Exploring Multimedia Vector Spaces with vitrivr-VR

Florian Spiess[1][0000−0002−3396−1516], Luca Rossetto[2][0000−0002−5389−9465], and Heiko Schuldt[1][0000−0001−9865−6371]

[1] Department of Mathematics and Computer Science
University of Basel, Basel, Switzerland
{firstname.lastname}@unibas.ch
[2] Department of Informatics, University of Zurich, Zurich, Switzerland
rossetto@ifi.uzh.ch

Abstract. Virtual reality (VR) interfaces are becoming more commonplace as the number of capable and affordable devices increases. However, VR user interfaces for common computing tasks often fail to take full advantage of the affordances provided by this immersive interface modality. New interfaces designed for VR must be developed in order to fully explore the possibilities for user interaction.
In this paper, we present vitrivr-VR and the improvements made for its participation in the Video Browser Showdown (VBS) 2024. We describe the current state of vitrivr-VR, with a focus on a novel point cloud browsing interface and improvements made to its text input methods.

Keywords: Video Browser Showdown · Virtual Reality · Interactive Video Retrieval · Content-based Retrieval.

1 Introduction

Virtual reality (VR) is becoming increasingly commonplace as a user interface modality as more VR hardware is brought to market. While many VR-only applications, such as games and interactive entertainment, implement and explore ways to fully utilize the immersion and interactive possibilities afforded by virtual reality, interfaces for tasks also performed on conventional interfaces, such as search and browsing, are often direct translations of their conventional interface counterparts.

vitrivr-VR is a virtual reality multimedia retrieval system research prototype that aims to explore how virtual reality can be utilized to improve multimedia retrieval and in particular user interaction with immersive query and results browsing interfaces. To evaluate how the methods and interfaces implemented in vitrivr-VR compare to other state-of-the-art systems, vitrivr-VR participates in multimedia retrieval campaigns such as the Video Browser Showdown (VBS) [2]. The VBS is a valuable source of insights, which has led to multiple important improvements of vitrivr-VR through previous participations [9,10,11].

In this paper, we describe the version of the vitrivr-VR system participating in the VBS 2024, with a focus on our novel point cloud results browsing interface and improvements made to text input. We describe the system and its components in Section 2, introduce our novel point cloud display in Section 3, describe our text input improvements in Section 4, and conclude in Section 5.

2 vitrivr-VR

vitrivr-VR is an experimental virtual reality multimedia retrieval and analytics system. Based on components of the vitrivr stack [8], the system architecture of vitrivr-VR consists of three parts: the vector database *Cottontail DB* [1], the feature extraction and retrieval engine *Cineast* [7], and the *vitrivr-VR*[3] virtual reality user interface. The vitrivr-VR user interface is developed in Unity with C# and is capable of running on any platform compatible with OpenXR. All parts of the vitrivr-VR stack are open-source and available on GitHub.[4]

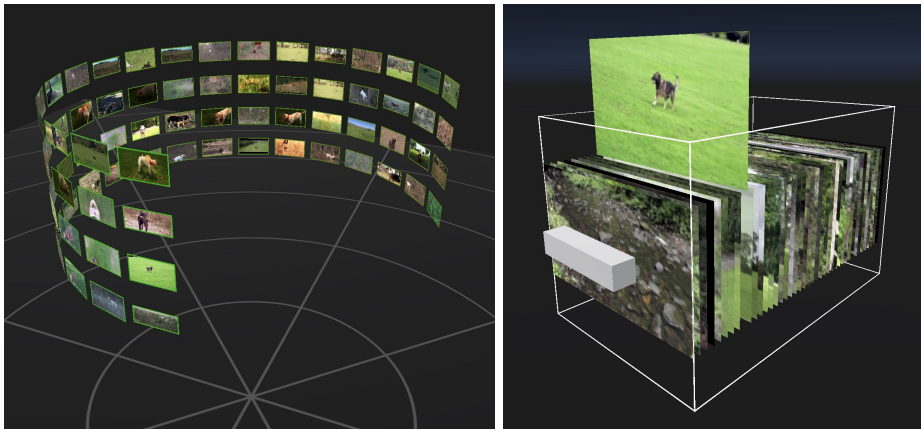
The goal of vitrivr-VR is to explore methods of multimedia retrieval and analytics with virtual reality user interfaces. To achieve this goal, vitrivr-VR implements query formulation, results browsing, and multimedia interaction methods developed with virtual reality user interfaces in mind, enabling an immersive user experience.

vitrivr-VR implements a number of query formulation methods for queries of different kinds. Due to the prevalence of semantic multimodal embedding features such as CLIP [5], text is an important query modality. Text input for VR interfaces is still subject to active research. vitrivr-VR implements a combination of speech-to-text and a word-gesture keyboard [12], which allows the input of entire sentences at a time with speech-to-text and fast word-level corrections using the word-gesture keyboard. Other query modalities supported by vitrivr-VR include query-by-concept, -pose, -sketch, and -example (frame). Queries can consist of individual query terms, combinations, or even temporally ordered query terms for temporal queries.

To browse query results, vitrivr-VR implements a number of results display and browsing methods. The main results display of vitrivr-VR is a cylindrical grid, an example of which can be seen in Figure 1a. This display method is a direct translation of conventional grid results displays into VR, with the only modification being that the results curve around the user, allowing more intuitive browsing through head movement. The cylindrical results display can be rotated to reveal more results, replacing already viewed results. Results can be selected from this display to be viewed in greater detail.

To allow easier browsing within a video vitrivr-VR implements a multimedia drawer view, as depicted in Figure 1b. This interactive display allows the temporally ordered browsing within the key frames of a video. By hovering a hand over a keyframe, the frame is selected and hovers above the drawer for easier viewing. In this way, a user can move their hand through the drawer to view

[3] https://github.com/vitrivr/vitrivr-vr
[4] https://github.com/vitrivr



(a) (b)

Fig. 1. Screenshots of the vitrivr-VR user interface: 1a cylindrical grid results display and 1b multimedia drawer segment view.

the key frames of a video in a fashion similar to a flip book. Keyframes can be selected to skip to that part of the video and view the respective video segment in greater detail. The handle at the front of the drawer allows it to be extended, increasing the spacing between keyframes and allowing easier frame selection for videos with many keyframes.

3 Point Cloud Browsing

Many of the theoretical advantages of VR interfaces over conventional interfaces in multimedia browsing and retrieval are enabled by the additional usable space for displaying and interacting with multimedia content. While vitrivr-VR already implements interfaces to explore these advantages for intra-video browsing, inter-video browsing in vitrivr-VR, such as browsing of query results, was previously only possible through interfaces heavily inspired by interfaces used on conventional displays.

Conventional grid-based displays for similarity search results have two properties that limit their effectiveness for browsing: due to the human cognitive limit, they are only able to present a very limited number of results at any given time, and since they only take into account the similarity of result items to the search query, results can only be displayed and traversed as sorted by this similarity measure. Furthermore, even though grid-based displays are inherently two-dimensional, the similarity of each result to the query is the only meaningful dimension of the data displayed. While wrapping the results into a grid allows more results to be displayed within a user's perceptive field, these grid-based displays use both dimensions available to them to express only a single quality of the data.

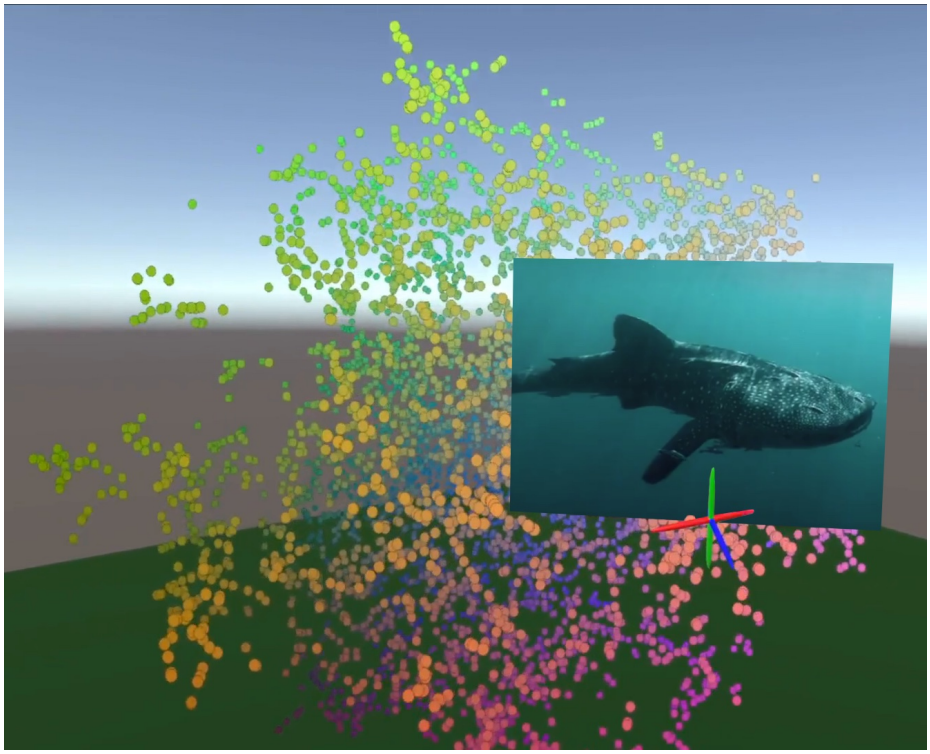


Fig. 2. The point cloud display shows a preview frame corresponding to the video segment represented by the currently hovered point. Points are colored by their coordinates in the reduced feature space to make distinguishing and keeping track of them easier.

To overcome these limitations of grid-based displays and further explore the possibilities of multimedia browsing in virtual reality, we have implemented a new point cloud-based results browsing interface for vitrivr-VR. Utilizing all three spatial dimensions available in virtual reality, this new point cloud display shows the highest-ranked results of a query as a point cloud in virtual space. To make the most use of the available dimensions, points in the point cloud are placed according to a dimensionality reduction of a high-dimensional feature space, allowing the similarity between results presented by the points to be estimated at a glance. By only including the highest-ranked results of the similarity query in the dimensionality reduction, the three reduced dimensions are fully utilized for differences within the result set rather than needing to include information about the collection as a whole. To reduce the cognitive strain of viewing a large number of multimedia items at once, results in the point cloud display are represented as points in space, only showing a preview of the respective query result when hovered by a user's hand. An example of the point cloud display is shown in Figure 2.

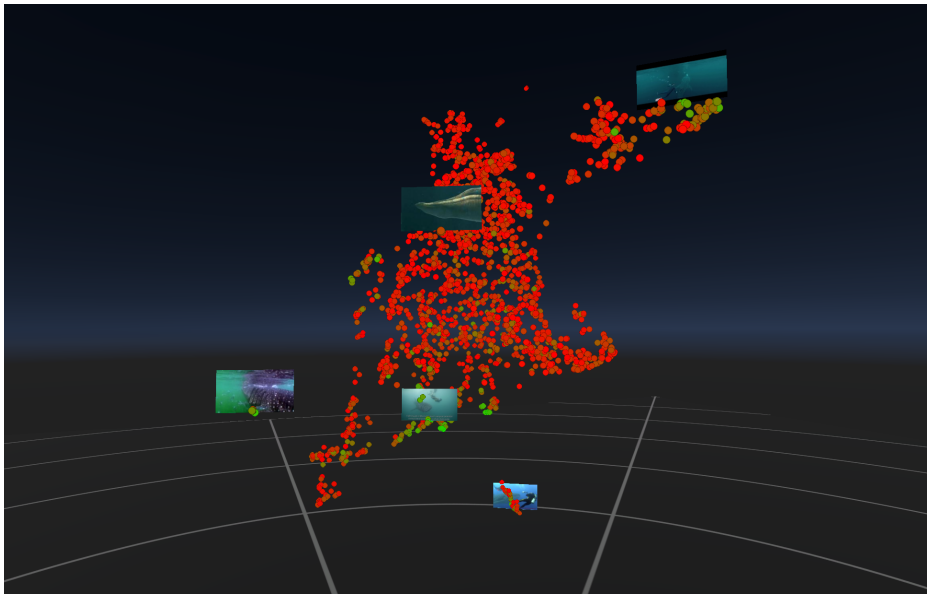


Fig. 3. The point cloud display for the query "A whale shark". The points are colored by their relative query score, where the best scoring result is colored green, and the worst scoring result is colored red.

The point cloud display can be used with any feature space, dimensionality reduction method, and point coloring, irrespective of the similarity query used to generate the results. This parameterization allows the point cloud display to be tuned to a specific task or use case. In our implementation of the point cloud display for the VBS 2024, we use a semantic co-embedding feature, experiment with t-SNE [3] and UMAP [4] dimensionality reduction, and color the points configurably by the relevance to the query (as shown in Figure 3) or the coordinates in the reduced feature space.

The point cloud display is not meant to replace conventional browsing displays, such as the cylindrical grid display implemented in vitriv-VR, but rather to complement them by providing alternative forms of user interaction and result display. While the point cloud display shows fewer result previews at once, it allows top-scoring results to be viewed in the context of their similarity to other results, making it easy to disregard entire clusters of irrelevant results and focus on areas of the point cloud containing more related results.

The most closely related browsing method previously used at a VBS is that implemented in EOLAS [13]. While this method also uses dimensionality reduction to show query results within the context of similar multimedia objects, it allows the viewing of only a very limited number of results at once and does not allow quick scanning of the result space.

4 Improved Text Input

In addition to the implementation of the point cloud display, this iteration of vitrivr-VR includes a number of further improvements. The two most notable of these improvements both concern how users of the system can input text. Analyses of the VBS 2023 indicate, once more, that the most successful systems rely on text queries and that text input is much slower in VR systems than in comparable systems using conventional interfaces [11].

The first improvement to vitrivr-VR text input is the replacement of our speech-to-text input method. Previously based on Mozilla DeepSpeech[5], our new speech-to-text input is now based on the much more capable OpenAI Whisper [6]. Whisper addresses some of the most common limitations of our implementation of DeepSpeech, namely the accuracy of transcription and support for languages other than English.

Our second improvement to text input is based on the observation that a query can be input by keyboard much more quickly than it can even be spoken for speech-to-text input. This puts VR systems at a disadvantage at the beginning of a search task when no results are available that can be scanned while refining the query. To give users the option to conveniently input their initial query using a physical keyboard, vitrivr-VR implements support for headsets with passthrough augmented reality capabilities, such as the Vive Focus 3.

5 Conclusion

This paper presents the state of the vitrivr-VR system in which we intend to participate in the VBS 2024. We focus particularly on our novel point cloud browsing interface, which presents results within the context of a similarity feature space, and on our improvements to text input.

Acknowledgements

This work was partly supported by the Swiss National Science Foundation through projects “Participatory Knowledge Practices in Analog and Digital Image Archives” (contract no. 193788) and “MediaGraph” (contract no. 202125).

References

1. Gasser, R., Rossetto, L., Heller, S., Schuldt, H.: Cottontail DB: An Open Source Database System for Multimedia Retrieval and Analysis. In: International Conference on Multimedia. pp. 4465–4468. Association for Computing Machinery, New York, NY, USA (2020). https://doi.org/10.1145/3394171.3414538

[5] https://github.com/mozilla/DeepSpeech

2. Lokoč, J., Andreadis, S., Bailer, W., Duane, A., Gurrin, C., Ma, Z., Messina, N., Nguyen, T.N., Peška, L., Rossetto, L., Sauter, L., Schall, K., Schoeffmann, K., Khan, O.S., Spiess, F., Vadicamo, L., Vrochidis, S.: Interactive Video Retrieval in the Age of Effective Joint Embedding Deep Models: Lessons from the 11^{th} VBS. Multimedia Systems (2023). https://doi.org/10.1007/s00530-023-01143-5
3. van der Maaten, L., Hinton, G.: Visualizing Data Using T-SNE. Journal of Machine Learning Research **9**(86), 2579–2605 (2008), http://jmlr.org/papers/v9/vandermaaten08a.html
4. McInnes, L., Healy, J., Melville, J.: UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. arXiv (2018). https://doi.org/10.48550/ARXIV.1802.03426
5. 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. arXiv (2021). https://doi.org/10.48550/ARXIV.2103.00020
6. Radford, A., Kim, J.W., Xu, T., Brockman, G., Mcleavey, C., Sutskever, I.: Robust Speech Recognition via Large-Scale Weak Supervision. In: Proceedings of the 40th International Conference on Machine Learning. vol. 202, pp. 28492–28518. PMLR (2023), https://proceedings.mlr.press/v202/radford23a.html
7. Rossetto, L., Giangreco, I., Schuldt, H.: Cineast: A Multi-feature Sketch-Based Video Retrieval Engine. In: IEEE International Symposium on Multimedia (2014). https://doi.org/10.1109/ISM.2014.38
8. Rossetto, L., Giangreco, I., Tanase, C., Schuldt, H.: vitrivr: A Flexible Retrieval Stack Supporting Multiple Query Modes for Searching in Multimedia Collections. In: Proceedings of the 2016 ACM Conference on Multimedia Conference, MM 2016, Amsterdam, The Netherlands, October 15-19, 2016. pp. 1183–1186. ACM (2016). https://doi.org/10.1145/2964284.2973797
9. 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. pp. 499–504. Springer (2022). https://doi.org/10.1007/978-3-030-98355-0_45
10. 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. pp. 441–447. Springer (2021). https://doi.org/10.1007/978-3-030-67835-7_42
11. Spiess, F., Gasser, R., Heller, S., Schuldt, H., Rossetto, L.: A Comparison of Video Browsing Performance between Desktop and Virtual Reality Interfaces. In: Proceedings of the 2023 ACM International Conference on Multimedia Retrieval. pp. 535–539. ACM (2023). https://doi.org/10.1145/3591106.3592292
12. Spiess, F., Weber, P., Schuldt, H.: Direct Interaction Word-Gesture Text Input in Virtual Reality. In: 2022 IEEE International Conference on Artificial Intelligence and Virtual Reality (AIVR). pp. 140–143. IEEE, CA, USA (2022). https://doi.org/10.1109/AIVR56993.2022.00028
13. Tran, L.D., Nguyen, M.D., Nguyen, T.N., Healy, G., Caputo, A., Nguyen, B.T., Gurrin, C.: A VR Interface for Browsing Visual Spaces at VBS2021. In: MultiMedia Modeling, vol. 12573, pp. 490–495. Springer International Publishing, Cham (2021). https://doi.org/10.1007/978-3-030-67835-7_50