Interactive Search or Sequential Browsing? A Detailed Analysis of the Video Browser Showdown 2018

JAKUB LOKOČ and GREGOR KOVALČÍK, Charles University
BERND MÜNZER and KLAUS SCHÖFFMANN, Klagenfurt University
WERNER BAILER, JOANNEUM RESEARCH
RALPH GASSER, University of Basel
STEFANOS VROCHIDIS, Centre for Research and Technology Hellas
PHUONG ANH NGUYEN, City University of Hong Kong
SITAPA RUJIKIETGUMJORN, National Electronics and Computer Technology Center
KAI UWE BARTHEL, HTW Berlin, Visual Computing Group

This work summarizes the findings of the 7th iteration of the Video Browser Showdown (VBS) competition organized as a workshop at the 24th International Conference on Multimedia Modeling in Bangkok. The competition focuses on video retrieval scenarios in which the searched scenes were either previously observed or described by another person (i.e., an example shot is not available). During the event, nine teams competed with their video retrieval tools in providing access to a shared video collection with 600 hours of video content. Evaluation objectives, rules, scoring, tasks, and all participating tools are described in the article. In addition, we provide some insights into how the different teams interacted with their video browsers, which was made possible by a novel interaction logging mechanism introduced for this iteration of the VBS. The results collected at the VBS evaluation server confirm that searching for one particular scene in the collection when given a limited time is still a challenging task for many of the approaches that were showcased during the event. Given only a short textual description, finding the correct scene is even harder. In ad hoc search with multiple relevant scenes, the tools were mostly able to find at least one scene, whereas recall was the issue for

This work was supported by Czech Science Foundation (GAČR) project no. 17-22224S. Parts of this work were supported by Universität Klagenfurt and Lakeside Labs GmbH, Klagenfurt, Austria, and funding from the European Regional Development Fund and the Carinthian Economic Promotion Fund (KWF) under grant KWF 20214 u. 3520/26336/38165. Part of this research received funding from the Horizon 2020 Research and Innovation Programme V4Design, under grant agreement no. 779962. Parts of this work were supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (CityU 11250716). Work on *VITRIVR* was partly supported by the CHIST-ERA project IMOTION, with contributions from the Swiss National Science Foundation (SNSF, contract no. 20CH21_151571).

Authors' addresses: J. Lokoč and G. Kovalčík, Faculty of Mathematics and Physics, Charles University, Malostranske nam. 25, Prague, 118 00, Czech Republic; emails: lokoc@ksi.mff.cuni.cz, gregor.kovalcik@gmail.com; B. Münzer and K. Schöffmann, Institute of Information Technology, Klagenfurt University, Austria; emails: {bernd, ks}@itec.aau.at; W. Bailer, Institute of Information and Communication Technology, JOANNEUM RESEARCH, Steyrergasse 17, Graz, 8010, Austria; email: werner.bailer@joanneum.at; R. Gasser, Department of Mathematics and Computer Science, University of Basel, Basel, 4051, Switzerland; email: ralph.gasser@unibas.ch; S. Vrochidis, Information Technologies Institute, Centre for Research and Technology Hellas, 6th Klm Charilaou-Thermi Rd, Thessaloniki, 57001, Greece; email: stefanos@iti.gr; P. Anh Nguyen, Department of Computer Science, City University of Hong Kong, Tatchee Ave, Kowloon Tong, Hong Kong; email: panguyen2-c@my.city.edu.hk; S. Rujikietgumjorn, National Electronics and Computer Technology Center, Thailand; email: sitapa.ruj@nectec.or.th; K.-U. Barthel, HTW Berlin, Visual Computing Group, Wilhelminenhofstraße 75a, Berlin, 12459, Germany; email: Kai-Uwe.Barthel@HTW-Berlin.de.

1551-6857/2019/02-ART29 $15.00
https://doi.org/10.1145/3295663

many teams. The logs also reveal that even though recent exciting advances in machine learning narrow the classical semantic gap problem, user-centric interfaces are still required to mediate access to specific content. Finally, open challenges and lessons learned are presented for future VBS events.

CCS Concepts: • **Information systems** → **Video search**; *Multimedia and multimodal retrieval*;

Additional Key Words and Phrases: Interactive video retrieval, video browsing, content-based methods, evaluation campaigns

ACM Reference format:
Jakub Lokoč, Gregor Kovalčík, Bernd Münzer, Klaus Schöffmann, Werner Bailer, Ralph Gasser, Stefanos Vrochidis, Phuong Anh Nguyen, Sitapa Rujikietgumjorn, and Kai Uwe Barthel. 2019. Interactive Search or Sequential Browsing? A Detailed Analysis of the Video Browser Showdown 2018. *ACM Trans. Multimedia Comput. Commun. Appl.* 15, 1, Article 29 (February 2019), 26 pages.
https://doi.org/10.1145/3295663

1 INTRODUCTION

Content-based search in video collections today is more important than ever because in many domains, a significant amount of videos are produced and added to video archives on a weekly or even daily basis. This is true for first-class video citizen domains, such as media and entertainment, but also for less visible but similarly important areas, such as education, lifelogging, medicine, sports, and surveillance. In these domains, videos are used more and more for purposes like teaching and training, visual analytics, or as visual prostheses, either to remember specific situations or for inspecting complicated procedures. Such scenarios often require finding specific target scenes of interest. The content of such a target scene may be known to the searcher in advance, who only lacks the knowledge about where to find it—this is called *known-item search* (KIS). In some cases, there could also be a rough textual description of target segments, without knowing their actual content (e.g., "*a car driving down a road*")—this is called *ad hoc video search* (AVS), and such search scenarios are important in situations, in which a third party describes the desired content.

With respect to interactive video retrieval, several different sub-disciplines are equally important to realize a highly competitive content search system: performing accurate and effective video content analysis [19, 39] (e.g., automatic detection of relevant semantics), indexing relevant content segments for efficient retrieval [42, 47], and providing a powerful user interface [37] that enables different user groups—both experts and non-experts—to perform content search in a simple, efficient, and effective way [46].

Evaluating interactive video retrieval systems is a challenge in its own right. One way is to perform a user study for a particular system, in which many subjects must solve specific search queries for a specific dataset while their time and interaction with the system is monitored and analyzed. Unfortunately, such studies are often performed with private datasets and/or with settings that cannot be recreated by others (e.g., different task descriptions, different devices, different light conditions). Consequently, such results are hard to compare to other studies. User simulations are an alternative evaluation method that are much easier to perform and less time consuming. However, they can only approximate a system's performance, which is often highly dependent on a human user, especially for interactive video retrieval. A third way involves evaluation campaigns, such as TRECVID [4], MediaEval [17], or the Video Browser Showdown (VBS) [8, 35]. This way offers a good alternative to the former two and makes the comparison of different systems more reliable. In this case, several users perform the same tasks, at the same time, on the same dataset and with the same setup (in terms of task description, task assessment, infrastructure, etc.). Such

an evaluation is more user centric and provides a comparable way for evaluating interactive video retrieval systems.

In this article, we present the most recent installment of the VBS (VBS 2018), which was performed at the International Conference on Multimedia Modeling (MMM 2018) in Bangkok, Thailand, in February 2018. We start with a quick overview of the objectives and evaluation metrics of the VBS, describe the challenging tasks of 2018, and report the results achieved by the nine different retrieval systems that participated. Furthermore, we investigate user actions that led to correct submissions and discuss observed search patterns, based on the logs collected during the VBS competition.

2 VIDEO BROWSER SHOWDOWN

The VBS is an annual, international video retrieval competition, in which participating teams present their systems in a highly competitive setting in front of an audience. The VBS is organized as part of the MMM. During a limited time frame, the teams simultaneously solve a set of tasks on a known, large dataset. Given each task, teams submit their results to the VBS evaluation server, which displays feedback on a data projector. The server also summarizes the current scores and ranking, which contributes to the entertaining atmosphere during the competition. Overall, the VBS aims at fostering research in the interactive video retrieval domain, and it represents a unique evaluation platform for the various different approaches to video retrieval. We will use the following sections to summarize the evaluation objectives, scenarios, and metrics at VBS 2018.

2.1 Evaluation Objectives and Scenarios at VBS 2018

After a first period accompanied by various limitations, the VBS has focused on unconstrained interactive video retrieval given a large video collection. The only constraint that remains is the forbidden usage of cameras to collect query examples. Ever since VBS 2017, three types of tasks are being evaluated on the IACC.3 dataset with 600 hours of video content. This dataset is provided by the U.S. National Institute for Standards and Technology (NIST) as part of their TRECVID benchmarking initiative [4]. The three types of tasks correspond to *Target Visual*, *Target Textual*, and *Class Textual* categories, according to the taxonomy table presented in Lokoč et al. [21]. More specifically, *Target Visual/Textual* categories are represented by Visual/Textual known-item search (KIS) tasks, in which teams search for one particular 20-second scene after it has been played (visual) or described (textual) on the data projector. The *Class Textual* category is represented by ad hoc video search (AVS) tasks, in which teams search for multiple scenes corresponding to a more general textual description. In addition, novice sessions were re-introduced for visual KIS and AVS tasks at VBS 2018 to also take the usability of the participating tools into account. The novice users for the session are selected from the audience, and they receive a short training on how to operate the tool. Once a novice session starts, the teams are no longer allowed to support their novices.

2.2 Evaluation Metrics at VBS 2018

After solving a $task_i$, a $team_j$ receives score points based on a correct submission time $t_j^i \in R_0^+$ and/or the number of correct and incorrect submissions represented by sets C_j^i and I_j^i [21]. A submitted frame is considered correct if it stems from the searched video and its frame number lies within the defined interval delimiting the desired scene.

For both KIS tasks, the score formula depends on the submission time of the correct result t_j^i and the number of incorrect submissions $ws_j^i = |I_j^i|$. Each task has a time limit $t_L \in R^+$. Once the time for a task is up, the teams without a correct submission receive zero points. Otherwise, the score is defined as $f_{KIS}^i(t, ws) = \lceil \max(0, s_C + (100 - s_C) \cdot f_{TS}^i(t) - f_{WSP}(ws)) \rceil$, where s_C is a

time-independent reward for solving the task, $f^i_{TS}(t)$ is a decreasing function representing a time-dependent reward, and $f_{WSP}(ws)$ is a function that penalizes wrong submissions. At VBS 2018, s_C was set to 50, $f^i_{TS}(t) = (t^i_L - t)/t^i_L$, and $f_{WSP}(ws) = 10 \cdot ws$. The time limit was 5 minutes for visual KIS and AVS tasks, whereas for textual KIS tasks the time limit was 7 minutes due to their higher difficulty. Note that to make scoring fairer, late correct submissions extended the time limit by an additional 30 seconds. For more details, refer to Lokoč et al. [21].

The ad hoc search tasks focus on finding as many correct scenes as possible given a fixed time limit t_L. Hence, the score formula is designed as a precision/recall-based function, whereas the submission time $t < t_L$ of each submitted shot is not considered. The recall is related to the pool P_j of all correct submissions by all teams in task$_j$. Given sets C^i_j, I^i_j , and P_j, the score is defined as $f_{AVS}(C, I, P) = \lceil \frac{100 \cdot |C|}{|C|+|I|/2} \cdot \frac{|q(C)|}{|q(P)|} \rceil$, where q returns a set of video time ranges containing a correct submission from a given set (C or P) [21]. For VBS 2018, time ranges were defined by fixed static partitioning of each video by 3-minute non-overlapping intervals. This mechanism is considered to prevent from too easy recall gains for submitting many correct shots adjacent to one another in a single video. Even though an easy recall gain is still permitted for adjacent range boundary shots, using range-based recall also provides a higher recall gain for correct shots that lie further apart.

Taking into account novice sessions for visual KIS and AVS tasks, five search session categories were considered at VBS 2018. Each team collected score points $score^c_{team_j}$ in each category c as the sum of its scores from all tasks within that category. The final score for each team $score_{team_j} = \lceil \frac{1}{5} \cdot \sum_{c=1}^{5} \frac{100 \cdot score^c_{team_j}}{\max_{j=1..k}(score^c_{team_j})} \rceil$ is the normalized average over the per-category scores.

3 TASKS AT VBS 2018

The task selection process for VBS 2018 was similar to previous years; however, formulation of KIS text queries was slightly modified due to the novel incremental presentation of texts. Task selection for KIS tasks is a somewhat tedious and delicate procedure, as it needs to be ensured that a unique and unambiguous match exists for a given description. For VBS 2018, the selection process was performed by two people, one selecting a set of candidates and filtering them based on suitability, and a second person reviewing them. The textual descriptions have been drafted by one person and rephrased after discussion with the reviewer. The following are task selection criteria.

Known duplicates. We curate a list of known (partial) duplicates, which was created from matching metadata and file size, content-based matches, and partial duplicates detected during browsing and reviewing the dataset. Any segment of a video that is part of the identified set of known duplicates is excluded from the list of candidates.

Uniqueness inside same and similar content. We ensure that the segment of interest is unique within the clip (e.g., not a returning presenter) and in the collection (e.g., repeated elements in series). In segments containing multiple shots, this uniqueness may also be used to ensure that the particular sequence of shots is unique, if one of the shots has multiple visually similar occurrences.

Complexity of segment. The segment should roughly exhibit a duration of 20 seconds but consist of a limited number of shots so that it can still be memorized by participants after seeing it.

Describability. In addition to the complexity, the textual KIS tasks require that the entire segment can be unambiguously described with the limited amount of text that fits on the screen and can be quickly read by participants. In practice, this means that for the textual KIS tasks, we tend to select

Fig. 1. Visualization of all selected Visual/Textual KIS tasks (test session tasks are on the right).

segments with fewer shots and salient background or foreground objects that can be captured with a noun and few descriptive attributes. In the practical workflow, a set of KIS tasks is selected and then those suitable for textual description are filtered, leaving the rest for visual tasks.

One novelty in VBS 2018 is the incremental presentation of KIS text tasks over three steps (i.e., the initial description is expanded twice, 100- and 200-seconds into the task. All selected visual and textual KIS tasks used at VBS 2018 are visualized in Figure 1. The texts are formulated so as to put a complete but superficial description in the first text, then add further descriptive attributes or secondary objects to the description later. Table 1 lists the textual tasks used in VBS 2018. It has been rightly criticized by participants that task 6 does not fully meet the design criteria for incremental queries, as some of the information evolving over time in the segment is only revealed in the additional texts.

The AVS tasks are easier to select because there may be many correct matches. They should not be too specific so as to avoid selecting very rare examples. However, the issue of near duplicates also exists for AVS, although it rather poses a problem for evaluation and as to whether and how to count such matches in the final score. The AVS queries used for VBS 2018 are listed in Table 2. Some of the AVS scenarios were defined by the international benchmarking activity TRECVID, whereas others were specifically created for the VBS. Selected AVS tasks are shared between TRECVID and the VBS. For VBS 2018, only tasks shared with TRECVID were used.

4 TEAMS AND TOOLS AT VBS 2018

Nine teams participated in VBS 2018 with their tools, which are briefly presented in the following sections with a focus on the most effective aspects in the context of the competition. Each tool is presented with a screenshot depicting that tool's setting to solve one particular VBS textual expert KIS task (ID = 6, see Table 2). Note that the tool setting in the screenshot is not related to the settings used during the competition.

4.1 SIRET Team, Charles University, Czech Republic

The SIRET team presented a major revision of its interactive video retrieval tool [22, 23] with the key features summarized as follows.

Video preprocessing. All employed retrieval models operate only on a set of selected frames. The SIRET team used its own frame selection method, which detected 906, 202 frames overall in

Table 1. Textual KIS Tasks for the Competition (3, 6, 9, 12) and for the Test Session (T21-T30)

ID	Query
3	A dark skinned man with a gray hat standing under a curved roof and talks into a microphone. His name is displayed at the beginning ... in a yellow box. He is called ... and wears a black shirt ... The roof is white. Houses and trees are visible behind the man on the left.
6	An Asian woman in a hardware store testing red paint on the wall. Male shop assistant cuts a wooden board. The women opens many exhibited doors ... Video is slightly fast forward, and has orange text overlay at the top and then left ... Woman carries a large orange bag, with white writing "B&Q".
9	A bearded man with glasses in a storehouse. He walks between boxes to the right and puts something on a shelf ... The camera follows the man closely, showing only head and shoulders ... He has dark hair and wears a orange/brownish pullover. There shelfs have yellow and red supports.
12	A man with a black jacket and glasses sitting at a desk, a secretary standing left of him, handing him papers to sign ... A wooden door is on the left in the background, a phone is on the right side of the desk ... A calendar with months forming a white/gray checkerboard is on the wall behind the desk.
T21	A student films his classmates in a lecture hall. An Asian girl jumps in front of the camera. Another classmate hides behind the door ... A blackboard is visible, and then a handrail on the left and a fire extinguisher on the right ... At first we see a man in a dark pullover, the Asian girl wears a pink vest over a black T-shirt.
T22	A shot with many food packages and then juice bottles and a glass with ice cubes ... In the first shot, rice flakes are poured into the hand from a packet ... In the second shot, red juice is poured into the glass.
T23	A truck carrying new cars out of a factory. A worker moving a steel plate into a press and operating it ... First shot shows a concrete factory building, with windows on the right side and a few trees ... The worker stands on the right, wearing a blue shirt and a cap.
T24	Street view with palm trees. People in tribal costumes dance along the street holding hands ... Dancers are first sitting on the floor, then each picks up the next one ... The camera follows a group of three dancers, then they pick up a fourth one.
T25	Shots of a silver toaster, a table with three glasses, and close-up of an electric can opener ... Two breads being removed from the toaster and put into a drawer ... Someone pouring juice from an orange container into the middle glass, greenish table cloth and a yellow napkin in front of a curtain.
T26	A cartoon sailboat seen from front right, sails moving, then sails roll up and the boat is towed to an ice shell ... The ship has white sails, and the bow wave is shown ... There are some dark clouds in the upper left corner.
T27	Dark close-up shots of turntables, view of a singer from below and side view of a guitar player ... Behind the guitar player there is a screen showing the singer ... A hand is visible over the turntable, then the hand adjusts mixer controls.
T28	Black/white shots of a large ship with the letters ARMAS on it going from left to right ... The background is just sea and sky ... The camera is slightly tilted, which makes the impression the ship goes "uphill".
T29	Shots of a factory hall from above. Workers transporting gravel with wheelbarrows. Other workers putting steel bars in place ... The hall has Cooperativa Agraria written in red letters on the roof ... There are 1950s style American cars and trucks visible in one shot.
T30	Indoor shots of a group of barefoot dancers in traditional costumes made of straw. In the right hand they have a rattle ... Sequence starts with a closeup of the dancers' feet ... An audience is standing around them.

Table 2. AVS Tasks Selected From the TRECVID List of Tasks

VBS ID	TRECVID ID	Users	Find shots of...
2	531	Experts	one or more people eating food at a table indoors
4	542	Experts	at least two planes both visible
7	551	Experts	a person riding a horse including horse-drawn carts
10	539	Experts	an adult person running in a city street
14	540	Novices	vegetables and/or fruits
16	547	Novices	a person with a gun visible
18	548	Novices	a chef or cook in a kitchen
20	557	Novices	a person holding, throwing, or playing with a balloon

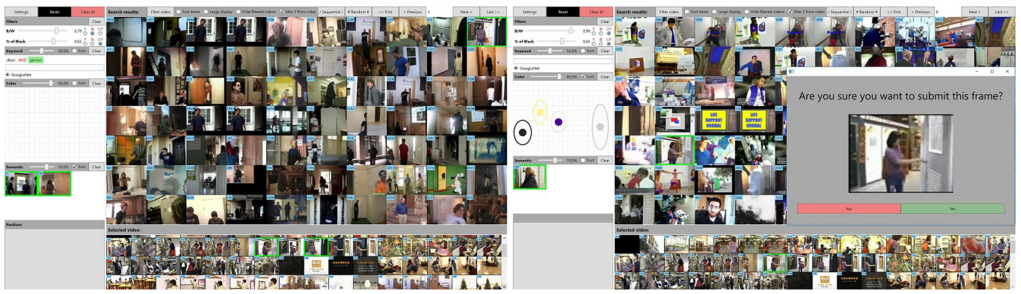

Fig. 2. On the left, the SIRET team's tool with the keyword query "door AND person" to filter 50% of all frames and two selected example frames to filter 70% of all frames. The intersection of the unfiltered frames is sorted by the relevance to frame examples. A frame from the searched video is on the second page of the results list (upper grid), showing a woman in a store. The summary of the video for inspection is below. The right figure shows an example of using a color sketch to solve the task in case of visual KIS.

the dataset. The method used the Euclidean distance over vectors obtained from RGB values in frame thumbnails (initially uniformly densely sampled). For each selected frame, the tool considered automatically extracted annotations (by a retrained GoogLeNet network [43] and an external annotation service [3]), color signatures (interpolation thumbnails), and activation features from the last pooling layer of the original GoogLeNet capturing semantic information in frames.

Retrieval models. Given the extracted frame representations and as detailed in Lokoč et al. [22, 23], the tool comprises a keyword-based ranking model (incorporating inverted files), ranking based on the cosine similarity on the activation features for content-based similarity search (including multi-query retrieval), and a model for simple color sketch retrieval. The color sketch consists of a set of colored ellipses covering region/pixels of interest, with an additional ALL/ANY specification.[1]

Tool interface. The interface of the tool (Figure 2) is organized into a query formulation panel on the left and result presentation grids in the remaining area. Users can provide few filters (e.g., black and white frames or pixel intensity filters), a limited set of keywords separated by AND/OR (with a prompt help, extended by hypernyms), color sketch consisting of ellipses (enabling easy interactive modifications), and a selected set of example frames for similarity search. Each model separately enables filtering of less similar frames and sorting with respect to the query. The results

[1] ALL/ANY option specifies whether all pixels are required or any pixel is required to have the specified color.

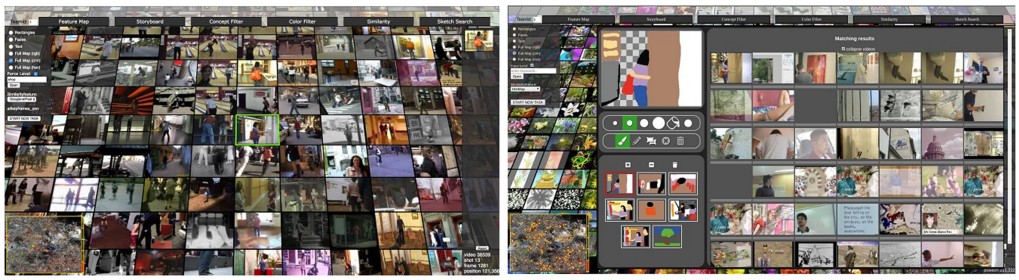

Fig. 3. The diveXplore system of ITEC1 provides a flexible feature map search with many additional filters. The sketch search interface of diveXplore allows users to perform image retrieval by simply drawing a sketch.

of each query are presented in a regular grid. Once a user selects a frame in the grid, all frames from the same video are displayed below. Using the mouse wheel over a displayed frame, users can play/inspect preceding/following frames in the corresponding video.

4.2 ITEC1 Team, Klagenfurt University, Austria

The ITEC1 team built on the so-called diveXplore system for VBS 2018 (Figure 3), which has been used in this competition for the second year in a row [27, 38]. The general idea of this system is to provide many different content search features that support several different search scenarios (query-by-browsing, text, filtering, example) [36, 37]. In particular, diveXplore is based on *browsable feature maps*, which allow to navigate in keyframes of shots that are arranged by some similarity (e.g., semantic similarity or visual similarity), inspired by the idea of Barthel et al. [6, 7]. The keyframes are arranged by a hierarchical self-organizing map [15] using color histograms as a similarity feature for visual presentation. However, the content of these maps varies: although a few maps contain all keyframes of all shots in the collection, there are several hundreds of maps containing content representing semantic classes recognized by Convolutional Neural Networks (CNN). In our current version, we use the classification by AlexNet [16], GoogLeNet [43], and the Inception-BN network [12]. All networks were trained with the classes from ImageNet [9]. The system also provides a *text-based search for available feature maps* containing the searched concept. In addition to the feature maps, however, the system also supports search by *color filtering*, by *concept-search*, and by *similarity search* based on a given example image. This image could be an existing keyframe obtained by browsing. The similarity search uses CNN features, such as weights of the last fully connected layer in the GoogLeNet architecture with the Manhattan distance.

4.3 ITEC2 Team, Klagenfurt University, Austria

Focusing on hand-drawn sketch-based search, the ITEC2 team extended the aforementioned diveXplore system by an additional view that allows a user to conduct similarity search on the basis of rudimentary drawings using a reduced color palette.

Video preprocessing. As the sketching functionality is integrated into the diveXplore system, it operates on the custom shot segmentation provided by the system. Specifically, it utilizes a manually designed color histogram descriptor, termed *HistMap* [18], which maps image regions to 18 bin color histograms[2] and is calculated in a sliding window fashion over an entire frame. This descriptor is precomputed for all available shot keyframes to enable fast online retrieval.

[2]HistMap considers primary colors, black and white, and some additional tones.

Retrieval models. Despite precomputing HistMap, acceptable runtime performance requires additional optimization measures: fast descriptor lookup is achieved via pivot table indexing—a feature that, as described in Primus et al. [27], is already part of diveXplore's similarity search mechanism and has been adapted to accommodate HistMap. Therefore, a single sketch query consists of calculating HistMap for an input sketch and utilizing pivot table lookup for retrieving a configurable amount of closest matches.

Tool interface. Figure 3 depicts the current interface of diveXplore's sketch search feature, demonstrating the retrieval of the aforementioned scene of a lady in a store leaving through a door by query by sketching. Since the query was given in a textual format, it is inconceivable to attempt finding the desired shot by sketching without any detailed information, as there are of course many possibilities to draw such a scene. Evidently, the example in Figure 3 is constructed after knowing what the scene looks like, which constitutes the typical use case for sketch-based querying, which is much more helpful for visual KIS. The view itself is partitioned into a drawing section and an area for matching results. Although the drawing section offers basic coloring tools, utilities for manipulating painted objects, and a sketch history, the results area lists matching shots from most to least similar and is updated with every action, so as to provide constant feedback to the user. Considering that the matching results list is presented in a format common among all diveXplore's interfaces, users may switch to different system features at any point on the basis of said results. For example, a specific shot can easily be located on the feature map, used for listing all shots of a corresponding video, or further analyzed by applying similarity search.

4.4 HTW Team, Visual Computing Group Of HTW Berlin, Germany

The HTW team extended their graph-based video search system [6, 7] with the possibility to search for 20,000 common keywords.

Video preprocessing. Initially, two frames per second were extracted from all clips. A ResNet was used to calculate visual feature vectors [11]. The 2,048 activations before the fully connected layer were L1-normalized followed by a PCA compression to 64 dimensions stored as bytes. Scene changes were detected by analyzing the changes of the feature vectors. Subsequently, a hierarchical image graph was constructed such that similar images/frames could be retrieved very quickly by traversing the edges of the graph. To produce visual feature vectors for keyword search, 1 million images from the Pixabay stock photo agency were used. To cover the great keyword variability, feature vectors were computed from up to 4,096 images per keyword. Each keyword is represented by 10 feature vectors, which are the cluster centers determined with the Generalized Lloyd algorithm.

Retrieval models. Queries can be started with sketches, example images, or single keywords. For a query, 4,096 result images are retrieved. On the one hand, this number turned out to be large enough to cover many variations of a search or a keyword concept. On the other hand, this number is not too large and can still be perceived on a three-level image pyramid with 16 x 16 images on the top level. Using a modified version of the self-sorting map algorithm [5], 4,096 images can be sorted in less than 250ms. The user can explore the search results by dragging and zooming the image pyramid. If an image close to the desired query was found, zooming in will display more related images. At the lowest pyramid level, this is achieved by "leaving the image pyramid" and switching to a mode, in which similar images are retrieved from the precomputed image graph.

Tool interface. On the right side of Figure 4, the HTW interface is shown. In the left part of the interface, the user can choose to either see the graph of all scenes, all scenes of a clip, or the search results displayed as a visually sorted image pyramid. In the middle, there is a scrollable results list. In the right part, a keyword search can be started. A query by example can be initiated by double

Fig. 4. The sequence of the search for a woman at a door. On the left, the top of the result pyramid of a "door" keyword search is shown. Zooming in on the image of the man at the door is shown in the middle. On the right, a similarity search was started with the second image to the left of the previous image.

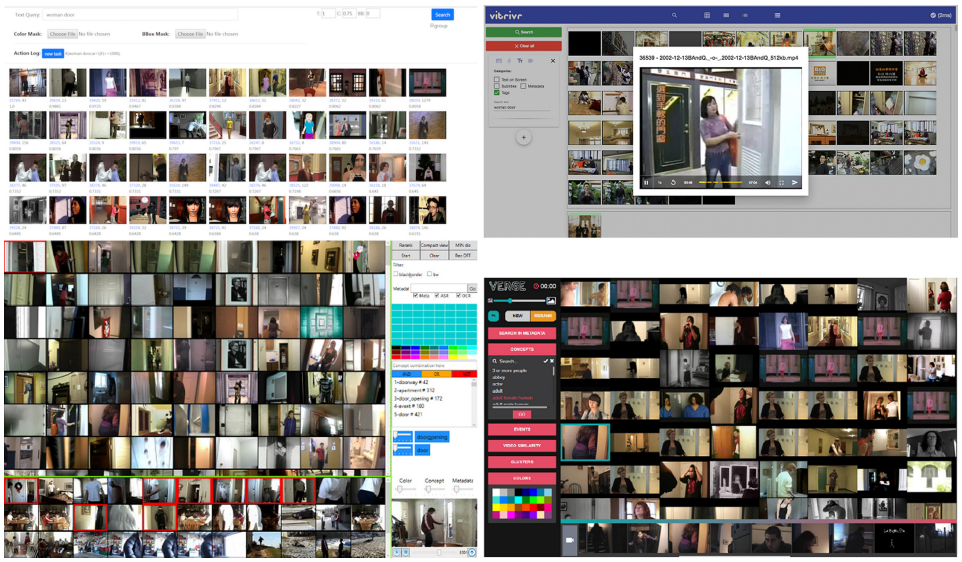

Fig. 5. At top left, the NECTEC team's tool with the text query "woman door" is shown. At bottom left, the VIREO team's tool with the concept query "door" and "opening door" is shown. The candidate frames are indicated by the red border. At top right, the VITRIVR user interface with a textual query for the tags "door" and "woman" is shown. The resulting keyframes are displayed in a grid view and sorted by relevance, which is also indicated by the green background color. At bottom right, the screenshot of the VERGE video retrieval engine is shown. Results from concept query "doorway" are re-ranked by concept query "adult female human." The image of a woman exiting through a door is highlighted to use it in a visual similarity query or to search in an entire video, presented at the bottom.

clicking a frame anywhere in the tool or in the video player in the lower right corner. It is possible to draw a sketch and to modify the example frame or the drawing. During the competition, it turned out to be a major drawback that a combined keyword search was not possible.

4.5 NECTEC Team, National Electronics and Computer Technology Center, Thailand

The NECTEC team presented a video retrieval system as an interactive browsing tool (see the top left screenshot in Figure 5), with which a video can be retrieved by either textual or visual queries [34]. The key features of this system are described as follows.

Video preprocessing. Each video sequence was extracted into keyframes derived from shot boundaries. NECTEC used the 335,944 keyframes provided with the dataset. The concept labels, color feature vectors, and object spatial feature vectors were extracted for each keyframe. Other additional data, such as metadata, OCR, and ASR, were not used in the indexing system. For concept labels, NECTEC used a CNN to extract objects, scenes, and image captions. The objects were recognized and localized using Faster R-CNN trained on the Open Images dataset [30] with 545 object classes. Scenes were extracted using Alexnet trained on the Place365 dataset [48]. An im2txt model [45] was used to generate image captions. The extracted labels were indexed using Elasticsearch [1] at a keyframe level. For visual query retrieval, each processed frame was converted into a color mask using color quantization in HSV color space, and object localization from Faster R-CNN [30] was extracted into a bounding box binary mask. The binary mask of each color and each object bounding box was resized to 16 x 16, which was then converted to a color feature vector and a spatial feature vector.

Retrieval models. For text-based retrieval, the images related to the search terms were ranked based on the Term Frequency/Inverse Document Frequency (TF/IDF) scoring model [28]. For fast sketch-based retrieval, the color and spatial feature vectors were hashed using locality sensitive hashes [2] and stored in Redis. The ranking was based on the cosine distance of the hashes.

Tool interface. NECTEC's tool has a simple interface, in which a user can search by either entering a keyword or uploading a sketch image, in which a user roughly sketches a color image or draws bounding boxes of a particular object. The sketched bounding box defines object location and the bounding box's color defines the object's class. Currently, the system only supports the top 10 object labels from 545 object classes. These identified object labels are represented by 10 different colors. A combined query can be performed with adjustable weights. Additionally, a user can switch between a display setting for viewing the ranked frames separately or grouped by video.

4.6 VIREO Team, City University of Hong Kong, China

In 2018, VIREO completely replaced its previous video retrieval tool [24] with a brand new version [26] focusing on three retrieval approaches: querying using a new color sketch retrieval model, keyword search in the provided metadata, and retrieval using automatically detected concepts.

Video preprocessing. For the color sketch–based retrieval, the colors of each cell (considering a fixed grid) in the video frame are calculated and stored in representations for individual frames and entire shots. For metadata, all video names, video descriptions, speech (provided by TRECVID), and on-screen texts (extracted using TesseractOCR [41]) are indexed on a video level using Lucene [25]. For concept-based retrieval, the previous concept bank [24] is upgraded to 14K concepts and the concept features are extracted for all of the shots. In addition, the ResNet50 pool 5 layer feature map is extracted and compacted with PCA for each shot keyframe and used for relevance feedback.

Retrieval models. The color sketch retrieval allows for similarity search using the Euclidean distance for two types of representations: keyframe based and shot based. In addition, a new color sketch recommendation module provides the user with prior knowledge about the color distribution of the dataset. For concept-based retrieval, the user can pick supported concepts from the concept bank and use AND, OR, NOT operators to combine them. Meanwhile, the related concept list helps the user to add relevant concepts to the query or filter out irrelevant ones. Additionally, free text retrieval can be performed for video metadata. Retrieval on the shot level is done by combining and weighting the results of the aforementioned three models. The top K final results are displayed as a ranked list for browsing. In the browsing phase, the user can pick positive examples and utilize a relevance feedback function to get semantically similar results.

Tool interface. The interface of the tool (see the bottom left screenshot in Figure 5) is divided into three panels: the right panel is used for query formulation, the top left panel shows the candidate list, and the bottom left panel displays the temporal video context of the selected item in the candidate list. In the query formulation panel, the interfaces for color sketch and text-based retrieval are displayed, including both the query input and the recommendation modules. The user can also use filters (e.g., black and white or black-framed frames) to specify the results and set weights for the partial results of the individual retrieval models. The tool provides two options to present the candidate list: shot based and video based. For the video-based option, all candidate shots that belong to the same video are grouped and presented as a dynamic image.

4.7 VITRIVR Team, University of Basel, Switzerland

The VITRIVR team competed with its equally named multimedia retrieval stack. The stack consists of three tiers: the ADAM$_{pro}$ storage engine [10], the Cineast feature extraction and retrieval engine [32], and the user interface. The architecture is further elaborated on in Rossetto et al. [31, 33]. The functionality of VITRIVR can be summarized as follows.

Video preprocessing. The videos in VITRIVR are organized around segments that, in the case of the VBS, were derived from the shot boundaries that came with the collection. In most cases, features are generated for a segment, and every type of feature maps to a separate entity in the persistence layer. For text retrieval, OCR data was extracted and concept labels were generated using Google Vision API. Moreover, the ASR information accompanying the dataset was imported. All textual data is stored in an Apache Solr index, which is part of the ADAM$_{pro}$ database. Furthermore, a multitude of edge, color, and interest point–based descriptors were generated to facilitate similarity search.

Retrieval models. VITRIVR allows for similarity search, based on either sketches (QbS) or image examples (QbE), and for (full-)text search in different domains, such as OCR or concept tags. The different query modes can be combined seamlessly. Similarity search is facilitated by different types of features, including local and global color descriptors, edge descriptors, and interest point descriptors like SURF and HOG. Depending on the type of query, different features are being employed. Retrieval involves a per-feature kNN lookup using a configurable distance measure. The top K results are selected and the per-feature results are subsequently merged in a two-step fusion process. This allows for flexible adjustment of the features that should be used, as well as offline re-ranking of the results.

Tool interface. The newly designed, Web-based user interface assists users in formulating queries that involve multiple modalities, including but not limited to visual QbE/QbS and textual queries. This can be done by setting up different query containers on the left-hand side of the UI (see the top right screenshot in Figure 5). The UI supports different types of results views, of which the grid view is showcased in Figure 5. Other results views include grouping segments by video. In addition, the UI offers different convenience features, such as loading neighboring segments, using existing segments for similarity search, or marking segments using colored tags.

4.8 VERGE Team, Information Technologies Institute, CERTH, Greece

The VERGE tool underwent significant changes this year (as compared to the 2017 version), which include the introduction of a novel graphical user interface (see the bottom right screenshot in Figure 5) with a new look and feel, and several new functionalities. In the following, we present the main components of VERGE.

Video analysis. The data layer for all retrieval modules comprises the 335,944 keyframes that were provided for the automatic AVS. Each keyframe is represented using the output of the last pooling layer of the GoogLeNet [43] deep neural network. Moreover, keyframes are indexed based on 1,000 ImageNet concepts, 345 TRECVID SIN concepts, 500 event-related concepts, and 205 place-related concepts using DCNN networks. Finally, the MPEG-7 Color Layout descriptor is used for capturing exclusively the color information of each image. Apart from the keyframe-based representations, several video-based representations have been developed that aim at organizing the dataset at the video level. Such representations include the exploitation of the video text metadata (i.e. its title, description, and tags) and the top K concepts of its video keyframes.

Retrieval models. Given the aforementioned representations, the VERGE tool allows for keyframe-based visual similarity search and advanced concept-based search that involves considering multiple concepts in a sequential or parallel manner. Moreover, automatic translation of free text to concepts is realized by using the pool of the concepts existing in the system. The color-based representation of the keyframes is used for color-based clustering in a set of 24 predefined color clusters. Retrieval on a video level is achieved either by using the video text representations or by combining the video metadata with the top K video concepts. Finally, the videos are clustered by considering the visual DCNN-based keyframe representations and by applying community detection. An additional functionality supports re-ranking of the results returned from a previous query, using any of the preceding models on the top N keyframes.

Tool interface. The graphical user interface of VERGE (see the bottom right screenshot in Figure 5) is designed to offer the user an intuitive experience when searching for an image or a video. Its main characteristic is that results are always displayed as image shots in a grid view, sorted by the retrieval scores, whereas all retrieval modalities are shown in a dashboard menu on the left. Various search capabilities, such as *Search in Metadata*, *Concepts*, *Events*, *Video Similarity*, *Clusters*, and *Colors*, are available as sliding boxes inside the menu. A valuable addition is a switch that toggles between "*NEW*" and "*RERANK*" (see Figure 5) and defines whether search modules should retrieve fresh results or re-rank the shots that have already been retrieved. Apart from the menu, hovering on a single shot reveals the *Visual Similarity* modality and clicking on the shot displays a film strip on the bottom of the screen with the complete set of video shots to which this frame belongs.

4.9 VNU Team, Vietnam National University, Vietnam

The VNU team presented a semantic concept-based video browsing system [44], focusing on spatial information of objects and action concepts. Several approaches to extract semantics were used, namely the YOLO detector [29], a high-saliency object detector based on DHSNet [20] in connection with VGG-16 net [40], scene attributes [49], relationships between objects [14], and provided video metadata. Color-based search was employed as well. Although the spatial information should further improve the retrieval effectiveness, the tool solved just e KIS tasks out of 22 (including the testing session) and also did not reach a high score in the AVS tasks. One of the hypotheses is that the employed semantic detection networks are still not effective enough for these types of tasks and videos with low resolution. However, given the only very brief logs, we do not have enough data to further investigate this hypothesis.

5 VBS 2018 EVALUATION AND RESULTS

Given the competition objectives, rules, selected tasks, and competing teams/tools that have been introduced in the previous sections, in this section we present and analyze the results of the seventh iteration of the VBS. The evaluation is based on data collected over 2 days. On the first day, a test

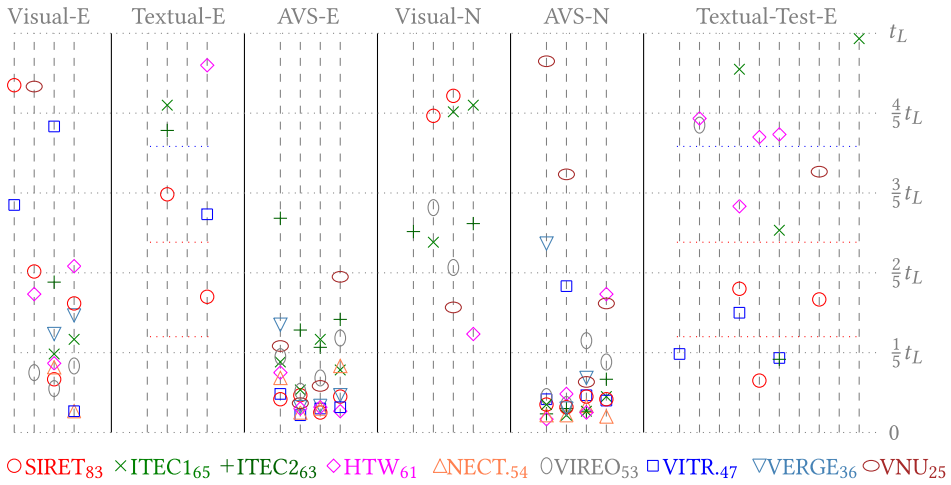

Fig. 6. Time elapsed until the first correct submission of each scoring team at VBS 2018 in KIS and AVS tasks (E = expert, N = novice). The time limit t_L was set to 5 minutes, except for textual KIS tasks where t_L was set to 7 minutes. The short blue horizontal lines indicate the 5-minute limit in both textual columns. Short red horizontal lines indicate the times at which the textual descriptions were extended. The overall VBS score is next to the team label (the test session score is not counted).

session was organized for the teams (expert users) consisting of 10 textual KIS tasks, T21-T30. The results of the test session were not considered for the final scoring. On the second day, the main VBS event—consisting of expert and novice sessions—took place during the welcome reception of MMM 2018 in front of an audience. First, the expert sessions were held for tasks 1-12, interleaving visual KIS, textual KIS, and AVS tasks. After the expert sessions, selected novice users competed in tasks 13-20, again interleaving visual KIS and AVS tasks. All tasks and results were presented using the VBS evaluation server, where all of the submissions were collected as well. Each submission received for a task consisted of a team ID, a shot/frame number, and an interaction log record (for log analysis, see Section 6). In addition, the VBS server appended the time that had passed and updated the team score for each correct submission. The analysis presented in this section focuses on the success rate and time required for KIS tasks, precision/recall reached in AVS tasks, and scoring of the teams.

5.1 Correct KIS and First AVS Submissions

In the first part of our analysis, we focus on the success rate and time at which (first) correct submissions arrived. A summary of the results for all KIS sessions is given in Figure 6. Each task in each of the categories is depicted as a dashed vertical time line, whereas solid lines delimit task categories and sessions. The correct submission of each tool in a given task is presented at the position corresponding to the time elapsed, where the bottom of the dashed line represents the start of the task. Please note that the dashed lines for textual KIS tasks depict 7-minute intervals, whereas the dashed lines for remaining task types depict 5-minute intervals. Therefore, a dotted blue horizontal line is employed to show the 5-minute mark in textual KIS task columns. For a comparison of the task complexity, the figure also shows the first correct submissions of each tool for both AVS sessions. The figure reveals several remarkable observations.

Textual KIS difficulty. Textual KIS tasks are still difficult to solve in a limited time frame given 600 hours of video. Five of 14 tasks were not solved by any of the competing tools, even though the initial time limit t_L was set to 7 minutes (for t_L only 5 minutes, seven tasks would have

remained unsolved). According to our experience, one of the main reasons is that often users have difficulties translating the textual description to a query, given just a few provided sentences. As a consequence, search initialization and browsing may lead to a wrong direction. Out of 14×9 possible correct submissions, only 22 (17.5%) were received. Except for the NECTEC and VERGE teams, all teams were able to solve at least one task. It contrasts with the visual KIS category tasks solved by expert users, where each team solved at least one task and the number of received correct submissions was 21 (58.3%) for four tasks and the shorter time limit.

Complexity of visual KIS. The complexity of KIS tasks varies even in the visual KIS task category with an ideal task specification. If the observed target scene is dark, blurry, or featureless, or if the scene is one of many similar ones in the dataset, it is more difficult to find the scene quickly. In connection with just a few evaluated tasks, it is not possible to draw significant conclusions for comparing expert and novice users. With that in mind, we observe for the visual KIS category that the expert users provided 21 (58.3%) correct submissions in 99 seconds on average, whereas the novice users provided just 11 (30.6%) correct submissions in 172 seconds on average. This observation corresponds to a natural expectation that experienced users should perform better. Interestingly, for both ITEC tools, the novice users found more correct submissions than the expert users.

AVS tasks. For TRECVID AVS textual topics with many relevant scenes in the IACC.3 dataset, it posed no problem to (quickly) find a first correct result for most of the tools. Only in one case (task ID 20, novice users, VERGE tool) was there no correct submission in the 5 minutes. The average first correct submission time for AVS tasks was significantly lower than the average correct submission times for visual or textual KIS tasks. These observations underline the key difference between finding any and finding one particular scene.

The score of the teams is affected also by the number of wrong submissions (i.e., users found a visually similar but different scene). Considering only the solved KIS tasks during the competition, the SIRET team had five wrong submissions in task ID 1 and one wrong submission in task ID 6; the VITRIVR team had one wrong submission in task ID 1; and the ITEC1 team had one wrong submission in tasks ID 6, ID 17, and ID 19. The five wrong submissions by the SIRET team were caused by a difficulty to recognize the one desired shot in the found correct video depicting a dark scene with very similar visual information.

5.2 AVS Task Precision and Recall

Table 3 presents details about correct and incorrect submissions as well as distinct submissions, such as frames that do not lie in the same, temporal range, and score per team and task, complemented by overall task statistics and average recall and precision values per team. As opposed to the log analysis, only one submission per shot and team is considered in this section. Overall, the server received 2,780 submissions across the eight AVS tasks, of which 2,288 (82%) were judged as correct, whereas 492 were considered incorrect by the live judges, who had to assess every single one of the 1,848 distinct submitted shots. The average precision ranged from 56% to 94% across tasks and from 69% to 84% across teams. The actual recall cannot be determined because only submitted shots were being assessed, so the absolute number of correct shots in the dataset is not known and not decidable with reasonable effort (TRECVID ground truth was not considered because it is not complete and therefore would arbitrarily distort the scoring). Therefore, the recall per team is computed with respect to the pool of found ranges from all teams. The values for the average recall over all tasks per team range from 11% (VNU) to 39% (NECTEC). The highest recall achieved within a single task is 49% (NECTEC in tasks 2 and 7, SIRET in task 4). This implies that there is a wide variety of shots found across the teams and only partial overlap. In the following, we describe a few selected observations in more detail.

Table 3. Summary of AVS Task Details

		Teams									Task Statistics	
		SIRET	ITEC1	ITEC2	HTW	NECTEC	VIREO	VITRIVR	VERGE	VNU		
Expert Tasks	2	7	6	13	16	32	19	33	9	10	217 (145/72) 104/51/47 71%	*Total submissions (correct/incorrect); Distinct shots/ranges/videos; Average precision*
		1	1	3	17	5	27	5	10	3		
		7	4	9	14	25	4	17	6	8		
		13	7	16	18	45	5	31	8	14		
	4	30	36	21	16	33	45	33	39	23	291 (276/15) 104/37/32 94%	
		0	0	2	2	1	2	6	0	2		
		18	15	10	7	13	11	8	14	11		
		49	41	26	18	35	29	20	38	28		
	7	35	28	36	33	75	47	26	20	10	339 (310/29) 159/45/38 92%	
		0	0	0	4	0	4	18	3	0		
		11	9	18	11	22	8	10	9	6		
		24	20	40	23	49	17	17	19	13		
	10	2	7	10	9	4	16	15	34	1	179 (98/81) 49/15/13 56%	
		0	4	6	14	13	1	17	20	6		
		2	3	6	7	4	3	5	7	1		
		13	16	31	26	10	19	21	36	2		
Novice Tasks	14	41	24	36	36	58	76	37	27	1	397 (336/61) 246/87/75 78%	
		16	5	10	8	5	7	3	4	3		
		32	21	21	27	33	8	23	14	1		
		31	22	21	28	36	9	25	15	0		
	16	36	52	38	35	77	60	12	83	5	505 (398/107) 284/95/82 77%	
		26	16	8	10	3	4	11	28	1		
		26	31	29	19	30	8	7	30	4		
		20	28	28	18	31	8	5	27	4		
	18	60	73	91	62	127	55	49	77	16	675 (610/65) 366/112/83 92%	
		13	13	10	4	17	0	8	0	0		
		45	38	43	37	52	11	18	27	15		
		36	31	36	32	44	10	15	24	13		
	20	25	9	26	5	22	20	4	0	4	177 (115/62) 89/38/31 56%	
		15	5	13	3	5	12	5	1	3		
		16	5	11	5	14	8	3	0	1		
		32	10	23	10	33	16	5	0	2		
∅*Recall*		31%	24%	31%	27%	39%	15%	21%	24%	11%		
∅*Prec.*		83%	82%	82%	73%	83%	84%	69%	70%	69%		

Note: Figures per task/team: Correct submissions (green), incorrect submissions (red), distinct found ranges (blue), task score (yellow).

Query ambiguity. We can observe that some tasks have a very high average precision (>90%), whereas others seem to be more ambiguous and thus have many more incorrect submissions. For task 10 ("adult person running in a city street"), only 55% of the submissions were considered correct (average precision across teams was 56%), presumably due to the very fuzzy target of a "city street" and the difficulty of deciding whether a person is or is not an adult and is actually running, walking, or standing. In this context, we also have to consider that due to time pressure in the competitive situation, users cannot thoroughly verify such details but rather rely on a quick first impression of a candidate shot. Another example is task 20 ("person holding, throwing, or playing with a balloon"), in which only 65% of the submissions were correct (average precision across teams was 56%), mainly due to confusion of a balloon with similar objects. However, task 4 ("at least two planes both visible") is very clear and therefore achieved an average precision of 94%.

Table 4. Normalized Scores of the Competing Tools in Different Task Categories and the Overall Tool Score

	Experts	Visual-E	Textual-E	AVS-E	Novices	Visual-N	AVS-N	Overall
1. SIRET	2	95	**100**	71	2	67	83	83
2. ITEC1	2	64	37	60	2	**100**	64	65
3. ITEC2	2	29	47	81	2	85	75	63
4. HTW	2	91	41	61	2	50	61	61
5. NECTEC	3	68	0	**100**	3	0	**100**	54
6. VIREO	1	**100**	0	50	1	86	30	53
7. VITRIVR	3	79	55	64	1	0	35	47
8. VERGE	1	62	0	72	1	0	46	36
9. VNU	1	21	0	41	1	48	14	25

Note: The number of users controlling the tools are presented as well.
Bold text indicates the best score in each task category.

Task difficulty. It seems that the novice tasks were "easier" than the expert tasks, considering that the average number of submissions per task is significantly higher (439 vs. 257) and also the overall percentage of correct submissions is slightly higher (83% vs. 81%). The task with the highest number of submissions is task 18 ("a chef or cook in a kitchen") with 675 submissions (on average, 75 per team), whereas the "hardest" task was task 20 ("person holding, throwing, or playing with a balloon") with only 177 submissions (about 20 per team). An interesting observation in this context is that although for most teams novices in total submitted many more shots (e.g., in the case of SIRET, 232 vs. 75), the opposite is true for VITRIVR and VNU (129 vs. 153 and 33 vs. 55, respectively). For the VITRIVR team, this was probably caused by only one novice user operating the tool.

Scoring. The new scoring function was designed to favor a larger variety of found shots compared to consecutive, non-distinct shots from the same video. This can be exemplified by a comparison between VIREO and ITEC1 in task 14. VIREO submitted 76 correct shots, but only from 8 distinct ranges (about 10 consecutive shots per range), although the overall pool includes at least 87 distinct ranges (recall 9%). However, ITEC1 only submitted 24 correct shots, but from 21 distinct ranges (recall 24%) and therefore scored 22 points compared to 9 points. Similar patterns can be observed across the entire competition, so finally VIREO only achieved an average recall of 15% and rather low scores, although the absolute number of correct shots is comparable to other teams in most tasks and the average precision of 84% outperforms all other teams. However, submitting a large number of correct shots from the same range still can be beneficial to compensate for incorrect submissions, as exemplified by VERGE compared to HTW in task 10: both teams found 7 of 15 distinct ranges, but VERGE submitted 34 correct shots (HTW only 9). Although VERGE also submitted a larger number of incorrect shots (20 vs. 14), the positive ratio between correct and incorrect submissions reduced the penalty and led to a considerably better score (36 vs. 26).

5.3 Overall Ranking of the Tools and Discussion

The max-normalized scores of each team in each session are presented in Table 4. We observe that the ranking reflects the first four tools scored in each session, whereas the remaining tools had zero points in at least one or even two sessions. The winning tool is from the SIRET team that focused on its own frame selection, automatic annotation, multi-modal query (re-)formulation, and asymmetric late fusion. The SIRET team reached a high score in expert KIS sessions and performed better than average in the remaining three sessions. The overall high score indicates a good versatility of the tool compared to the remaining participants. The tools controlled by the

two ITEC teams reached second and third place, of which the ITEC2 expert team focused more on hand-drawn sketches. The interesting observation is that the novice users performed better than the expert users (in these cases). The HTW team performed well in the expert visual KIS session, but the tool reached just an average score in the remaining sessions, which yielded the overall fourth place. However, considering its performance in the textual test expert session, the overall performance of the HTW tool could be considered on par with the ITEC tool. Both tools rely on browsing semantically organized 2D maps of images and use similar query initialization approaches. Fifth place was reached by the NECTEC team, who won both AVS sessions while only reaching zero points in the two KIS sessions. Their good performance in the AVS tasks, however, demonstrates the need for efficient interfaces to browse results and select many relevant items quickly. However, the number of users (three in both expert and novice NECTEC teams) seems to correlate with the overall score in generally formulated AVS tasks with many relevant shots. The overall score in AVS tasks contrasts with the VIREO tool controlled just by one person. Nevertheless, even with just one operator, the VIREO tool performed well in both visual KIS sessions (with the help of its new color sketching model and interface) and reached sixth place. The VITRIVR team (winner of VBS 2017) reached seventh place. After the expert sessions, the score of the team was probably negatively affected by having just one novice user. Surprisingly, the novice user was unable to solve any of the visual novice KIS tasks. The VERGE team solved only two KIS tasks in the visual expert session, which resulted in significant score losses and the overall eightth place. Despite the respected deep learning approaches employed by the VNU team, they only reached last place. In Section 6, the logged features used for solving the tasks are presented, if available.

To conclude the scoring and ranking, VBS 2018 had a clear winner that outperformed all other tools in terms of the overall score. However, the number of users seems to be an issue for future VBS events. Although a superior tool can neglect the effect of having more users, for similar tools and especially considering the importance of time, the number of users affects the overall score, as more users can investigate more search options and inspect result sets in parallel.

6 INTERACTION LOGGING: FIRST PROMISING ATTEMPT AT THE VBS

The results presented in Section 5 are only weakly coupled with the methods employed by the tools presented in Section 4. Therefore, the teams were asked to implement logging of a high-level sequence of (inter-)actions that were performed when solving a task. Owing to the failure of such attempts in previous VBS iterations, a highly simplified format was proposed. For example, if a user employs keyword search, browses few pages, then tries a color sketch, and browses on an image map, the system should log a high-level sequence "K; P; ...; P; C; B; ..., B;" and provide that sequence with a subsequent submission for the given task. Considering that each tool has its own set of query initialization, filtering, and browsing options, the vocabulary of actions was designed to consist of a generalized (mandatory) and a tool-specific (optional) part. The unified mandatory part is represented by a capital letter, corresponding to the type of action that was issued:

- K = using keyword search in automatically detected annotations (most teams use DCNNs), automatically detected concepts, or provided metadata (but not audio)
- A = using extracted audio data (usually searched by keywords as well)
- O = using optical character recognition (usually searched by keywords as well)
- C, E, M = ranking by color, edge, or motion sketches
- I = similarity search by an example image (from results or external server)
- F = explicit filtering using various attributes or actual dataset ordering (top K results)
- P = paging, visiting a next/previous page in the actual ordering, scrolling
- B = using a tool-specific browsing system (e.g., zoom in/out in a hierarchical image map)

—V = frames from a selected video are inspected (e.g., video summary or player)
—X = reset the interface and start searching from the scratch

In brackets, additional, brief, tool-specific details could be provided for each letter (e.g., the time of the action, used keyword, parameters of filtering) for further analysis and verification.

The simplicity of the logging resulted in a first successful attempt at collecting interaction logs during the VBS. Most teams implemented the logging at least to some extent. Only the HTW team did not implement the logging at all due to time constraints. The VNU team apparently had some problems with the implementation, as the submitted log records only contained the letter "K." Even though the collected logs are just a projection of all performed interactions, the records obtained during VBS 2018 can already be used to get a preliminary insight into what actions/features were used to solve a task and general (expected) search patterns discussed in the following sections. In addition, we present challenges that should be addressed in future installments of the VBS.

6.1 Employed Actions

Considering that the time at which an interaction was issued was not mandatory, we first present Table 5, which details a list of actions employed during the tasks by teams that had at least one submission (represented as "!"). The capital letters for action types are sorted based on their first occurrence in the log record for a given task, and the number of observed actions of the same type is presented in the upper index. Please note that implicitly issued actions, such as filters that were automatically applied after a keyword search, are not included. Owing to late and partially ambiguous specification, some teams did not implement the logging to the full extent. For example, the SIRET team did not log letter "V" for temporal context and video summary inspection, which was used quite often. Similarly, ITEC1 and ITEC2 did not log interaction with their video player, which can be opened from any part of the system to inspect the temporal context. This context proved to be very helpful in practice and is one of the reasons for the high diversity of shots found in AVS tasks as compared to some other teams. Moreover, they only logged entering the browsing map but not browsing around therein, as this would have bloated the log. These examples demonstrate that browsing took more actions than presented in Table 5. From the tool-specific details, we also observe that browsing actions (P, V, B) were interpreted differently by different teams, as each team relies on its own browsing interface.

Despite incomplete log records, however, the table reveals different query formulation complexities for successfully solved tasks. Whereas only a few tasks were solved by a single keyword, some tasks required many interactions and query reformulation. Different interaction complexity was observed even for one task (e.g., see visual KIS task 8 and VITRIVR compared to SIRET). Text-based approaches are most commonly used to initialize search in textual KIS tasks, whereas color-based search is often used to initialize the search in visual KIS tasks. As time elapses, teams also try query-by-image similarity search to proceed with the tasks. Most of the AVS log records show an expected search pattern, where keyword search is interleaved with browsing the list of candidates and subsequent submission attempts. In some cases, similarity or color search was used, probably to identify new clusters of searched frames.

6.2 Interaction Heat Map

Considering that teams SIRET, ITEC1, ITEC2, VITRIVR, VIREO, and VERGE also submitted timestamps alongside each logged action, we were able to analyze how the usage of those actions changed over the course of a task. The outcome of this analysis is depicted in Figure 7, where we summarize the interactions per task type. The contribution of the individual teams is presented in the Appendix. For the sake of simplicity, we aggregated the atomic actions introduced in Section 6

Table 5. Summary of Logged Actions Up to the Last Correct Submission of a User in a Task, Including Cardinalities Presented in the Upper Index

Task ID	VIREO	VITRIVR	ITEC1	ITEC2	SIRET	NECTEC	VERGE
KIS Visual (Expert)							
1 (V, E)		$A^1V^{11}K^2B^1!^1$ $A^6V^9B^5K^3!^1$			$C^{12}K^1I^3!^6$		
5 (V, E)	$C^6F^1!^1$				$K^2C^2P^7I^3!^1$		
8 (V, E)	$C^{10}K^1!^1$	$K^2C^3E^3B^{12}I^2V^3!^1$	$K^1F^1!^1$	$F^6B^1!^1$	$K^1!^1$	$K^1V^2P^1!^1$	$C^1I^1!^1$
11 (V, E)	$K^1!^1$	$K^1O^1!^1$	$K^1F^1!^1$		$C^4K^1I^1!^1$	$K^1V^2!^1$	$K^1!^1$
KIS Textual (Expert)							
3 (T, E)		$K^1B^6V^2!^1$					
6 (T, E)			$K^{10}F^2!^2$	$C^{11}K^2F^1!^1$	$K^3P^4C^4I^5!^2$ $K^3C^{11}P^{26}B^3I^1!^1$		$K^8C^1P^4!^6$
9 (T, E)				$C^9K^2F^2!^1$			
12 (T, E)	$K^3!^1$	$K^2V^9B^6!^1$			$K^1I^1!^1$		$K^3!^2$
AVS (Expert)							
2 (A, E)	$K^4!^{45}$	$K^1B^{16}!^7$ $K^4B^{62}V^3!^{29}$ $B^8!^4$	$K^1!^1$ $K^{11}B^1!^6$	$K^2I^1B^6F^4!^{10}$ $C^8K^3F^1!^4$	$K^2I^2P^3!^2$ $K^7P^{14}B^2I^1!^5$	$K^2V^7X^1P^{13}!^5$ $K^5P^{15}X^4!^{16}$ $K^2P^{19}X^1!^{17}$	$K^7P^1!^{16}$
4 (A, E)	$K^1!^{47}$	$K^8B^5V^{24}!^{25}$ $K^4V^{14}B^5!^{16}$ $K^1B^2!^5$	$K^2I^2!^{13}$ $K^3B^1!^{23}$	$K^4B^2!^{14}$ $C^3F^1!^9$	$K^1I^1P^{35}!^{26}$ $K^2P^7!^{13}$	$K^3V^7P^{13}!^{14}$ $K^3P^{17}X^2V^2!^{22}$ $K^4P^{15}X^3!^{10}$	$K^3P^2I^1!^{64}$
7 (A, E)	$K^1!^{51}$	$K^2B^{17}V^{19}!^{25}$ $K^4B^7V^6!^{27}$ $!^1$	$K^2F^1B^1!^{14}$ $K^4B^1!^{14}$	$K^3F^3B^3!^{15}$ $K^4F^1!^{21}$	$K^1P^3!^{18}$ $K^1P^8!^{35}$	$K^2V^3P^{20}!^{54}$ $K^2X^1P^{20}V^8!^{17}$ $K^3P^{12}X^2V^3!^{15}$	$K^1P^1!^{23}$
10 (A, E)	$K^{12}!^{17}$	$K^1!^{11}$ $K^9B^{28}V^7I^1!^{17}$ $B^6K^1!^{14}$	$K^5!^8$ $K^2B^1!^1$	$K^4B^2F^1!^7$ $K^{19}F^2!^7$	$K^2P^6!^2$	$K^5X^4P^9V^1!^2$ $K^4X^3P^7V^2!^6$ $K^3V^5X^1P^3!^2$	$K^2!^{57}$
KIS Visual (Novice)							
13 (V, N)				$K^3B^2C^1!^1$			$K^5P^3C^1!^1$
15 (V, N)	$F^2C^{45}K^2!^1$	$K^7C^1E^1X^1!^1$	$K^4!^1$		$C^{22}I^2P^{23}K^7!^1$		
17 (V, N)	$F^6K^2!^1$		$K^6!^1$ $K^2!^1$	$K^3I^2F^1B^2!^1$	$K^3C^{21}P^{17}I^3!^1$		
19 (V, N)			$K^4!^2$	$K^3F^1!^1$			
AVS (Novice)							
14 (A, N)	$K^2F^2!^{83}$	$K^{11}A^{11}B^1!^{52}$	$K^9I^1!^{18}$ $C^1K^4!^{10}$	$K^7I^1!^{30}$ $K^{10}I^2B^2!^{15}$	$K^2P^{40}!^{38}$ $K^6P^{15}!^{27}$	$K^{23}V^8P^{16}X^{22}!^{21}$ $K^7V^9P^{33}X^6!^{33}$ $K^{21}V^{16}P^{28}X^{19}!^{18}$	$K^3C^1P^1!^{37}$
16 (A, N)	$K^1!^{64}$	$K^6A^6B^1!^{23}$	$K^3!^{50}$ $K^3B^1!^{18}$	$K^1!^{24}$ $K^1!^{22}$	$K^2P^{30}!^{43}$ $K^4P^{14}!^{24}$	$K^{15}V^4P^{30}X^{14}!^{31}$ $K^4X^3P^{23}!^{59}$ $K^{13}V^2P^{23}X^{12}!^{22}$	$C^1K^1P^2!^{138}$
18 (A, N)	$K^1!^{55}$	$K^5A^5!^{62}$	$K^4!^{61}$ $K^2!^{25}$	$K^1B^1!^{71}$ $K^6!^{27}$	$K^2P^7!^{40}K^2P^7!^{59}$	$K^{11}V^7P^{20}X^{10}!^{52}$ $K^4V^2P^{40}X^3!^{49}$ $K^2V^1X^1P^{36}!^{78}$	$K^5P^1!^{101}$
20 (A, N)	$K^1F^1!^{31}$	$K^4O^4A^4B^3!^9$	$K^5I^1!^{13}$ $K^6B^2!^1$	$K^5!^{14}$ $K^6F^3!^{25}$	$K^4P^8!^{15}$ $K^4P^6I^1!^{41}$	$K^6V^8P^{33}X^5!^{17}$ $K^{11}V^{10}X^{10}P^{16}!^{10}$ $K^7V^2P^{11}X^6!^7$	$K^3!^1$

(Continued)

Table 5. Continued

Task ID	VIREO	VITRIVR	ITEC1	ITEC2	SIRET	NECTEC	VERGE
KIS Textual (Expert, test session)							
21 (T, E)		$K^1B^2!^1$					$K^1!^1$
22 (T, E)	$K^7!^1$	$K^{13}B^6!^1$				$K^3V^7P^7X^2!^1$ $K^2V^6P^5X^1!^1$	$K^3P^3I^2!^2$
23 (T, E)	$K^3!^2$				$K^{15}P^5C^7!^1$		$I^1X^1K^7P^1!^1$
24 (T, E)	$K^1!^{12}$	$K^4B^4!^2$ $K^1B^2!^1$	$K^{11}F^4B^1!^1$ $K^3F^3B^5!^1$	$K^6B^7F^{16}I^1!^6$ $K^{19}I^1B^2F^2!^3$	$K^1I^2!^2$ $K^2C^3!^1$	$K^3V^{22}P^{27}X^2!^5$ $K^2V^4P^4X^1!^1$	$K^4P^2I^1!^7$
25 (T, E)	$K^{17}F^1!^2$				$K^2!^1$		$K^2P^3!^1$
26 (T, E)	$F^4K^3!^4$	$K^3B^2!^1$	$K^5I^2B^1F^2!^2$ $K^3F^3B^1!^1$	$K^2!^1$	$K^1I^1!^2$		$I^3X^1K^7P^{11}!^2$
27 (T, E)					$K^4I^2P^{10}!^1$	$K^4X^3V^{13}P^{12}!^2$ $K^7V^{12}P^{13}X^6!^2$	$K^2I^4P^1!^5$
28 (T, E)		$O^4B^{24}K^4M^1!^2$			$K^1P^{20}I^3!^1$	$K^4V^{13}P^{20}X^3!^1$	$K^1C^1I^1!^3$
29 (T, E)	$K^5!^8$					$K^{10}V^{17}P^{20}X^9!^1$	
30 (T, E)	$K^8F^1!^1$	$K^1B^8!^1$	$K^6F^5B^4!^3$		$K^6I^4P^{25}!^1$	$K^{14}V^7P^{11}X^{13}!^1$ $K^7X^6V^{13}P^{13}!^2$	$K^6P^3!^2$

Note: The letters are sorted according to their first occurrence. Each cell row contains interactions submitted by one user. Available unsuccessful submissions are shown in red.

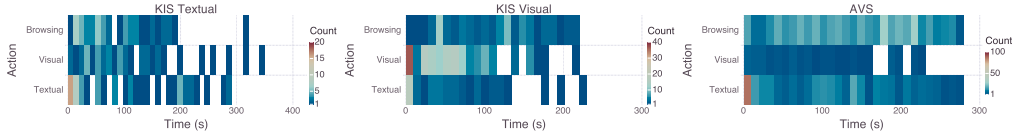

Fig. 7. Heat map of logged actions up to the last correct submission of a user in a task for the three different task types, aggregated across all teams and tasks of the respective type. For the sake of readability, actions were merged into three categories.

into categories as follows: Textual query (K, A, O), Visual query (C, E, M, I, F), and Browsing (P, B, V). Furthermore, all actions were aligned relative to the time of the first action in the submission, and only correct submissions were considered. Blank spots indicate the absence of any action, which can be attributed to fragmentary/incomplete logging and also the time spent during content inspection.

Even though the logs can only be considered to be a projection—that is, not all interactions were logged by all teams in a consistent fashion—we can still observe certain trends. As expected, textual queries dominate the initialization phase of the AVS and KIS Textual tasks, whereas visual query interactions are frequently used in the early phase of KIS Visual tasks. However, textual queries also seem to play an important role during KIS Visual tasks, even though they are slightly less dominant than their visual counterparts. The initialization phase lasts for 10 to 20 seconds and is followed by a phase in which query reformulation takes place. We can also observe a steady decline of interactions toward the end of a task, which is due to teams submitting correct results and therefore fulfilling the objective. Owing to its nature, this trend cannot be observed for AVS tasks.

The data offers some insight into what constitutes a successful query strategy. The presence of repeated query reformulation interactions during the KIS tasks indicate that, indeed, the interactive decisions of a human in the loop, accompanied by browsing, were behind most of the successful submissions for more difficult tasks. Additionally for the AVS tasks, we observe a high rate of query reformulation as the users tried to achieve high recall by trying different textual

queries. However, using just a single keyword query and subsequent browsing/paging proved to be sufficient to find a great amount of relevant segments, at least in the cases in which the automatic annotation was effective (e.g., horse riding). It also appears that users seem to have refrained from using visual queries as the AVS tasks approached their end. This makes sense insofar, because speed matters very much for AVS tasks, and losing time on formulating complex visual queries cannot be afforded.

6.3 Future Challenges and Lessons Learned

Collecting logging data to understand how users interact with their system during a competition like VBS seems to be a viable way to go. Nevertheless, an important challenge for future events is to better understand both the performance and usability of the systems [13]. To that end, logging can only be one piece of the puzzle. It further requires evaluation of more tasks and user questionnaires after the competition. It is also necessary to provide a clearer specification of the interaction logging format including mandatory timestamps for each action. Even with the current specification, it seems that there was too much leeway for interpretation and actions were not used consistently across teams. There must be also a more precise definition as to how frequently interactions should be logged. Finally, we must foresee a validation period prior to the next competition, during which logs submitted by the different teams are checked for their compliance with the intended structure. All of these aspects are important to better compare the complexity of tools and tasks. In addition, to better understand whether the winning team won due to a smart tool design or thanks to more effective ranking models, the VBS server should also evaluate ranked lists of query initialization results. At least for KIS with a known ground truth, this could be realized by a special session, reproduction of actions in logs, or tighter integration of participating systems with the evaluation server (e.g., by automatic forwarding of top K ranked results to the VBS server).

7 CONCLUSION

This article presents the settings and findings of the seventh VBS, in which nine teams competed in known-item and ad hoc search tasks over a large, common video collection. During the event, both expert and novice users were involved in the competition and influenced the score. The results confirm different complexities of tasks discussed in Lokoč et al. [21]. The interaction logs collected at the VBS for the first time reveal that in many cases, the teams had to rely on query reformulation and interactive browsing to solve a task. Considering that novel machine learning approaches effectively narrow the semantic gap, most queries were initialized by keyword search. The visual KIS tasks were often solved in combination with sketch-based retrieval. The overall winner—the SIRET tool—used multi-modal search relying on its own representative frame selection, automatic annotation using deep neural networks, position-color information in frames, and query-by-image provided by deep features. The models in combination with a user interface enabling simple query reformulations and visualization of resulting or selected video frames proved to be effective enough to win this iteration of the VBS. Second place was reached by the ITEC tool, integrating also a navigation in an image map and collaborative search. Surprisingly, the novice users focusing mostly on simple keyword search outperformed the expert ITEC users, who often tried leveraging advanced features of their tool. Hence, an effective incorporation of advanced browsing approaches represents an open challenge for future VBS events. Regarding VBS organization, we want to limit the number of users in each team and focus on more advanced approaches to better understand the effectiveness and usability of participating tools [13]. We also plan to change the presentation of visual tasks to prevent potentially unrealistic performance of sketch-based methods that profit from the fact that the scene is playing in an infinite loop.

APPENDIX

Figure 8 presents interaction heat maps for each task type and teams that logged time of actions. Please note that the source logs do not contain all interactions. Specifically, the SIRET tool did not log frequently used video inspection (letter V). The ITEC tool did not log interactions with the video player and logged just entering their browsing map (but not browsing actions). The VIREO tool also did not log video inspection and simple browsing implemented as scrolling of result lists.

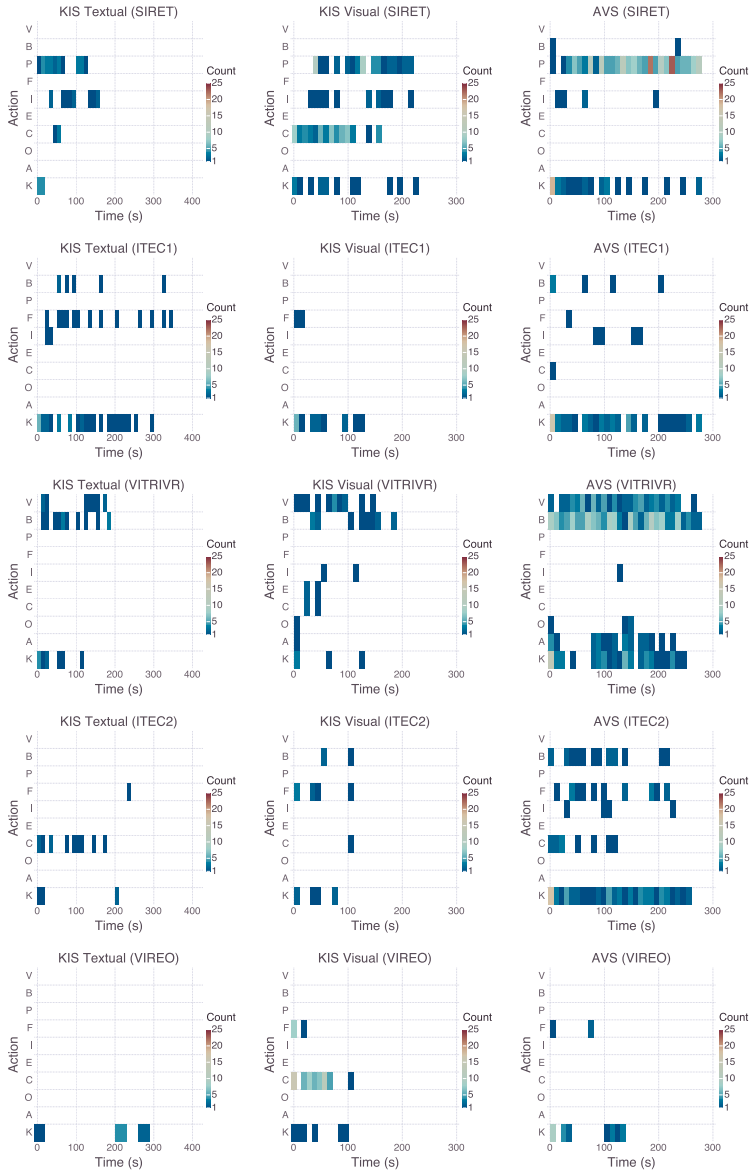

Fig. 8. Heat map of logged actions up to the last correct submission of a user in a task for all of the different task types and the different teams that logged the time of actions.

ACKNOWLEDGMENTS

This paper has been supported by Czech Science Foundation (GAČR) project Nr. 17-22224S. Parts of this work are supported by Universität Klagenfurt and Lakeside Labs GmbH, Klagenfurt, Austria and funding from the European Regional Development Fund and the Carinthian Economic Promotion Fund (KWF) under grant KWF 20214 u. 3520/26336/38165. Part of this research has received funding from the Horizon 2020 Research and Innovation Programme V4Design, under Grant Agreement No 779962. Parts of this work are supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (CityU 11250716). Work on *vitrivr* was partly supported by the CHIST-ERA project IMOTION with contributions from the Swiss National Science Foundation (SNSF, contract no. 20CH21_151571).

REFERENCES

[1] Elasticsearch: RESTful, Distributed Search & Analytics. Home Page. Retrieved March 30, 2018, from https://www.elastic.co/products/elasticsearch.

[2] NearPy. Home Page. Retrieved March 30, 2018, from https://github.com/pixelogik/NearPy.

[3] Giuseppe Amato, Fabrizio Falchi, Claudio Gennaro, and Fausto Rabitti. 2017. Searching and annotating 100M Images with YFCC100M-HNfc6 and MI-File. In *Proceedings of the 15th International Workshop on Content-Based Multimedia Indexing (CMBI'17)*. 26:1–26:4.

[4] George Awad, Asad Butt, Jonathan Fiscus, Martial Michel, David Joy, Wessel Kraaij, et al. 2017. TRECVID 2017: Evaluating ad-hoc and instance video search, events detection, video captioning and hyperlinking. In *Proceedings of the 17th AnnualTREC Video Retrieval Evaluation (TRECVID'17)*.

[5] Kai Uwe Barthel and Nico Hezel. 2018. Visually exploring millions of images using image maps and graphs. In *Big Data Analytics for Large-Scale Multimedia Search*, B. Huet, S. Vrochidis, and E. Chang (Eds.). John Wiley & Sons, New Jersey, 251–275.

[6] Kai Uwe Barthel, Nico Hezel, and Radek Mackowiak. 2015. Graph-based browsing for large video collections. In *MultiMedia Modeling*, X. He, S. Luo, D. Tao, C. Xu, J. Yang, and M. A. Hasan (Eds.). Springer International Publishing, Cham, Switzerland, 237–242.

[7] Kai Uwe Barthel, Nico Hezel, and Radek Mackowiak. 2016. Navigating a graph of scenes for exploring large video collections. In *MultiMedia Modeling*, Q. Tian, N. Sebe, G.-J. Qi, B. Huet, R. Hong, and X. Liu (Eds.). Springer International Publishing, Cham, Switzerland, 418–423.

[8] Claudiu Cobârzan, Klaus Schoeffmann, Werner Bailer, Wolfgang Hürst, Adam Blažek, Jakub Lokoč, et al. 2017. Interactive video search tools: A detailed analysis of the Video Browser Showdown 2015. *Multimedia Tools and Applications* 76, 4, 5539–5571.

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'09)*. IEEE, Los Alamitos, CA, 248–255.

[10] Ivan Giangreco and Heiko Schuldt. 2016. ADAM pro: Database support for big multimedia retrieval. *Datenbank-Spektrum* 16, 1, 17–26.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Identity mappings in deep residual networks. arXiv:1603.05027. http://arxiv.org/abs/1603.05027.

[12] Sergey Ioffe and Christian Szegedy. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of the International Conference on Machine Learning*. 448–456.

[13] Melody Y. Ivory and Marti A. Hearst. 2001. The state of the art in automating usability evaluation of user interfaces. *ACM Computing Surveys* 33, 4, 470–516. DOI: http://dx.doi.org/10.1145/503112.503114

[14] Justin Johnson, Andrej Karpathy, and Fei-Fei Li. 2015. DenseCap: Fully convolutional localization networks for dense captioning. arXiv:1511.07571. http://arxiv.org/abs/1511.07571.

[15] Teuvo Kohonen. 1998. The self-organizing map. *Neurocomputing* 21, 1-3, 1–6.

[16] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*. 1097–1105.

[17] Martha Larson, Mohammad Soleymani, Guillaume Gravier, Bogdan Ionescu, and Gareth J. F. Jones. 2017. The benchmarking initiative for multimedia evaluation: MediaEval 2016. *IEEE MultiMedia* 24, 1, 93–96.

[18] Andreas Leibetseder, Sabrina Kletz, and Klaus Schoeffmann. 2018. Sketch-based similarity search for collaborative feature maps. In *MultiMedia Modeling*, K. Schoeffmann, T. H. Chalidabhongse, C. W. Ngo, S. Aramvith, N. E. O'Connor, Y.-S. Ho, et al. (Eds.). Springer International Publishing, Cham, Switzerland, 425–430.

[19] Michael S. Lew, Nicu Sebe, Chabane Djeraba, and Ramesh Jain. 2006. Content-based multimedia information retrieval: State of the art and challenges. *ACM Transactions on Multimedia Computing, Communications, and Applications* 2, 1, 1–19.

[20] N. Liu and J. Han. 2016. DHSNet: Deep hierarchical saliency network for salient object detection. In *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'16)*. 678–686. DOI: http://dx.doi.org/10.1109/CVPR.2016.80

[21] J. Lokoč, W. Bailer, K. Schoeffmann, B. Muenzer, and G. Awad. 2018. On influential trends in interactive video retrieval: Video Browser Showdown 2015-2017. *IEEE Transactions on Multimedia* 20, 12, 3361–3376.

[22] Jakub Lokoč, Gregor Kovalčík, and Tomáš Souček. 2018. Revisiting SIRET video retrieval tool. In *Proceedings of the 24th International Conference on Multimedia Modeling (MMM'18), Part II*. 419–424.

[23] Jakub Lokoč, Tomáš Souček, and Gregor Kovalčík. 2018. Using an interactive video retrieval tool for lifelog data. In *Proceedings of the 2018 ACM Workshop on the Lifelog Search Challenge (LSC'18)*. ACM, New York, NY, 15–19.

[24] Yi-Jie Lu, Phuong Anh Nguyen, Hao Zhang, and Chong-Wah Ngo. 2017. Concept-based interactive search system. In *MultiMedia Modeling*, K. Schoeffmann, T. H. Chalidabhongse, C. W. Ngo, S. Aramvith, N.E. O'Connor, Y.-S. Ho, et al. (Eds.). Springer International Publishing, Cham, Switzerland, 463–468.

[25] Michael McCandless, Erik Hatcher, and Otis Gospodnetic. 2010. *Lucene in Action, Second Edition: Covers Apache Lucene 3.0*. Manning Publications, Greenwich, CT.

[26] Phuong Anh Nguyen, Yi-Jie Lu, Hao Zhang, and Chong-Wah Ngo. 2018. Enhanced VIREO KIS at VBS 2018. In *MultiMedia Modeling*, K. Schoeffmann, T. H. Chalidabhongse, C. W. Ngo, S. Aramvith, N. E. O'Connor, Y.-S. Ho, et al. (Eds.). Springer International Publishing, Cham, Switzerland, 407–412.

[27] Manfred Jürgen Primus, Bernd Münzer, Andreas Leibetseder, and Klaus Schoeffmann. 2018. The ITEC collaborative video search system at the Video Browser Showdown 2018. In *MultiMedia Modeling*, K. Schoeffmann, T. H. Chalidabhongse, C. W. Ngo, S. Aramvith, N. E. O'Connor, Y.-S. Ho, et al. (Eds.). Springer International Publishing, Cham, Switzerland, 438–443.

[28] Marek Rogozinski Rafal Kuc. 2013. *Mastering ElasticSearch*. Packt Publishing.

[29] Joseph Redmon and Ali Farhadi. 2016. YOLO9000: Better, faster, stronger. arXiv:1612.08242. http://arxiv.org/abs/1612.08242

[30] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Neural Information Processing Systems (NIPS)*.

[31] Luca Rossetto, Ivan Giangreco, Ralph Gasser, and Heiko Schuldt. 2018. Competitive video retrieval with vitrivr. In *Proceedings of the 24th International Conference on MultiMedia Modeling (MMM'18), Part II*. 403–406.

[32] Luca Rossetto, Ivan Giangreco, and Heiko Schuldt. 2014. Cineast: A multi-feature sketch-based video retrieval engine. In *Proceedings of the 2014 IEEE International Symposium on Multimedia*. 18–23.

[33] Luca Rossetto, Ivan Giangreco, Claudiu Tanase, and Heiko Schuldt. 2016. Vitrivr: A flexible retrieval stack supporting multiple query modes for searching in multimedia collections. In *Proceedings of the 2016 ACM Conference on Multimedia (MM'16)*. ACM, New York, NY, 1183–1186.

[34] Sitapa Rujikietgumjorn, Nattachai Watcharapinchai, and Sanparith Marukatat. 2018. Sloth search system. In *Proceedings of the 24th International Conference on MultiMedia Modeling (MMM'18), Part II*. 431–437.

[35] Klaus Schoeffmann. 2014. A user-centric media retrieval competition: The Video Browser Showdown 2012-2014. *IEEE MultiMedia* 21, 4, 8–13.

[36] Klaus Schoeffmann, Frank Hopfgartner, Oge Marques, Laszlo Boeszoermenyi, and Joemon M. Jose. 2010. Video browsing interfaces and applications: A review. *SPIE Reviews* 1, 1, 018004. DOI: http://dx.doi.org/10.1117/6.0000005

[37] Klaus Schoeffmann, Marco A. Hudelist, and Jochen Huber. 2015. Video interaction tools: A survey of recent work. *ACM Computing Surveys* 48, 1, Article 14 (Sept. 2015), 34 pages.

[38] Klaus Schoeffmann, Manfred Jürgen Primus, Bernd Muenzer, Stefan Petscharnig, Christof Karisch, Qing Xu, et al. 2017. *Collaborative Feature Maps for Interactive Video Search*. Springer International Publishing, Cham, Switzerland, 457–462.

[39] Mei-Ling Shyu, Zongxing Xie, Min Chen, and Shu-Ching Chen. 2008. Video semantic event/concept detection using a subspace-based multimedia data mining framework. *IEEE Transactions on Multimedia* 10, 2, 252–259.

[40] Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. arxiv:1409.1556. http://arxiv.org/abs/1409.1556.

[41] R. Smith. 2007. An overview of the tesseract OCR engine. In *Proceedings of the 9th International Conference on Document Analysis and Recognition—Volume 02 (ICDAR'07)*. IEEE, Los Alamitos, CA, 629–633.

[42] Cees G. M. Snoek and Marcel Worring. 2005. Multimedia event-based video indexing using time intervals. *IEEE Transactions on Multimedia* 7, 4, 638–647.

[43] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott E. Reed, Dragomir Anguelov, et al. 2015. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'15)*. 1–9.

[44] Thanh-Dat Truong, Vinh-Tiep Nguyen, Minh-Triet Tran, Trang-Vinh Trieu, Tien Do, Thanh Duc Ngo, et al. 2018. Video search based on semantic extraction and locally regional object proposal. In *MultiMedia Modeling*, K. Schoeffmann, T. H. Chalidabhongse, C. W. Ngo, S. Aramvith, N. E. O'Connor, Y.-S. Ho, et al. (Eds.). Springer International Publishing, Cham, Switzerland, 451–456.

[45] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. 2017. Show and tell: Lessons learned from the 2015 mscoco image captioning challenge. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39, 4, 652–663.

[46] Marcel Worring, Paul Sajda, Simone Santini, David A. Shamma, Alan F. Smeaton, and Qiang Yang. 2012. Where is the user in multimedia retrieval? *IEEE MultiMedia* 19, 4, 6–10.

[47] Zheng-Jun Zha, Meng Wang, Yan-Tao Zheng, Yi Yang, Richang Hong, and Tat-Seng Chua. 2012. Interactive video indexing with statistical active learning. *IEEE Transactions on Multimedia* 14, 1, 17–27.

[48] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. 2017. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence 40, 6, 1452–1464.*

[49] Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. 2014. Learning deep features for scene recognition using places database. In *Proceedings of the 27th International Conference on Neural Information Processing Systems—Volume 1 (NIPS'14).* 487–495.

Received July 2018; revised November 2018; accepted November 2018