On the User-centric Comparative Remote Evaluation of Interactive Video Search Systems

Luca Rossetto* Ralph Gasser¶ Silvan Heller,¶ Mahnaz Parian-Scherb¶‡ Loris Sauter¶ Florian Spiess¶
Heiko Schuldt¶ Ladislav Peška§ Tomáš Souček§ Miroslav Kratochvíl§ František Mejzlík§ Patrik Veselý§
Jakub Lokoč§
* Department of Informatics, University of Zurich, Switzerland
¶ Department of Mathematics and Computer Science, University of Basel, Switzerland
‡ Numediart Institute, University of Mons, Belgium
§ Department of Software Engineering, Charles University, Prague

Abstract—In the research of video retrieval systems, comparative assessments during dedicated retrieval competitions provide priceless insights into the performance of individual systems. The scope and depth of such evaluations is unfortunately hard to improve, due to the limitations by the set-up costs, logistics and organization complexity of large events. We show that this easily impairs the statistical significance of the collected results, and the reproducibility of the competition outcomes. In this paper, we present a methodology for remote comparative evaluations of content-based video retrieval systems and demonstrate that such evaluations scale-up to sizes that reliably produce statistically robust results, and propose additional measures that increase the replicability of the experiment. The proposed remote evaluation methodology forms a major contribution towards open science in interactive retrieval benchmarks. At the same time, the detailed evaluation reports form an interesting source of new observations about many subtle, previously inaccessible aspects of video retrieval.

I. Introduction

With the ever-growing amount of information available digitally, we increasingly depend on effective search methods and systems that enable users to find items of interest. The problem of finding specific items in large datasets is addressed by research in information retrieval (IR). While IR systems already handle volumes of textual information that literally span the entire web, the content-based search in other types of media, such as images or video, poses a more challenging problem. Current research on video retrieval methods suggests that the best results are achieved interactively, with a human operator of the retrieval system working in a feedback loop, examining intermediate results and formulating and refining queries iteratively [1], resulting in a hybrid human-machine approach.

The reproducibility of the results obtained in experiments with modern IR systems is often difficult to guarantee, or outright impossible, when depending on the input of a human operator. Ferro [2] summarized the main challenges as follows: First, even with open source software, there are countless opaque parameters and configurations that heavily influence the performance of an IR system. Second, the data collections used in experiments are often inaccessible or incomplete, such as in case of a text corpus relying on Tweets. Moreover, the IR researchers usually meta-evaluate their systems to improve their own methodology, often using publicly unavailable data, thus making their key observations and motivation for a specific configuration of the system irreproducible. Finally, with interactive IR we must also consider the human operators themselves, whose performance is derived from experience, domain knowledge, or simply from the momentary well-being. Data collection methodology needs to be adjusted to account for this variability, for example, by collecting additional data on the search progress instead of just the final result.

The Video Browser Showdown (VBS) [3], which celebrates its 10^{th} anniversary in 2021, is an annual competitive evaluation event for interactive video retrieval systems aimed at advancing the state of the art in this field. The VBS is held as a special session during the International Conference on Multimedia Modeling, to provide the same conditions for all on-site participants. Since there are no restrictions on how the retrieval tasks are to be solved, participants are free to bring whatever hardware best suits their respective approaches, introducing another opaque parameter into the process. During the VBS, the participating teams solve various kinds of search tasks, usually with two operators per team, and are scored by a central evaluation server for successfully finishing the tasks within the time limit. Currently, three types of tasks are used: Visual Known-Item search (VKIS), in which participants are shown a unique 20-second sequence of a video that needs to be found, Textual Known-Item Search (TKIS) in which participants are given only a textual description of the unique sequence, and Ad-hoc Video Search (AVS), in which the participants need to find as many sequences as possible that match a more general, textual description. The correctness of submitted AVS sequences is not predetermined, but judged on the fly by human referees.

All participants are provided with the competition dataset in advance, in order to be able to perform the data pre-processing necessary for their retrieval approaches; but are only introduced to the actual task descriptions during the event. Currently, VBS uses the open V3C1 [4] dataset,

which combines around 1000 total hours of video that accurately represents the arbitrary videos found on the Web.

While the VBS undoubtedly provides a valuable opportunity to evaluate state-of-the-art video search systems and draw conclusions about their performance, the scope of the performed evaluation is limited for many reasons: First, VBS results are only a single-day snapshot of the performance of a very limited number of operators, and momentary variations in individual performance may influence the outcome disproportionally, especially given the very small margins by which a team is deemed to be the 'winner'. Second, due to the competition structure, there is only a limited number of tasks that may be solved on a given day, which may be detrimental to the significance of the results. Finally, and despite recent advances in logging, there are many aspects to the competition result that cannot be reproduced, such as the crowd-sourced verdicts by human judges during the AVS tasks.

Technically, the competitions measure performance of a hybrid human-machine search approach, as the evaluation reflects both the performance of the human operator and the capability of the retrieval system. In this paper, we present the insights gained from an extensive dedicated VBS-style evaluation conducted in a distributed setting, that focused on a comprehensive comparison of teams operating SOMHunter [5] and vitrivr [6]. The systems have both won the VBS in one of the last two years. For the evaluation, we utilized a new specialized Distributed Retrieval Evaluation Server (DRES) [7], see dres.dev. DRES is orchestrating the evaluation by presenting the tasks and hints to all participants, collecting the retrieved results, evaluating their correctness, and assigning the scores. All systems involved, as well as the server itself, were made available as open source. Similarly to VBS, the evaluation used the V3C1 dataset, with tasks prepared solely for the purpose of this evaluation. However, as opposed to VBS, only visual and textual KIS tasks were performed.

The contribution of this paper is twofold: First, we report new insights on statistical significance of the collected results. In particular, we provide a base estimate of the evaluation size required for producing reliable results that can distinguish even teams that perform so similarly as SOMHunter and vitrivr, and we give an overview of interesting observations derived from the collected data. Both of these will serve as a reference point for designing future evaluations. Second, we highlight the requirements and methodologies needed to achieve better evaluation reproducibility. Most importantly, we provide access to all data collected during the evaluations that are required to reproduce the outcome (see github.com/lucaro/siretVSvitrivr2020). We further point out a set of systematic procedures, such as the advanced logging capabilities of the new evaluation server, which increase the evaluation transparency and provide a robust data source for post-evaluation analysis. We showcase the results obtainable from competition logs on an observation of a rather surprising difference between retrieval and browsing capabilities of both systems.

II. Participating Video Search Systems

In this evaluation, we compare SOMHunter [5], the winner of VBS 2020 developed within the SIRET research group of the Charles University in Prague, with vitrivr [6], which has its primary home at the University of Basel in Switzerland and won VBS in 2019. vitrivr is an advanced, universal multimedia retrieval stack that comprises many multimedia retrieval models and media types. SOMHunter is, in contrast, a light-weight tool with a minimalist set of components for interactive video retrieval. Both systems are available as open source software. In the following, we will briefly introduce the two systems in detail.

A. SOMHunter

Designed from scratch by the team that previously developed the VIRET tool [8], SOMHunter was first presented at VBS 2020 [5]. The main design objective was to create a minimalist, fast and simple interactive search tool on top of a state-of-the-art (but replaceable) video-text matching model. Currently, a variant of the W2VV++ model (Word2VisualVec) is used [9], including the visual features, as recommended by Mettes et al. [10]. For both text descriptions and selected representative video frames extracted in advance, the trained model provides mappings to a single feature space, which allows the system to freely mix the evaluation of textual queries and visual feedback from the user.

The current open source version of the tool [11] (available from github.com/siret/somhunter) supports the following main features:

- Search with temporal text queries. Given an option to specify several consecutive frames or shots, users can try to target searched items with a specially constructed text query, and browse a ranked result list that shows the matching frame thumbnails. Specifically, all temporal text query elements are mapped to the feature space; for each query element vector, database frames are scored by making use of their feature space vector representations. Temporal queries then use a score aggregation function that accounts for several neighboring frames in the video sequence, as described by Lokoč et al. [8].
- Score refinement based on relevance feedback. For query reformulation, the users can select positive example images from the observed result set (as opposed to rewriting the text query), and ask the system to re-score the database accordingly. This type of refinement allows for more precise targeting of searches that do not have an immediately obvious textual description.
- SOM-based coarse browsing. To avoid browsing through near duplicate results, potentially provided by the previous two types of searches, the engine supports dynamic training of a self-organizing map

(currently 8×8 nodes), which displays a comprehensive and convenient topologically organized selection of the current, best-scoring results.
- Easily accessible exploitation views. Once the user finds a promising frame, either the k-nearest neighbors from the feature space or frames from the same video can be quickly displayed and sequentially browsed.

B. vitrivr

The open-source content-based multimedia retrieval stack vitrivr [6] is focused on multi-modal search in large multimedia collections. In particular, users can choose from a plethora of query formulation modalities to express their search and freely combine various modalities, including Query-by-Sketch, Query-by-Semantic-Sketch, Query-by-Example, various textual query modes, and many more. Textual queries can be used to query for text on screen, audio transcripts, scene captions, and detected concepts.

An example combination of these query modes could be find all images or video sequences of mountains with green in the bottom quarter, where mountain would be a text-based query executed first and green would be represented by a sketch. Alternatively, users may employ combinations of the modalities based on deep-learning, such as OCR, ASR and concept detection, aggregating individual scores, e.g., all planes (a concept) with EasyJet (OCR) on them, shown while someone talks about a flight to London. For videos, users may also arrange the queries in a temporal sequence, e.g., a scene depicting a lion, followed by scene with a giraffe, followed by an elephant. The front-end then aggregates results so that videos that contain these element in the specified temporal order are ranked higher than videos that contain only some of these elements [12]. In a competitive setup such as VBS, the visual (e.g., Query-by-Semantic-Sketch) and especially textual query modes have been proven to be very useful.

vitrivr's result presentation is separated in multiple views, each with its own ranking model. There is no one-size-fits-all view in vitrivr and in practice, a user may switch between the views depending on a concrete need and their personal preference. This is easily possible, since the same result-set can be accessed through different views without issuing a new query.

- In VBS style competitions, the mini gallery view that ranks the video segments by their similarity score in a tile layout is used most often. The top left segment has the highest score, and the bottom right one the lowest.
- The list view groups segments from the same object together (in the case of VBS, from the same video), and ranks them by max-pooling their scores. This view is beneficial for loading and inspecting the neighboring segment thumbnails.
- The temporal scoring view is used to consider the temporal context in a sequence, usually for browsing the results obtained from a temporal query. For example, given the task description 'A deer is looking directly into the camera. The next shot shows a tractor driving across a field.', a temporal query for deer first and then tractor could be used. The temporal scoring view then showcases sequences that match both query terms, allowing user to evaluate the match of both concepts at once.

All views incorporate late filtering and late fusion, to further refine the displayed segments. These functions allow a user to quickly filter and constrain the displayed segments using concepts and metadata occurring in the query results, enabling fast, local and result dependent query refinement.

Efficient browsing of large result sets is enabled through dynamic data loading. Result sets are only displayed partially and more data is loaded as the user scrolls down. This enables users to quickly browse through a large number of results. This feature is available in all parts of vitrivr. To allow large numbers of segments to be displayed at once, a segment's video is only loaded upon interaction. Hovering over a tile plays the video at the segment's temporal location or, if the preview tile is too small, a video player shows the segment in detail.

Submissions to the competition are possible from the video player as well as from the result presentation views. This enables users to submit segments which might not have been in the result set, but where a different segment from the same video was returned. This feature further strengthens vitrivr's browsing capabilities.

III. Distributed Comparison Setup

We performed the evaluation in July 2020 with 7 participants operating SOMHunter (two of whom had to leave before the end of the competition, missing a few tasks) and 8 working with vitrivr. In contrast to VBS, where all members of a team collectively contribute to a shared score, we scored each participant separately, resulting in a total of 15 'teams'. For both systems, some of the participants were experts who are highly familiar with both the system and the VBS-style competitions, while others lacked this experience completely. All participants were from computer science, as is usual at VBS. Due to the COVID-19 pandemic that made international travel difficult, we performed the evaluation in a distributed setting, using the newly developed evaluation server 'DRES'. An impression of the server interface used during the evaluation can be seen in Figure 1.

Both groups of participants gathered in one location each, (in Prague and Basel, respectively) in a room with a large display for providing a comparable presentation of the tasks. The setup is shown in Figure 2. DRES was managed from an independent third location (in Zurich), to prevent any group from gaining preliminary access to any task-related information. While this distributed setting introduces a communication delay, it does not affect the used evaluation metrics, as the network time

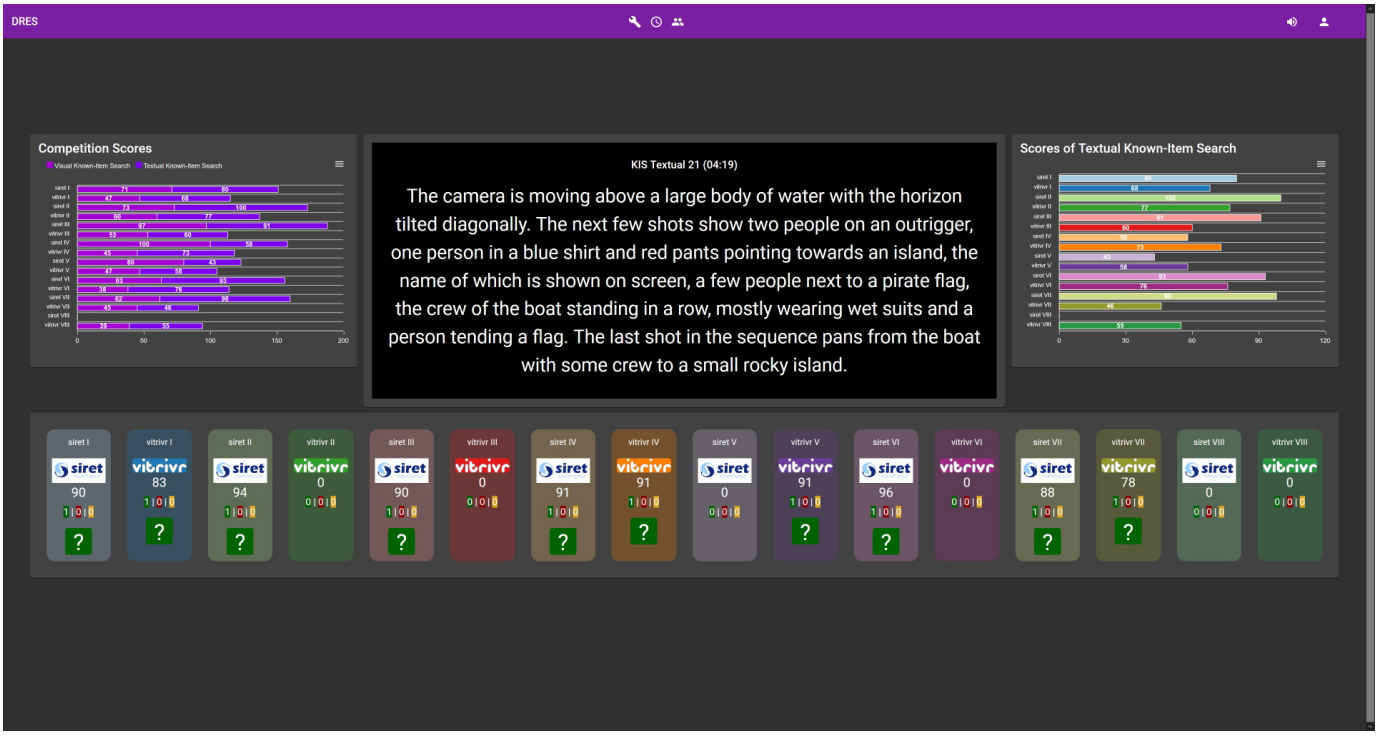

Fig. 1: Screenshot of the evaluation system during a textual known-item search task.

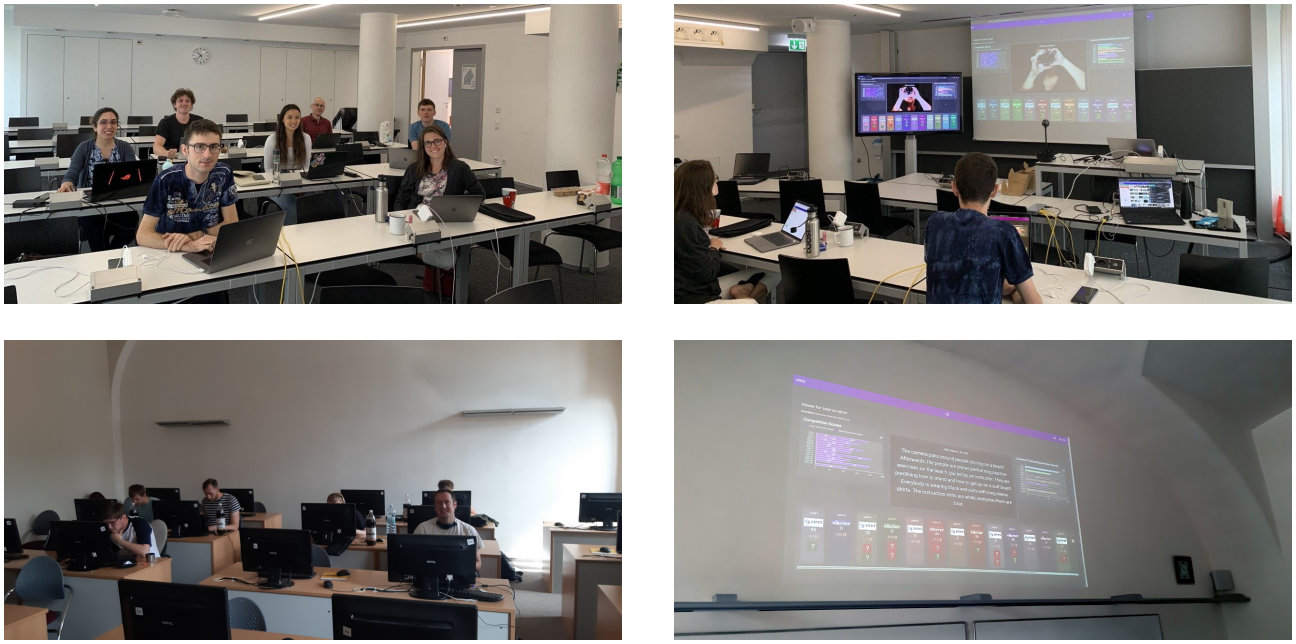

Fig. 2: Top: the participants and setup of vitrivr, participating from Basel, Switzerland. Bottom: the participants and setup of SIRET research group (using SOMHunter), participating from Prague, Czech Republic.

overhead is orders of magnitude smaller than the actual time needed for participants to solve a task. To remove the need for external judges, we only evaluated Known-Item Search (KIS) tasks using both the visual and textual query description. Three evaluation metrics were considered: A binary evaluation of whether the participant solved the task within the given time limit, an inverse linear scoring which rewards quicker solutions to tasks, and the VBS score [1], which provides a finer-grained metric of solution quality, accounting for the time to a correct submission as well as the number of incorrect submissions prior to the correct one. The latter metric has been used successfully during the VBS for several years now. It is a good fit, since it rewards correct responses that were found quickly while effectively discouraging the submission of many incorrect results, hence capturing precision, recall and time.

In total, we evaluated 42 unique tasks, 21 of which were visual and another 21 were textual. Each task was presented only once and no query targets were shared between the two task types. Considering only these two task types, this resulted in roughly twice as many tasks as would have been possible during a typical VBS. To the best of our knowledge, the event was the largest VBS-like comparative known-item video search evaluation recorded so far.

IV. Results

In the aggregate results, participants operating vitrivr were successful in 170 out of 336 tasks in total, and participants operating SOMHunter were successful in 190 out of 282 tasks. The overall success rates were thus 0.51 and 0.67, respectively. The difference between both teams was slightly higher on the visual KIS tasks (0.53 vs. 0.72) than on the textual KIS tasks (0.48 vs. 0.62). Within the successful searches, the mean VBS scores obtained per task were 75.2 vs. 77.7 for visual tasks and 69.1 vs. 75.9 for textual tasks, but the differences were not statistically significant. Considering the overall distribution of the scores, differences between SOMHunter and vitrivr participants existed on both ends of the score spectrum, i.e., a much lower volume of scores equal to zero and a higher volume of scores close to 100 for SOMHunter. The first corresponds to the binary evaluation, while the latter indicates that a considerable portion of tasks were solved very quickly with SOMHunter. Other than that, the distribution of scores was similar.

A. Significance analysis

First, we assess the statistical significance of the overall results and their subsets. We specifically consider the question of how large the evaluation should be to reliably select the best-performing team. There are two variables that contribute to the size of the evaluation: The number of evaluated tasks, and the number of participants per team. For the sake of simplicity, we focused only on the binary 'solved tasks' indicators. Statistically, SOMHunter significantly outperformed vitrivr ($p < 10^{-4}$ in Fisher exact test), which also held separately for the textual and visual task subsets (with $p = 0.016$ and $p < 10^{-3}$, respectively). The evaluation can therefore be assumed to be of sufficient size to reliably detect the difference between the SOMHunter and vitrivr competition participants.

To determine the necessary size of the future experiments, we have used the standard statistical bootstrapping techniques. Specifically, we performed a 2D bootstrap by first randomly selecting $1 \leq k \leq 42$ tasks (with possible repetition) and $1 \leq l \leq 7$ users of each retrieval system (again with repetition). For each k, the random selection was repeated 100 times, while for each list of selected tasks and each l, we repeated the selection of users 20 times. Overall, this gave 2000 bootstrap runs per (k, l) pair. Figure 3 shows the results obtained in bootstrapping for all (k, l) configurations as heatmaps.

In particular, Figure 3 (top) shows the percentage of bootstrap runs that ended with the same experimental result as the full experiment, i.e., the probability that any given subset of tasks and teams would show a higher overall performance for SOMHunter than for vitrivr. We observe that in order to deduce valid results reasonably often, larger experiment than the ones conducted at VBS are indeed necessary. For instance, if we want to receive a correct result in at least 95% of cases, the competition must evaluate at least $4-6$ participants each solving $20-25$ tasks, or two participants solving 40 tasks each. The obtained data may be further used to choose a sufficient minimal size of the task set for any number of available users.

In some cases, we may require a stronger evidence of differences between individual user groups, such as a statistically significant measurement of the difference. Figure 3 (bottom) summarizes the percentage of bootstrap runs required for obtaining frequentist-style significance (p-value < 0.05 for the Fisher exact test). Again, we observe that VBS-scale experiments would only rarely conclude with statistically significant results. Notably, the fraction of significant results obtained in the largest considered scenarios was still below 90%.

We would like to emphasize that the main motivation for this experiment is to point out the statistical limitations of in-place comparative evaluations, with a limited number of performed tasks and users. Assuming the collected outcomes from our remote comparative evaluation are representative enough and the assumptions for the statistical testing are sufficiently fulfilled, the bootstrap runs reveal that even 20 tasks performed by two users might not be enough to test for differences between systems like SOMHunter and vitrivr.

B. Score aggregation

In contrast to VBS evaluations, we did not collect the submissions to the retrieval tasks in groups of two or more participating system operators, but rather for each user individually. This enabled us to study the effects of the aggregation of operators in a team of two, which is used commonly at VBS but the effects of which have not been extensively studied so far. Figure 4 shows the sum of the scores over all tasks for all individual participants (on the diagonal) compared to the scores that would be obtained by all possible teams of two, independently of the system used. The combined scores are computed using the scoring function applied to the aggregated submissions of both team members.

The figure compares three different scoring functions: a binary function which simply awards a point for a solved task, a linear function which awards up to 100 points depending on the time remaining to solve a task, and a function which rewards early correct submissions and penalizes incorrect ones, as used at VBS [1]. While the first counting function for solved tasks already shows some differences between systems and participants, it provides

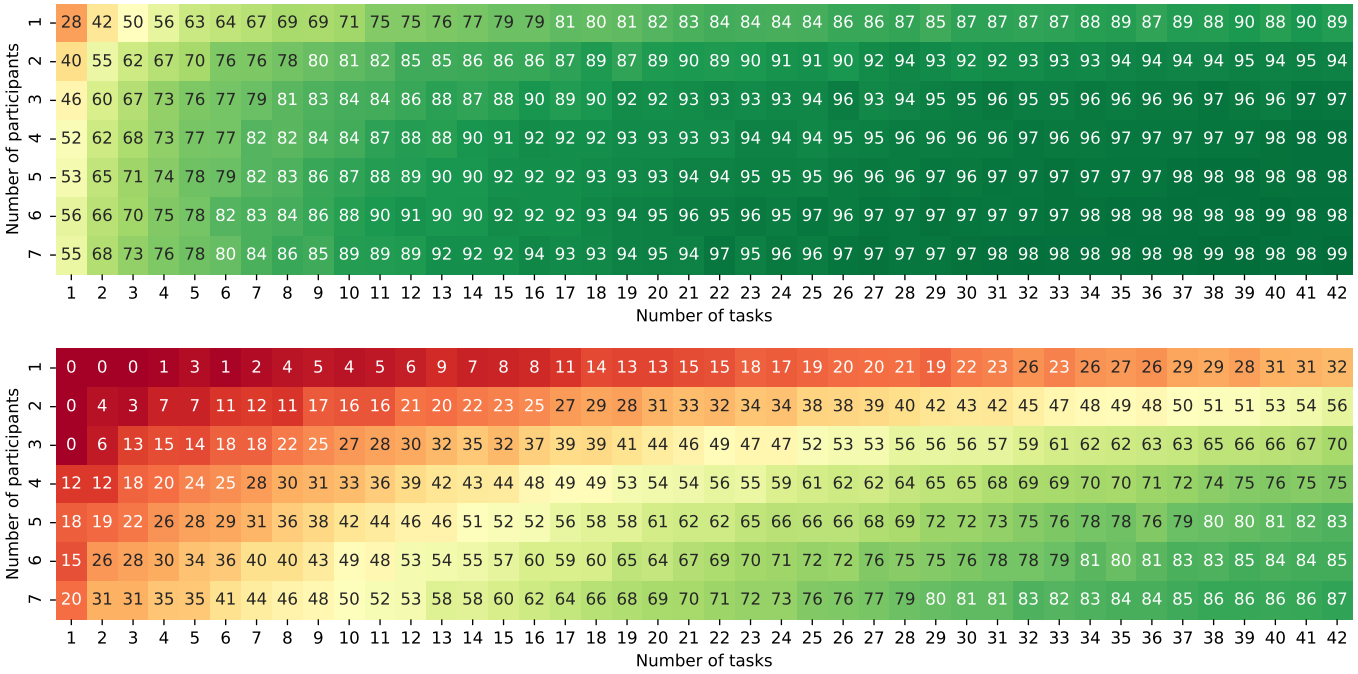

Fig. 3: 2D bootstrap results - top: percentage of the bootstrap runs that ended with the same results as the full test (i.e., SOMHunter outperformed vitrivr). bottom: percentage of the bootstrap runs with statistically significant results (p-value < 0.05) considering the Fisher exact test.

very little insight into how the results were achieved. The second scoring function refines the distinction by highlighting the differences in task submission speed, but fails to clearly distinguish between tasks that were solved very late, and tasks not solved within the limit. Notably, the first two functions only consider recall, resulting in a trivial way to achieve high scores by submitting as many (plausible) results as quickly as possible, thus increasing the probability of a correct submission. The third scoring function, also used at VBS, penalizes such exploitation by only partially discounting the score over time, and explicitly penalizing incorrect submissions in order to encourage retrieval precision as well as recall.

The comparison of scores of the hypothetical combined 2-participant teams has shown that the combined scores often vastly exceed the score of both individual participants. Interestingly, despite the overall higher individual scores of the participants with SOMHunter, the highest combined score (using the VBS scoring function) was generated by the combination of 'SOMHunter IV' and 'vitrivr VI'. This indicates that the two systems (or teams) likely possess complementary capabilities that can be leveraged in combination; presumably by reducing the probability of failing on a task that is, for some reason, hard to solve with a certain system. We believe that this provides an interesting case that supports collection of the results of all individual participants, where the detailed data might provide more useful insights into the qualities of the systems.

C. Retrieval versus Browsing

Both evaluated systems included preliminary support of detailed logging of user actions and retrieval sub-results, which we aimed to utilize for exploring the interactive search process. We have managed to collect a significant portion of the logging information that was sufficient for providing an illustrative view of some details (unfortunately, some logs from vitrivr instances were lost due to technical issues, and logs of two SOMHunter instances were incomplete).

The results are summarized in Figure 5, which shows the distributions of the time required until a correct submission in each system, and the relative distributions of the best ranks of the target video sequence (or any of its representative frames, in case of SOMHunter) in the internal retrieval model of the system. The data for the latter was recorded specifically in the logs for each user and task. Results are considered independently of whether they were actually noticed by the system operator, and the ranks do not necessarily correspond to the position of the retrieved frame on the display, due to possible rearrangements by the display logic. For completeness, we also included the distributions of the best whole-video ranks, i.e., the best ranks of any frame from the same video as the target sequence.

The plots reveal interesting insights into the effectiveness and potential of the search strategies and the underlying retrieval engines accompanied with rich browsing options. The result logs show that during the interactive search process, the best achieved ranks of target frames

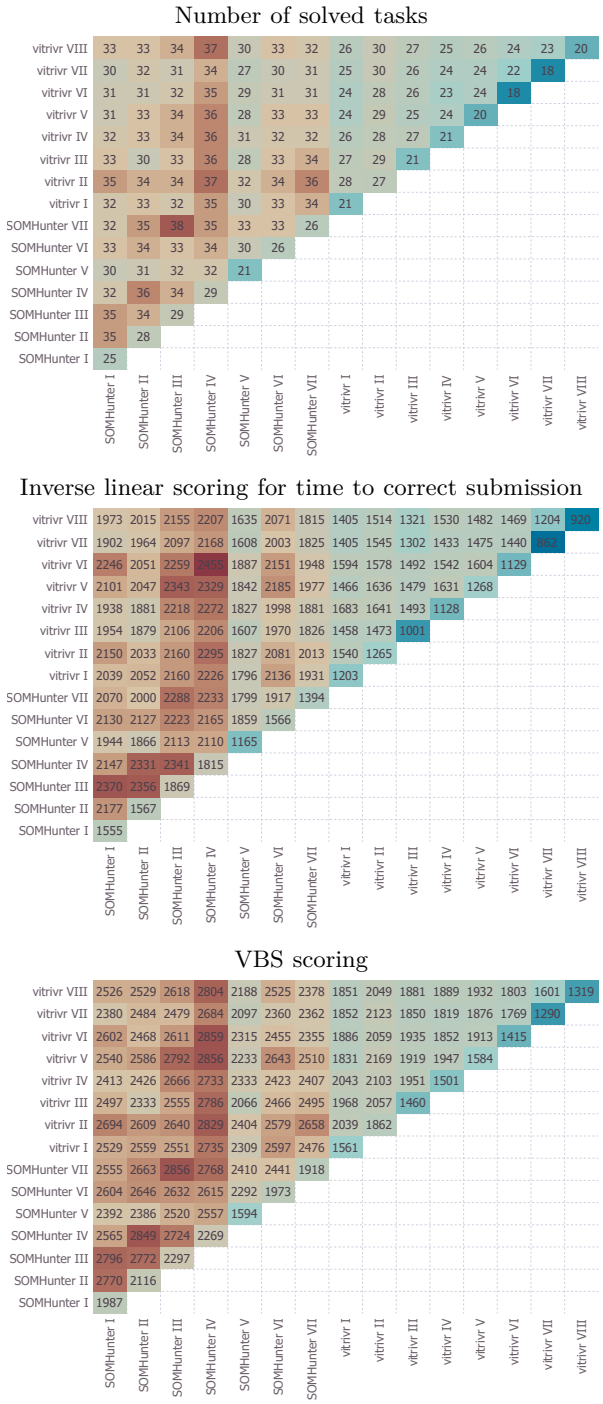

Fig. 4: Overview of the total scores that would be achieved by all possible team combinations with two participants, for 3 different scoring functions. Diagonals show the original scores of single-participant teams.

and videos were (on average) better in SOMHunter (when compared to vitrivr). This may be caused by the more effective ranking models and the different search strategy, as the SOMHunter-using participants chose to re-formulate the queries more often in order to increase the chances of a correct result appearing between top-ranked results. vitrivr users, on the other hand, generally spent more time by carefully examining a much larger portion of the result set, utilizing the browsing capabilities of vitrivr. We hypothesize that this behavior has often prevented vitrivr users from 'missing' the target with a sufficient rank in a result set, at the cost of longer browsing times and reduced number of inspected result sets. In contrast, SOMHunter users were often able to successfully solve tasks even if they failed to notice the displayed relevant target frames during the inspection of some promising candidate result sets. Testing and verifying such hypotheses will be possible in future evaluations that will use a more controllable log collection setting.

V. Discussion

Our results mainly show that for obtaining rigorous comparison results in video retrieval, large remote evaluations provide an interesting option. Here, we briefly summarize the main benefits.

First of all, with the number of newly emerging retrieval and interactive search options, it is essential to support efficient comparative evaluations, accessible for a large number of teams and users without the necessity of co-location. We demonstrated that 15 users from two teams may easily compete in a large number of known-item search tasks. As the evaluation ran without hitting any obvious long-term scalability limits and already generated new insights in the effectiveness of both systems' retrieval, we expect that similar events will become a major source of statistically significant performance measurements. The achievable significance of results, summarized in Figure 3, may serve as a starting point for future competitions. As an interesting benefit, the remote evaluations may be scaled to span multiple days, expanded by additional participants or run asynchronously with little effort; and even extended dynamically in case the collected results do not reach the desired level of statistical certainty.

More generally, our results indicate that while VBS serves well as a yearly benchmark for video retrieval systems and is a great opportunity to inspire new ideas, the number of participants and tasks is not sufficient for a statistically significant comparison in case of systems of similar performance. Since the organizational, provisional, financial and technical challenges of hosting a sufficiently large competition are prohibitive; we suggest to complement such 'small-scale' in-person evaluation events with long-term, larger-scale remote evaluations that collect the sufficiently precise statistics of systems' performance. The option to organize asynchronous evaluation events, the support of which is planned for future versions of DRES, could offer a new level of flexibility for the cooperation of teams from different time zones and participants who cannot attend in person. Despite the benefits of remote evaluations, the small-scale co-located competitions should not be dropped, as they provide the much required space for academic networking, and give the verifiable in-person 'credibility' to any produced competition results.

Systematic collection of precise interaction logs produces a large quantity of data useful for subsequent

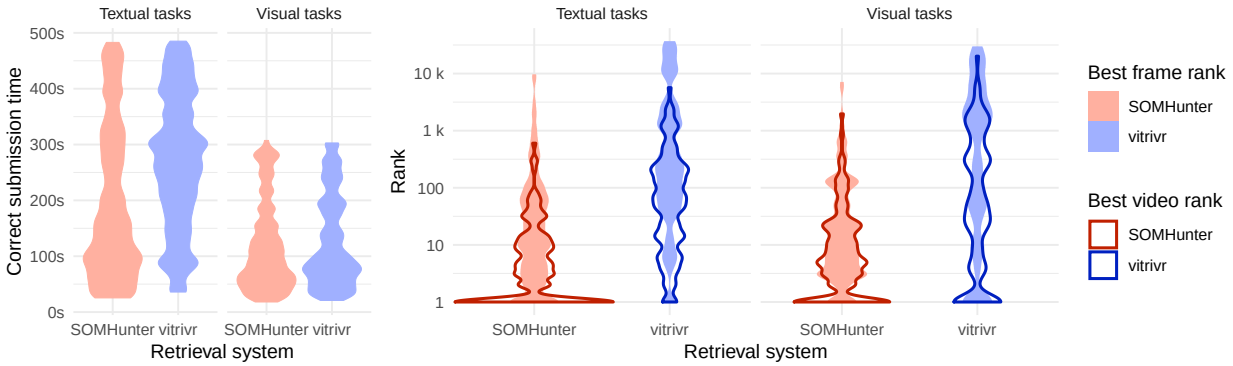

Fig. 5: Left: Distributions of the correct submission times. Right: Relative distributions of the best recorded positions of the target frames and target videos achieved in the internal ranking systems of the retrieval engines for each task and user. The plots provide a combined view of the effectiveness of ranking models, browsing options and used search strategies. Horizontal scaling is adjusted to equal area between corresponding measurements pairs.

analyses. In our case, the preliminary logging support has produced sufficient data to illustrate the difference between the utilized text search models and/or search strategies (see Figure 5). We expect that this approach will generate new results relevant in human-computer interaction research, such as uncovering the root causes of performance differences between the users, and allow us to accurately focus on effective querying components of the retrieval systems. In order to make logging controllable, statistically relevant and reproducible in the future, the competition servers (such as DRES) should readily provide options to properly test the client logging implementations for conformance and reliability. Additionally, because the log post-processing is a time demanding task, we hope to standardize a sufficiently extensible log format that will allow the analysts to simplify or even automate the log processing.

Replicability and reproducibility are important goals in systems research, particularly for the comparative evaluations. Although the required presence of human operators in the 'framework' vastly complicates the possibility to repeat an experiment exactly, we should still guarantee the reproducibility at the process level. Accordingly, we believe that the future evaluations should adhere to the following principles:

- Not only the systems, but also the exact configurations used for the evaluation should be released as a properly documented open source software. This would guarantee that the participating systems can be reconstructed for re-evaluation, and that their outputs and logged information can be easily interpreted.
- The detailed log analysis conducted for this paper was only possible because both systems took great care to adhere to the specification and test the compliance with the logging standard. For evaluations with more teams, we suggest multiple qualification stages, in which the teams are required to participate in 'dry runs' during which the logged data is subjected to validity and consistency checks.
- Any data artifacts (such as the image features extracted by neural networks) should be made available, as it is often unfeasible to re-run the entire feature extraction on large collections. Most importantly, it is necessary that the artifacts are deposited to be accessible long after the evaluation event reports are published.

In the future, we aim to integrate these principles into the peer review process for evaluation campaigns like VBS and the Lifelog Search Challenge (LSC). These principles are however not limited to interactive retrieval evaluations and adherence to them would also provide a benefit in other campaigns, such as TRECVID and MediaEval.

VI. Conclusion

In this paper, we have reported the experience gained from organizing a large remote evaluation of two content-based video retrieval systems (SOMHunter and vitrivr) in a VBS-style comparative setting, controlled by the recently released distributed retrieval evaluation server DRES.

We showed that it is easier, both organizationally and financially, to run evaluations in the remote setting, allowing to collect more data from a larger number of participants and tasks, especially when compared to on-site, in-person evaluations that often suffer from space and time constraints. Most importantly, with all participating systems (including the evaluation server) being available as open source, and with detailed information about the use of the systems available via logging, the concepts of open science, in particular the reproducibility of the entire evaluation process, are significantly strengthened.

While the evaluation has demonstrated the feasibility of conducting the experiments in a distributed setting, there is still room for improvement. We expect that improvements of the data collection methodology, such as a framework to precisely evaluate and compare the internal state of the search engines, may generate better insight into the tested retrieval approaches, and provide

more detailed feedback to drive their further development. Ultimately, these improvements have to reflect the human-machine hybrid approach and provide data about the human interaction with the systems.

We have additionally summarized several practical recommendations that may help to design future evaluations. Mainly, our data suggest a baseline of sample sizes (participant and task counts) required for a practical competition to deliver a decisive result. We additionally recommended a set of guidelines that improve the process-level replicability of the evaluation process.

Acknowledgment

This paper has been supported by Czech Science Foundation (GAČR) project 19-22071Y and by Charles University grant SVV-260588. MK was supported by ELIXIR CZ (MEYS), grant number LM2018131. The UNIBAS work was partly supported by the Hasler Foundation, project City-Stories (contract no. 17055) and by the Swiss National Science Foundation, project Polypheny-DB (contract no. 200021_172763) .

References

[1] L. Rossetto, R. Gasser, J. Lokoč, W. Bailer, K. Schoeffmann, B. Muenzer, T. Souček, P. A. Nguyen, P. Bolettieri, A. Leibetseder et al., "Interactive Video Retrieval in the Age of Deep Learning – A Detailed Evaluation of VBS 2019," IEEE Transactions on Multimedia, 2020.

[2] N. Ferro, "Reproducibility challenges in information retrieval evaluation," Journal of Data and Information Quality (JDIQ), vol. 8, no. 2, pp. 1–4, 2017.

[3] K. Schoeffmann, "Video browser showdown 2012-2019: A review," in 2019 International Conference on Content-Based Multimedia Indexing (CBMI). IEEE, 2019, pp. 1–4.

[4] L. Rossetto, H. Schuldt, G. Awad, and A. A. Butt, "V3c– a research video collection," in International Conference on Multimedia Modeling. Springer, 2019, pp. 349–360.

[5] M. Kratochvíl, P. Veselý, F. Mejzlík, and J. Lokoč, "Somhunter: Video browsing with relevance-to-som feedback loop," in International Conference on Multimedia Modeling. Springer, 2020, pp. 790–795.

[6] L. Rossetto, M. A. Parian, R. Gasser, I. Giangreco, S. Heller, and H. Schuldt, "Deep learning-based concept detection in vitrivr," in International Conference on Multimedia Modeling. Springer, 2019, pp. 616–621.

[7] L. Rossetto, R. Gasser, L. Sauter, A. Bernstein, and H. Schuldt, "A system for interactive multimedia retrieval evaluations," in International Conference on Multimedia Modeling. Springer, 2021, pp. 385–390.

[8] J. Lokoč, G. Kovalčík, T. Souček, J. Moravec, and P. Čech, "A framework for effective known-item search in video," in Proceedings of the 27th ACM International Conference on Multimedia, ser. MM '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 1777–1785. [Online]. Available: https://doi.org/10.1145/3343031.3351046

[9] X. Li, C. Xu, G. Yang, Z. Chen, and J. Dong, "W2VV++: fully deep learning for ad-hoc video search," in Proceedings of the 27th ACM International Conference on Multimedia, MM 2019, Nice, France, October 21-25, 2019, 2019, pp. 1786–1794. [Online]. Available: https://doi.org/10.1145/3343031.3350906

[10] P. Mettes, D. C. Koelma, and C. G. M. Snoek, "Shuffled imagenet banks for video event detection and search," ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM), vol. 16, no. 2, pp. 1–21, 2020.

[11] M. Kratochvíl, F. Mejzlík, P. Veselý, T. Souček, and J. Lokoč, "Somhunter: Lightweight video search system with som-guided relevance feedback," in Proceedings of the 28th ACM International Conference on Multimedia, 2020, pp. 4481–4484.

[12] S. Heller, L. Sauter, H. Schuldt, and L. Rossetto, "Multi-stage queries and temporal scoring in vitrivr," in 2020 IEEE International Conference on Multimedia & Expo Workshops (ICMEW). IEEE, 2020, pp. 1–5.