Free-form Multi-Modal Multimedia Retrieval (4MR)

Rahel Arnold[0000−0002−5881−4432], Loris Sauter[0000−0001−8046−0362], and Heiko Schuldt[0000−0001−9865−6371]

Databases and Information Systems Research Group
University of Basel, Basel, Switzerland
{firstname}.{lastname}@unibas.ch

Abstract. Due to the ever increasing amount of multimedia data, efficient means for multimedia management and retrieval are required. Especially with the rise of deep-learning-based analytics methods, the semantic gap has shrunk considerably, but a human in the loop is still considered mandatory. One of the driving factors of video search is that humans tend to refine their queries after reviewing the results. Hence, the entire process is highly interactive. A natural approach to interactive video search is using textual descriptions of the content of the expected result, enabled by deep learning-based joint visual text co-embedding. In this paper, we present the Multi-Modal Multimedia Retrieval (4MR) system, a novel system inspired by vitrivr, that empowers users with almost entirely free-form query formulation methods. The top-ranked teams of the last few iterations of the Video Browser Showdown have shown that CLIP provides an ideal feature extraction method. Therefore, while 4MR is capable of image and text retrieval as well, for VBS video retrieval is based primarily based on CLIP.

Keywords: Video Browser Showdown · Interactive Video Retrieval · Content-based Retrieval

1 Introduction

With the ever-growing volume of multimedia data, means for efficient and effective search in such multimedia collections are a necessity. At the annual Video Browser Showdown (VBS) [15] – a competition-style evaluation campaign in the domain of interactive video search – interactive multimedia retrieval systems compete against each other in a pseudo-realistic setting. Particularly, Known-Item Search (KIS) and Ad-hoc Video Search (AVS) are task categories of VBS [8]. The former category consists of two sub-categories, each with a single target video shot of about 20 seconds with textual and visual hints, respectively. The latter features broader query terms without known ground truth, and human judges decide whether a shot meets the task criteria or not. VBS operates on the Vimeo Creative Commons Collection (V3C) [14], in particular V3C's first and second shards [1,13], culminating in approximately 2300 h video content. In

addition, a new, more homogeneous "marine video" dataset [17] of more than 11 h complements previous video data for more challenging tasks.

Inspired by vitrivr [3,6], a long-running participant at VBS, we present in this paper 4MR (*M*ulti-*M*odal *M*ulti*M*edia *R*etrieval), a novel open-source multi-modal multimedia retrieval system with a focus on retrieval blocks. In particular, the 4MR system empowers users to express multi-modal queries and to freely combine these with Boolean logic. Like the top-ranked teams from previous iterations [7] that have implemented CLIP [10], we also rely on CLIP as the primary feature extraction method. Furthermore, we support deep learning-based OCR and ASR methods and extracted concept labels with ResNet50 [5].

The remainder of this paper is structured as follows: Section 2 introduces the concepts of retrieval blocks and how to formulate queries, Section 3 gives insights on the implementation, and Section 4 concludes.

2 Retrieval Blocks

In order to efficiently and effectively search in video collections, three search paradigms have proven to be successful [7]: (i) (extended) Boolean search, (ii) vector-based text search, and (iii) kNN search. We employ all three paradigms and build multi-modal queries on them: Boolean search in order to efficiently use provided metadata such as collection affiliation. We plan to use this feature to specify V3C shards one and two or the Marine Video Kit. However, vector search is used to search in deep learning-based extracted feature data such as CLIP representations. In addition, textual vector search is applied to search in OCR data, for example.

At its core, our multi-modal query model consists of so-called *Query Statements*. A Query Statement is the smallest unit in our model and might be one of the following: (i) textual search, either Boolean or kNN search, (ii) metadata search, using Boolean search; or (iii) visual search, using kNN search.

Two Query Statements can be linked with a so-called *Statement Linking Operator*, either the logical AND or OR. One or two Query Statements form a *Query Block* with or without a Statement Linking Operator. Query Blocks can subsequently be arbitrarily nested, and their relationship to each other is also defined by Statement Linking Operators. Ultimately, one or more Query Blocks are called *Retrieval Statement*, put into relation with the so-called *Retrieval Linking Operator*. Retrieval Linking Operators define an order so that the prerequisites of one Retrieval Statement must be met first before the next one can be applied, effectively creating stages. Another such Retrieval Linking Operator could represent some temporal relation to the Retrieval Statements.

This nesting of Query Blocks and their defined relation to one another is sufficient to formulate arbitrarily complex queries as needed for interactive search. While the model would allow for interactive search within text-based and inherently multi-media collections, this functionality cannot be used in VBS as the competition is video search only. However, we intend to use the Query Blocks in order to express complex queries.

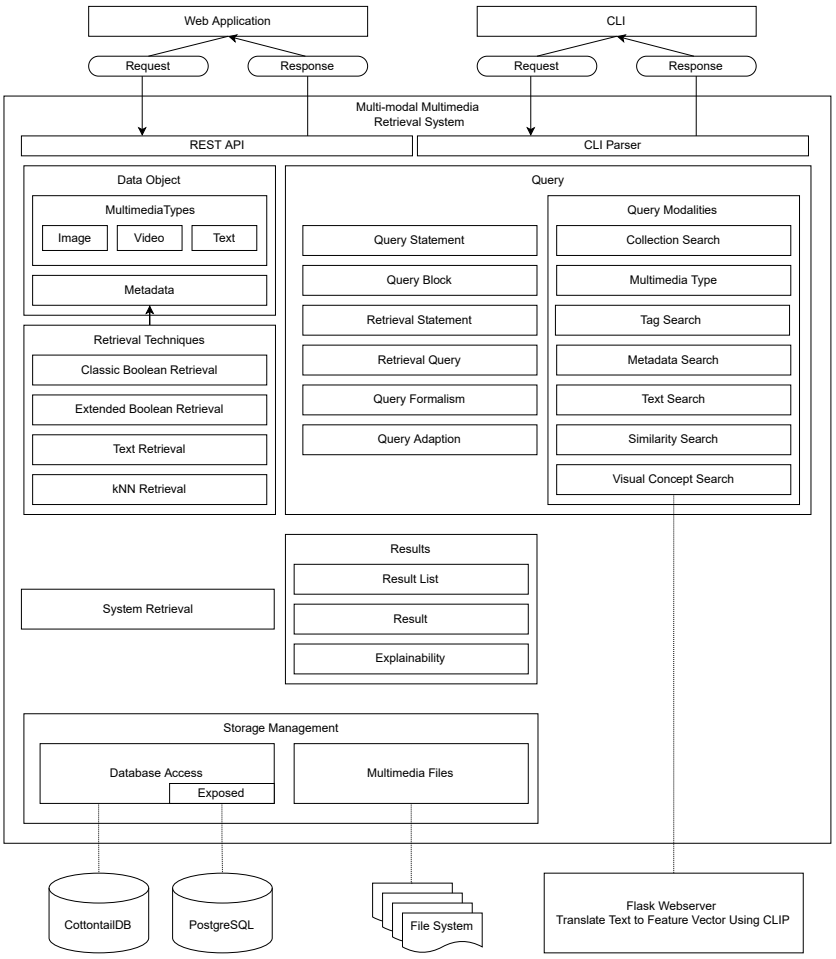

Fig. 1. Architecture diagram of the 4MR system.

3 Implementation

The 4MR system follows a three-tier architecture [16], as shown in Figure 1. The storage layer consists of a Postgres database [1] (textual and Boolean search) for textual information, a CottontailDB database [2] for vectors (and kNN search) and the file system to store the actual media files. Multiple storage systems are used to exploit their strengths: Postgres for efficient Boolean search and CottontailDB for kNN search. In what follows, we describe the retrieval engine (Section 3.1), explain which features we search for (Section 3.2) and finally introduce the user interface (Section 3.3).

[1] https://www.postgresql.org

3.1 Retrieval Engine

Written in Kotlin, the retrieval engine communicates with the storage tier using a JDBC adapter and a gRPC client for Postgres and CottontailDB, respectively. The entire parsing of queries, as well as relaying the appropriate parts to the underlying storage systems in correspondence to the query type, is handled in the retrieval engine. Ultimately, the results of the storage components are fused into a single ranked result list and sent to the user interface. All functionality of the retrieval engine is made accessible to the front end by a REST API.

3.2 Video Analysis

Due to the success of deep learning-based video analysis methods, or feature extraction, our system employs four such models. In particular, we use Contrastive Language–Image Pre-training, better known as CLIP, based on recognising image elements using our natural language to describe them [9,10,11]. It builds on a large body of work on zero-shot transfer, natural language supervision, and multi-modal learning. We use CLIP, or more precisely the "ViT-B/32"-model from Open-AI, to provide visual concept and similarity search. The visual concept search encodes the text input with the model. The resulting vector is then used for a kNN search, where the best fitting images with small distances are returned. In the concept of similarity search, kNN search is directly performed on the 512-feature-long vectors. Besides CLIP, a video's textual and audio content is often interesting for queries. We support optical character recognition (OCR) queries in these domains and automatic speech recognition (ASR), respectively. The features used for these features are the same as presented by our inspiration, vitrivr [12] and rely on text search entirely.

Last but not least, we support the notion of concepts recognised within videos, so-called tags. Using a residual network with 50 layers, ResNet50 [4], we retain tags and the confidence that the neural net classified them, which we can use in the query formulation.

3.3 User Interface

Heavily inspired by vitrivr's frontend vitrivr-ng[2], we also have an Angular[3] based frontend divided into a query sidebar and central result presentation area (see Figure 2).

4 Conclusion

We introduce 4MR, a new system focusing on query formulation based on vitrivr, to participate in the Video Browser Showdown 2023. The contribution is twofold: On one hand, we describe a query formulation concept which empowers users

[2] https://github.com/vitrivr/vitrivr-ng
[3] https://angular.io

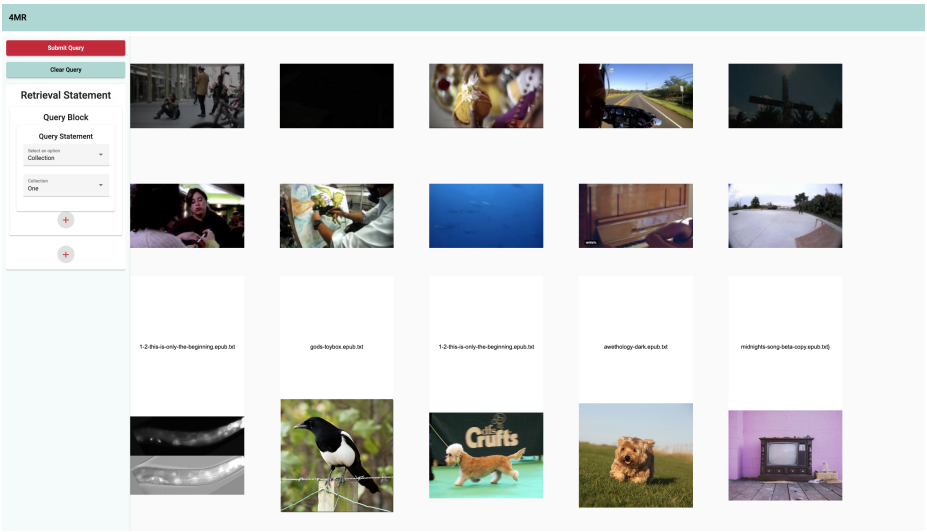

Fig. 2. A screenshot of the 4MR system in action. The left features the query formulation area and the center is used for result presentation.

to combine query blocks and freely define their relationship. On the other hand, we provide an implementation of the concept in order to be able to evaluate our system in the competitive setting of VBS. Our query formulation methodology uses deep learning-based features, particularly CLIP, like the top-ranked teams in the last instances of VBS.

References

1. Berns, F., Rossetto, L., Schoeffmann, K., Beecks, C., Awad, G.: V3C1 Dataset: An Evaluation of Content Characteristics. In: International Conference on Multimedia Retrieval. ACM (2019). https://doi.org/10.1145/3323873.3325051
2. Gasser, R., Rossetto, L., Heller, S., Schuldt, H.: Cottontail DB: An Open Source Database System for Multimedia Retrieval and Analysis, p. 4465–4468. Association for Computing Machinery, New York, NY, USA (2020), https://doi.org/10.1145/3394171.3414538
3. Gasser, R., Rossetto, L., Schuldt, H.: Multimodal Multimedia Retrieval with vitrivr. In: International Conference on Multimedia Retrieval (2019)
4. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition (2015). https://doi.org/10.48550/ARXIV.1512.03385, https://arxiv.org/abs/1512.03385
5. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016. pp. 770–778. IEEE Computer Society (2016). https://doi.org/10.1109/CVPR.2016.90, https://doi.org/10.1109/CVPR.2016.90

6. Heller, S., Arnold, R., Gasser, R., Gsteiger, V., Parian-Scherb, M., Rossetto, L., Sauter, L., Spiess, F., Schuldt, H.: Multi-modal Interactive Video Retrieval with Temporal Queries. In: MultiMedia Modeling. Springer (2022). https://doi.org/10.1007/978-3-030-98355-0_44
7. Heller, S., Gasser, R., Illi, C., Pasquinelli, M., Sauter, L., Spiess, F., Schuldt, H.: Towards Explainable Interactive Multi-modal Video Retrieval with Vitrivr. In: MultiMedia Modeling. Springer (2021). https://doi.org/10.1007/978-3-030-67835-7_41
8. Lokoč, J., Bailer, W., Barthel, K.U., Gurrin, C., Heller, S., þór Jónsson, B., Peška, L., Rossetto, L., Schoeffmann, K., Vadicamo, L., Vrochidis, S., Wu, J.: A Task Category Space for User-Centric Comparative Multimedia Search Evaluations. In: MultiMedia Modeling (2022). https://doi.org/10.1007/978-3-030-98358-1_16
9. OpenAI: Github repository clip. https://github.com/openai/CLIP, accessed 2022-10-10
10. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., Sutskever, I.: Learning transferable visual models from natural language supervision (2021). https://doi.org/10.48550/ARXIV.2103.00020, https://arxiv.org/abs/2103.00020
11. Radford, A., Sutskever, I., Kim, J.W., Krueger, G., Agarwal, S.: Clip: Connecting text and images. https://openai.com/blog/clip/, accessed 2022-10-10
12. Rossetto, L., Amiri Parian, M., Gasser, R., Giangreco, I., Heller, S., Schuldt, H.: Deep learning-based concept detection in vitrivr. In: Kompatsiaris, I., Huet, B., Mezaris, V., Gurrin, C., Cheng, W.H., Vrochidis, S. (eds.) MultiMedia Modeling. pp. 616–621. Springer International Publishing, Cham (2019)
13. Rossetto, L., Schoeffmann, K., Bernstein, A.: Insights on the V3C2 dataset. CoRR **abs/2105.01475** (2021), https://arxiv.org/abs/2105.01475
14. Rossetto, L., Schuldt, H., Awad, G., Butt, A.A.: V3C - A research video collection. In: MultiMedia Modeling. Springer (2019). https://doi.org/10.1007/978-3-030-05710-7_29
15. Schoeffmann, K.: Video Browser Showdown 2012-2019: A Review. In: International Conference on Content-Based Multimedia Indexing (2019). https://doi.org/10.1109/CBMI.2019.8877397
16. Schuldt, H.: Multitier Architecture, pp. 2443–2446. Springer New York, New York, NY (2018). https://doi.org/10.1007/978-1-4614-8265-9_652, https://doi.org/10.1007/978-1-4614-8265-9_652
17. Truong, Q.T., Vu, T.A., Ha, T.S., Lokoc, J., Tim, Y.H.W., Joneja, A., Yeung, S.K.: Marine video kit: A new marine video dataset for content-based analysis and retrieval. In: MultiMedia Modeling - 29th International Conference, MMM 2023, Bergen, Norway, January 9-12, 2023. Lecture Notes in Computer Science, Springer (2023)