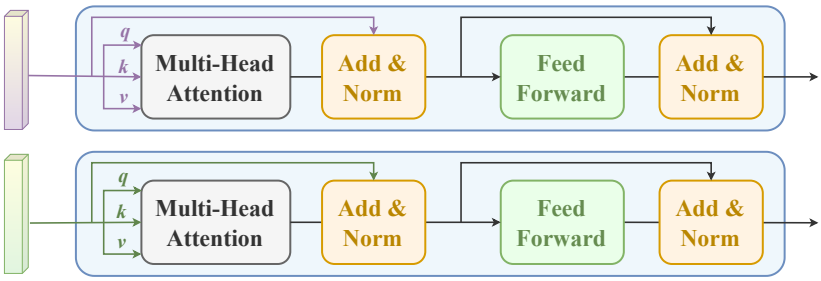
5.2.3.2 Attention Module

The attention mechanism serves as a fundamental component in the transformer architecture, functioning as a mapping between queries and key-value pairs to an output. At its core, the attention module produces a weighted sum of the value, where the weight matrix, or affinity matrix, is determined by the query and its corresponding key. The key differences between the Self-Attention and the Cross-Attention module are the input modality and operational mechanisms. While the self-attention unit takes only one modality as input and computes the attention weights for itself to model the intra-modal refinement, the cross-attention unit considers interactions between different input modalities. In this dissertation, we integrate attention mechanisms as a feature encoder [10] to model the relationship between visual and textual modalities of input data. Specifically, we use a stack of N identical layers, each consisting of a multi-head attention sub-layer and a position-wise feed-forward layer, illustrated in Figure 5.5.

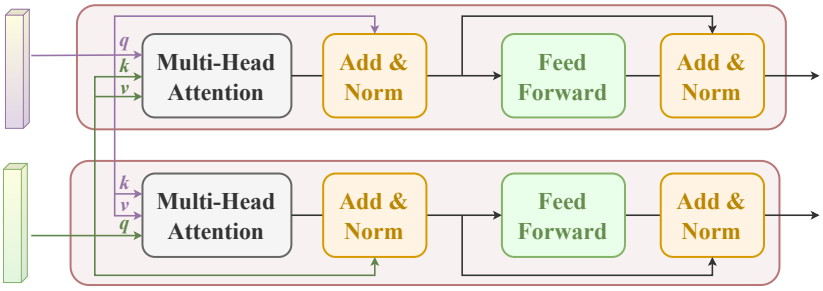
Self-Attention Module

The Self-Attention module is implemented to capture the intra-modality relationship between the visual and textual features. Figure 5.5a visualises the self-attention mechanism unit, processing one single input modality and computing the attention weights through self-interactions.

Given an input sequence of fragments $\mathbf{F} = \{f_1; f_2; ...; f_k\}$, where each fragment $f_i \in \mathbb{R}^{1\times d_f}$ and the stack $\mathbf{F} \in \mathbb{R}^{k\times d_f}$ represent either image regions or sentence words. We first project them into query, key, and value: $\mathbf{Q}_F = \mathbf{F}\mathbf{W}_i^Q$, $\mathbf{K}_F = \mathbf{F}\mathbf{W}_i^K$, $\mathbf{V}_F = \mathbf{F}\mathbf{W}_i^V$, with i denoting the i-th head. The "Scaled Dot-Product Attention" weight matrix, visualised in Figure 5.6a, is calculated by the dot product of the



(a) Self-Attention unit

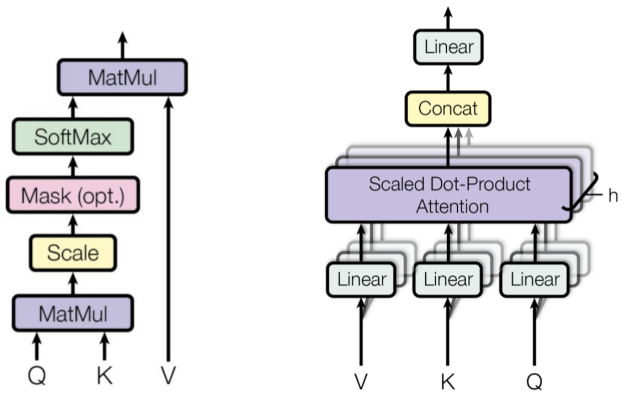


(b) Cross-Attention unit

Figure 5.5: Attention mechanisms used in the PaTF architecture.

query and all keys, then scaled by the $\sqrt{d_k}$, where d_k is the dimension of the key, and applied a softmax function to normalise the weights as:

$$\text{Attention}(\mathbf{Q}_F, \mathbf{K}_F, \mathbf{V}_F) = \text{softmax}\left(\frac{\mathbf{Q}_F \mathbf{K}_F^T}{\sqrt{d_k}}\right)\mathbf{V}_F \tag{5.7}$$



(a) Scaled Dot-Product Attention (b) Multi-Head Attention

Figure 5.6: Scaled Dot-Product Attention and Multi-Head Attention [10].

Multi-head attention allows the model to simultaneously attend to information from different representation subspaces at different positions. After that, the multi-head attention is obtained by computing the values of all single attention heads and concatenating them together using the following equations:

$$\text{MultiHead}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{concat}(\text{head}_1, ..., \text{head}_h)\mathbf{W}^O \tag{5.8}$$

$$\text{where head}_i = \text{Attention}(\mathbf{F}\mathbf{W}_i^Q, \mathbf{F}\mathbf{W}_i^K, \mathbf{F}\mathbf{W}_i^V) \tag{5.9}$$

where the projections are parameter matrices $\mathbf{W}_i^Q \in \mathbb{R}^{d_f \times d_k}$, $\mathbf{W}_i^K \in \mathbb{R}^{d_f \times d_k}$, $\mathbf{W}_i^V \in \mathbb{R}^{d_f \times d_v}$, and the output matrix $\mathbf{W}^O \in \mathbb{R}^{hd_v \times d_k}$, and h is the number of the heads.

Cross-Attention Module

While the intra-modality relationship of visual-textual-feature pairs is captured by the Self-attention module, their inter-modality relationship is modeled by the Cross-attention module. The cross-attention module enables fine-grained alignment between different input modalities, which is crucial to matching the visual content with the textual query [219]. The cross-attention mechanism is visualised in Figure 5.5b. Building upon the foundations of the self-attention mechanism, the cross-attention module also considers the interaction between the visual and textual features, focusing on the inter-modality relationship. Specifically in our framework, the input sequence of fragments consists of two modalities, visual features $\mathbf{F}_{VE}$ and textual features $\mathbf{F}_{SG}$, is defined as:

$$\mathbf{Y} = \begin{pmatrix} \mathbf{F}_{VE} \\ \mathbf{F}_{SG} \end{pmatrix} = \{f_{ve1}; ...; f_{vek}; f_{sg1}; ...; f_{sgn}\} \tag{5.10}$$

where $\mathbf{Y} \in \mathbb{R}^{(k+n)\times d_x}$, is passed into the cross-attention module. The query, key, and value for the fragments are formed as:

$$\mathbf{Q}_Y = \mathbf{Y}\mathbf{W}^{\mathbf{Q}} = \begin{pmatrix} \mathbf{F}_{VE} \cdot \mathbf{W}^{\mathbf{Q}} \\ \mathbf{F}_{SG} \cdot \mathbf{W}_i^{\mathbf{Q}} \end{pmatrix} = \begin{pmatrix} \mathbf{Q}_{VE} \\ \mathbf{Q}_{SG} \end{pmatrix}, \tag{5.11}$$

$$\mathbf{K}_Y = \mathbf{Y}\mathbf{W}^{\mathbf{K}} = \begin{pmatrix} \mathbf{F}_{VE} \cdot \mathbf{W}^{\mathbf{K}} \\ \mathbf{F}_{SG} \cdot \mathbf{W}_i^{\mathbf{K}} \end{pmatrix} = \begin{pmatrix} \mathbf{K}_{VE} \\ \mathbf{K}_{SG} \end{pmatrix}, \tag{5.12}$$

$$\mathbf{V}_Y = \mathbf{Y}\mathbf{W}^{\mathbf{V}} = \begin{pmatrix} \mathbf{F}_{VE} \cdot \mathbf{W}^{\mathbf{V}} \\ \mathbf{F}_{SG} \cdot \mathbf{W}_i^{\mathbf{V}} \end{pmatrix} = \begin{pmatrix} \mathbf{V}_{VE} \\ \mathbf{V}_{SG} \end{pmatrix} \tag{5.13}$$

Following the same procedure as the self-attention module in Equation 5.7, the cross-attention module computes the attention weights as:

$$\text{Attention}(\mathbf{Q}_Y, \mathbf{K}_Y, \mathbf{V}_Y) = \text{softmax}\left(\frac{\mathbf{Q}_Y \mathbf{K}_Y^T}{\sqrt{d}}\right)\mathbf{V}_Y \tag{5.14}$$

To simplify the notation, we drop the softmax function and the scaling factor from the above equation without loss of generality. The Scaled-Dot Product Attention is calculated as:

$$\mathbf{Q}_Y \mathbf{K}_Y^T \cdot \mathbf{V}_Y = \begin{pmatrix} \mathbf{Q}_{VE} \\ \mathbf{Q}_{SG} \end{pmatrix} \begin{pmatrix} \mathbf{K}_{VE}^T & \mathbf{K}_{SG}^T \end{pmatrix} \cdot \begin{pmatrix} \mathbf{V}_{VE} \\ \mathbf{V}_{SG} \end{pmatrix} \tag{5.15}$$

$$= \begin{pmatrix} \mathbf{Q}_{VE}\mathbf{K}_{VE}^T & \mathbf{Q}_{VE}\mathbf{K}_{SG}^T \\ \mathbf{Q}_{SG}\mathbf{K}_{VE}^T & \mathbf{Q}_{SG}\mathbf{K}_{SG}^T \end{pmatrix} \cdot \begin{pmatrix} \mathbf{V}_{VE} \\ \mathbf{V}_{SG} \end{pmatrix} \tag{5.16}$$

$$= \begin{pmatrix} \mathbf{Q}_{VE}\mathbf{K}_{VE}^T\mathbf{V}_{VE} + \mathbf{Q}_{VE}\mathbf{K}_{SG}^T\mathbf{V}_{SG} \\ \mathbf{Q}_{SG}\mathbf{K}_{SG}^T\mathbf{V}_{SG} + \mathbf{Q}_{SG}\mathbf{K}_{VE}^T\mathbf{V}_{VE} \end{pmatrix}. \tag{5.17}$$

As previously mentioned, $\begin{pmatrix} \mathbf{F}_{VE}^* \\ \mathbf{F}_{SG}^* \end{pmatrix} = \mathbf{Q}_Y \mathbf{K}_Y^T \cdot \mathbf{V}_Y$, the updated output of the cross-attention module is:

$$\mathbf{F}_{VE}^* = \{f_{ve1}^*; ...; f_{vek}^*\} = \mathbf{Q}_{VE}\mathbf{K}_{VE}^T\mathbf{V}_{VE} + \mathbf{Q}_{VE}\mathbf{K}_{SG}^T\mathbf{V}_{SG} \tag{5.18}$$

$$\mathbf{F}_{SG}^* = \{f_{sg1}^*; ...; f_{sgn}^*\} = \mathbf{Q}_{SG}\mathbf{K}_{SG}^T\mathbf{V}_{SG} + \mathbf{Q}_{SG}\mathbf{K}_{VE}^T\mathbf{V}_{VE} \tag{5.19}$$

Position-wise Feed-Forward Module

In addition to the attention mechanism, the position-wise feed-forward sub-layer is applied to each position separately and identically in our encoder to capture long-term dependencies. This sub-layer consists of two fully connected layers, with a ReLU activation function [220] in between, defined as follows:

$$\mathrm{FFN}(x) = \mathrm{ReLU}(x\mathbf{W}_1 + b_1)\mathbf{W}_2 + b_2, \tag{5.20}$$

where $x \in \mathbb{R}^{1\times d_x}$, $\mathbf{W}_1 \in \mathbb{R}^{d_x\times d_x}$, $\mathbf{W}_2 \in \mathbb{R}^{d_x\times d_x}$, $b_1 \in \mathbb{R}^{1\times d_x}$, and $b_2 \in \mathbb{R}^{1\times d_x}$. While the linear transformations remain the same across different positions within the same layer, the parameters of the feed-forward sub-layer vary from layer to layer, allowing the model to learn different features at different levels of abstraction. Following both multi-head attention and position-wise feed-forward network, the architecture employs residual connections [131] and layer normalisation [221] to facilitate optimisation.

5.2.4 Regression Stage

5.2.4.1 Feature Pyramid Network (FPN)

Leveraging the pyramidal structure of ConvNet features, the Feature Pyramid Network (FPN) [222] creates a feature pyramid with strong semantics from various levels. Concretely, the FPN's construction involves a bottom-up pathway and a top-down pathway. While the former, with low-resolution levels, captures more global content and represents richer semantic meanings of the data, the latter, with high-resolution levels, highlights local information and more accurate spatial information. Then, lateral connections are used as bridges to connect the two pathways, allowing the model to merge the features of the same spatial size from different levels.

As described, FPN helps extract multi-scale features from the input, here, FPN is implemented to predict candidate moments by combining the features from different

levels and estimating the start and end times of the relevant segment. Particularly, we used six layers of feature pyramids, each a convolution layer with different strides and kernel sizes, to obtain features from various levels.

5.2.4.2 Moment Boundary Regression Head

The objective of the regression head is to examine all levels of features given in the feature pyramid and predict the distance to the onsets and offsets of the moments. The regression head, guided by relevance scores, provides valuable insights into how the model weighs different cues to determine the temporal boundaries of moments in the retrieval task. To that end, the regression head is implemented with a 1D convolutional network, including three 1D convolutional layers, two normalisation layers [221], and a ReLU activation [220].

5.2.4.3 Postprocessing

The postprocessing step is responsible for transforming the raw frame-level predictions from Regression Head into meaningful temporal moments. Once the distances are obtained, we align all frame outputs into the start and end frame orders. The moment segments are obtained as follows:

$$(s, e)_i = (\max(i - d_i^s, 0), \min(i + d_i^e, n)) \tag{5.21}$$

where n is the number of frames from that video and $(s, e)_i$ is the predicted start and end frame of the moment, d_i^s and d_i^e are the distances from the current frame position i to the start and end of the predicted moment, respectively. In doing so, a list of candidates (s, e) is ranked based on their relevance score.

As the regression head predicts the distance to the moment boundaries via multiple levels of features, we noticed several overlapping predictions. To obtain multiple potential candidate moments (top k) and avoid overlapped predictions, we adopt Soft Non-Maximum Suppression (SoftNMS) [223], a postprocessing

algorithm, as a postprocessing step to select moments with the highest relevance score. Unlike traditional NMS [224] that completely eliminates the overlapped predictions using a hard threshold, SoftNMS simply reduces the confidence score of the overlapped predictions. This allows the model to keep the overlapped predictions with lower confidence scores, which may still be relevant to the target moments.

5.2.5 Loss Function

For the VMR task, the final loss function $\mathcal{L}_{final}$ is formulated as a weighted combination of two distinct components: (1) Focal loss $\mathcal{L}_{rel}$ for relevance score and (2) $L1$ loss $\mathcal{L}_{reg}$ for the regression head. The former is used to handle the problem of class imbalance in object detection, while the latter computes the loss value based on the absolute distance between the predicted output and the ground-truth moment boundaries. The final loss function for each input is defined below:

$$\mathcal{L}_{final} = \mathcal{L}_{rel} + \lambda \times \mathcal{L}_{reg} \tag{5.22}$$

where λ is a weight constraint to balance between the two aforementioned losses. λ is practically suggested as 1.

5.3 Experiments and Results

5.3.1 Implementation Details

5.3.1.1 Dataset and Evaluation

We evaluated our framework on the Charades-STA dataset [48], an extension of the original Charades dataset [225], designed especially for VMR tasks by collecting the sentence temporal annotations for video segments. This dataset captures daily activities at home[1], containing **6,672** videos (**5,338** video for training and **1,334**

[1] `https://prior.allenai.org/projects/charades`

videos for testing) and **16,128** pairs of textual description and action segments (**12,408** pairs for training and **3,720** pairs for testing). Each video has a duration of approximately 30 seconds and an average of 2.4 moments (each lasting approximately 8.2 seconds). Regarding the evaluation metric, we leverage the Recall at k with the $IoU = v$ metric, as described in Equation 3.9. In this work, we employ the recall with k values of 1 and 5, and with IoU thresholds of 0.5 and 0.7 for evaluation.

We decided to validate our framework on this dataset since it is widely used in the VMR community and the only VMR dataset that has scene graph annotations. Nevertheless, since the dataset is related to daily activities, it is closely related to the lifelog data that we are interested in. Therefore, we believe that the Charades-STA dataset is a suitable benchmark for evaluating our framework.

5.3.1.2 Model Setting

Our model is developed as a proposal-free VMR technique, inspired by ActionFormer [180] with PyTorch [226] for the VMR task. Our model is trained on one NVIDIA GTX 1080ti GPU with a total batch size of 16 for 10 epochs. During the training process, we use the Adam optimiser [227] to minimise the final loss (calculated in Equation 5.22) with the initial learning rate of 0.001. The visual features are extracted by the pre-trained OpenCLIP model [150] and the I3D model [228], and the text features are extracted by the pre-trained OpenCLIP model. Motifs [206] is used as the model for scene graph generation (SGG) and trained on the Action Genome dataset [207]. Within this research, both training losses are equally weighted, which means the value of λ in Equation 5.22 is 1.

Sampling techniques adopted from [180] with a fixed number of 128 evenly spaced frames sampled from each video in the Charades-STA dataset. The regression range is specifically set for each level of the feature pyramid. For this purpose, we set the range of the regression head to be $[0, 4], [4, 8], [8, 16], [16, 32], [32, 64], [64, \infty]$, from the lowest to the highest level of the feature pyramid.

5.3.2 Framework Performance

5.3.2.1 Comparison with SOTA methods

To investigate the effectiveness of our framework, we make a comparison between the performance of PaTF and other state-of-the-art approaches conducted on the Charades-STA dataset. Particularly, we compared our framework with the proposal-based techniques, CTRL [48], 2D-TAN [175], MMRG [50], FVMR [54], and with the proposal-free techniques, such as VSLNet [52], Moment-DETR [56], QD-DETR [58], MH-DETR [57], UVCOM [179], LGI [177], UnLoc [178]. CTRL uses a sliding window strategy to generate video moment candidates, while MMRG and FVMR apply a graph technique to capture moment relationships. 2D-TAN constructs the 2-D temporal map to model the temporal relationships between video frames. In the meantime, Moment-DETR, QD-DETR, and MH-DETR exploit the DETR transformer architecture [182] as their backbone, while UnLoc is a method that uses a transformer-based architecture with a visual-semantic embedding. UVCOM unifies the visual and linguistic information to localise the target moments, while LGI integrates multi-level architecture to capture bi-modal interactions between the video segments and the textual queries. VSLNet considers the input video as a text passage and uses a multimodal span-based QA framework to solve the VMR task.

The performance comparison presented in Table 5.1 demonstrates the effectiveness of our PaTF framework compared to existing state-of-the-art methods in the Charades-STA dataset. In this table, the best results are highlighted in bold, while the suboptimal values are underlined. In general, the proposal-free methods, including our framework, show an advantage over the proposal-based methods in terms of both $R@1$ and $R@5$ with both $IoU = 0.5$ and $IoU = 0.7$. Notably, our framework outperforms all existing methods in all metrics, with the highest $R@1 - IoU = 0.5$ of 65.8 and $R@1 - IoU = 0.7$ of 45.1. Diving deeper into comparative analysis, our model achieves a substantial improvement of roughly 3%

Table 5.1: Performance comparison between PaTF and SOTA methods on Charades-STA dataset. There are four types of visual features: SlowFast (SF), VGG, CLIP, OpenCLIP, and I3D features. The best and suboptimal values are highlighted in bold and underlined, respectively.

Type	Method	Visual Features	Scene Graphs	$R@1$		$R@5$	
				$IoU = 0.5$	$IoU = 0.7$	$IoU = 0.5$	$IoU = 0.7$
Proposal-based	CTRL [48]	I3D	–	22.6	8.9	58.9	29.5
	2D-TAN [175]	VGG	–	40.9	22.9	83.8	50.4
	MMRG [50]	I3D	–	44.3	–	60.2	–
	FVMR [54]	I3D	–	55.0	33.7	89.2	57.2
Proposal-free	VSLNet [52]	I3D	–	54.2	35.2	–	–
	Moment-DETR [56]	CLIP	–	55.7	34.2	–	–
	QD-DETR [58]	SF+CLIP	–	57.3	32.6	–	–
	MH-DETR [57]	I3D	–	56.4	35.8	–	–
	UVCOM [179]	SF+CLIP	–	59.3	36.6	–	–
	LGI [177]	I3D	–	59.5	35.5	–	–
	UnLoc-L [178]	CLIP	–	60.8	38.4	88.2	61.1
	PaTF (Ours)	OpenCLIP	✓	63.6	40.8	<u>90.7</u>	65.3
	PaTF (Ours)	I3D	✓	<u>64.0</u>	<u>43.4</u>	**90.9**	<u>66.5</u>
	PaTF (Ours)	I3D + OpenCLIP	✓	**65.8**	**45.1**	90.6	**68.2**

for $R@1 - IoU = 0.5$ by incorporating scene graph representation in one additional branch, along with similar visual features and backbone with the latest approach, UnLoc. Note that when using the combination of the I3D and OpenCLIP features as visual features, we obtain the best results ($R@1 - IoU = 0.5$ of 65.8 and $R@1 - IoU = 0.7$ of 45.1). Notably, this version surpasses UnLoc, with an increase of 5% for $R@1 - IoU = 0.5$, and approximately 7% for $R@1 - IoU = 0.7$. These large performance gaps prove our stronger localisation ability as compared to the SOTA methods. Extensive ablation studies are conducted to evaluate the contribution of each component in our framework, detailed in the Appendix C. In particular, those studies investigate the impact of the visual features and scene graph representations, the scene graph generation, the transformer module's architecture, and the technique of vision-text fusion on the overall performance of the framework.

To the best of our knowledge, our framework is the first to use two parallel branches with the standard transformer backbone to achieve competitive performance with other techniques. This indicates the fact that beyond the advance of the transformer model, which has already been proven in other work, the semantics represented via scene graphs is a key factor that benefits the process of localising the target video moments in our framework.

5.3.2.2 Qualitative Analysis

To obtain an intuitive perception of the impact of the scene graph representations, we show two visualisations of the prediction of the backbone and PaTF framework, which can be seen in Figure 5.7. These qualitative examples show the failure cases of the backbone without the scene graphs. One potential reason for this could be that it only identifies the objects in the frame but not the relations or interactions among them. Nevertheless, we demonstrate that our framework can predict the moment boundaries more accurately by leveraging the combination of visual feature representations and scene graph representations. The backbone model's

limitations become apparent in cases where the objects are visible throughout the video, but the action only occurs at some points. In the first example, "*the person*" and "*the sandwich*" are visible in almost every frame throughout the video, but the action of putting the sandwich only occurs at the beginning of the video. The backbone model, however, fails to do so, as it only recognises the objects without considering when the person actually "*put*" the sandwich on the table. On the other hand, the PaTF enriched the visual features with the semantic cues from the scene graph representations and successfully highlighted the relational information, resulting in more accurate boundaries. A similar challenge is presented in the second example, where the backbone cannot identify the completion of action "*open the bag*", leading to the wrong offset predictions. These qualitative observations determine the integration of scene graphs, providing a richer understanding of not just what objects are presented but also how they interact with each other.

Figure 5.7: Qualitative comparison of top-1 examples on Charades-STA dataset (best viewed in colors). The three colored boxes are the moment boundaries corresponding to the input query. The ground truth is in blue, while the predictions from the baseline and PaTF are in red and green, respectively.

5.4 Discussion and Conclusion

With these outperformed results, we demonstrated the effectiveness of the proposed framework in localising the target moments in the video data. The incorporation of scene graph representation in the visual-textual joint features has significantly

improved the performance of the model. It is worth noting that the integration of the parallel transformer module also contributes to the success of the framework. This framework can be a promising approach to the VMR task, which can be further extended to other lifelog retrieval tasks thanks to their similarity in terms of the data format and retrieval objective. The integration of scene graph representations in the visual-textual joint features, together with the parallel transformer module, has significantly improved the performance of the model by more than 5% and 7% in terms of $R@1-IoU = 0.5$ and $R@1-IoU = 0.7$, as compared to the latest methods. With this promising results and insights gained from this research, we addressed the Research Question 2 (**RQ2**): *How can a state-of-the-art transformer-based Video Moment Retrieval technique be designed to effectively localise target moments in video sequences?* by proposing an attention-based parallel transformer framework for the Video Moment Retrieval task, named **PaTF**.

5.5 Chapter Summary

In this chapter, we attempt to look for the answer to Research Question 2 (**RQ2**): *How can a state-of-the-art transformer-based Video Moment Retrieval technique be designed to effectively localise target moments in video sequences?*. To address this question, we proposed an attention-based parallel transformer framework for the Video Moment Retrieval task, named **PaTF**, which the **RQ2** is addressed. This framework's innovative architecture leverages the combination of two streams of data, visual content and semantic cues, to search for the target moments in the video data. Specifically, the scene graph representations are used as one stream of visual content to model the high-level semantic information alongside low-level visual features extracted from the video frames. The parallel processing architecture further enhances the relationship between the visual content and the textual query, capturing both intra-modality and inter-modality relationships between them. The comprehensive experiments and ablation studies on the

Charades-STA dataset have shown that the adoption of scene graph representation to enrich the visual representation results in performance improvement as compared to the latest methods. The promising results and insights gained from this research demonstrate the potential of combining transformer-based architectures with rich semantic representations for improved video content analysis, making it a promising approach to adopt to lifelog retrieval tasks. Subsequently, the next chapter discusses the implementation of the proposed framework in to lifelog retrieval field.

Chapter 6

Moment-based Lifelog Retrieval System

6.1 Introduction

Recent research in cognitive science has shown that human memory operates on the level of sequences of events or moments, rather than individual images [15, 60]. This understanding suggests that human memory naturally segments continuous experiences into coherent events or moments. However, the majority of existing lifelog retrieval systems are constructed based on retrieving individual images, which somehow misaligns with natural human memory processes. To bridge this gap, the moment-based lifelog retrieval system is increasingly important to provide a more intuitive and contextually rich method for accessing personal digital memories. The demand for moment-based retrieval systems stems from cognitive theories of human memory and practical application requirements. When users interact with their personal archive to seek past moments or activities, they are more likely to be seeking the complete events with contextual information about what they were doing, where they were, and who they were with, rather than just viewing standalone images. Image-based approaches, however, only concentrate on that image's content but fail to capture the context of the moment or the event that the image belongs to. Considering an action of "*making coffee in the kitchen*", one single image of the coffee machine might not be informative enough to recall the whole event. However, a sequence of images showing the process of making coffee is more informative and

detailed, describing multiple steps and objects involved.

In Chapter 5, we have examined the effectiveness of the proposed transformer-based video moment retrieval (VMR) approach applied to enriched data representations in the context of retrieving moments from video data. The importance of semantic information in the form of scene graphs has been highlighted as a key factor in capturing the context of the moments along with the visual embedding features, which result in a more informative and detailed representation of the moments. The parallel transformer module has been introduced to model the joint representation of the visual and textual features, which has shown promising results in the VMR tasks as compared to the existing state-of-the-art techniques. As lifelog data has a similar nature of continuous and chronological format to video data, adopting advanced VMR techniques to lifelog data is expected to improve the retrieval performance of the current lifelog retrieval systems, which are mainly based on image retrieval.

A key challenge in this adaptation lies in scaling moment retrieval techniques from retrieving moments in one single video to handling large-scale lifelog collections. To address this challenge, I propose a hierarchical moment-based retrieval approach modified from the PaTF framework developed in Chapter 5, named **PaTFLifelog**. This framework consists of two stages: the period retrieval stage, which identifies the top periods of the day that correspond to the query, and the moment based, which finds the moment's boundaries among those selected periods. The following sections detail the technical approach, experimental study, and the comparative evaluation of the proposed system against the baseline lifelog retrieval system.

The result of this chapter's experiment will answer the **RQ3**: *To what extent can the proposed transformer-based retrieval approaches, when applied to lifelog retrieval, improve the lifelog retrieval performance of past moments when compared to existing SOTA interactive retrieval systems?*

6.2 Lifelog Corpus Moment Retrieval

This chapter indicates the integration of the proposed transformer-based video moment retrieval approach into the lifelog retrieval system. To give a formulating idea of the system, I suppose the input lifelog corpus $\mathbf{D}$ as a sequence of days $\mathbf{D} = \{\mathbf{D}_1, \mathbf{D}_2, \ldots, \mathbf{D}_M\}$, where M is the total number of collected days in the collection and each day $\mathbf{D}_i$ is represented as a sequence of pre-segmented periods $\mathbf{D}_i = [\mathbf{P}_l]_{l=0}^{L_i-1}$. Precisely, L_i is the number of extracted periods of the i-th day, in which each period consists of a fixed number of frames. Given an input query $\mathbf{Q}$, our key objective is to retrieve the most relevant moments $\mathbf{Y} = (s_j, e_j)$ across the lifelog data collection $\mathbf{D}$ that semantically correspond to the query $\mathbf{Q}$. Following the formulation in Equation 5.1, the moment retrieval task can be defined as:

$$\mathbf{D} = \{\mathbf{D}_1, \ldots, \mathbf{D}_M\} = \{[\mathbf{P}_l]_{l=0}^{L_1-1}, \ldots, [\mathbf{P}_l]_{l=0}^{L_M-1}\} \rightarrow \mathbf{Y} = (s_j, e_j) \tag{6.1}$$

6.2.1 Proposed System

With the detailed descriptions shown in the previous chapter, the PaTF framework has demonstrated its promising performance at identifying the most relevant moments within one single video in the VMR tasks. However, its direct application to large lifelog collections for lifelog retrieval tasks would be inefficient and not scalable. The main reason for this lies in the original design of the PaTF framework, which processes each video as a single entity and performs the retrieval operations across all videos, making it computationally expensive and time-consuming. To address this limitation, I introduce a hierarchical moment-based retrieval approach, built upon the PaTF framework developed in Chapter 5, named **PaTFLifelog**. This framework consists of two main components, such as the period retrieval stage and the moment retrieval stage, as illustrated in Figure 6.1. Following the sampling strategy established in the PaTF framework, each period is sampled as a sequence of 128 frames, which is considered

as a single entity for the moment retrieval stage. I will use the same number of 128 frames (approximately 1 hour period) for sampling lifelog periods to maintain consistency with the PaTF framework. Once the periods are sampled, the period retrieval stage is applied to analyse the input query to identify the most relevant periods of the day, effectively narrowing down the search space. In doing so, just the selected periods are passed to the moment retrieval stage, instead of the entire lifelog collection, which significantly reduces the computational complexity and time required for the retrieval process. The moment retrieval stage is in charge of locating the specific moments within the selected periods, leveraging the PaTF framework. Following this, a re-ranking strategy is applied to get the final moments based on the combination of the period relevance score and the moment relevance score. To this end, there are two sub-objectives to be achieved: (1) period retrieval to search for relevant periods of the day based on the input query (finding $\mathbf{P}^*$ from $\mathbf{D}$) and (2) moment retrieval to localise the most relevant moments within the selected periods (locating $\mathbf{Y}$ in $\mathbf{P}^*$). Hence, the Equation 6.1 can be further decomposed into two sub-tasks as follows:

$$\mathbf{D} = \{\mathbf{D}_1, \ldots, \mathbf{D}_M\} = \{[\mathbf{P}_l]_{l=0}^{L_1-1}, \ldots, [\mathbf{P}_l]_{l=0}^{L_M-1}\} \rightarrow \mathbf{P}^* = \{\mathbf{P}_1^*, \ldots, \mathbf{P}_N^*\} \tag{6.2}$$

$$\mathbf{P}^* = \{\mathbf{P}_1^*, \ldots, \mathbf{P}_N^*\} \rightarrow \mathbf{Y} = (s_j, e_j) \tag{6.3}$$

6.2.1.1 Period Retrieval

The period retrieval step ranks the periods of the day among the collection of periods, which serves as an initial filter to narrow down the search space for the PaTF framework. Given the query, the system operates by ranking the periods based on their relevance to the query content and then selecting the top-N periods for the next stage of moment retrieval. While the core ranking mechanism relies on measuring the similarity between the query and the periods in the joint embedding space, the challenge lies in how to represent the period's features. There are several

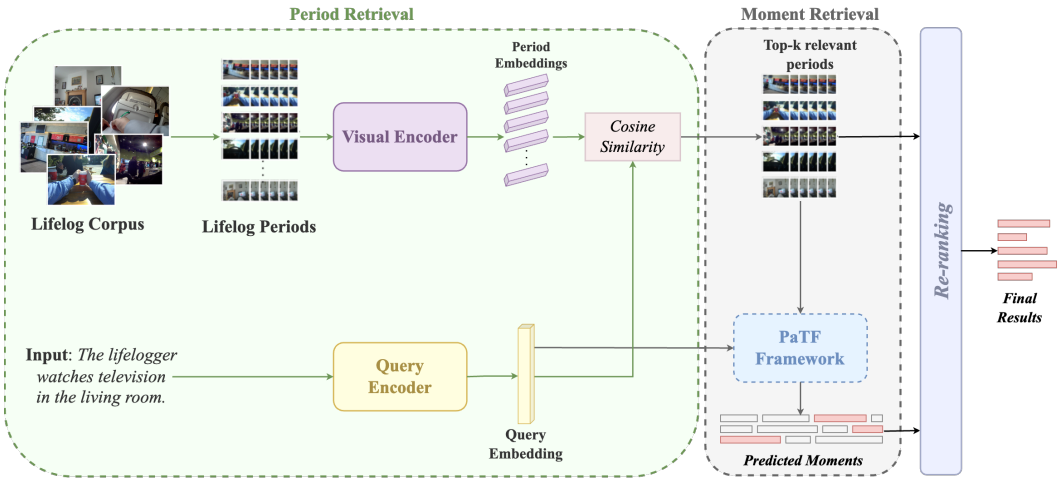


Figure 6.1: An overview of PaTFLifelog framework, which consists of three main components, the period retrieval stage, the moment retrieval stage, and the re-ranking stage. The period retrieval stage begins with ranking the lifelog periods based on its relevance to the query, followed by the moment retrieval stage that is responsible for retrieving the most relevant moments within the selected period. The final output is then re-ranked based on the relevance of the two stages.

approaches to obtain the features of the period by aggregating the features of all frames within the period, such as mean pooling, weighted pooling [229], clustering [230] or transformer encoders [10]. Although these methods show promising results in aggregate features at the period level, they might not be appropriate for the lifelog data due to the sparse nature of the lifelog data itself. Each period, as aforementioned, is considered as a sequence of 128 frames, which last approximately one hour. During this one hour, the lifelogger might have experienced various activities and events, which are not necessarily continuous. Therefore, to simplify the period representation, I propose to maintain the individual features of all frames as the period's features. Then, the similarity between the query and the period is calculated based on the cosine similarity between the query features and each frame feature within the period, and the highest score is represented as the period's similarity score. This approach is expected to capture the most relevant frames within the period of the query.

Particularly, queries and images are embedded into the same joint embedding space using the pre-trained (OpenCLIP ViT - L/14@336) [145] model. They are

ranked using the ranking techniques discussed in Chapter 4, with additional filtering options based on the concepts of time and location if available in the query. Hence, the output of this stage is the top-N periods of the day, consisting of the highest relevant frames to the query. In this dissertation, I use the top-10 periods for the next stage of moment retrieval, although this number can be adjusted and optimised in the future to improve retrieval performance.

6.2.1.2 Moment Retrieval

The Moment Retrieval stage is responsible for precisely localising the most relevant moments within the selected periods obtained from the period retrieval stage. This stage is inherited from the PaTF framework developed in Chapter 5, which leverages transformer-based video moment retrieval approaches while incorporating rich semantic information through scene graph representations alongside conventional visual features. In particular, this architecture leverages the parallel transformer module to process dual streams of feature representations, one combining the visual features of the frames and the query and the other being between the scene graph representations and the query. This process also involves three main steps: feature extraction, parallel transformer module, and regression (as illustrated in Figure 5.1), ultimately outputting the start and end keyframes of the relevant moment for the query.

Feature Extraction

The feature extraction step is to obtain the visual features of the frames, the scene graph representations, and the query embedding features to be used as input to the parallel transformer module. The following steps are taken to extract the features:

- *Visual Features*: As previously described, the lifelog images are passed through the pre-trained OpenCLIP image embedding model. In this case, each period, consisting of a sequence of N continuous frames, is considered as a "*video*" in the PaTF framework to represent that period of the day. Precisely, N is set to

128 frames, approximately one hour of the lifelog data, maintaining alignment with the sampling strategy for videos in the PaTF framework.

- *Scene Graph Encoder*: Similar to the PaTF implementation, the scene graph representations of lifelog images are generated using Neural-Motifs [206].
- *Query Encoder*: The query embedding features are extracted from the pre-trained OpenCLIP model. All query tokens are passed through the Query Encoder to get the query embedding features.

Then, the two branches of visual-textual joint features are obtained: one combines the visual features and query features and the other concatenates the scene graph representations and the query features. The construction of two branches exploits visual meaning and semantic relations between the query and the visual content.

The Parallel Transformer Module and Regression

Following the PaTF framework architecture described in Chapter 5, the two streams of data are passed through the parallel transformer module to model both intra-modality and inter-modality relationships between them. This module is comprised of multiple transformer layers with the help of attention mechanisms to capture the interactions between the two sets of features.

The output of the PaTF framework for this stage is the moment boundaries with the start and end keyframes, which can be converted into the list of frames within that moment. The regression is applied to predict the start and end keyframes of the moment, which is expected to capture the most relevant moments within the selected periods of the query.

6.2.1.3 Moment Re-ranking Strategy

As illustrated in Figure 6.1, the moment retrieval stage is applied to each period obtained from the period retrieval stage to localise the target moments. Once the moment retrieval stage is applied to each of the periods identified from the initial stage, the final moments are then re-ranked based on the combination of the

similarity scores of the two stages. Hence, the final similarity score is calculated as the weighted sum of the period relevance score S_{p_i} and the moment relevance score $S_m^{p_i}$ as follows:

$$\text{Final Score} = \alpha \times S_{p_i} + (1 - \alpha) \times S_m^{p_i} \tag{6.4}$$

where α is the weight parameter to balance the importance of the period relevance score and the moment relevance score, S_{p_i} is the period relevance score of the i-th period, and $S_m^{p_i}$ is the moment relevance score of the p_i-th period. In this dissertation, I set $\alpha = 0.5$ to balance the importance of the two stages equally. The final moments are then selected based on the final score, which is expected to capture the most relevant moments within the selected periods of the query.

6.2.2 Baseline System

As a reminder, I utilise the E-LifeSeeker developed in LSC'23 as the baseline system, as described in Chapter 4. This system is an image-based lifelog search tool that retrieves images from the lifelog data based on the content of the input query. E-LifeSeeker represents the state-of-the-art in lifelog image retrieval, which exploits the advances of the pre-trained language-vision model, OpenCLIP [150]. In doing so, the system can bridge the gap between the textual and visual semantic meanings by embedding both the images and the queries into the same joint embedding space. Moreover, the system also incorporates the facet filters to refine the search results based on non-visual information, such as time and location. With several functionalities and features, E-LifeSeeker provides an interactive user interface for the lifelog retrieval tasks, which allows users to interact with the system and evaluate the retrieval performance. The system is evaluated on several benchmarking challenges, including the Lifelog Search Challenges (LSC) [47, 82, 98] and the NTCIR Lifelog Task [203] to demonstrate its effectiveness in lifelog image retrieval tasks.

6.2.3 Comparison

The key differences between the two systems can be analysed across three main components: the retrieval unit, the content representations, and the core retrieval engine. Since the two systems will be compared in an automatic manner, the other non-visual information, such as time and location, is removed from the queries and converted into facet filters. Then, only the visual-related information is used as the input query to retrieve the moments. By doing so, the comparison focuses on the retrieval performance of the two systems themselves, especially the core search engine while processing the moment retrieval task rather than the user interaction and interface. As discussed earlier, given the input query, the baseline system looks for the most relevant images, in contrast, the PaTF framework localises the moments within the periods of the day with start and end times. Regarding visual representation, the PaTFLifelog provides a richer representation of image content by considering both visual features and semantic information via the scene graph representation, while LifeSeeker only relies on the visual features of the images. For the core engine, the baseline system first encodes the textual and visual content to the same latent space and then simply uses the cosine similarity to measure the relevance between them. The PaTFLifelog framework, on the other hand, utilises the parallel transformer module to model the relationship between the rich visual features (including visual features and scene graph representations) and the textual features before regressing the moment boundaries. As for the automatic retrieval task, the User Interface and Interaction module is not considered to compare as the focus is on the retrieval performance of the system itself. Therefore, the UI of the PaTFLifelog system remains the same as the baseline system, which is described in Chapter 4.5.

6.3 Experiments and Results

To investigate the effectiveness of our proposed moment-based retrieval approach in the lifelog domain, I conduct a user experiment comparing the performance of the proposed system with the baseline system in automatic settings. With this settings, there are no human involvement in the retrieval process. This provides insights into the retrieval performance of the two systems themselves and the potential of the proposed approach in improving the retrieval performance of past moments in lifelog data, which indicates the answer to the **RQ3**.

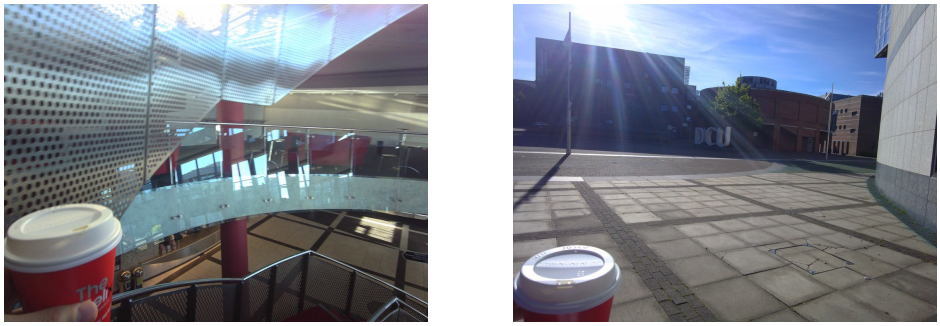
6.3.1 Dataset

To better compare the proposed approach and the baseline system, we utilise two datasets, one from the Lifelog Search Challenge 2020 (LSC'20)[1] and the other from the Lifelog Question Answering (LLQA) dataset. The LSC'20 dataset is a collection of lifelog data captured from one lifelogger, including the 191,439 egocentric images captured with OMG Autographer and Narrative Clip devices, metadata, and corresponding annotations.

For the lifelog image descriptions, I inherited the dataset from the Lifelogg Question Answering Dataset (LLQA) [11], built upon the LSC'20 lifelog data collection [81]. This dataset contains 85 days of lifelog data, 26 days in 2015 and 59 days in 2016. Leveraging the event segmentation approach developed by Doherty et al. [71], each day was systematically segmented into short events based on changes in the lifelogger's locations (such as work, home, airport, etc.) and activities (including walking, driving, etc.). The annotators were tasked with providing detailed descriptions for each event using a specialised annotation system. These descriptions should include specific actions or activities being performed, objects that the lifelogger interacted with, location information, and the presence of other

[1] `http://www.lifelogsearch.org/lsc/2020/lsc_data/`

people[2]. In addition, the dataset also includes the starting and ending times for each event, which are used to define the boundaries of the moments.



(a) (b)

Figure 6.2: Example images in the LLQA dataset [11] with the description "*The lifelogger is walking back to the office while holding a cup of coffee by hands.*", and the corresponding time segments $[09:09-09:11]$.

Query Selection

Regarding the query selection in this experiment design, I consider two types of queries, the LSC'20 topic queries and LLQA (Lifelog Question Answering) descriptions. The LSC'20 topics are generated by the LSC organisers and are described as a set of sub-queries that are related to the events in the lifelog data. Those queries are detailed with the time and location information to give the context of the events. For this specific automatic experiment, I made a slight modification to the queries by extracting only the visual-related information to be the input query to the model and converting non-visual concepts related to time and location into facet filters for the results. Moreover, other redundant information, such as the lifelogger's emotional state or the weather, is removed to simplify the queries. This step is needed as those information would not contribute to the visual content in the joint embedding matching. For instance, the original query "*I am looking for my yellow staff card in May 2018. I remember putting it in my wallet. At work on Tuesday afternoon. I had been working in my office for a*

[2]For the full list of descriptions, please refer to `https://github.com/allie-tran/LLQA/blob/main/published/descriptions.jsonl` and `https://github.com/allie-tran/LLQA/blob/main/published/new/descriptions.jsonl`

few hours. Then I went to a computer lab." is simplified to "*I am looking for my yellow staff card. I remember putting it in my wallet.*" for the input query to the model, and "*Tuesday afternoon, May 2018*" and "*office*" are used as time and location facet filters for the results. The full list of 22 LSC'20 topics used in this experiment can be found in Appendix D.

The LLQA descriptions (with details described above) are captured by the annotators. Most annotations are single sentences describing events, actions, objects, and surrounding context of the image. The query modification is not required for this dataset as the descriptions are only related to the visual content, while the date, start, and end time information of the events are already included in the annotations and the location information is not mentioned. For this dataset, the date is used as a time filter for the queries.

Furthermore, to better align with the moment retrieval task, those queries having only one image are removed from the dataset, as single-image queries conflict with the temporal nature of the moment we aim to retrieve. Those queries with the wrong moment boundaries annotations (end time is sooner than the start time) are also eliminated because the segments have to satisfy the temporal order of the events. The queries with more than 128 ground-truth frames are also dropped, as our input period is set to 128 frames, the largest possible moment we can retrieve. The final LLQA dataset consists of 7257 queries with the corresponding ground-truth moments. After preprocessing the descriptions, I split the dataset into training and test sets with a ratio of 85:15 with an equal distribution of the segment's length.

6.3.2 Evaluation Metrics

For this automatic experiment, I consider two evaluation metrics: Hit Rate at k ($H@k$) and Mean Average Precision at k ($MAP@k$) for lifelog retrieval tasks. As described in Section 3.3, the $H@k$ measures the possibility of successfully retrieving at least one relevant answer in the top k retrieved items, and the $MAP@k$ indicates the average precision of the relevant moments retrieved in the

top-k results, considering the order of the retrieved moments. For the PaTFLifelog, a moment is counted as relevant if at least one frame in the moment matches the judgments. With regard to the value of k, I set $k \in [1, 10, 50, 100]$ for both tasks to evaluate the performance of the systems in retrieving the relevant moments.

6.3.3 Experiment Results

6.3.3.1 Comparison with the Baseline system

A detailed performance comparison between the PaTFLifelog and the baseline system is presented in Table 6.1 for the LSC'20 topics, and the LLQA descriptions, in terms of two aforementioned metrics. While the Hit Rate values indicate the ability to successfully find at least one relevant target, the MAP values measure the average precision of the relevant moments with respect to the ranking order. The higher $MAP@k$ values the system achieves, the more optimal the positions of the relevant moments are. When examining 22 queries in the LSC'20 challenge, both systems demonstrated strong retrieval capabilities, with the baseline system showing higher values for the hit rate metric ($H@1 = 54.54$ and $H@10 = 77.27$ compared to $H@1 = 50.00$ and $H@10 = 72.72$ for the PaTFLifelog system). However, the PaTFLifelog system performs better at higher k values, particularly achieving a hit rate of 95.45% compared to the baseline's 90.90% at $k = 100$. Those queries generated by the lifelogger for the competition are more detailed and specific, which makes it easier for the system to retrieve the target frames with over 90% success in finding relevant content in the top 100 frames. In terms of the Mean Average Precision metric, the two systems showed comparable performance. Particularly, the baseline obtains $MAP@1 = 54.54\%$, which is 4% higher than that of the PaTFLifelog system. And the other $MAP@k$ values of the PaTFLifelog system are slightly higher than the baseline, with an approximately 1% to 2% difference.

For the LLQA dataset which contains 1089 queries describing more general and frequent activities, both systems struggle to retrieve the relevant moments. This

Table 6.1: Performance comparison between PaTFLifelog and the baseline systems on the LSC'20 query and LLQA test sets in terms of Hit Rate at k ($R@k$) and Mean Average Percision at k ($MAP@k$). The best values are highlighted in bold.

Query Set	No. queries	Method	Hit Rate				MAP			
			$k=1$	$k=10$	$k=50$	$k=100$	$k=1$	$k=10$	$k=50$	$k=100$
LSC query	22	Baseline	**54.54**	**77.27**	90.90	90.90	**54.54**	53.48	49.82	49.01
		PaTFLifelog	50.00	72.72	90.90	**95.45**	50.00	**54.61**	**51.47**	**50.14**
LLQA query	1089	Baseline	15.88	46.01	75.39	85.22	15.88	22.39	20.36	18.72
		PaTFLifelog	**31.67**	**63.47**	**82.07**	**88.54**	**31.67**	**41.69**	**39.41**	**40.61**

larger and more diverse query set presents more challenging retrieval scenarios for the systems when compared to the LSC'20 dataset. The challenge can be seen through the value of the $H@1$ metric, where the baseline system only achieves $H@1 = 15.88\%$ and the PaTFLifelog system achieves $H@1 = 31.67\%$ to find the correct frames in the first prediction. The hit rate values continue to increase as the value of k increases, however, these of the baseline system are still significantly lower than the PaTFLifelog system. The PaTFLifelog system reaches a hit rate of $H@10 = 63.47\%$ and $H@50 = 82.07\%$, which are 17.46% and 6.68% higher than the baseline system, respectively. Although the hit rate values of the baseline at $k = 50$ and $k = 100$ are competitive with values higher than 75%, the $MAP@k$ values are significantly lower with less than 20%. This indicates that while the baseline has a high chance of retrieving at least one relevant frame that satisfies the query, it struggles to position itself in the rankings. This limitation can be due to the nature of the lifelog data, where relevant moments are often distributed across the day, making precise ranking more challenging. The PaTFLifelog system, on the other hand, has a better performance in terms of $MAP@k$ values as it is designed to localise moments within the periods of the day, which has a higher chance of ranking the relevant moments in the top ranks.

In summary, the PaTFLifelog system demonstrates better performance over the baseline system in terms of hit rate values, particularly at higher k values. Importantly, the PaTFLifelog system surpasses the baseline system in ranking effectiveness as indicated by the higher $MAP@k$ values for most of the k values. This result demonstrates the effectiveness of the proposed moment-based retrieval approach in the lifelog domain, which leverages the transformer-based video moment retrieval technique to localise the relevant moments within the periods of the day.

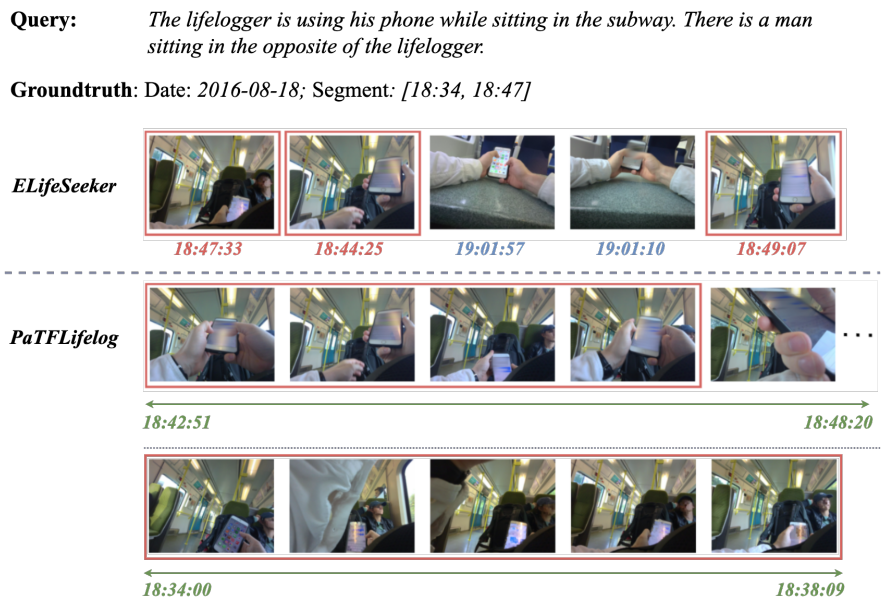
6.3.3.2 Qualitative Analysis

Figure 6.3 shows a detailed visualisation comparison of the retrieval results between the baseline system and the PaTFLifelog system on the LLQA dataset. This qualitative analysis aims to provide insights into the retrieval performance of the two systems in retrieving relevant moments based on the input queries. For each example, the query input and its corresponding ground-truth moments are shown, along with the top-5 retrieved frames from the baseline system and the first retrieved moment from PaTFLifelog system. While the results of the baseline are shown separately with its corresponding taken time, the PaTFLifelog system displays the retrieved moments with some representative frames together with the start and end times of the moments. Regarding the query with detailed descriptions (example in Figure 6.3a), the PaTFLifelog system successfully retrieves the relevant moments within the selected periods.

On the other hand, for queries with less information and too general like "*The lifelogger is in the living room*", the baseline faces difficulties in handling these queries, as shown in Figure 6.3b. The system struggles to find the relevant frames due to the frequent occurrence of these events in lifelog data. Hence, matching one single prediction with the ground-truth is challenging. However, the PaTFLifelog system demonstrates its effectiveness in managing to retrieve the relevant moments under top-5 moments thanks to the moment-based retrieval approach, which provides a broader temporal context for each predicted event.

6.4 Discussion and Conclusion

The user study investigated the retrieval performance of the proposed PaTFLifelog system against the baseline system in the automatic settings. In particular, the PaTFLifelog system implemented a hierarchical moment-based retrieval approach that leverages period retrieval and transformer-based moment retrieval stages to identify relevant moments within periods of the day in lifelog data. The experiment



(a) First example

(b) Second example

Figure 6.3: Qualitative results of PaTFLifelog on the LLQA dataset (best viewed in color). Top 5 retrieved frames from E-LifeSeeker are displayed in the top row, while the bottom row shows the first predicted moment from PaTFLifelog. The correct relevant answers are highlighted with orange boxes.

results revealed that the PaTFLifelog surpasses the baseline by a large margin of MAP values in the LLQA dataset (approximately 20% higher), showing its ability

in handling general queries and localising the relevant moments within the periods of the day compared to the baseline system. For the LSC'20 queries, while both systems showed competitive performance, PaTFLifelog demonstrated better precision at higher k values (approximately 1% to 2% higher), which indicates a more optimal ranking capability. This improvement can be attributed to the moment-based retrieval approach that effectively captures the event context and temporal relationships and the integration of scene graph representations that enrich the understanding of visual content.

The system's current performance, while significantly advancing the state-of-the-art in lifelog retrieval, also suggested promising directions for future research, such as incorporating additional modalities of lifelog data, optimising the hierarchical retrieval strategy, and exploring adaptive ranking strategies for different types of queries. This experiment results provide the answer to my last research question, **RQ3**, which is "*To what extent can the proposed transformer-based moment retrieval approaches, when applied to lifelog retrieval, improve the lifelog retrieval performance of past moments when compared to existing SOTA interactive retrieval systems?*".

6.5 Chapter Summary

This chapter presented a novel framework that integrated the proposed transformer-based video moment retrieval approach to lifelog data and its evaluation against the conventional image-based lifelog retrieval system in automatic settings. The proposed framework, PaTFLifelog, was built upon the PaTF framework developed in Chapter 5, consisting of three stages such as period retrieval, moment retrieval, and moment re-ranking. Through comprehensive experimental evaluation conducted using two distinct query sets, the LSC'20 topics and the LLQA descriptions, I make a detailed comparison between the PaTFLifelog system and the baseline system in retrieving the queries' relevant moments. The experiment results demonstrated that the proposed hierarchical lifelog moment retrieval approach, PaTFLifelog system, has

the ability to enhance the retrieval performance of past moments in terms of hit rate and mean average precision, particularly in the LLQA dataset, demonstrating the effectiveness of the proposed moment-based retrieval approach in the lifelog domain. In the end, with comprehensive experiments and evaluations, this chapter brought the answer to the **RQ3**: "*To what extent can the proposed transformer-based moment retrieval approaches, when applied to lifelog retrieval, improve the lifelog retrieval performance of past moments when compared to existing SOTA interactive retrieval systems?*".

Chapter 7

Conclusion

7.1 Summary

In this dissertation, we concentrated on developing a novel transformer-based moment retrieval approach to enhance the efficiency of retrieving one's past experiences from lifelog data. A summary of the investigations, research findings, and proposed solutions is presented as follows.

Chapter 1 established the background of our research by introducing the concept of lifelogging, lifelog retrieval tasks, and the key motivations behind this research. This chapter also covered the hypothesis regarding the potential of transformer-based techniques in enhancing the performance of lifelog retrieval systems in moment retrieval tasks and presented three research questions to investigate this hypothesis. We also outlined the major contributions of our research work, including the development of a state-of-the-art lifelog retrieval system, the design of a novel transformer-based Video Moment Retrieval (VMR) technique, and the integration of the VMR technique into the lifelog retrieval system.

Chapter 2 outlined the history of lifelogging, and the evolution of lifelog search engines and platforms. The chapter also provides a comprehensive review of existing lifelog retrieval systems, especially the Lifelog Search Challenge (LSC) systems. The chapter concludes with the lack of efficient moment-based retrieval techniques for lifelog data. According to the literature review, the research gap in existing lifelog retrieval systems is identified as focusing only on image-based

retrieval techniques, which may not be efficient for retrieving past moments from lifelog data. Therefore, we proposed a novel moment-based retrieval approach that integrates transformer-based techniques to enhance the efficiency of retrieving one's past experiences. The approaches in Video Moment Retrieval (VMR) are explored to investigate the potential of developing a moment-based retrieval system for lifelog data.

Chapter 3 detailed our research methodology, operating constraints, and evaluation metrics used in this research. We explained the experimental methodology schema to investigate the research questions and evaluate the proposed solutions. Operating constraints, including time limitations, query generation processes, data collection constraints, and system evaluation constraints, are also acknowledged in this chapter. Regarding different retrieval tasks, we used different evaluation metrics to evaluate the system performance or compare the system with other systems.

Chapter 4 described the development of our lifelog interactive retrieval system, LifeSeeker, a state-of-the-art image-based lifelog retrieval system that supports users in retrieving their personal lifelog images. We provided an in-depth analysis of all components of the system, including the core search engine, user interface, and evaluation metrics. Our evaluation using the Lifelog Search Challenge (LSC) dataset demonstrated that LifeSeeker ś competitive performance against those of other systems. As such, the system is adequately used as a baseline for further improvements and comparison with the novel retrieval system developed in the later phase.

Chapter 5 introduced a novel approach to the Video Moment Retrieval (VMR) tasks, named the Parallel Transformer Framework (PaTF). The PaTF framework exploited both low-level visual cues and high-level relational contexts of video-query pairs, together with the textual information, to localise the target moment in the video. A parallel transformer module was implemented to take advantage of the two data streams simultaneously. A comprehensive evaluation of the PaTF framework

was conducted on the Charades-STA dataset, highlighting the effectiveness of the proposed approach in localising the target moment in the video. In addition, ablation studies were also performed to clarify the importance of relational information and query-irrelevant masking for the scene graph features.

Chapter 6 presented a novel moment-based retrieval approach for lifelog data, named PaTFLifelog. This chapter described our success in adapting and integrating the PaTF framework into the lifelog retrieval system, addressing the situations of retrieving past moments from personal lifelog data. The integration process involved significant modifications to the core search engine by processing lifelog data and incorporating a period retrieval module to find the top relevant periods initially. This was then followed by the PaTF module to localise the target moment within the chosen periods. The evaluation of the PaTFLifelog system was conducted on the LSC, coupled with the queries from the LSC and LLQA query sets. The experimental results demonstrated the effectiveness of the PaTFLifelog system in retrieving past moments from lifelog data, outperforming the baseline lifelog retrieval system, LifeSeeker, in terms of hit rate and mean average precision. This helps to support the hypothesis that the integration of transformer-based techniques into a moment-based lifelog retrieval approach enhances the performance of retrieving one's past experiences when compared to conventional image-based lifelog retrieval systems.

7.2 Revisiting of Hypothesis, Research Questions, and Contributions

The focus of this thesis is to explore approaches to answering questions from lifelog data, as well as to incorporate these approaches into interactive lifelog retrieval systems. In this section, I will revisit the hypothesis and research questions, and examine them with respect to the proposed solutions and the experimental results.

Hypothesis

The integration of transformer-based techniques into a moment-based lifelog retrieval approach enhances the performance of retrieving one's past experiences when compared to conventional image-based lifelog retrieval systems.

Associated with this hypothesis, we have formulated three main research questions in order to effectively investigate this hypothesis, as follows:

Research Question 1 (RQ1): How can a state-of-the-art baseline lifelog interactive retrieval system be developed to effectively support users in retrieving their past experiences?

For this research question, the dissertation covered the development of a conventional lifelog interactive retrieval system with a straightforward user interface to help users retrieve their lifelog images. E-LifeSeeker the latest versions of our lifelog retrieval system was then used as a baseline for comparison with the novel retrieval system developed in the later phase. The system was evaluated through multiple research iterations in the Lifelog Search Challenge (LSC) competition and the NTCIR Lifelog Challenge. In addition, comprehensive user studies were also conducted to investigate different aspects of the system's performance, including the system's usability, user interaction, core engine performance, and system effectiveness. Based on the evaluation results, system performance, as well as the experiences gained from both our team and the others in the competitions, we analysed our system's strengths and weaknesses for further improvements in the later phase. This was proven by the adaptation of visual-textual embedding models, new user interface designs, and functionalities in the latest versions of the system after recognising the importance of joint representation learning for the lifelog retrieval task. The system was evaluated using the LSC dataset, and the results show that the system achieves competitive

performance compared to other systems [106, 6, 102].

Research Question 2 (RQ2): How can a state-of-the-art transformer-based Video Moment Retrieval technique be designed to effectively localise target moments in video sequences?

This research question focused on exploring the state-of-the-art techniques for Video Moment Retrieval tasks. Our objective is to propose a novel approach that better localises the target onset and offset of the moment in the video. The transformer-based approach appears to be effective in capturing the temporal relations between the video and the query, which is in agreement with the works done by [56, 57, 58]. However, our approach also utilises scene graph embeddings to enhance the relational information between objects in the video, which has not been explored in previous works. The proposed approach, the Parallel Transformer Framework (PaTF), advanced the moment retrieval task by simultaneously exploring two parallel feature streams of video-query pairs. The PaTF framework was constructed as a parallel transformer architecture: a visual-textual stream that extracts the links between global visual features and textual information, and a semantic-textual stream that emphasises the relations between objects via scene graph representations. The effectiveness of the proposed approach was evaluated on the Charades-STA dataset, demonstrating the capability of the PaTF framework in localising the target moment in the video. This approach can be promising in doing moment retrieval tasks in video data, but can also be extended to other data types such as lifelog data.

Research Question 3 (RQ3): To what extent can the proposed transformer-based moment retrieval approaches, when applied to lifelog retrieval, improve the lifelog retrieval performance of past moments when compared to existing SOTA interactive retrieval systems?

As for the similarity in the continuous and chronological format between lifelog data and video data, we explored the transferability of these techniques onto the

lifelog dataset. The objective of this research question was to investigate the integration of transformer-based moment retrieval techniques into the lifelog retrieval system to enhance the efficiency of retrieving one's past experiences. This was followed by the evaluation of the proposed system compared to the existing state-of-the-art lifelog retrieval system on two different query sets, the LSC query and the LLQA query.

Research Question 3 is particularly relevant to the hypothesis as it demonstrates the effectiveness of incorporating transformer-based techniques, PaTF, into lifelog retrieval systems, PaTFLifelog. This is proven by the outperformance of the PaTFLifelog system compared to the baseline lifelog retrieval system, E-LifeSeeker, in terms of hit rate and mean average precision when retrieving past moments from lifelog data in the LSC dataset. The findings of the user studies showed that the proposed pipeline can improve the performance of interactive lifelog retrieval tasks when compared to the conventional image-based search tool. The findings of this research work supported the hypothesis.

7.3 Limitations

- **Limited dataset**: Although this limitation does not affect the work in the scope of this research, where we currently focus on data collected from a single lifelogger, more data from more lifeloggers is still needed in the future to diversify the dataset as well as to improve the generalisability of the system.

- **Small number of users**: The number of participants in the user study is limited due to challenges in recruiting volunteers. As a result, this research focuses only on investigating the system's usability and user interaction with a small number of participants, without considering the vast diversity in the user group.

- **Frame rate**: Although having a similar continuous and chronological format,

lifelog images are sparser and less informative than video data. For this reason, the moment retrieval techniques may not perform as well as expected. We can further improve by setting the lifelog camera taken time to be less spare (current is one photo per 30 seconds).

- **Image quality**: As the current wearable cameras are built to be lightweight and easy to carry, the image quality is not as good as the professional cameras. Moreover, the nature of capturing images in daily life is not always in the best condition, such as low light, blurry, or occluded images. Although our approaches perform well in the LSC dataset, further techniques in pre-processing image data need to be developed to overcome the image quality limitations. Some potential solutions include using better cameras with higher resolution and implementing better stabilisation techniques [231, 232].

- **Evaluation metrics**: The lifelog retrieval system is evaluated based on the LSC score, which may not be the best metric to evaluate the system's performance, especially for the moment-based retrieval system. We can consider other metrics (Recall@k with IoU, Mean Average Precision, etc.) that closely align with the system's objective, which is to support individuals in retrieving their past experiences. Therefore, metrics measuring memory recall should be considered as well.

- **Scene Graph Embedding**: The scene graph is the pre-trained model I integrated to extract the relationships between objects in the image. The scene graph embedding merely flattens the confidence matrix to obtain the scene graph features, which may overlook important information during feature extraction, such as spatial information. More advanced techniques could be explored to address this limitation, such as using graph neural networks to capture the spatial information in the scene graph.

- **Rely strongly on the annotations**: The lifelog data and event/description

are generated by the annotators, which sometimes may be biased to the annotators' perspective. We can consider other ways to generate the event description, such as using image captioning techniques where event generation can be done via advanced deep learning techniques [135, 233, 132] then can be verified by the annotators.

- **Running time**: Since the PaTF is built for the VMR task in which the moment is retrieved within a single video, the running time and cost of PaTFLifelog will be significantly increased when running across a corpus of lifelog data or collection of lifelog periods. Meanwhile, if we merely concentrated on a subset of the video collection, the system might run faster, but the accuracy might not be guaranteed in some circumstances. With our current implementation, we accepted the trade-off between the running time and the accuracy.

7.4 Future Works

Looking forward, there are several directions for future research that can be explored to further improve the lifelog retrieval system.

Improving the Data Collection

- **Collecting more diverse and representative lifelog data**: The current lifelog dataset is collected from a single lifelogger, which may not be representative of the general population. Therefore, it may be necessary to collect more diverse and representative lifelog data to improve the generalisability of the lifelog retrieval system.

- **Improving the image quality**: Better cameras with higher resolution and better stabilisation can be a potential solution to improve image quality. Besides, implementing better stabilisation techniques can also help to address the image quality limitations.

Improving the Moment Retrieval Techniques

- **Video Corpus Moment Retrieval (VCMR) task:** The VCMR task is more challenging than the VMR task as it requires the system to retrieve the moment across a collection of videos. Therefore, it is necessary to develop new moment retrieval techniques that are specifically designed for the VCMR task.
- **Improving the moment retrieval techniques for lifelog data:** The current moment retrieval techniques are based on moment retrieval techniques for video data, which is denser and richer in information. However, lifelog data is different from video data in terms of content and context. Therefore, it is necessary to develop new moment retrieval techniques that are specifically designed for lifelog data.

Improving the Lifelog Retrieval System

- **Enhancing the user interface and user interaction**: Result presentation in clusters of moments and more interactive visualisation options for the user to explore the lifelog data.
- **User Experiments**: Conducting more user experiments to evaluate the system's performance and user experience. Although the current automatic settings showed promising results, the system's usage and interaction can be further improved by conducting more user experiments in interactive settings.

Chapter 8

Publication List

In this chapter, I would like to highlight the following publications as I have contributed as first or co-author during my PhD study. The publications are related to the research topics of my Ph.D. study, including lifelog retrieval, video retrieval, and interactive retrieval systems, which are listed in reverse chronological order as follows:

- **Thao-Nhu Nguyen**, Zongyao Li, Yamazaki Satoshi, Jianquan Liu, Cathal Gurrin. A Parallel Transformer Framework for Video Moment Retrieval. In Proceedings of the 2024 International Conference on Multimedia Retrieval, ICMR '24, page 460–468, New York, NY, USA, 2024. Association for Computing Machinery.

- **Thao-Nhu Nguyen**, Le Minh Quang, Graham Healy, Binh T. Nguyen, and Cathal Gurrin. VideoCLIP 2.0: An Interactive CLIP-Based Video Retrieval System for Novice Users VBS2024. In MultiMedia Modeling: 30th International Conference, MMM 2024, Amsterdam, The Netherlands, January 29 February 2, 2024, Proceedings, Part IV, page 394399, Berlin, Heidelberg, 2024. Springer-Verlag.

- **Thao-Nhu Nguyen**, Bunyarit Puangthamawathanakun, Chonlameth Arpnikanondt, Cathal Gurrin, Annalina Caputo, and Graham Healy. Efficient Search with an Interactive Video Retrieval System for Novice Users in IVR4B. In Proceedings of the 20th International Conference on Content-Based Multimedia Indexing, CBMI '23, page 168172, New York, NY,

USA, 2023. Association for Computing Machinery.

- Jakub Lokoč, Stelios Andreadis, Werner Bailer, Aaron Duane, Cathal Gurrin, Zhixin Ma, Nicola Messina, **Thao-Nhu Nguyen**, Ladislav Peška, Luca Rossetto, Loris Sauter, Konstantin Schall, Klaus Schoeffmann, Omar Shahbaz Khan, Florian Spiess, Lucia Vadicamo, and Stefanos Vrochidis Interactive video retrieval in the age of effective joint embedding deep models: lessons from the 11th VBS. Multimedia Syst., 29(6):34813504, August 2023.

- **Thao-Nhu Nguyen**, Tu-Khiem Le, Van-Tu Ninh, Annalina Caputo, Graham Healy, Sinéad Smyth, Minh-Triet Tran, and Nguyen Thanh Binh. LifeSeeker: an interactive concept-based retrieval system for lifelog data. Multimedia Tools Appl., 82(24):3785537876, August 2023.

- **Thao-Nhu Nguyen**, Tu-Khiem Le, Van-Tu Ninh, Cathal Gurrin, Minh-Triet Tran, Thanh Binh Nguyen, Graham Healy, Annalina Caputo, and Sinead Smyth. E-LifeSeeker: An Interactive Lifelog Search Engine for LSC23. In Proceedings of the 6th Annual ACM Lifelog Search Challenge, LSC '23, page 1317, New York, NY, USA, 2023. Association for Computing Machinery.

- Maria Tysse Hordvik, Julie Sophie Teilstad Østby, Manoj Kesavulu, **Thao-Nhu Nguyen**, Tu-Khiem Le, and Duc-Tien Dang-Nguyen. LifeLens: Transforming Lifelog Search with Innovative UX/UI Design. In Proceedings of the 6th Annual ACM Lifelog Search Challenge, LSC '23, page 16, New York, NY, USA, 2023. Association for Computing Machinery.

- **Thao-Nhu Nguyen**, Bunyarit Puangthamawathanakun, Annalina Caputo, Graham Healy, Binh T. Nguyen, Chonlameth Arpnikanondt, and Cathal Gurrin. VideoCLIP: An Interactive CLIP-based Video Retrieval System at VBS2023. In Duc-Tien Dang-Nguyen, Cathal Gurrin, Martha Larson, Alan F. Smeaton, Stevan Rudinac, Minh-Son Dao, Christoph Trattner, and

Phoebe Chen, editors, MultiMedia Modeling, pages 671–677, Cham, 2023. Springer International Publishing.

- 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 ór Jónsson, Luca Rossetto, An-Zi Yen, Ahmed Alateeq, Naushad Alam, Minh-Triet Tran, Graham Healy, Klaus Schoeffmann, and Cathal Gurrin. Comparing Interactive Retrieval Approaches at the Lifelog Search Challenge 2021. IEEE Access, 11:30982–30995, 2023.

- **Thao-Nhu Nguyen**, Tu-Khiem Le, Van-Tu Ninh, Minh-Triet Tran, Thanh Binh Nguyen, Graham Healy, Sinéad Smyth, Annalina Caputo, and Cathal Gurrin. LifeSeeker 4.0: An Interactive Lifelog Search Engine for LSC'22. In Proceedings of the 5th Annual on Lifelog Search Challenge, LSC '22, page 1419, New York, NY, USA, 2022. Association for Computing Machinery.

- **Thao-Nhu Nguyen**, Tu-Khiem Le, Van-Tu Ninh, Ly-Duyen Tran, Manh-Duy Nguyen, Minh-Triet Tran, Thanh-Binh Nguyen, Annalina Caputo, Sinéad Smyth, Graham Healy. DCU and HCMUS at NTCIR-16 Lifelog-4. In Proceedings of the 16th NTCIR Conference on Evaluation of Information Access Technologies. NTCIR, 2022.

- Manh-Duy Nguyen, **Thao-Nhu Nguyen**, Binh Thanh Nguyen, Annalina Caputo, and Cathal Gurrin. DCU Team at the NTCIR-16 RCIR Task. In Proceedings of the 16th NTCIR Conference on Evaluation of Information Access Technologies. NTCIR, 2022.

- **Thao-Nhu Nguyen**, Bunyarit Puangthamawathanakun, Graham Healy, Binh T. Nguyen, Cathal Gurrin, and Annalina Caputo. VideoFall - A Hierarchical Search Engine for VBS2022. In Björn ór Jónsson, Cathal Gurrin, Minh-Triet Tran, Duc-Tien Dang-Nguyen, Anita Min-Chun Hu, Binh Huynh

Thi Thanh, and Benoit Huet, editors, MultiMedia Modeling, pages 518–523, Cham, 2022. Springer International Publishing.

- **Thao-Nhu Nguyen**, Tu-Khiem Le, Van-Tu Ninh, Minh-Triet Tran, Graham Healy, Binh T. Nguyen, Annalina Caputo, and Cathal Gurrin. LifeSeeker 3.0: Interactive Lifelog Search Engine at LSC 2021. In Proceedings of the Third Annual Workshop on Lifelog Search Challenge, LSC '21, page 5762, New York, NY, USA, 2021. Association for Computing Machinery.

- Ly-Duyen Tran, Manh-Duy Nguyen, **Thao-Nhu Nguyen**, Graham Healy, Annalina Caputo, Binh T. Nguyen, and Cathal Gurrin. A VR Interface for Browsing Visual Spaces at VBS2021. In Jakub Lokoč, Tomáš Skopal, Klaus Schoeffmann, Vasileios Mezaris, Xirong Li, Stefanos Vrochidis, and Ioannis Patras, editors, MultiMedia Modeling, pages 490–495, Cham, 2021. Springer International Publishing.

Appendix A

User Experiment

A.1 Experiment Design

To further evaluate the two search mechanisms' performances, we conducted an experiment for individuals with no prior knowledge about lifelogging and lifelog retrieval systems, referred to as novice users. By doing so, we were able to choose the more suitable core engine for those specific users. A total of 8 participants, including three undergraduate students, three postgraduate students, and two researchers, were recruited for this study without any specific technical requirement. They were equally separated into two groups: one interacts with the concept-based system (**Group 1**), and one explores the semantic-based engine (**Group 2**) to locate the target. Regarding the task selection process, there were ten queries randomly sampled from the query collection given by the LSC's organisers (as shown in Section A.1). The study used the format adopted from the LSC competition in which users are required to seek lifelog moments from a description in a limited time.

Prior to the experiment, we introduced the volunteers to how the system operates, how to navigate the searching tool and how to get useful information from the result display. There are 10 queries collected from the query set given by the LSC organisers with a full description shown in Section A.1. Every initial information is provided as a piece of text, which is then followed by 5 further hints each displayed at intervals of 30 seconds. Consequently, the maximum search time allowed for one query is 3 minutes, leading to a duration of 1 hour per user. There is no limitation on the

number of submissions, either relevant or irrelevant. Users are able to submit as many images as they like until the correct one is found. However, wrong attempts do have a negative impact on the overall results in terms of score, precision and recall, and in addition, they result in a 10-point penalty on the official total score.

For further analysis purposes, the statistics related to searching time, score and correctness have been recorded at the end of each query. Particularly, solving time and score are calculated for the first correct answer only. The score is calculted following the scoring scheme used at the LSC challenge, shown in Equation 3.1.

Queries used in the User Experiment

The ten queries used in the experiments are defined as follows:

Q1 Planning a thesis/dissertation on a whiteboard with my PhD student, who was wearing a blue and black stripey top... in my office in 2016. We were using blue, black and green pens. After this I went back to work at my computer. It was on the 27th September.

Q2 I was organizing technology devices (phones, iPads, etc) on the wooden floor at home in an attempt to show a lifeloggers toolkit. There was a phone, an iPad, an iPad Mini, a book, and other devices on a Sunday evening in 2016.

Q3 I was taking a photo of a lake with a DSLR camera. It was my Sony camera. I was driving outside of Sheffield before and after stopping at the lake. It was in 2015 on a Saturday.

Q4 I was taking a photo of grandfather clocks while shopping in the UK. It was a Saturday in an antiques store in March 2015. I had driven a rental car to the store.

Q5 I was going into Northside Shopping Centre. I was there to get new keys. I drove to the shopping centre from work and then I drove home. It was in 2015 in the morning time.

Q6 Drinking a bottle of Budweiser beer at home. This was during a BBQ in the evening in summer 2018. I had driven back home in someone else's car before putting on the BBQ and getting the beer on a dull evening.

Q7 I was lost and looking for directions on a street, close to an Asian restaurant called Maple Leaf. It was in the late afternoon or evening and it was in Wexford. I had driven there in 2015.

Q8 Colleague in my office; she was carrying a large paper envelope full of documents. The envelope looked very heavy. She was wearing red trousers, a white shirt and a polkadot top. I remember my office door was open. It was in September in 2016. On the 27th I think, in the afternoon.

Q9 Eating a large plate of scrambled egg at home, alone in the late afternoon. I was in my living room, with the TV on and using my phone. I was sitting on my red chair with a green exercise mat visible. It was in 2016.

Q10 Birds in a cage, a yellow one on the lower left. There was also one box with a small, GREEN old car (Beetle-like). No, the car was BLUE! It was in 2018 in May. I think it was a Sunday.

A.2 Experimental Results

We will consider two components of making an efficient retrieval system: speed and correctness. In general, users from **Group 2** have outperformed those from **Group 1** in terms of both criteria, fast and precise. People using the semantic-based system have achieved nearly double the total points compared to those using the conventional system (Table A.1). While the former group has been able to resolve all queries, the latter failed to answer 40% number of the total queries (Q2, Q6, Q8, and Q9). On the other hand, the elapsed time until the first correct submission, including and excluding the unsolved tasks, were drawn in Figure A.2a and Figure A.2b, respectively. Newcomers from **Group 2** need a shorter time on average to seek the target compared to those from **Group 1**. To summarise, in this experimental

Table A.1: Total score of all users over 10 queries. The best values are highlighted in bold.

	Group 1				**Group 2**			
	User 1	User 2	User 3	User 4	User 5	User 6	User 7	User 8
Q1	66.94	65.83	43.61	50.28	70.83	76.11	**76.94**	67.78
Q2	0.00	0.00	0.00	0.00	0.00	53.33	**75.00**	72.22
Q3	58.89	56.39	46.67	68.33	71.39	67.22	**87.78**	78.33
Q4	73.61	77.78	15.00	66.94	77.22	78.33	**91.67**	78.89
Q5	71.67	0.00	86.94	0.00	83.61	58.06	**91.94**	80.56
Q6	0.00	0.00	0.00	0.00	0.00	**62.78**	0.00	0.00
Q7	0.00	60.28	0.00	0.00	**67.78**	0.00	0.00	0.00
Q8	0.00	0.00	0.00	0.00	55.56	84.17	80.28	**82.22**
Q9	0.00	0.00	0.00	0.00	77.22	**79.72**	54.44	51.11
Q10	58.61	86.67	57.50	74.44	90.00	0.00	**90.56**	53.33
Total score	329.72	346.94	249.72	260.00	593.61	559.72	**648.61**	564.44

study, using the CLIP-embedding-model-based search mechanism is more efficient for novice users to solve retrieval tasks in terms of both time and accuracy.

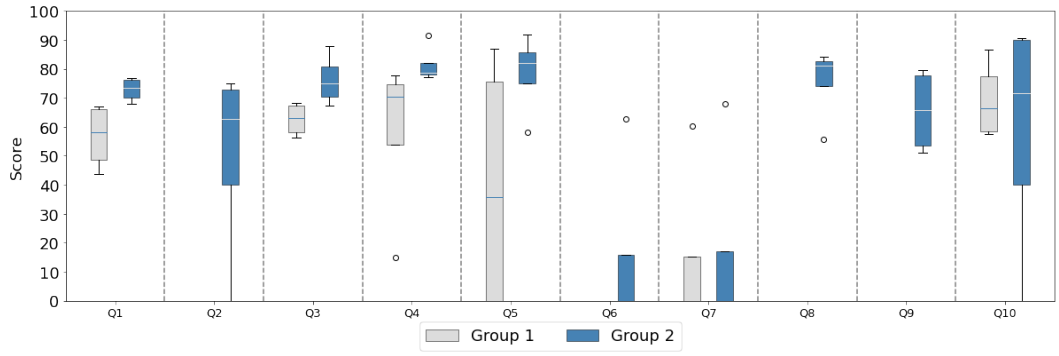
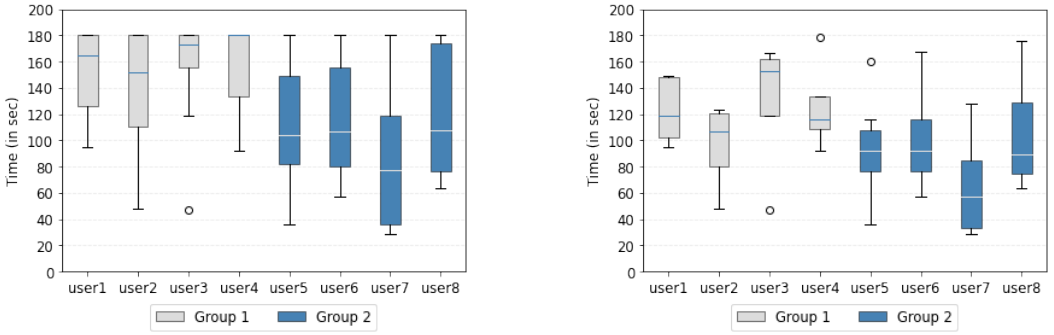


Figure A.1: Score of 2 novice user groups divided by query (best viewed in colors). Group 1 and Group 2 are denoted for participants using the concept-based system and the semantic-based system, respectively.



(a) Include unsolved task

(b) Exclude unsolved tasks

Figure A.2: Elapsed time until the first correct submission of all novice users (best viewed in colors). Group 1 and Group 2 are denoted for participants using the concept-based system and the semantic-based system, respectively.

Appendix B

NTCIR-16 Lifelog Moment Retrieval Task Results

Table B.1: All results of all runs submitted by LifeSeeker. We omit precision scores P@K where $k > 100$ as the limit of submissions for each query is 100. "Ux - Ay" stands for "User x using approach y". (There are 6 runs corresponding to 2 users with 3 post-processing ranking methods).

Run	**U1-A1**	**U2-A1**	**U1-A2**	**U2-A2**	**U1-A3**	**U2-A3**
num_q	48	48	48	48	48	48
num_ret	4629	4677	4530	4708	3714	3846
num_rel	2993	2993	2993	2993	2993	2993
num_rel_ret	320	334	238	365	229	275
map	0.0299	0.0211	0.0236	0.0237	0.0286	0.0168
gm_map	0.0007	0.0007	0.0001	0.0003	0.0002	0.0003
Rprec	0.0351	0.0380	0.0346	0.0410	0.0253	0.0306
bpref	0.1634	0.2014	0.1184	0.1493	0.0992	0.1373
recip_rank	0.1338	0.0971	0.1468	0.1475	0.1207	0.1219
iprec_at_recall_0.00	0.1421	0.1186	0.1491	0.1570	0.1290	0.1379
iprec_at_recall_0.10	0.0621	0.0650	0.0550	0.0737	0.0602	0.0625
iprec_at_recall_0.20	0.0341	0.0296	0.0449	0.0348	0.0389	0.0321
iprec_at_recall_0.30	0.0336	0.0255	0.0427	0.0314	0.0382	0.0238
iprec_at_recall_0.40	0.0335	0.0253	0.0424	0.0247	0.0171	0.0055
iprec_at_recall_0.50	0.0142	0.0083	0.0221	0.0225	0.0170	0.0055
iprec_at_recall_0.60	0.0128	0.0050	0.0018	0.0023	0.0150	0.0055
iprec_at_recall_0.70	0.0128	0.0050	0.0018	0.0023	0.0150	0.0025
iprec_at_recall_0.80	0.0128	0.0050	0.0018	0.0023	0.0150	0.0025
iprec_at_recall_0.90	0.0128	0.0050	0.0018	0.0023	0.0150	0.0025
iprec_at_recall_1.00	0.0128	0.0050	0.0018	0.0023	0.0150	0.0025
P_5	0.0833	0.0583	0.0500	0.0792	0.0792	0.0583
P_10	0.0750	0.0583	0.0583	0.0729	0.0667	0.0625
P_15	0.0750	0.0611	0.0514	0.0806	0.0611	0.0681
P_20	0.0740	0.0656	0.0490	0.0844	0.0677	0.0688
P_30	0.0681	0.0618	0.0535	0.0792	0.0646	0.0708
P_100	0.0667	0.0696	0.0496	0.0760	0.0477	0.0573

Appendix C

Ablation Studies for Video Moment Retrieval

To evaluate the contribution of each module in our method and make the best choices of the modules, we conduct a series of ablation studies. All the experiments of ablation studies, if the visual type is not mentioned, are conducted with I3D features on the test split.

C.1 Contributions of visual features and scene graphs

To give deeper insight into the contributions of the visual features and scene graphs, we conducted experiments with different combinations of visual features and scene graphs as well as each of them individually, resulting in Table C.1. We attempt to use the ground truth scene graphs during both training and test to demonstrate the potential for further performance improvement using a more powerful SGG model. Note that the Action Genome dataset annotates the scene graphs for only some keyframes of the videos, and the missing frames' features will be filled with zeros. It is possible to estimate that the performance will be further improved by accurate scene graphs for all frames.

As indicated in Table C.1, introducing the predicted scene graphs increases the recall score by roughly $2-3\%$ except for $R@1-IoU=0.7$, when compared to using only I3D or OpenCLIP features. The reason behind this increase in recall when using OpenCLIP features could be the loss of temporal information during

Table C.1: Ablation study for evaluating the contributions of the visual features (I3D, OpenCLIP) and scene graphs in our framework (Pred: generated by SGG model, GT: ground truth). The best values are highlighted in bold.

Visual features		Scene graphs		R@1	
I3D	OpenCLIP	Pred	GT	$IoU = 0.5$	$IoU = 0.7$
✓				60.2	41.1
✓		✓		64.0	43.4
✓			✓	73.9	55.2
	✓			61.2	40.5
	✓	✓		63.6	40.8
	✓		✓	75.0	**56.6**
✓	✓			62.7	42.8
✓	✓	✓		65.8	45.1
✓	✓		✓	**75.6**	56.3
		✓		53.9	32.8
			✓	71.0	51.0

the feature extraction. To reach a high temporal IoU, the temporal information needs to be preserved well in the extracted features. Compared to the I3D features each of which is extracted from a clip of continuous frames, the OpenCLIP features are extracted from single sampled frames, leading to more temporal information loss. Moreover, when using I3D and OpenCLIP features together via the parallel transformer (dual-stream or triple-stream if using scene graphs), the performance is better than that with either of the two kinds of visual features. This indicates the efficacy of ensembling different kinds of features. The results using only scene graph features, shown in the last two rows in Table C.1, degrade compared to those of the combinations with visual features. For the predicted scene graphs, the drop in performance without visual features implies that the performance of our current SGG model is unsatisfactory for the VMR task. As to the ground truth, the performance degradation is due to the information loss of the missing frames.

Table C.2: Ablation study for evaluating the importance of relational information and query-irrelevant masking for the scene graph features. The best values are highlighted in bold.

Relationship	Masking	R@1	
		$IoU = 0.5$	$IoU = 0.7$
✓	✗	63.1	41.8
✓	✓	**64.0**	**43.4**
✗	✗	60.7	41.2
✗	✓	62.6	42.2
Baseline w/o scene graphs		60.2	41.1

C.2 Scene graph features

Table C.2 shows the results of the ablation study for the scene graph features. Our baseline is the PaTF framework without the scene graph branch. As mentioned in Section 5.2.2.2, the confidence score matrix is masked to filter out irrelevant noise. Comparing the first and second rows in Table C.2, the query-irrelevant masking clearly enhances the performance of using scene graphs. Moreover, to prove the importance of the relational information, we remove the relationship dimension of the matrix and replace the scene graph feature with a $|O|$-length vector of object scores only indicating the presence of objects, with the results displayed in the third and fourth rows. Additionally, the object-score vector slightly improves the baseline performance. Although the improvement becomes clearer with the query-irrelevant masking, the recalls of using the scene graph features are higher by 1.4% and 1.2%, highlighting the importance of the relational information.

Table C.3: Comparison on transformer module's architecture. The best values are highlighted in bold.

Transformer module	Block	R@1	
		$IoU = 0.5$	$IoU = 0.7$
Vanilla	Self-Attention	63.3	42.1
Dual	Self-Attention	**64.0**	43.4
Dual	Cross-Attention	63.1	44.1
Dual	Combination	63.7	**43.6**

C.3 Parallel transformer module

We further make a comparison between the vanilla and parallel transformer modules, shown in Table C.3. The vanilla one takes concatenation of the visual and scene graph features as input while the parallel one processes the two feature streams separately. We also compare three different kinds of attention blocks for the parallel module, a self-attention block, a cross-attention block, and a combination of the two blocks, as depicted in Figure 5.4. The results show that the three parallel modules perform similarly, but all outperform the vanilla module with concatenated features, which indicates: (1) ensembling the features processed by two separate branches can benefit the integration of the visual cues and relational contexts as compared to a single branch with feature concatenation; (2) interaction between the two branches might be not essential in this specific case.

C.4 Vision-text fusion

In Table C.4, we compare several approaches for the fusion between the textual features and the visual features. The results are obtained using only the I3D features. The first one in Table C.4, "*Aggr.*" used in the proposed method, aggregates all the tokens output by OpenCLIP's text encoder with a trainable aggregation layer, while the second one aggregates the tokens with the average pooling. The third one uses only the EOT token (the last token in the text embedding sequence) of OpenCLIP

Table C.4: Comparison on approaches of vision-text fusion. The best values are highlighted in bold.

Module	Tokens	Concat dim	R@1 $IoU = 0.5$	R@1 $IoU = 0.7$
Aggr.	All	Channel	**60.2**	**41.1**
AvgPool	All	Channel	**60.2**	40.1
–	EOT	Channel	59.2	39.8
–	All	Temporal	59.0	38.9
Cross-att	All	–	59.3	38.9

without an additional module. The first three approaches concatenate the visual and text features along the dimension of the feature channel. The fourth one, used in UnLoc [178], concatenates all the tokens with the visual features along the temporal dimension. The last one, used in QD-DETR [58], fuses the features with a cross-modal attention module. Among the approaches, our aggregation module performs the best. While the previous work [58] reported performance improvement by the cross-modal attention, this module fails to demonstrate any improvement for our method. Possible explanations could be the difference in framework, prediction based on temporal-sequence features (ours), or learned queries (QD-DETR).

Appendix D

A Novel Moment-based Lifelog Retrieval System

D.1 Queries used in the User Experiment

D.1.1 Lifelog Search Challenge 2020 Queries

The Lifelog Search Challenge 2020 (LSC'20) queries [1] used in the experiments are listed as follows:

Table D.1: The modified LSC'22 queries and their corresponding filters used in the experiment. The query consists of location and time information which is separated by the ";".

Query ID	**Query**	**Filter**
LSC-66	I was shaving for a TV recording in front of my bathroom mirror. The TV crew came to record some video.	;Wednesday 2016
LSC-49	The Red House. I remember there were lots of cars that day and the weather was very nice with a beautiful blue sky. I was just after eating an icecream by the sea.	;Monday evening 2018

Continued on next page

[1] For the detailed descriptions of the LSC 2020 topics, please refer to `http://www.lifelogsearch.org/lsc/2021/resources/lsc20-topics-qrels.txt`

Table D.1 – continued from previous page

Query ID	**Query**	**Filter**
LSC-51	There was a Chandelier (large glass light) hanging from the ceiling, and red chairs. Actually there were lots of chandeliers.	Dublin, Trinity College; Tuesday 2015
LSC-52	Getting a taxi from the Railway station. It was a red taxi with a very rude driver. I was just after getting off a short (but very fast) maglev train journey.	China; 2015
LSC-53	The diamond shaped wooden sign was high up in the air. It was when I was walking beside the sea.	;Tuesday
LSC-54	I am looking for my yellow staff card. I remember putting it in my wallet.	work; Tuesday afternoon May 2018
LSC-55	Feeding the Dog. He was at the table waiting to be given food in the garden.	; Saturday night 2018
LSC-57	I was having beer after a long day of meetings. It was a 'corona extra' beer in a bottle.	Wuhan; May 2018
LSC-58	I was buying a wicker picnic basket. It was a straw/wicker picnic basket with a white bottom. I also bought some salt lamps at Carraig Donn.	Carraig Donn; May 2018
LSC-59	Just looking at coffee machines one evening in a luxury store in Dublin called 'Brown Thomas'. Afterwards I had a birthday dinner in Sole restaurant.	Dublin, Sole; evening 31/05/2018

Continued on next page

Table D.1 – continued from previous page

Query ID	**Query**	**Filter**
LSC-60	My old car needed a wheel repair so I brought it to a repair store/garage. A man (not me) worked on the front left wheel.	; Monday March 2015
LSC-61	Looking at ancient Chinese vases in a museum. There were two of them. They were blue and white in a wood and glass case.	China; May 2018
LSC-62	A VR rollercoaster on a long staircase controlled using a handheld games controller. It was black and white, almost ghostly.	Science Gallery; Tuesday evening in 2015
LSC-63	Passing by a clocktower while running in a park near my home.	Saturday morning in February
LSC-64	I was alone drinking tea and eating cake in an antiques store.	store, UK; Saturday morning
LSC-68	Flying on a BAE 146. I was looking out the aircraft window to see two under-wing jet engines on a flight from Dublin to London.	airport; Monday 16/03
LSC-69	A driver's viewpoint from a train on the TV at home.	;Monday evening in May
LSC-70	Ordering Fast Food from McDonalds Restaurant. I was queuing to order the food for less than two minutes.	McDonalds, airport, China; 2018
LSC-71	Checking out of the Yeats Country Hotel after having breakfast with one other person before driving home (via some shops).	Yeats Country Hotel, Sligo; Sunday 18/09/2016

Continued on next page

Table D.1 – continued from previous page

Query ID	**Query**	**Filter**
LSC-72	Taking a photograph of an A380 airplane in Germany before boarding a flight in the late afternoon in 2015 on the 19th March.	Germany; 19/03/2015
LSC-73	Speaking with a microphone in my hand at a lunch-time lecture to a crowd in a large industrial building.	Dublin; Wednesday, 2015
LSC-74	Four red figures, maybe they are aliens. It looked like a painting of aliens. There were walking on the desert. There was a big red wall behind the painting.	; March 2015

Bibliography

[1] Steve Hodges, Lyndsay Williams, Emma Berry, Shahram Izadi, James Srinivasan, Alex Butler, Gavin Smyth, Narinder Kapur, and Ken Wood. SenseCam: A Retrospective Memory Aid. In Paul Dourish and Adrian Friday, editors, *UbiComp 2006: Ubiquitous Computing*, pages 177–193, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[2] Jim Gemmell, Gordon Bell, Roger Lueder, Steven Drucker, and Curtis Wong. MyLifeBits: Fulfilling the Memex Vision. In *Proceedings of the Tenth ACM International Conference on Multimedia*, MULTIMEDIA '02, page 235–238, New York, NY, USA, 2002. Association for Computing Machinery.

[3] Aleks Aris, Jim Gemmell, and Roger Lueder. Exploiting Location and Time for Photo Search and Storytelling in MyLifeBits. Technical Report MSR-TR-2004-102, October 2004.

[4] Hyowon Lee, Alan Smeaton, Noel O'Connor, Gareth Jones, Michael Blighe, Daragh Byrne, Aiden Doherty, and Cathal Gurrin. Constructing a SenseCam visual diary as a media process. *Multimedia Systems*, 14:341–349, 12 2008.

[5] Silvan Heller, Ralph Gasser, Mahnaz Parian-Scherb, Sanja Popovic, Luca Rossetto, Loris Sauter, Florian Spiess, and Heiko Schuldt. Interactive Multimodal Lifelog Retrieval with Vitrivr at LSC 2021. In *Proceedings of the 4th Annual on Lifelog Search Challenge*, LSC '21, page 35–39, New York, NY, USA, 2021. Association for Computing Machinery.

[6] Ly-Duyen Tran, Manh-Duy Nguyen, Binh Nguyen, Hyowon Lee, Liting Zhou, and Cathal Gurrin. E-myscéal: Embedding-based interactive lifelog retrieval system for lsc'22. In *Proceedings of the 5th Annual on Lifelog Search Challenge*,

LSC '22, page 32–37, New York, NY, USA, 2022. Association for Computing Machinery.

[7] Maria Tysse Hordvik, Julie Sophie Teilstad Østby, Manoj Kesavulu, Thao-Nhu Nguyen, Tu-Khiem Le, and Duc-Tien Dang-Nguyen. LifeLens: Transforming Lifelog Search with Innovative UX/UI Design. In *Proceedings of the 6th Annual ACM Lifelog Search Challenge*, LSC '23, page 1–6, New York, NY, USA, 2023. Association for Computing Machinery.

[8] Aaron Duane, Björn Þór Jónsson, and Cathal Gurrin. VRLE: Lifelog Interaction Prototype in Virtual Reality: Lifelog Search Challenge at ACM ICMR 2020. In *Proceedings of the Third Annual Workshop on Lifelog Search Challenge*, LSC '20, page 7–12, New York, NY, USA, 2020. Association for Computing Machinery.

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

[10] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[11] Ly-Duyen Tran, Thanh Cong Ho, Lan Anh Pham, Binh Nguyen, Cathal Gurrin, and Liting Zhou. LLQA - Lifelog Question Answering Dataset. In *MultiMedia Modeling: 28th International Conference, MMM 2022, Phu Quoc, Vietnam, June 6–10, 2022, Proceedings, Part I*, page 217–228, Berlin, Heidelberg, 2022. Springer-Verlag.

[12] Aiden R. Doherty, Katalin Pauly-Takacs, Niamh Caprani, Cathal Gurrin, Chris J. A. Moulin, Noel E. O'Connor, and Alan F. Smeaton. Experiences

of Aiding Autobiographical Memory Using the SenseCam. *Human–Computer Interaction*, 27(1-2):151–174, 2012.

[13] Arthur W. Melton. Implications of short-term memory for a general theory of memory. *Journal of Verbal Learning and Verbal Behavior*, 2(1):1–21, 1963.

[14] Alan Baddeley. *Your Memory: A User's Guide*. Carlton Books, 2004.

[15] Endel Tulving and Donald M. Thomson. Encoding specificity and retrieval processes in episodic memory. *Psychological Review*, 80:352–373, 1973.

[16] Jesse Purdy, Michael Markham, Bennett Schwartz, and William Gordon. *Learning and Memory, 2nd. Ed.* 01 2001.

[17] David Elsweiler. Supporting human memory in personal information management. *SIGIR Forum*, 42:75–76, 2008.

[18] Endel Tulving. *Relation between encoding specificity and levels of processing*, pages 405–428. Lawrence Erlbaum Associates, 1979.

[19] Margaret Jean Intons-Peterson and George L. Newsome III. External memory aids: Effects and effectiveness. In *Memory improvement: Implications for memory theory*, pages 101–121. Springer, 1992.

[20] Margaret Jean Intons-Peterson. External memory aids and their relation to memory. In *Cognitive psychology applied*, pages 135–158. Psychology Press, 2014.

[21] Stanley B Klein. What memory is. *Wiley Interdisciplinary Reviews: Cognitive Science*, 6(1):1–38, 2015.

[22] Cathal Gurrin, Alan F. Smeaton, and Aiden R. Doherty. LifeLogging: Personal Big Data. *Found. Trends Inf. Retr.*, 8(1):1–125, jun 2014.

[23] E. Loftus and K. Ketcham. *Witness for the Defense: The Accused, the*

Eyewitness, and the Expert Who Puts Memory on Trial. St. Martin's Publishing Group, 1991.

[24] Vannevar Bush. As We May Think. *Atl. Mon.*, 176(1):101–108, 1945.

[25] Cathal Gurrin, Hideo Joho, Frank Hopfgartner, Liting Zhou, Rami Albatal, Graham Healy, and Duc-Tien Nguyen. *Experiments in Lifelog Organisation and Retrieval at NTCIR*, pages 187–203. Springer Singapore, Singapore, 2021.

[26] Martin Dodge and Rob Kitchin. Outlines of a world coming into existence: Pervasive computing and the ethics of forgetting. Environment and Planning B: Planning and Design, 34, 431-445. *Environment and Planning B: Planning and Design*, 34:431–445, 05 2007.

[27] Steve Mann. Wearable Computing: A First Step Toward Personal Imaging. *Computer*, 30(2):25–32, 1997.

[28] Kiyoharu Aizawa, Datchakorn Tancharoen, Shinya Kawasaki, and Toshihiko Yamasaki. Efficient retrieval of life log based on context and content. *CARPE'04 - Proceedings of the First ACM Workshop on Continuous Archival and Retrieval of Personal Experiences*, pages 22–31, 10 2004.

[29] Jim Gemmell, Gordon Bell, and Roger Lueder. MyLifeBits: a personal database for everything. *Communications of the ACM (CACM)*, 49:88–95, January 2006. also as MSR-TR-2006-23.

[30] Cathal Gurrin, Daragh Byrne, Noel O'Connor, Gareth Jones, and Alan Smeaton. Architecture and challenges of maintaining a large-scale, context-aware Human Digital Memory. pages 158 – 163, 09 2008.

[31] Cathal Gurrin, Zhengwei Qiu, Mark Hughes, Niamh Caprani, Steve Hodges, and Alan Smeaton. The Smartphone As a Platform for Wearable Cameras in Health Research. *American Journal of Preventive Medicine*, March 2013.

[32] Zaher Hinbarji, Rami Albatal, Noel O'Connor, and Cathal Gurrin. LoggerMan, a Comprehensive Logging and Visualization Tool to Capture Computer Usage. pages 342–347, 01 2016.

[33] Emma Woodberry, Georgina Browne, Steve Hodges, Peter Watson, Narinder Kapur, and Ken Woodberry. The use of a wearable camera improves autobiographical memory in patients with Alzheimer's disease. *Memory (Hove, England)*, 23, 02 2014.

[34] Paulina Piasek, Kate Irving, and Alan Smeaton. Case study in SenseCam use as an intervention technology for early-stage dementia. *International Journal of Computers in Healthcare*, 1:304–319, 01 2012.

[35] Abdullah Mahmud, Jeffrey Braun, and Jean-bernard Martens. Designing to capture and share life experiences for persons with aphasia. pages 391–392, 09 2010.

[36] Emma Berry, Narinder Kapur, Lyndsay Williams, Steve Hodges, Peter Watson, Gavin Smyth, James Srinivasan, Reg Smith, Barbara Wilson, and Ken Woodberry. The use of a wearable camera, SenseCam, as a pictorial diary to improve autobiographical memory in a patient with limbic encephalitis: A preliminary report. *Neuropsychological rehabilitation*, 17:582–601, 08 2007.

[37] Abigail J. Sellen, Andrew Fogg, Mike Aitken, Steve Hodges, Carsten Rother, and Ken Wood. Do Life-Logging Technologies Support Memory for the Past? An Experimental Study Using Sensecam. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '07, page 81–90, New York, NY, USA, 2007. Association for Computing Machinery.

[38] Vaiva Kalnikaitė and Steve Whittaker. Beyond being there? Evaluating augmented digital records. *International Journal of Human-Computer Studies*, 68(10):627–640, 2010.

[39] Aiden R. Doherty, Cathal Gurrin, and Alan F. Smeaton. An Investigation into Event Decay from Large Personal Media Archives. In *Proceedings of the 1st ACM International Workshop on Events in Multimedia*, EiMM '09, page 49–56, New York, NY, USA, 2009. Association for Computing Machinery.

[40] Daniel L. Schacter and Endel Tulving. What Are the Memory Systems of 1994? In *Memory Systems 1994*, pages 1–38. MIT Press, Cambridge, MA, USA, 1994.

[41] Abigail Sellen and Steve Whittaker. Beyond Total Capture: A Constructive Critique of Lifelogging. *Commun. ACM*, 53:70–77, 05 2010.

[42] Morgan Harvey, Marc Langheinrich, and Geoff Ward. Remembering through lifelogging: A survey of human memory augmentation. *Pervasive and Mobile Computing*, 27:14–26, 2016.

[43] E. Berry, A. Hampshire, J. Rowe, Steve Hodges, N. Kapur, Peter Watson, Georgina Browne, G. Smyth, Ken Woodberry, and Adrian Owen. The neural basis of effective memory therapy in a patient with limbic encephalitis. *Journal of Neurology, Neurosurgery & Psychiatry*, 80, 12 2009.

[44] Cathal Gurrin, Hideo Joho, Frank Hopfgartner, Liting Zhou, and Rami Albatal. Overview of NTCIR-12 Lifelog Task. In *NTCIR*, 2016.

[45] Duc Tien Dang Nguyen, Luca Piras, Michael Riegler, G. Boato, Liting Zhou, and Cathal Gurrin. Overview of ImageCLEFlifelog 2017: Lifelog Retrieval and Summarization. 08 2017.

[46] Cathal Gurrin, Klaus Schoeffmann, Hideo Joho, Bernd Münzer, Rami Albatal, Frank Hopfgartner, Liting Zhou, and Duc Tien Dang Nguyen. *A Test Collection for Interactive Lifelog Retrieval*, pages 312–324. 01 2019.

[47] Cathal Gurrin, Liting Zhou, Graham Healy, Björn Þór Jónsson, Duc-Tien Dang-Nguyen, Jakub Lokoć, Minh-Triet Tran, Wolfgang Hürst, Luca Rossetto,

and Klaus Schöffmann. Introduction to the Fifth Annual Lifelog Search Challenge, LSC'22. In *Proceedings of the 2022 International Conference on Multimedia Retrieval*, ICMR '22, page 685–687, New York, NY, USA, 2022. Association for Computing Machinery.

[48] Jiyang Gao, Chen Sun, Zhenheng Yang, and Ram Nevatia. Tall: Temporal activity localization via language query. In *Proceedings of the IEEE international conference on computer vision*, pages 5267–5275, 2017.

[49] Shaoning Xiao, Long Chen, Songyang Zhang, Wei Ji, Jian Shao, Lu Ye, and Jun Xiao. Boundary Proposal Network for Two-Stage Natural Language Video Localization. *CoRR*, abs/2103.08109, 2021.

[50] Yawen Zeng, Da Cao, Xiaochi Wei, Meng Liu, Zhou Zhao, and Zheng Qin. Multi-Modal Relational Graph for Cross-Modal Video Moment Retrieval. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2215–2224, 2021.

[51] Jingyuan Chen, Xinpeng Chen, Lin Ma, Zequn Jie, and Tat-Seng Chua. Temporally Grounding Natural Sentence in Video. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii, editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 162–171, Brussels, Belgium, October-November 2018. Association for Computational Linguistics.

[52] Hao Zhang, Aixin Sun, Wei Jing, and Joey Tianyi Zhou. Span-based Localizing Network for Natural Language Video Localization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6543–6554, Online, July 2020. Association for Computational Linguistics.

[53] Zhu Zhang, Zhijie Lin, Zhou Zhao, and Zhenxin Xiao. Cross-Modal Interaction Networks for Query-Based Moment Retrieval in Videos. In *Proceedings of the*

42nd International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, jul 2019.

[54] J. Gao and C. Xu. Fast Video Moment Retrieval. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1503–1512, Los Alamitos, CA, USA, oct 2021. IEEE Computer Society.

[55] Haoyu Tang, Jihua Zhu, Lin Wang, Qinghai Zheng, and Tianwei Zhang. Multi-Level Query Interaction for Temporal Language Grounding. *IEEE Transactions on Intelligent Transportation Systems*, 23(12):25479–25488, 2022.

[56] Jie Lei, Tamara L. Berg, and Mohit Bansal. QVHighlights: Detecting Moments and Highlights in Videos via Natural Language Queries. *CoRR*, abs/2107.09609, 2021.

[57] Yifang Xu, Yunzhuo Sun, Yang Li, Yilei Shi, Xiaoxiang Zhu, and Sidan Du. Mh-detr: Video moment and highlight detection with cross-modal transformer. *arXiv preprint arXiv:2305.00355*, 2023.

[58] WonJun Moon, Sangeek Hyun, SangUk Park, Dongchan Park, and Jae-Pil Heo. Query-dependent video representation for moment retrieval and highlight detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23023–23033, 2023.

[59] Jeffrey Zacks, Barbara Tversky, and Gowri Iyer. Perceiving, remembering, and communicating structure in events. *Journal of experimental psychology. General*, 130:29–58, 04 2001.

[60] Jeffrey Zacks, Nicole Speer, Khena Swallow, Todd Braver, and Jeremy Reynolds. Event Perception: A Mind/Brain Perspective. *Psychological bulletin*, 133:273–93, 04 2007.

[61] Cathal Gurrin, Alan F. Smeaton, Zhengwei Qiu, and Aiden Doherty. Exploring the technical challenges of large-scale lifelogging. In *Proceedings of the 4th*

International SenseCam & Pervasive Imaging Conference, SenseCam '13, page 68–75, New York, NY, USA, 2013. Association for Computing Machinery.

[62] Lydia Dubourg, Ana Rita Silva, Christophe Fitamen, Chris Moulin, and Souchay Céline. SenseCam: A new tool for memory rehabilitation? *Revue Neurologique*, 172, 11 2016.

[63] Mark Hughes, Eamonn Newman, Alan F. Smeaton, and Noel E. O'Connor. A lifelogging approach to automated market research. 2012.

[64] Sasank Reddy, Andrew Parker, Josh Hyman, Jeff Burke, Deborah Estrin, and Mark Hansen. Image Browsing, Processing, and Clustering for Participatory Sensing: Lessons from a DietSense Prototype. In *Proceedings of the 4th Workshop on Embedded Networked Sensors*, EmNets '07, page 13–17, New York, NY, USA, 2007. Association for Computing Machinery.

[65] Kiyoharu Aizawa, Yuto Maruyama, He Li, and Chamin Morikawa. Food Balance Estimation by Using Personal Dietary Tendencies in a Multimedia Food Log. *IEEE Transactions on Multimedia*, 15(8):2176–2185, 2013.

[66] Vangelis Metsis, Dimitris Kosmopoulos, Vassilis Athitsos, and Fillia Makedon. Non-invasive analysis of sleep patterns via multimodal sensor input. *Personal and Ubiquitous Computing*, 18, 01 2014.

[67] Floriano Zini, Martin Reinstadler, and Francesco Ricci. Life-logs Aggregation for Quality of Life Monitoring. pages 131–132, 05 2015.

[68] Paul Kelly, Aiden Doherty, Emma Berry, Steve Hodges, Alan Batterham, and Charles Foster. Can we use digital life-log images to investigate active and sedentary behavior? Results from a pilot study. *The international journal of behavioral nutrition and physical activity*, 8:44, 05 2011.

[69] Joshua McVeigh-Schultz, Jennifer Stein, Jacob Boyle, Emily Duff, Jeff Watson, Avimaan Syam, Amanda Tasse, Simon Wiscombe, and Scott Fisher. Vehicular

lifelogging: New contexts and methodologies for human-car interaction. 05 2012.

[70] Jana Machajdik, Allan Hanbury, Angelika Garz, and Robert Sablatnig. Affective Computing for Wearable Diary and Lifelogging Systems: An Overview. 01 2011.

[71] Aiden Doherty and Alan Smeaton. Automatically Segmenting LifeLog Data into Events. pages 20–23, 06 2008.

[72] Marti A. Hearst and Christian Plaunt. Subtopic structuring for full-length document access. In *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '93, page 59–68, New York, NY, USA, 1993. Association for Computing Machinery.

[73] Michael Blighe, Aiden Doherty, Alan F. Smeaton, and Noel E. O'Connor. Keyframe detection in visual lifelogs. In *Proceedings of the 1st International Conference on PErvasive Technologies Related to Assistive Environments*, PETRA '08, New York, NY, USA, 2008. Association for Computing Machinery.

[74] Aiden R. Doherty and Alan F. Smeaton. Combining Face Detection and Novelty to Identify Important Events in a Visual Lifelog. In *Proceedings of the 2008 IEEE 8th International Conference on Computer and Information Technology Workshops*, CITWORKSHOPS '08, page 348–353, USA, 2008. IEEE Computer Society.

[75] Cathal Gurrin, Hideo Joho, Frank Hopfgartner, Liting Zhou, and Rami Albatal. NTCIR Lifelog: The First Test Collection for Lifelog Research. In *NTCIR*, pages 705–708, 07 2016.

[76] Cathal Gurrin, H. Joho, Frank Hopfgartner, Liting Zhou, Tu Ninh, Tu-Khiem Le, Rami Albatal, Duc-Tien Dang-Nguyen, and Graham Healy. Overview of the NTCIR-14 Lifelog-3 task. In *CLEF*, 06 2019.

[77] Duc-Tien Dang-Nguyen, Luca Piras, Michael Riegler, Liting Zhou, Mathias Lux, and Cathal Gurrin. Overview of ImageCLEFlifelog 2018: Daily Living Understanding and Lifelog Moment Retrieval. In *CLEF (working notes)*, 2018.

[78] Van-Tu Ninh, Tu-Khiem Le, Liting Zhou, Luca Piras, Michael Alexander Riegler, Pål Halvorsen, Mathias Lux, Minh-Triet Tran, Cathal Gurrin, and Duc Tien Dang Nguyen. Overview of ImageCLEF Lifelog 2020: Lifelog Moment Retrieval and Sport Performance Lifelog. 2020.

[79] Duc Tien Dang Nguyen, Luca Piras, Michael Riegler, Liting Zhou, Mathias Lux, Minh Triet Tran, Tu-Khiem Le, Van-Tu Ninh, and Cathal Gurrin. Overview of ImageCLEFlifelog 2019: solve my life puzzle and lifelog moment retrieval. In *CLEF*. CEUR Workshop Proceedings, 07 2019.

[80] Cathal Gurrin, Klaus Schoeffmann, Hideo Joho, Andreas Leibetseder, Liting Zhou, Aaron Duane, Duc Tien Dang Nguyen, Michael Riegler, Luca Piras, Minh-Triet Tran, Jakub Lokoč, and Wolfgang Hürst. Comparing Approaches to Interactive Lifelog Search at the Lifelog Search Challenge (LSC2018). *ITE Transactions on Media Technology and Applications*, 7:46–59, 04 2019.

[81] Cathal Gurrin, Tu-Khiem Le, Van-Tu Ninh, Duc-Tien Dang-Nguyen, Björn ór Jónsson, Jakub Lokoč, Wolfgang Hürst, Minh-Triet Tran, and Klaus Schöffmann. *Introduction to the Third Annual Lifelog Search Challenge (LSC'20)*, page 584–585. Association for Computing Machinery, New York, NY, USA, 2020.

[82] Cathal Gurrin, Björn ór Jónsson, Klaus Schöffmann, Duc-Tien Dang-Nguyen, Jakub Lokoč, Minh-Triet Tran, Wolfgang Hürst, Luca Rossetto, and Graham Healy. Introduction to the Fourth Annual Lifelog Search Challenge, LSC'21. In *Proceedings of the 2021 International Conference on Multimedia Retrieval*, ICMR '21, page 690–691, New York, NY, USA, 2021. Association for Computing Machinery.

[83] Liting Zhou, Duc-Tien Dang-Nguyen, and Cathal Gurrin. A Baseline Search Engine for Personal Life Archives. *Proceedings of the 2nd Workshop on Lifelogging Tools and Applications*, 2017.

[84] Bahjat Safadi, Philippe Mulhem, Georges Quénot, and Jean-Pierre Chevallet. LIG-MRIM at NTCIR-12 Lifelog Semantic Access Task. In *12th NTCIR Conference on Evaluation of Information Access Technologies*, Proceedings of the 12th NTCIR Conference on Evaluation of Information Access Technologies, Tokyo, Japan, June 2016.

[85] Hsiang Lun Lin, Tzu-Chieh Chiang, Liang-Pu Chen, and Ping-Che Yang. Image Searching by Events with Deep Learning for NTCIR-12 Lifelog. In Noriko Kando, Tetsuya Sakai, and Mark Sanderson, editors, *Proceedings of the 12th NTCIR Conference on Evaluation of Information Access Technologies, National Center of Sciences, Tokyo, Japan, June 7-10, 2016*. National Institute of Informatics (NII), 2016.

[86] Gabriel de Oliveira Barra, Alejandro Cartas Ayala, Marc Bolaños, Mariella Dimiccoli, Xavier Giró i Nieto, and Petia Radeva. LEMoRe: A Lifelog Engine for Moments Retrieval at the NTCIR-Lifelog LSAT Task. In *NTCIR*, 2016.

[87] Harrisen Scells, Guido Zuccon, and Kirsty Kitto. QUT at the NTCIR Lifelog Semantic Access Task. In *NTCIR*, 2016.

[88] Ana Garcia del Molino, Bappaditya Mandal, Jie Lin, Joo-Hwee Lim, Vigneshwaran Subbaraju, and Vijay Ramaseshan Chandrasekhar. VC-I2R@ImageCLEF2017: Ensemble of Deep Learned Features for Lifelog Video Summarization. In *Conference and Labs of the Evaluation Forum*, 2017.

[89] Mihai Dogariu and Bogdan Ionescu. Multimedia Lab @ ImageCLEF 2018 Lifelog Moment Retrieval Task. In *Conference and Labs of the Evaluation Forum*, 2018.

[90] Ergina Kavallieratou, Carlos R. del Blanco, Carlos Cuevas, and Narciso García. Retrieving Events in Life Logging. In *Conference and Labs of the Evaluation Forum*, 2018.

[91] Minh-Triet Tran, Tung Dinh-Duy, Truong Dat, Khoa Vo, Quoc Luong, and Vinh-Tiep Nguyen. Lifelog Moment Retrieval with Visual Concept Fusion and Text-based Query Expansion. 06 2018.

[92] Fatma Abdallah, Ghada Feki, Ben ammar Anis, and Chokri Ben Amar. Big Data For Lifelog Moments Retrieval Improvement. 07 2019.

[93] Ricardo F. Ribeiro, António J. R. Neves, and José Luís Oliveira. PT Bioinformatics at ImageCLEF 2019 : Lifelog Moment Retrieval based on Image Annotation and Natural Language Processing. 2019.

[94] Nguyen-Khang Le, Dieu-Hien Nguyen, Vinh-Tiep Nguyen, and Minh-Triet Tran. Lifelog Moment Retrieval with Advanced Semantic Extraction and Flexible Moment Visualization for Exploration. In *Conference and Labs of the Evaluation Forum*, 2019.

[95] Minh Dao, Khoa Vo, Trong-Dat Phan, and Koji Zettsu. BIDAL@imageCLEFlifelog2019: The Role of Content and Context of Daily Activities in Insights from Lifelogs. 11 2019.

[96] Pengfei Zhou, Cong Bai, and Jie Xia. Zjutcvr team at imagecleflifelog2019 lifelog moment retrieval task. In *Conference and Labs of the Evaluation Forum*, 2019.

[97] Maxime Tournadre, Guillaume Dupont, Vincent Pauwels, Bezeid Cheikh, Mohamed Lmami, and Alexandru Ginsca. A Multimedia Modular Approach to Lifelog Moment Retrieval. 2380, 08 2019.

[98] *LSC '23: Proceedings of the 6th Annual ACM Lifelog Search Challenge*, New York, NY, USA, 2023. Association for Computing Machinery.

[99] Thao-Nhu Nguyen, Tu-Khiem Le, Van-Tu Ninh, Minh-Triet Tran, Nguyen Thanh Binh, Graham Healy, Annalina Caputo, and Cathal Gurrin. LifeSeeker 3.0: An Interactive Lifelog Search Engine for LSC'21. In *Proceedings of the 4th Annual on Lifelog Search Challenge*, LSC '21, page 41–46, New York, NY, USA, 2021. Association for Computing Machinery.

[100] Thao-Nhu Nguyen, Tu-Khiem Le, Van-Tu Ninh, Cathal Gurrin, Minh-Triet Tran, Thanh Binh Nguyen, Graham Healy, Annalina Caputo, and Sinead Smyth. E-LifeSeeker: An Interactive Lifelog Search Engine for LSC'23. In *Proceedings of the 6th Annual ACM Lifelog Search Challenge*, LSC '23, page 13–17, New York, NY, USA, 2023. Association for Computing Machinery.

[101] Nhat Hoang-Xuan, Hoang-Phuc Trang-Trung, E-Ro Nguyen, Thanh-Cong Le, Mai-Khiem Tran, Tu-Khiem Le, Van-Tu Ninh, Cathal Gurrin, and Minh-Triet Tran. Flexible Interactive Retrieval SysTem 3.0 for Visual Lifelog Exploration at LSC 2022. In *Proceedings of the 5th Annual on Lifelog Search Challenge*, LSC '22, page 20–26, New York, NY, USA, 2022. Association for Computing Machinery.

[102] Nhat Hoang-Xuan, Thang-Long Nguyen-Ho, Cathal Gurrin, and Minh-Triet Tran. Lifelog Discovery Assistant: Suggesting Prompts and Indexing Event Sequences for FIRST at LSC 2023. In *Proceedings of the 6th Annual ACM Lifelog Search Challenge*, LSC '23, page 47–52, New York, NY, USA, 2023. Association for Computing Machinery.

[103] Luca Rossetto, Oana Inel, Svenja Lange, Florian Ruosch, Ruijie Wang, and Abraham Bernstein. Multi-Mode Clustering for Graph-Based Lifelog Retrieval. In *Proceedings of the 6th Annual ACM Lifelog Search Challenge*, LSC '23, page 36–40, New York, NY, USA, 2023. Association for Computing Machinery.

[104] Tien-Thanh Nguyen-Dang, Xuan-Dang Thai, Gia-Huy Vuong, Van-Son Ho, Minh-Triet Tran, Van-Tu Ninh, Minh-Khoi Pham, Tu-Khiem Le, and

Graham Healy. LifeInsight: An Interactive Lifelog Retrieval System with Comprehensive Spatial Insights and Query Assistance. In *Proceedings of the 6th Annual ACM Lifelog Search Challenge*, LSC '23, page 59–64, New York, NY, USA, 2023. Association for Computing Machinery.

[105] Naushad Alam, Yvette Graham, and Cathal Gurrin. Memento 2.0: An Improved Lifelog Search Engine for LSC'22. In *Proceedings of the 5th Annual on Lifelog Search Challenge*, LSC '22, page 2–7, New York, NY, USA, 2022. Association for Computing Machinery.

[106] Naushad Alam, Yvette Graham, and Cathal Gurrin. Memento 3.0: An Enhanced Lifelog Search Engine for LSC'23. In *Proceedings of the 6th Annual ACM Lifelog Search Challenge*, LSC '23, page 41–46, New York, NY, USA, 2023. Association for Computing Machinery.

[107] Ricardo Ribiero, Alina Trifan, and Antonio J. R. Neves. MEMORIA: A Memory Enhancement and MOment RetrIeval Application for LSC 2022. In *Proceedings of the 5th Annual on Lifelog Search Challenge*, LSC '22, page 8–13, New York, NY, USA, 2022. Association for Computing Machinery.

[108] Ricardo Ribeiro, Luísa Amaral, Wei Ye, Alina Trifan, António J. R. Neves, and Pedro Iglésias. MEMORIA: A Memory Enhancement and MOment RetrIeval Application for LSC 2023. In *Proceedings of the 6th Annual ACM Lifelog Search Challenge*, LSC '23, page 18–23, New York, NY, USA, 2023. Association for Computing Machinery.

[109] Silvan Heller, Luca Rossetto, Loris Sauter, and Heiko Schuldt. vitrivr at the Lifelog Search Challenge 2022. In *Proceedings of the 5th Annual on Lifelog Search Challenge*, LSC '22, page 27–31, New York, NY, USA, 2022. Association for Computing Machinery.

[110] Ly Duyen Tran, Binh Nguyen, Liting Zhou, and Cathal Gurrin. MyEachtra: Event-Based Interactive Lifelog Retrieval System for LSC'23. In *Proceedings*

of the 6th Annual ACM Lifelog Search Challenge, LSC '23, page 24–29, New York, NY, USA, 2023. Association for Computing Machinery.

[111] Florian Spiess and Heiko Schuldt. Multimodal Interactive Lifelog Retrieval with vitrivr-VR. In *Proceedings of the 5th Annual on Lifelog Search Challenge*, LSC '22, page 38–42, New York, NY, USA, 2022. Association for Computing Machinery.

[112] Florian Spiess, Ralph Gasser, Heiko Schuldt, and Luca Rossetto. The Best of Both Worlds: Lifelog Retrieval with a Desktop-Virtual Reality Hybrid System. In *Proceedings of the 6th Annual ACM Lifelog Search Challenge*, LSC '23, page 65–68, New York, NY, USA, 2023. Association for Computing Machinery.

[113] Ahmed Alateeq, Mark Roantree, and Cathal Gurrin. Voxento 3.0: A Prototype Voice-Controlled Interactive Search Engine for Lifelog. In *Proceedings of the 5th Annual on Lifelog Search Challenge*, LSC '22, page 43–47, New York, NY, USA, 2022. Association for Computing Machinery.

[114] Ahmed Alateeq, Mark Roantree, and Cathal Gurrin. Voxento 4.0: A More Flexible Visualisation and Control for Lifelogs. In *Proceedings of the 6th Annual ACM Lifelog Search Challenge*, LSC '23, page 7–12, New York, NY, USA, 2023. Association for Computing Machinery.

[115] Andreas Leibetseder, Daniela Stefanics, and Klaus Schoeffmann. lifeXplore at the Lifelog Search Challenge 2022. In *Proceedings of the 5th Annual on Lifelog Search Challenge*, LSC '22, page 48–52, New York, NY, USA, 2022. Association for Computing Machinery.

[116] Klaus Schoeffmann. lifeXplore at the Lifelog Search Challenge 2023. In *Proceedings of the 6th Annual ACM Lifelog Search Challenge*, LSC '23, page 53–58, New York, NY, USA, 2023. Association for Computing Machinery.

[117] Quang-Linh Tran, Ly-Duyen Tran, Binh Nguyen, and Cathal Gurrin. MemoriEase: An Interactive Lifelog Retrieval System for LSC'23. In

Proceedings of the 6th Annual ACM Lifelog Search Challenge, LSC '23, page 30–35, New York, NY, USA, 2023. Association for Computing Machinery.

[118] Ly-Duyen Tran, Manh-Duy Nguyen, Nguyen Thanh Binh, Hyowon Lee, and Cathal Gurrin. Myscéal 2.0: A Revised Experimental Interactive Lifelog Retrieval System for LSC'21. In *Proceedings of the 4th Annual on Lifelog Search Challenge*, LSC '21, page 11–16, New York, NY, USA, 2021. Association for Computing Machinery.

[119] Thao-Nhu Nguyen, Tu-Khiem Le, Van-Tu Ninh, Minh-Triet Tran, Thanh Binh Nguyen, Graham Healy, Sinéad Smyth, Annalina Caputo, and Cathal Gurrin. LifeSeeker 4.0: An Interactive Lifelog Search Engine for LSC'22. In *Proceedings of the 5th Annual on Lifelog Search Challenge*, LSC '22, page 14–19, New York, NY, USA, 2022. Association for Computing Machinery.

[120] Leland McInnes, John Healy, and S. Astels. hdbscan: Hierarchical density based clustering. *J. Open Source Softw.*, 2:205, 2017.

[121] Basel Kikhia, Andrey Boytsov, Josef Hallberg, Zaheer Sani, Håkan Jonsson, and Kåre Synnes. Structuring and Presenting Lifelogs Based on Location Data. volume 100, pages 133–144, 05 2014.

[122] Ly-Duyen Tran, Dongyun Nie, Liting Zhou, Binh Nguyen, and Cathal Gurrin. VAISL: Visual-Aware Identification Semantic Locations Lifelog. In *MultiMedia Modeling: 29th International Conference, MMM 2023, Bergen, Norway, January 9–12, 2023, Proceedings, Part II*, page 659–670, Berlin, Heidelberg, 2023. Springer-Verlag.

[123] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft COCO: Common Objects in Context, 2015.

[124] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio

Torralba. Places: A 10 million Image Database for Scene Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[125] Mingxing Tan and Quoc V. Le. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. *CoRR*, abs/1905.11946, 2019.

[126] Chien-Yao Wang, Alexey Bochkovskiy, and Hong-Yuan Mark Liao. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors, 2022.

[127] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *CoRR*, abs/1506.01497, 2015.

[128] Mingxing Tan, Ruoming Pang, and Quoc V. Le. EfficientDet: Scalable and Efficient Object Detection. *CoRR*, abs/1911.09070, 2019.

[129] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering, 2018.

[130] Chien-Yao Wang, Alexey Bochkovskiy, and Hong-Yuan Mark Liao. Scaled-YOLOv4: Scaling Cross Stage Partial Network, 2021.

[131] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[132] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual Captions: A Cleaned, Hypernymed, Image Alt-text Dataset For Automatic Image Captioning. pages 2556–2565, 01 2018.

[133] Youngmin Baek, Bado Lee, Dongyoon Han, Sangdoo Yun, and Hwalsuk Lee. Character Region Awareness for Text Detection. *CoRR*, abs/1904.01941, 2019.

[134] Alexander Theus, Luca Rossetto, and Abraham Bernstein. HyText - A Scene-Text Extraction Method for Video Retrieval. In *Conference on Multimedia Modeling*, 2022.

[135] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron C. Courville, Ruslan Salakhutdinov, Richard S. Zemel, and Yoshua Bengio. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. *CoRR*, abs/1502.03044, 2015.

[136] WordNet. `https://wordnet.princeton.edu/`. [accessed 28/11/2024].

[137] nltk library. `https://www.nltk.org/`. [accessed 28/11/2024].

[138] KAREN SPARCK JONES. A STATISTICAL INTERPRETATION OF TERM SPECIFICITY AND ITS APPLICATION IN RETRIEVAL. *Journal of Documentation*, 28(1):11–21, Jan 1972.

[139] Claude Sammut and Geoffrey I. Webb, editors. *TF–IDF*, pages 986–987. Springer US, Boston, MA, 2010.

[140] Jakub Lokoč, František Mejzlik, Patrik Veselý, and Tomáš Souček. Enhanced SOMHunter for Known-Item Search in Lifelog Data. In *Proceedings of the 4th Annual on Lifelog Search Challenge*, LSC '21, page 71–73, New York, NY, USA, 2021. Association for Computing Machinery.

[141] Ralph Gasser, Luca Rossetto, Silvan Heller, and Heiko Schuldt. Cottontail DB: An Open Source Database System for Multimedia Retrieval and Analysis. In *Proceedings of the 28th ACM International Conference on Multimedia*, MM '20, page 4465–4468, New York, NY, USA, 2020. Association for Computing Machinery.

[142] Luca Rossetto, Ivan Giangreco, and Heiko Schuldt. Cineast: A Multi-feature Sketch-Based Video Retrieval Engine. *2014 IEEE International Symposium on Multimedia*, pages 18–23, 2014.

[143] Anh-Vu Mai-Nguyen, Trong-Dat Phan, Anh-Khoa Vo, Van-Luon Tran, Minh-Son Dao, and Koji Zettsu. BIDAL-HCMUS@LSC2020: An Interactive Multimodal Lifelog Retrieval with Query-to-Sample Attention-based Search Engine. In *Proceedings of the Third Annual Workshop on Lifelog Search Challenge*, LSC '20, page 43–49, New York, NY, USA, 2020. Association for Computing Machinery.

[144] Minh-Triet Tran, Thanh-An Nguyen, Quoc-Cuong Tran, Mai-Khiem Tran, Khanh Nguyen, Van-Tu Ninh, Tu-Khiem Le, Hoang-Phuc Trang-Trung, Hoang-Anh Le, Hai-Dang Nguyen, Trong-Le Do, Viet-Khoa Vo-Ho, and Cathal Gurrin. FIRST - Flexible Interactive Retrieval SysTem for Visual Lifelog Exploration at LSC 2020. In *Proceedings of the Third Annual Workshop on Lifelog Search Challenge*, LSC '20, page 67–72, New York, NY, USA, 2020. Association for Computing Machinery.

[145] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning Transferable Visual Models From Natural Language Supervision. *CoRR*, abs/2103.00020, 2021.

[146] Norman Mu, Alexander Kirillov, David A. Wagner, and Saining Xie. SLIP: Self-supervision meets Language-Image Pre-training. *ArXiv*, abs/2112.12750, 2021.

[147] Evgeny Izutov. LIGAR: Lightweight General-purpose Action Recognition. *ArXiv*, abs/2108.13153, 2021.

[148] Ron Mokady, Amir Hertz, and Amit H. Bermano. ClipCap: CLIP Prefix for Image Captioning. *ArXiv*, abs/2111.09734, 2021.

[149] Naushad Alam, Yvette Graham, and Cathal Gurrin. Memento: A Prototype Lifelog Search Engine for LSC'21. In *Proceedings of the 4th Annual on Lifelog*

Search Challenge, LSC '21, page 53–58, New York, NY, USA, 2021. Association for Computing Machinery.

[150] Gabriel Ilharco, Mitchell Wortsman, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. Open Clip, 7 2021.

[151] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation, 2022.

[152] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, 2020.

[153] Model Card for CLIP ViT-H/14 - LAION-2B. `https://huggingface.co/laion/CLIP-ViT-H-14-laion2B-s32B-b79K`. [accessed 28/11/2024].

[154] Model Card for CLIP ViT-L/14 - LAION-5B. `https://huggingface.co/laion/CLIP-ViT-L-14-DataComp.XL-s13B-b90K`. [accessed 28/11/2024].

[155] Model Card for CLIP ViT-B/32 xlm roberta base - LAION-5B. `https://huggingface.co/laion/CLIP-ViT-B-32-xlm-roberta-base-laion5B-s13B-b90k`. [accessed 28/11/2024].

[156] CLIP. `https://github.com/openai/CLIP`. [accessed 28/11/2024].

[157] BLIP. `https://github.com/salesforce/BLIP`. [accessed 25/01/2024].

[158] Florian Spiess, Ralph Gasser, Silvan Heller, Mahnaz Parian-Scherb, Luca Rossetto, Loris Sauter, and Heiko Schuldt. Multi-modal Video Retrieval in Virtual Reality with vitrivr-VR. In *MultiMedia Modeling: 28th International*

Conference, MMM 2022, Phu Quoc, Vietnam, June 6–10, 2022, Proceedings, Part II, page 499–504, Berlin, Heidelberg, 2022. Springer-Verlag.

[159] Xirong Li, Chaoxi Xu, Gang Yang, Zhineng Chen, and Jianfeng Dong. W2VV++: Fully Deep Learning for Ad-hoc Video Search. In *Proceedings of the 27th ACM International Conference on Multimedia*, MM '19, page 1786–1794, New York, NY, USA, 2019. Association for Computing Machinery.

[160] Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with GPUs. *CoRR*, abs/1702.08734, 2017.

[161] Jianguo Wang, Xiaomeng Yi, Rentong Guo, Hai Jin, Peng Xu, Shengjun Li, Xiangyu Wang, Xiangzhou Guo, Chengming Li, Xiaohai Xu, Kun Yu, Yuxing Yuan, Yinghao Zou, Jiquan Long, Yudong Cai, Zhenxiang Li, Zhifeng Zhang, Yihua Mo, Jun Gu, Ruiyi Jiang, Yi Wei, and Charles Xie. Milvus: A Purpose-Built Vector Data Management System. In *Proceedings of the 2021 International Conference on Management of Data*, SIGMOD '21, page 2614–2627, New York, NY, USA, 2021. Association for Computing Machinery.

[162] J. J. Rocchio. Relevance feedback in information retrieval. 1971.

[163] Antoine Yang, Antoine Miech, Josef Sivic, Ivan Laptev, and Cordelia Schmid. Zero-Shot Video Question Answering via Frozen Bidirectional Language Models, 2022.

[164] Bernd Münzer, Andreas Leibetseder, Sabrina Kletz, Manfred Primus, and Klaus Schoeffmann. lifeXplore at the Lifelog Search Challenge 2018. pages 3–8, 06 2018.

[165] Jakub Lokoč, Tomáš Souček, and Gregor Kovalčik. Using an Interactive Video Retrieval Tool for LifeLog Data. In *Proceedings of the 2018 ACM Workshop on The Lifelog Search Challenge*, LSC '18, page 15–19, New York, NY, USA, 2018. Association for Computing Machinery.

[166] Giuseppe Amato, Paolo Bolettieri, Fabio Carrara, Franca Debole, Fabrizio Falchi, Claudio Gennaro, Lucia Vadicamo, and Claudio Vairo. VISIONE at VBS2019. In Ioannis Kompatsiaris, Benoit Huet, Vasileios Mezaris, Cathal Gurrin, Wen-Huang Cheng, and Stefanos Vrochidis, editors, *MultiMedia Modeling*, pages 591–596, Cham, 2019. Springer International Publishing.

[167] Aaron Duane, Cathal Gurrin, and Wolfgang Huerst. Virtual Reality Lifelog Explorer: Lifelog Search Challenge at ACM ICMR 2018. In *Proceedings of the 2018 ACM Workshop on The Lifelog Search Challenge*, LSC '18, page 20–23, New York, NY, USA, 2018. Association for Computing Machinery.

[168] Aaron Duane and Bjorn Þór Jónsson. ViRMA: Virtual Reality Multimedia Analytics at LSC 2021. In *Proceedings of the 4th Annual on Lifelog Search Challenge*, LSC '21, page 29–34, New York, NY, USA, 2021. Association for Computing Machinery.

[169] Aaron Duane and Björn Pór Jónsson. ViRMA: Virtual Reality Multimedia Analytics. In *Proceedings of the 2022 International Conference on Multimedia Retrieval*, ICMR '22, page 211–214, New York, NY, USA, 2022. Association for Computing Machinery.

[170] Jihye Shin, Alexandra Waldau, Aaron Duane, and Björn Þór Jónsson. PhotoCube at the Lifelog Search Challenge 2021. In *Proceedings of the 4th Annual on Lifelog Search Challenge*, LSC '21, page 59–63, New York, NY, USA, 2021. Association for Computing Machinery.

[171] Glen Shires. Voice Driven Web Apps: Introduction to the Web Speech API, 2013.

[172] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust Speech Recognition via Large-Scale Weak Supervision, 2022.

[173] Sally Sisi Qu, Mattia Soldan, Mengmeng Xu, Jesper Tegnér, and Bernard Ghanem. VLG-Net: Video-Language Graph Matching Network for Video Grounding. *CoRR*, abs/2011.10132, 2020.

[174] Zongmeng Zhang, Xianjing Han, Xuemeng Song, Yan Yan, and Liqiang Nie. Multi-Modal Interaction Graph Convolutional Network for Temporal Language Localization in Videos. *CoRR*, abs/2110.06058, 2021.

[175] Songyang Zhang, Houwen Peng, Jianlong Fu, and Jiebo Luo. Learning 2D Temporal Adjacent Networks for Moment Localization with Natural Language. *CoRR*, abs/1912.03590, 2019.

[176] Chujie Lu, Long Chen, Chilie Tan, Xiaolin Li, and Jun Xiao. DEBUG: A Dense Bottom-Up Grounding Approach for Natural Language Video Localization. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5144–5153, Hong Kong, China, November 2019. Association for Computational Linguistics.

[177] Jonghwan Mun, Minsu Cho, and Bohyung Han. Local-Global Video-Text Interactions for Temporal Grounding. *CoRR*, abs/2004.07514, 2020.

[178] Shen Yan, Xuehan Xiong, Arsha Nagrani, Anurag Arnab, Zhonghao Wang, Weina Ge, David Ross, and Cordelia Schmid. UnLoc: A Unified Framework for Video Localization Tasks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13623–13633, 2023.

[179] Yicheng Xiao, Zhuoyan Luo, Yong Liu, Yue Ma, Hengwei Bian, Yatai Ji, Yujiu Yang, and Xiu Li. Bridging the Gap: A Unified Video Comprehension Framework for Moment Retrieval and Highlight Detection, 2023.

[180] Chen-Lin Zhang, Jianxin Wu, and Yin Li. ActionFormer: Localizing Moments

of Actions with Transformers. In *European Conference on Computer Vision*, volume 13664 of *LNCS*, pages 492–510, 2022.

[181] Songyang Zhang, Jinsong Su, and Jiebo Luo. Exploiting Temporal Relationships in Video Moment Localization with Natural Language. *CoRR*, abs/1908.03846, 2019.

[182] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-End Object Detection with Transformers. *CoRR*, abs/2005.12872, 2020.

[183] João Carreira and Andrew Zisserman. Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset. *CoRR*, abs/1705.07750, 2017.

[184] Daniel P.W. Ellis and Keansub Lee. Minimal-Impact Audio-Based Personal Archives. In *Proceedings of the the 1st ACM Workshop on Continuous Archival and Retrieval of Personal Experiences*, CARPE'04, page 39–47, New York, NY, USA, 2004. Association for Computing Machinery.

[185] Estefania Talavera, Mariella Dimiccoli, Marc Bolaños, Maedeh Aghaei, and Petia Radeva. R-Clustering for Egocentric Video Segmentation, 2017.

[186] Mariella Dimiccoli, Marc Bolaños, Estefania Talavera, Maedeh Aghaei, Stavri G. Nikolov, and Petia Radeva. SR-clustering: Semantic regularized clustering for egocentric photo streams segmentation. *Computer Vision and Image Understanding*, 155:55–69, 2017.

[187] M.M. Yeung and Boon-Lock Yeo. Time-constrained clustering for segmentation of video into story units. In *Proceedings of 13th International Conference on Pattern Recognition*, volume 3, pages 375–380 vol.3, 1996.

[188] Wei-Hao Lin and Alexander Hauptmann. Structuring continuous video recordings of everyday life using time-constrained clustering. In *Electronic Imaging*, 2006.

[189] Rashmi Gupta. Considering Documents in Lifelog Information Retrieval. In *Proceedings of the 2018 ACM on International Conference on Multimedia Retrieval*, ICMR '18, page 497–500, New York, NY, USA, 2018. Association for Computing Machinery.

[190] Rashmi Gupta and Cathal Gurrin. *Approaches for Event Segmentation of Visual Lifelog Data*, pages 581–593. 01 2018.

[191] Junghyun Bum, Joyce Whang, and Hyunseung Choo. Sentiment-based sub-event segmentation and key photo selection. *Journal of Visual Communication and Image Representation*, 74:102973, 01 2021.

[192] Alan F. Smeaton and Paul Browne. A usage study of retrieval modalities for video shot retrieval. *Information Processing and Management*, 42(5):1330–1344, 2006.

[193] Photchara Ratsamee, Yasushi Mae, Amornched Jinda-apiraksa, Mitsuhiro Horade, Kazuto Kamiyama, Masaru Kojima, and Takero Arai. Keyframe Selection Framework Based on Visual and Excitement Features for Lifelog Image Sequences. *International Journal of Social Robotics*, 7, 07 2015.

[194] John Creswell and Timothy Guetterman. *Educational Research: Planning, Conducting, and Evaluating Quantitative and Qualitative Research, 6th Edition*. 02 2018.

[195] Silvan Heller, Viktor Gsteiger, Werner Bailer, Cathal Gurrin, Björn Jónsson, Jakub Lokoč, Andreas Leibetseder, František Mejzlík, Ladislav Peska, Luca Rossetto, Konstantin Schall, Klaus Schoeffmann, Heiko Schuldt, Florian Spiess, Duyen Tran, Lucia Vadicamo, Patrik Veselý, Stefanos Vrochidis, and Jiaxin Wu. Interactive video retrieval evaluation at a distance: comparing sixteen interactive video search systems in a remote setting at the 10th Video Browser Showdown. *International Journal of Multimedia Information Retrieval*, 11, 03 2022.

[196] Tu-Khiem Le, Van-Tu Ninh, Duc-Tien Dang-Nguyen, Minh-Triet Tran, Liting Zhou, Pablo Redondo, Sinead Smyth, and Cathal Gurrin. LifeSeeker: Interactive Lifelog Search Engine at LSC 2019. In *Proceedings of the ACM Workshop on Lifelog Search Challenge*, LSC '19, page 37–40, New York, NY, USA, 2019. Association for Computing Machinery.

[197] Tu-Khiem Le, Van-Tu Ninh, Minh-Triet Tran, Thanh-An Nguyen, Hai-Dang Nguyen, Liting Zhou, Graham Healy, and Cathal Gurrin. LifeSeeker 2.0: Interactive Lifelog Search Engine at LSC 2020. In *Proceedings of the Third Annual Workshop on Lifelog Search Challenge*, LSC '20, page 57–62, New York, NY, USA, 2020. Association for Computing Machinery.

[198] Thao-Nhu Nguyen, Tu-Khiem Le, Van-Tu Ninh, Ly-Duyen Tran, Manh-Duy Nguyen, Minh-Triet Tran, Binh T Nguyen, Annalina Caputo, Cathal Gurrin, Graham Healy, et al. DCU and HCMUS at NTCIR-16 Lifelog-4. In *Proceedings of the 16th NTCIR Conference on Evaluation of Information Access Technologies*. NTCIR, 2022.

[199] Bharat Singh, Mahyar Najibi, and Larry S. Davis. SNIPER: Efficient Multi-Scale Training, 2018.

[200] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Fei-Fei Li. Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations, 2016.

[201] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade W Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa R Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. LAION-5b: An open large-scale dataset for training next generation image-text models. In

Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track, 2022.

[202] Angel X. Chang and Christopher D. Manning. SUTime: A library for recognizing and normalizing time expressions. In *LREC*, 2012.

[203] Liting Zhou, Cathal Gurrin, Graham Healy, Hideo Joho, Binh T. Nguyen, Rami Albatal, Perfogram Ltd, Ireland Frank, Hopfgartner, and Duc-Tien Dang-Nguyen. Overview of the NTCIR-16 Lifelog-4 Task.

[204] DRES. `https://github.com/dres-dev/DRES`. [accessed 25/11/2024].

[205] Luca Rossetto, Ralph Gasser, Loris Sauter, Abraham Bernstein, and Heiko Schuldt. A System for Interactive Multimedia Retrieval Evaluations. In Jakub Lokoč, Tomáš Skopal, Klaus Schoeffmann, Vasileios Mezaris, Xirong Li, Stefanos Vrochidis, and Ioannis Patras, editors, *MultiMedia Modeling*, pages 385–390, Cham, 2021. Springer International Publishing.

[206] Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. Neural Motifs: Scene Graph Parsing with Global Context. *CoRR*, abs/1711.06640, 2017.

[207] Jingwei Ji, Ranjay Krishna, Li Fei-Fei, and Juan Carlos Niebles. Action Genome: Actions as Composition of Spatio-temporal Scene Graphs. *CoRR*, abs/1912.06992, 2019.

[208] P.J. Werbos. Backpropagation through time: what it does and how to do it. *Proceedings of the IEEE*, 78(10):1550–1560, 1990.

[209] Sepp Hochreiter and Jürgen Schmidhuber. Long Short-term Memory. *Neural computation*, 9:1735–80, 12 1997.

[210] Qiang Wang, Bei Li, Tong Xiao, Jingbo Zhu, Changliang Li, Derek F. Wong, and Lidia S. Chao. Learning Deep Transformer Models for Machine Translation, 2019.

[211] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, 2019.

[212] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter, 2020.

[213] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. RoBERTa: A Robustly Optimized BERT Pretraining Approach, 2019.

[214] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language Models are Unsupervised Multitask Learners. 2019.

[215] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeff Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Ma teusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language Models are Few-Shot Learners. *ArXiv*, abs/2005.14165, 2020.

[216] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows, 2021.

[217] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is Space-Time Attention All You Need for Video Understanding?, 2021.

[218] Huaishao Luo, Lei Ji, Ming Zhong, Yang Chen, Wen Lei, Nan Duan, and Tianrui Li. CLIP4Clip: An Empirical Study of CLIP for End to End Video Clip Retrieval, 2021.

[219] Xi Wei, Tianzhu Zhang, Yan Li, Yongdong Zhang, and Feng Wu. Multi-modality cross attention network for image and sentence matching. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10941–10950, 2020.

[220] Abien Fred Agarap. Deep learning using rectified linear units (relu), 2019.

[221] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer Normalization, 2016.

[222] Tsung-Yi Lin, Piotr Dollár, Ross B. Girshick, Kaiming He, Bharath Hariharan, and Serge J. Belongie. Feature Pyramid Networks for Object Detection. *CoRR*, abs/1612.03144, 2016.

[223] Navaneeth Bodla, Bharat Singh, Rama Chellappa, and Larry S. Davis. Soft-NMS – Improving Object Detection With One Line of Code. 2017.

[224] Jan Hosang, Rodrigo Benenson, and Bernt Schiele. Learning non-maximum suppression, 2017.

[225] Gunnar A. Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. Hollywood in Homes: Crowdsourcing Data Collection for Activity Understanding. *CoRR*, abs/1604.01753, 2016.

[226] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Z. Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An Imperative Style, High-Performance Deep Learning Library. *CoRR*, abs/1912.01703, 2019.

[227] Ilya Loshchilov and Frank Hutter. Fixing Weight Decay Regularization in Adam. *CoRR*, abs/1711.05101, 2017.

[228] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017.

[229] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. A CLIP-Hitchhiker's Guide to Long Video Retrieval, 2022.

[230] Jesús Andrés Portillo-Quintero, José Carlos Ortiz-Bayliss, and Hugo Terashima-Marín. A Straightforward Framework For Video Retrieval Using CLIP. *CoRR*, abs/2102.12443, 2021.

[231] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. Multi-Stage Progressive Image Restoration, 2021.

[232] Yu-Lun Liu, Wei-Sheng Lai, Ming-Hsuan Yang, Yung-Yu Chuang, and Jia-Bin Huang. Hybrid Neural Fusion for Full-frame Video Stabilization, 2021.

[233] Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, Yejin Choi, and Jianfeng Gao. Oscar: Object-Semantics Aligned Pre-training for Vision-Language Tasks, 2020.