A new Retrieval Engine for vitrivr

Ralph Gasser[1][0000−0002−3016−1396], Rahel Arnold[1][0000−0002−5881−4432], Fynn Faber[1][0009−0004−3486−0209], Heiko Schuldt[1][0000−0001−9865−6371], Raphael Waltenspül[1][0009−0004−9622−7265], and Luca Rossetto[2][0000−0002−5389−9465]

[1] University of Basel, Basel, Switzerland
{firstname}.{lastname}@unibas.ch
[2] University of Zurich, Zurich, Switzerland
rossetto@ifi.uzh.ch

Abstract. While the vitrivr stack has seen many changes in components over the years, its feature extraction and query processing engine traces its history back almost a decade. Some aspects of its architecture and operation are no longer current, limiting the entire stack's applicability in various use cases. In this paper, we present the first glimpse into vitrivr's next-generation retrieval engine and our plan to overcome previously identified limitations.

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

1 Introduction

The content-based multimedia retrieval stack vitrivr [18] has been a long-time participant in the Video Browser Showdown (VBS). Including its predecessor, the iMotion System [17], from which vitrivr ultimately emerged, the system participates for the 10^{th} year in a row. Architecturally, the vitrivr stack consists of three primary components: a database, a retrieval engine, and a user interface. While both the database and the user interface have been replaced several times over the years, the retrieval engine *Cineast* traces its origins back almost a decade [16]. Despite much work and many contributions made to Cineast over the years, many insights gained through past activities and technological changes cannot easily be retrofitted to its underlying architecture, which is why an overhaul of the retrieval engine has been long overdue.

In this light, we, therefore, use our 10^{th} VBS participation as an opportunity to take the first steps towards Cineast's successor, a system we simply call the *vitrivr-engine*. The remainder of this paper is structured as follows: Section 2 starts with a brief history of the vitrivr stack and its components and capabilities, especially in the context of VBS. It then continues by looking forward and discussing some of our future plans with the new vitrivr-engine. Section 3 then goes into more detail about the exact state of the system that will participate in the VBS in 2024. Finally, Section 4 offers some concluding remarks.

2 vitrivr

This section first provides a brief history of the development of the vitrivr stack before outlining the goals we are trying to achieve with the vitrivr-engine.

2.1 The "vitrivr-Story" Thus Far

The vitrivr stack grew out of the iMotion system, which participated in the VBS in 2015 [17] and 2016 [15]. The iMotion system used a database system called *ADAM* [4], which was an extension of the PostgreSQL [28] database system, augmented with the ability to perform vector space operations. The retrieval engine was Cineast, which used a tightly coupled browser-based user interface.

The *ADAM* database was replaced with its successor, $ADAM_{pro}$ [5], for the 2017 participation [19], which would be the last under the iMotion name and the first time the system scored highest among all participants [9]. In contrast to the purely relational *ADAM* database, $ADAM_{pro}$ took a polyglot approach to accommodate the different types of queries needed for multimedia retrieval.

2018 saw a substantial extension in the capabilities of Cineast, turning it from a video-only retrieval engine into one capable of processing several media types, including images, audio, and 3D models [3]. That year's VBS participation [14] happened under the name *vitrivr* for the first time, for which we also introduced a new user interface called vitrivr-ng. With the official transition to the new name, we made a consistent effort to open-source the entire stack. 2019 [12] saw various improvements throughout the system but no fundamental changes. Nevertheless, the system outperformed all other VBS participants [13].

In 2020 [23], we introduced the 3rd database system to be used in the stack: Cottontail DB [2]. This was the last more significant architectural change to the vitrivr stack since then. Even so, in 2021 [7], vitrivr was the highest-scoring challenge participant for the third time [8]. That year also saw the first participation of a new user interaction approach, using virtual reality rather than a traditional desktop interface. vitrivr-VR [26] uses the same database and retrieval engine backend but introduces a completely new user interface with much more effective browsing capabilities. In 2022 [6,25] and 2023 [22,27], both vitrivr and vitrivr-VR participated without any fundamental architectural changes.

Throughout all this time, the only system component that, despite benefiting from various improvements, stayed the same was the feature extraction and query processing engine Cineast. While it served us well over the years in a broad range of applications, the implicit and explicit assumptions baked into its architecture are increasingly at odds with today's requirements. Especially in recent years, we have encountered more and more applications, e.g., in the context of the projects XReco[3] and Archipanion,[4] that could benefit from vitrivr's multimedia retrieval capabilities but that were not easily integrated due to Cineast's rather monolithic setup. The following section, therefore, provides a brief overview of some of our plans and gives a glimpse into Cineast's successor, the vitrivr-engine.

[3] See https://xreco.eu/
[4] See https://dbis.dmi.unibas.ch/research/projects/archipanion/

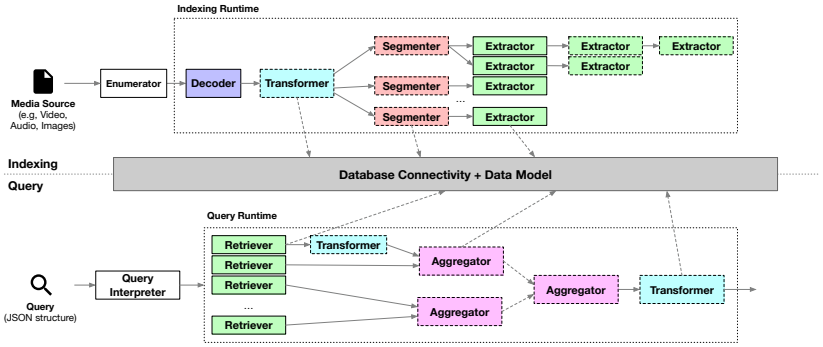

Fig. 1. A high-level depiction of vitrivr-engine's internals. The runtime environments for indexing and querying are strictly separated, and both are modeled as a pipeline of operators. The two modules share a common database connectivity and data model.

2.2 Where To Go From Here

With the lessons learned over the past years, we have identified several weaknesses in Cineast that we plan to address in the new vitrivr-engine to prepare vitrivr as a whole for a future in a wider range of different setups. An overview of vitrivr-engine's high-level architecture is provided in Figure 1.

We plan to make vitrivr-engine more modular at different levels. At the highest level, there will be separate modules for facilities used for executing queries – the *query runtime* – and those used for indexing media collections – the *indexing runtime*. Both facilities will share a common core that defines the basic processing and data model. Within these facilities, the indexing and retrieval processes are explicitly modeled as configurable pipelines that consist of operators generating, processing, and transforming a stream of common entities. A plug-able database abstraction layer will back all this.

Data Model Cineast's original data model is very strict in how information units are structured and connected. Firstly, it expects the main *media object* entity to always refer to individual files (with associated metadata) that are then temporally divided into *segments*, from which all features are derived. This is a remnant of Cineast's roots as a video retrieval engine. Consequently, some use cases could not easily be covered, e.g., having multiple independent or non-temporal segmentation schemes. Secondly, more complex relationships between parts of the media items could not be modeled.

The vitrivr-engine data model is more flexible in that regard. Fundamentally, it processes a stream of *content* from some source (e.g., a video file), from which one (or many) *retrievable(s)* are derived, e.g., through segmentation. The segmentation process can be temporal, spatial, or anything that can be applied to a particular type of content stream. In addition, vitrivr-engine allows for the modeling of explicit relationships between *retrievable*s, which leads to a graph-like data structure with much more descriptive power than a static data model.

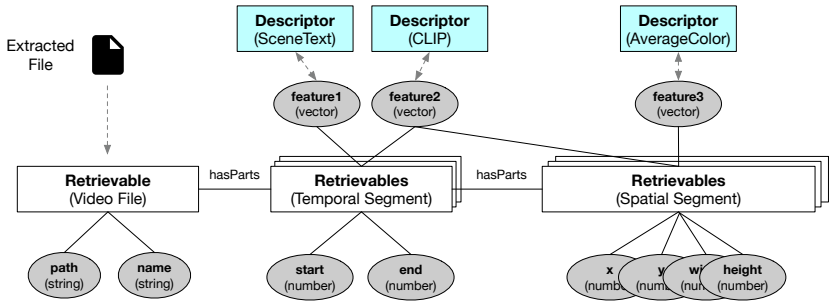

Fig. 2. An indexed video file is represented as three types of retrievables: One retrievable for the file itself, one for the temporal segments thereof, and one for the spatial segments generated for every temporal one. Each level of retrievable can hold its own fields with various types of attributes. A descriptor explicitly backs most fields.

Fields can then describe any *retrievable*, which can hold anything from a feature vector to a scalar value. This Entity-Attribute-Value (EAV) approach to describing *retrievables* gives us more flexibility in how instances of vitrivr can be tailored to a particular use case (with their own data model). Most fields are backed by some type of *descriptor*, an algorithm that extracts the *feature* from the *retrievable*. An example of the new data model's expressive power is provided in Figure 2.

Query Model Cineast's query model has always been tied to the functionality provided by the vitrivr-ng and (later) the vitrivr-VR user interfaces. The query model was extended whenever a new feature emerged, leading to a complex data structure.

In vitrivr-engine, we model queries as a pipeline of operators, as is illustrated in Figure 1. vitrivr-engine's query model simply describes the shape and elements of such a pipeline in a declarative fashion, leading to a richer and more expressive query model. We see this as a necessary step to generalizing mechanisms for query formulation and execution and to decoupling them from particular implementations offered by some user interfaces.

Extensibility and Modularity Cineast has always been extensible in that new feature descriptors (and other types of functionality) could be added by implementing a set of defined interfaces. However, such extensions were constantly added to the core codebase, leading to many feature implementations that are rarely ever used.

In vitrivr-engine, we have conceived a plugin architecture that facilitates adding functionality using a set of *extension points*. In addition to feature descriptors, these extension points involve all elements along the indexing and query pipeline, particularly all the operators depicted in Figure 1, down to the database connectivity layer itself.

All the relevant feature descriptors existing in Cineast today will be bundled in a *plugin* that can be used to bootstrap a new installation. Consequently, the new vitrivr-engine can still be used as an off-the-shelf solution by using the default plugins if one wishes to do so. However, using the vitrivr-engine core facilities is also possible exclusively with custom implementations of decoders, extractors, and feature descriptors, leading to a leaner project better tailored to a particular application.

Decoupling the Engine from the Interface Various use cases require different public programming interfaces that facilitate interaction with the extraction and retrieval engine. That is why Cineast featured three types of interfaces – gRPC, HTTP/REST, and WebSocket – that have seen major and breaking changes over the years to accommodate new use cases. In Cineast, the engine facilitating the execution of all these processes was strongly intertwined with these different interfaces.

In vitrivr-engine, we will decouple the processing facilities and the public interfaces completely. While vitrivr-engine will come with a default server that allows access via an OpenAPI REST interface, it will also be possible to use vitrivr-engine's processing components merely as libraries, without the overhead that comes with the default API infrastructure. This is especially useful in projects such as XReco, which rely on vitrivr's retrieval capabilities but come with their own design paradigm for public APIs.

3 Implementation

The vitrivr setup used at VBS 2024 will be similar to the one used in previous installments [6,7,22]. However, instead of using Cineast as the feature extraction and query engine, we will rely on the first version of the new vitrivr-engine, written in Kotlin. This middleware will be backed by the multimedia DBMS Cottontail DB [2] and will use a new, minimalistic UI that debuted at CBMI 2023 [24]. That UI is focused on a simple, efficient user interaction rather than a rich set of functionality.

Due to some architectural changes in vitrivr-engine, we will be able to serve all three independent collections[5] used during VBS within a common deployment of vitrivr-engine. For this year's installment of VBS and based on the experiences gathered in previous years, we restrict ourselves to a minimal set of features, which we know to be effective in this highly competitive setting: We use OpenClip [1] for text-based querying and DinoV2 [10] for similarity search queries (more-like-this). As these two features rely on neural networks, they will be hosted in an external feature server written in Python and are accessed via a RESTful API. In addition, we will index the V3C speech transcripts [20] generated using whisper [11] as well as OCR data already used in previous years.

[5] V3C (shards 1 and 2) [21], the Marine Video Kit [29], and LapGyn100 laparoscopic gynaecology video dataset.

4 Conclusion

In this paper, we have provided an overview of vitrivr's history, focusing on its retrieval engine, Cineast. While Cineast has served us well over the years, some of its architectural decisions, made about a decade ago, have become limiting. We therefore offer a first glimpse into its successor, the vitrivr-engine, and outline some of our plans towards vitrivr's future.

Acknowledgements

This work was partly supported by the Swiss National Science Foundation through project MediaGraph (contract no. 202125), the InnoSuisse project Archipanion (contract no. 2155012542), and the Horizon Europe project XR mEdia eCOsystem (XReco), based on a grant from the Swiss State Secretariat for Education, Research and Innovation (contract no. 22.00268).

References

1. Cherti, M., Beaumont, R., Wightman, R., Wortsman, M., Ilharco, G., Gordon, C., Schuhmann, C., Schmidt, L., Jitsev, J.: Reproducible Scaling Laws for Contrastive Language-Image Learning. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023. pp. 2818–2829. IEEE (2023)
2. Gasser, R., Rossetto, L., Heller, S., Schuldt, H.: Cottontail DB: An Open Source Database System for Multimedia Retrieval and Analysis. In: MM'20: The 28th ACM International Conference on Multimedia, Virtual Event / Seattle, WA, USA, October 12-16, 2020. pp. 4465–4468. ACM (2020)
3. Gasser, R., Rossetto, L., Schuldt, H.: Multimodal Multimedia Retrieval with vitrivr. In: Proceedings of the 2019 on International Conference on Multimedia Retrieval, ICMR 2019, Ottawa, ON, Canada, June 10-13, 2019. pp. 391–394. ACM (2019)
4. Giangreco, I., Al Kabary, I., Schuldt, H.: ADAM - A Database and Information Retrieval System for Big Multimedia Collections. In: 2014 IEEE International Congress on Big Data, Anchorage, AK, USA, June 27 - July 2, 2014. pp. 406–413. IEEE Computer Society (2014)
5. Giangreco, I., Schuldt, H.: ADAM$_{pro}$: Database Support for Big Multimedia Retrieval. Datenbank-Spektrum **16**(1), 17–26 (2016)
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 – 28th International Conference, MMM 2022, Phu Quoc, Vietnam, June 6-10, 2022, Proceedings, Part II. Lecture Notes in Computer Science, vol. 13142, pp. 493–498. Springer (2022)
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 – 27th International Conference, MMM 2021, Prague, Czech Republic, June 22-24, 2021, Proceedings, Part II. Lecture Notes in Computer Science, vol. 12573, pp. 435–440. Springer (2021)

8. Heller, S., Gsteiger, V., Bailer, W., Gurrin, C., Jónsson, B.Þ., Lokoc, J., Leibetseder, A., Mejzlík, F., Peska, L., Rossetto, L., Schall, K., Schoeffmann, K., Schuldt, H., Spiess, F., Tran, L., Vadicamo, L., Veselý, P., Vrochidis, S., Wu, J.: 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**(1), 1–18 (2022)
9. Lokoc, J., Bailer, W., Schoeffmann, K., Münzer, B., Awad, G.: On Influential Trends in Interactive Video Retrieval: Video Browser Showdown 2015-2017. IEEE Trans. Multim. **20**(12), 3361–3376 (2018)
10. Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., Assran, M., Ballas, N., Galuba, W., Howes, R., Huang, P., Li, S., Misra, I., Rabbat, M.G., Sharma, V., Synnaeve, G., Xu, H., Jégou, H., Mairal, J., Labatut, P., Joulin, A., Bojanowski, P.: DINOv2: Learning Robust Visual Features without Supervision. CoRR **abs/2304.07193** (2023)
11. Radford, A., Kim, J.W., Xu, T., Brockman, G., McLeavey, C., Sutskever, I.: Robust Speech Recognition via Large-Scale Weak Supervision. In: International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA. Proceedings of Machine Learning Research, vol. 202, pp. 28492–28518. PMLR (2023)
12. Rossetto, L., Amiri Parian, M., Gasser, R., Giangreco, I., Heller, S., Schuldt, H.: Deep Learning-Based Concept Detection in vitrivr. In: MultiMedia Modeling – 25th International Conference, MMM 2019, Thessaloniki, Greece, January 8-11, 2019, Proceedings, Part II. Lecture Notes in Computer Science, vol. 11296, pp. 616–621. Springer (2019)
13. Rossetto, L., Gasser, R., Lokoc, J., Bailer, W., Schoeffmann, K., Münzer, B., Soucek, T., Nguyen, P.A., Bolettieri, P., Leibetseder, A., Vrochidis, S.: Interactive Video Retrieval in the Age of Deep Learning – Detailed Evaluation of VBS 2019. IEEE Trans. Multim. **23**, 243–256 (2021)
14. Rossetto, L., Giangreco, I., Gasser, R., Schuldt, H.: Competitive Video Retrieval with vitrivr. In: MultiMedia Modeling – 24th International Conference, MMM 2018, Bangkok, Thailand, February 5-7, 2018, Proceedings, Part II. Lecture Notes in Computer Science, vol. 10705, pp. 403–406. Springer (2018)
15. Rossetto, L., Giangreco, I., Heller, S., Tanase, C., Schuldt, H., Dupont, S., Seddati, O., Sezgin, T.M., Altiok, O.C., Sahillioglu, Y.: IMOTION – Searching for Video Sequences Using Multi-Shot Sketch Queries. In: MultiMedia Modeling - 22nd International Conference, MMM 2016, Miami, FL, USA, January 4-6, 2016, Proceedings, Part II. Lecture Notes in Computer Science, vol. 9517, pp. 377–382. Springer (2016)
16. Rossetto, L., Giangreco, I., Schuldt, H.: Cineast: A Multi-feature Sketch-Based Video Retrieval Engine. In: 2014 IEEE International Symposium on Multimedia, ISM 2014, Taichung, Taiwan, December 10-12, 2014. pp. 18–23. IEEE Computer Society (2014)
17. Rossetto, L., Giangreco, I., Schuldt, H., Dupont, S., Seddati, O., Sezgin, T.M., Sahillioglu, Y.: IMOTION – A Content-Based Video Retrieval Engine. In: MultiMedia Modeling – 21st International Conference, MMM 2015, Sydney, NSW, Australia, January 5-7, 2015, Proceedings, Part II. Lecture Notes in Computer Science, vol. 8936, pp. 255–260. Springer (2015)
18. Rossetto, L., Giangreco, I., Tanase, C., Schuldt, H.: vitrivr: A Flexible Retrieval Stack Supporting Multiple Query Modes for Searching in Multimedia Collections. In: ACM Conference on Multimedia (2016)

19. Rossetto, L., Giangreco, I., Tanase, C., Schuldt, H., Dupont, S., Seddati, O.: Enhanced Retrieval and Browsing in the IMOTION System. In: MultiMedia Modeling - 23rd International Conference, MMM 2017, Reykjavik, Iceland, January 4-6, 2017, Proceedings, Part II. Lecture Notes in Computer Science, vol. 10133, pp. 469–474. Springer (2017)
20. Rossetto, L., Sauter, L.: Vimeo Creative Commons Collection (V3C) Whisper Transcripts (Nov 2022)
21. Rossetto, L., Schuldt, H., Awad, G., Butt, A.A.: V3C - A Research Video Collection. In: MultiMedia Modeling – 25th International Conference, MMM 2019, Thessaloniki, Greece, January 8-11, 2019, Proceedings, Part I. Lecture Notes in Computer Science, vol. 11295, pp. 349–360. Springer (2019)
22. Sauter, L., Gasser, R., Heller, S., Rossetto, L., Saladin, C., Spiess, F., Schuldt, H.: Exploring Effective Interactive Text-Based Video Search in vitrivr. In: MultiMedia Modeling - 29th International Conference, MMM 2023, Bergen, Norway, January 9-12, 2023, Proceedings, Part I. Lecture Notes in Computer Science, vol. 13833, pp. 646–651. Springer (2023)
23. Sauter, L., Parian, M.A., Gasser, R., Heller, S., Rossetto, L., Schuldt, H.: Combining Boolean and Multimedia Retrieval in vitrivr for Large-Scale Video Search. In: MultiMedia Modeling – 26th International Conference, MMM 2020, Daejeon, South Korea, January 5-8, 2020, Proceedings, Part II. Lecture Notes in Computer Science, vol. 11962, pp. 760–765. Springer (2020)
24. Sauter, L., Schuldt, H., Waltenspül, R., Rossetto, L.: Novice-Friendly Text-based Video Search with vitrivr. In: 20th International Conference on Content-based Multimedia Indexing (CBMI 2023), September 20–22, 2023, Orléans, France. ACM (2023)
25. Spiess, F., Gasser, R., Heller, S., Parian-Scherb, M., Rossetto, L., Sauter, L., Schuldt, H.: 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. Lecture Notes in Computer Science, vol. 13142, pp. 499–504. Springer (2022)
26. Spiess, F., Gasser, R., Heller, S., Rossetto, L., Sauter, L., Schuldt, H.: Competitive Interactive Video Retrieval in Virtual Reality with vitrivr-VR. In: MultiMedia Modeling – 27th International Conference, MMM 2021, Prague, Czech Republic, June 22-24, 2021, Proceedings, Part II. Lecture Notes in Computer Science, vol. 12573, pp. 441–447. Springer (2021)
27. Spiess, F., Heller, S., Rossetto, L., Sauter, L., Weber, P., Schuldt, H.: Traceable Asynchronous Workflows in Video Retrieval with vitrivr-VR. In: MultiMedia Modeling – 29th International Conference, MMM 2023, Bergen, Norway, January 9-12, 2023, Proceedings, Part I. Lecture Notes in Computer Science, vol. 13833, pp. 622–627. Springer (2023)
28. Stonebraker, M., Rowe, L.A.: The Design of Postgres. In: Proceedings of the 1986 ACM SIGMOD International Conference on Management of Data, Washington, DC, USA, May 28-30, 1986. pp. 340–355. ACM Press (1986)
29. Truong, Q., Vu, T., Ha, T., Lokoc, J., Tim, Y.H.W., Joneja, A., Yeung, S.: 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, Proceedings, Part I. Lecture Notes in Computer Science, vol. 13833, pp. 539–550. Springer (2023)