Simplifying Accessibility Without Data Loss: An Exploratory Study on Object Preserving Keyframe Culling

Marc Ritter[1](✉), Danny Kowerko[1], Hussein Hussein[1], Manuel Heinzig[1],
Tobias Schlosser[1], Robert Manthey[1], and Gisela Susanne Bahr[2]

[1] Junior Professorship Media Computing, Technische Universität Chemnitz,
09107 Chemnitz, Germany
{marc.ritter,danny.kowerko,hussein.hussein,manuel.heinzig,
tobias.schlosser,robert.manthey}@informatik.tu-chemnitz.de

[2] Department of Biomedical Engineering, Florida Institute of Technology,
Melbourne, FL 32901, USA
gbahr@fit.edu

Abstract. Our approach to multimedia big data is based on data reduction and processing techniques for the extraction of the most relevant information in form of instances of five different object classes selected from the TRECVid Evaluation campaign on a shot-level basis on 4 h of video footage from the BBC EastEnders series. In order to reduce the amount of data to be processed, we apply an adaptive extraction scheme that varies in the number of representative keyframes. Still, many duplicates of the scenery can be found. Within a cascaded exploratory study of four tasks, we show the opportunity to reduce the representative data, i.e. the number of extracted keyframes, by up to 84 % while maintaining more than 82 % of the appearing instances of object classes.

Keywords: Multimedia analysis · Duplicate detection · Human inspired data reduction algorithms · Data reduction strategies · Big data · Object detection · Instance Search · Rapid evaluation

1 Introduction

A more recent challenge in the area of accessibility engineering is how to cope with large volumes of data. Approaches to the analysis of audiovisual big data are usually based on data reduction and processing techniques that focus on the extraction of the most relevant information. On closer consideration, such kind of information usually appears in the form of objects that occur within a specific shot, whereas we denote a shot as a continuous recording in time and space [1]. In order to retrieve such valuable objects, two essential steps need to be accomplished: A decomposition of the structure of a video into homogeneous sequences of images (shots) that is often followed by a more distinct content-based analysis focusing on the detection or recognition of specific objects. However, this

M. Antona and C. Stephanidis (Eds.): UAHCI 2016, Part III, LNCS 9739, pp. 492–504, 2016.
DOI: 10.1007/978-3-319-40238-3_47

procedure opens a spot for a vast reduction of data by allowing the selection of representative keyframes on a shot-level basis that are further processed as a surrogate in place for the remaining frames in the very same shot. Over more than a decade, a lot of efforts within the scientific community around the international *Text Retrieval Evaluation Campaign on Videos* (TRECVid) [2], annually organized and held by the *National Institute of Standards and Technologies* (NIST, US), have led to advanced methods in structural and content-based analysis in the task of *Instance Search* that nowadays yield reasonable results. The major objective is to locate instances of a given object class. This becomes outstandingly challenging, since the object class is mainly provided by four sample images and a descriptive text string of the class. Since the requested object class is unknown prior to the appearance of the search query and the search systems are not allowed to be trained on the class requested, there is a need to adapt the knowledge base of the system on-line once the search query becomes available. A principle capability of adaption is also necessary, since the object class requested may alter their appearance far beyond the four given query samples: E.g. the period of TRECVid 2014 contained an object query (topic ID 9103) that was named *a curved plastic-bottle of ketchup*. During the course of the series the ketchup container changed from a tall and slim bottle of ketchup into a small and bulbous bottle. A lot of the participants use a variety of different methods to reduce the number of frames for object extraction. All have in common that they try to greatly reduce the amount of data. For example, *Alvi et al.* [3] extract one image per second and *Feng et al.* [4] resize it to 75 %, while *Yao et al.* [5] select every 15th frame. In contrast to those methods that extract a constant number of keyframes in a specific interval, our contribution focuses on a more adaptive scheme (Sect. 2) that tries to reduce the number of frames in a shot by selecting a different amount of keyframes with respect to the length of the shots and their overall distribution. This method has already been successfully applied within past evaluation periods (cf. to *Ritter et al.* [6,7]). The necessary master shot boundaries are officially provided by NIST as a result of an automated boundary shot detection algorithm on video footage of 464 h from the British soap *BBC EastEnders*. Despite the application of this adaptive method, still a vast number of duplicates from the same scenery with a lot of overlapping contents is extracted. In order to investigate such shot-based duplicate keyframes, we use a large sample from the TRECVid 2015 BBC *Instance Search* dataset. An intellectual way to identify duplicated keyframes is to investigate the extracted frames for common objects. This can be regarded as a challenging task, since no objects in the fore- or background should be removed by accident in order to prevent a loss of information for further processing. Within a cascaded empirical study in Sect. 3, we are going to ask the participants to identify the keyframes with the same objects and to remove the duplicated ones respectively. We will use multiple sequential tasks that build on each other in order to intensify the efforts of the cognitive workload of the participants. Furthermore, in Sect. 4 we are interested to learn about the criteria for the intellectual process of duplicate removal with removal constraints in order to compare them to results from tra-

Table 1. Summary of the data set that builds the base of our experiments.

Data set	Subset of TRECVid BBC EastEnders	
Video resolution	768 × 576 px anamorphic	
Omnibus video files	# 114	# 163
Duration	01:54:20 h	01:57:49 h
Size	1.25 GB	1.29 GB
Keyframe resolution	928 × 512 px	
Keyframe format	24 Bit JPEG	
# Keyframes	5,428	4,921

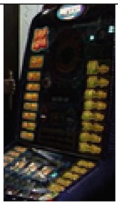
ditional methods from the field of image similarity computation using different measures. The general prevention of a loss of objects will be limited to a number of five different object/topic categories that originate from the past evaluation campaign period. Our developed annotation and evaluation tools [8–10] create a convenient baseline for the investigations involved.

2 System Architecture, Data Extraction Scheme, and Data Setup

In the following paragraphs, we introduce the origin, the construction scheme, and characteristic properties for the image data that provides the base for the further experiments of the study.

Data Setup: Our primary data source is derived from the video footage of the TRECVid 2015 *Instance Search* task. This data set consists of recordings from the *BBC Series EastEnders*, a daily soap running in the UK since 1985 featuring various indoor and outdoor settings in more than 26 filming locations with a huge variety of objects used as background decoration. As the production and release of such daily episodes tends to be a bit hasty and time pressured, the footage is not very well edited from time to time. As a consequence, even basic cineastic standards for image quality are missed, resulting for instance in an imperfect white balance and brightness level. The given video collection contains 244 so called omnibus episodes, which means that a number of episodes are glued together to form one big two hour long sequence without any interruptions caused by intro, outro or advertisements. The videos are recorded with 25 frames per second in an anamorphic format, which has the effect of standard algorithms grabbing a 4:3 image that appears visually distorted and therefore has to be stretched to 16:9 to achieve visual regularity. We also remove black borders that appear around the image content in order to prevent distractions in the visual cognition of the test subjects. The main goal in the TRECVid *Instance Search* task is to automatically detect appearances of objects ("instances") in the given data set. In each years evaluation period, there are 30 so called "topics" consisting of four example pictures and a very brief description of what can be

Table 2. Five object topics from the TRECVid *Instance Search 2015 task* [12] to be retrieved by the participants in Task 3 and 4 of the study.

9130	9131	9150	9154	9156
				
a chrome napkin holder	a green and white iron	this IMPULSE game	this neon Kathy's sign	a 'DEVLIN' lager logo

seen in the pictures or whether a particular specification of the object is needed. This also comprises additional properties like slight differences that manifest for instance in different colors of a shirt or vest. The search targets can be of any kind and are not limited by any means. Persons as well as small physical things or even particular houses or landmarks can be of interest. Some of them are easy to perceive by a human, some appear quite challenging. Whereas the first two tasks are concerned with duplicate removal, the latter tasks focus on the intellectual retrieval of five different topics. Therefore, we must assure that those objects appear in the video footage chosen for our experiments. Therefore, we analyze the ground-truth distribution of object appearances in specific shots for all 30 topics provided by NIST for the last years iteration of TRECVid. Due to a very time-consuming and mentally exhausting annotation process, the hand-truth distribution only contains results from an inspected fraction of all available shots. However, the intellectually annotated instances of all topics are to be considered as rare cases and are not spread uniformly in the data set. By using analytical methods, we found that the five relevant object categories in this contribution (see Table 2) mostly occur together in the videos with the numbers 114 and 163 with a total of 111 instances. This is why we select both videos for our experiment. The data properties are shown in Table 1.

Adaptive Keyframe Extraction Scheme: Since the data is recorded with 25 frames per second, it leads to potentially 90,000 images per hour that potentially contain objects of interest. Given the limits of human attention, evaluating such huge numbers of pictures turns out as a very challenging task. Therefore, we developed an approach to automatically reduce the data by a significant factor while losing a minimum of semantic information. The essential knowledge we have to keep are objects that are present in the video footage. When looking at the given data from that point of view, a cinematic characteristic can be noticed: During a shot, the objects that appear are mostly stable. That is, because in the small time window where a camera is directed approximately at the same scenery, usually things in the background do not move and are present for a plenty of subsequent frames. This leads to the assumption, that it is sufficient to look at one representative picture (that we refer to as keyframe) per shot. In practice this assumption does not hold strictly due to camera twists, slow turns or object movements, which lead to a change of the depicted area and consequently the objects in it. To respect this factor, we propose a scheme that extracts a various number of keyframes from each shot to represent temporal object variances. Following a simple but logical approach, the most representative frame to extract is the one in the middle of a shot. As stated before, this is only sufficient for very short shot durations. To represent the whole temporal outline, we also extract a frame from the start and the end of each shot. Sometimes the duration of a shot gets extensively large (20 s and more). In such cases we can no longer be sure that the filmed scenery does only change slightly, which could lead to lost objects when only extracting the three frames mentioned above. In order to still keep all existing objects, we introduce yet another level of extraction, where two more frames are picked at 25 % and 75 %, in between the existing positions. We finally end up with a scheme, that extracts one keyframe when the length of the shot is lower than two seconds, 5 keyframes when its 5 s or longer and 3 keyframes in between. The selected intervals are resulting from an analysis of the shot length distribution (cmp. to *Ritter et al.* [6, p. 3]). Applied to the selected video footage, this leads to a reduction from 167,750 potential to 5,428 representative frames for the first test video 114 while reducing 171,075 frames to 4,921 keyframes on video 163, which is a diminution by approximately a factor of more than 30.

3 Experiments and Results

In the following, we investigate the extracted keyframe data set from the previous section by our study in order to explore the potential for the reduction of duplicates while preserving relevant object instances.

Method: Five persons (age $\mu = 32.6$ and $\sigma = 7.9$, male, expertise in computer science or physics) participated in the exploratory study that consists of four major tasks (the first three tasks are building upon and complementing each other). Due to its dependent nature, the amount of data varies between the participants in both of the inner tasks. *Task 1* uses the total amount of 10,349 keyframes whereas *Task 4* operates on shots that contain at least two representative frames per shot adding up to 8,881 keyframes in total. The first task intents to eliminate unusable keyframes in which no objects are found due to poor quality, monochromaticity or blurred pictures as well as to keep usable keyframes with identifiable objects. The second task aims to sort out duplicate keyframes in shots from which at least more than one keyframe was extracted. The keyframes with the same objects are removed (culled), whereas keyframes with different objects are preserved. The third task attempts to search for objects in the remaining keyframes from the second task. We used five different objects from the TRECVid 2015 *Instance Search* evaluation campaign database. The fourth task seeks for the same objects which are used in the third task.

- *Task 1* removes keyframes which don't contain any useful information, i.e. no objects can be found in the pictures. The maximal search time is limited to 45 min per video.
- *Task 2* comprises the removal (culling) of shot-based duplicate keyframes that seem to contain the same objects without changes. The search time is limited to multiples of 20 min sprints being followed by a break of 5 min.
- After removing unusable and duplicated keyframes in the previous tasks, *Task 3* aims to spot five instances of different object categories on a shot-level basis. Every participant searched for three different objects; in total each object category is searched by three different users. The search time is limited to 10 min sprints followed by a break of 5 min. The objects used in this study are shown in Table 2.
- *Task 4* aims to retrieve the object topics from *Task 3* by using the complete set of keyframes used before the *Task 1* in order to compare the working speed as well as the quality & quantity between this task on the one and the other three tasks on the other hand. Every participant searched for instances of exactly one specific object category. The search mode is equal to the previous task.

In addition, answers to the following questions are retrieved from the participants for each task:

- Information comprising criteria or individual reasoning about the elimination of unusable keyframes and the preservation of useful keyframes.
- Identification or individual reasoning of criteria that led to the maintenance or deletion of shot-based duplicate keyframes.
- Impressions showing the experience in object recognition as well as the advantages and disadvantages of the search process.
- Discovery of individual differences or opinions between the test people.

Humans usually adapt to patterns while dealing with larger amounts of data resulting in an acceleration of the task completion time. In order to counter such effects and also relieve fatigue, we split the participants into two groups: We join two participants (P2 and P4) in the first group and the other three (P1, P3, and P5) into a second group and reverse the processing order of keyframe data sets for the both groups in *Task 1, 2,* and *4*. Due to the large reduction in the number of keyframes in the previous tasks, we didn't provide any order for *Task 3*.

Evaluation of the Study: The completion times, numbers of culled images, and remaining images for the four tasks of the exploratory study are shown in Tables 3 and 4. *Task 1* detected a very small number of unusable keyframes with an average of 16 and a standard deviation of 5 as shown in Table 3. The participants performed the task with a varying accuracy. P1 and P5 are the slowest participants, however, P1 removed only one keyframe on average, whereas P5 found the largest number

Table 3. Completion times (t_1, t_2) of *Task 1+2* in seconds as well as numbers of culled images (c_1, c_2) and remaining images (r_1, r_2) for the data sets *Video 114* and *Video 163* of the participants (P1–P5). The initial data set contained the same 10,349 images for every participant being reduced by the results of the first task.

	Video 114			Video 163			Task 1					
	t_1	c_1	r_1	t_2	c_2	r_2	μ_t	σ_t	μ_c	σ_c	μ_r	σ_r
P1	1,144	0	5,428	1,237	2	4,919	1,190.5	65.8	1.0	1.4	5,173.5	359.9
P2	690	6	5,422	678	10	4,911	684.0	8.5	8.0	2.8	5,166.5	361.3
P3	513	18	5,410	774	23	4,898	643.5	184.6	20.5	3.5	5,154.0	362.0
P4	736	10	5,418	725	17	4,904	730.5	7.8	13.5	4.9	5,161.0	363.5
P5	1,260	44	5,385	900	30	4,887	1,080.0	254.6	37.0	9.9	5,136.0	352.1
						Ø	865.7	104.2	16.0	4.5	5,158.2	359.8

	Video 114			Video 163			Task 2					
	t_1	c_1	r_1	t_2	c_2	r_2	μ_t	σ_t	μ_c	σ_c	μ_r	σ_r
P1	7,139	2,675	1,923	5,754	2,649	1,532	6,446.5	979.3	2,662.0	18.4	1,727.5	276.5
P2	4,900	2,752	1,846	3,950	2,707	1,475	4,425.0	671.8	2,729.5	31.8	1,660.5	262.3
P3	3,418	2,856	1,736	1,534	2,693	1,485	2,476.0	1,332.2	2,774.5	115.3	1,610.5	177.5
P4	3,883	2,876	1,719	2,538	2,750	1,428	3,210.5	951.1	2,813.0	89.1	1,573.5	205.8
P5	3,760	2,949	1,606	2,740	2,818	1,357	3,250.0	721.2	2,883.5	92.6	1,481.5	176.1
						Ø	3,961.6	931.1	2,772.5	69.4	1,610.7	219.6

of unusable keyframes with an average of 37 keyframes. Some reasons given by test subjects for the deletion of keyframes are unsharp pictures, blurred objects, and compression artifacts. The number of keyframes detected as duplicates and therefore being removed from the data set in *Task 2* is 2,772 keyframes on average. The completing time of this task varied strongly between participants. With 2,476 s, P3 appears as the fastest participant in almost all tasks. The participants detected duplicate keyframes that for instance don't contain changes in objects or don't show other objects in the fore- and background by slight camera movements. Individual results showing the distribution of remaining keyframe numbers of P1 to P5 in absolute values are depicted in Fig. 1 which tends to be homogeneous between all participants with a standard deviation of less than 15 % on average. Moreover, participants tend to have individual preferences for a specific keyframe number, like 0 and 2. The domination of keyframes 0, 2 and 4 over 1 and 3 results from the fact that a vast number of shots consist of only three keyframes enumerated with the labels "0", "2" and "4". As each keyframe is selected using the keys 1 to 5 on the keyboard, the participants use individual favourite keys in case of similar images for the sake of time. In conclusion, the average of 1,611 remaining keyframes approximately equals to 16 % of the data set used in the study yielding a data reduction potential of more than 84 %. With 3,961 s on average, the working time is about 4.5 times higher than in the previous task greatly reflecting the cognitive challenge of finding duplicates while preserving any objects. In addition, *Task 3* searched for instances of the five topics in the data set remainder from the previous tasks and

Table 4. Completion times (t_1, t_2) of *Task 3+4* in seconds as well as the numbers of shots containing object instances found (h_1, h_2) of the participants (P1–P5). The initial data sets for *Task 3* contained the remaining individual sets from *Task 2* of every participant in contrast to the 8,881 keyframes that contain at least two keyframes per shot in *Task 4*.

		Video 114		Video 163		Task 3					
	TopicID	t_1	h_1	t_2	h_2	$\sum_t$	μ_t	σ_t	$\sum_h$	μ_h	σ_h
P1	9150	666	8	840	17	1,506	753.0	123.0	25	12.5	6.4
	9154	836	1	536	4	1,372	686.0	212.1	5	2.5	2.1
	9156	847	4	812	16	1,659	829.5	24.7	20	10.0	8.5
P2	9130	570	21	425	4	995	497.5	102.5	25	12.5	12.0
	9131	580	0	420	7	1,000	500.0	113.1	7	3.5	4.9
	9156	516	2	440	14	956	478.0	53.7	16	8.0	8.5
P3	9131	423	4	291	10	714	357.0	93.3	14	7.0	4.2
	9154	351	1	284	0	635	317.5	47.4	1	0.5	0.7
	9156	296	4	232	11	528	264.0	45.3	15	7.5	4.9
P4	9130	467	19	256	3	723	361.5	149.2	22	11.0	11.3
	9131	371	10	312	6	683	341.5	41.7	16	8.0	2.8
	9150	329	7	236	11	565	282.5	65.8	18	9.0	2.8
P5	9130	344	20	308	4	652	326.0	25.5	24	12.0	11.3
	9150	304	8	297	15	601	300.5	4.9	23	11.5	4.9
	9154	250	1	181	2	431	215.5	48.8	3	1.5	0.7
					Ø	868.0	434.0	76.7	15.6	7.8	5.7

		Video 114		Video 163		Task 4					
	TopicID	t_1	h_1	t_2	h_2	$\sum_t$	μ_t	σ_t	$\sum_h$	μ_h	σ_h
P1	9130	1,391	20	1,124	3	2,515	1,257.5	188.8	23	11.5	12.0
P2	9154	713	1	1,080	5	1,793	896.5	259.5	6	3.0	2.8
P3	9150	577	6	599	17	1,176	588.0	15.6	23	11.5	7.8
P4	9156	533	4	643	14	1,176	588.0	77.8	18	9.0	7.1
P5	9131	956	16	543	9	1,499	749.5	292.0	25	12.5	4.9
					Ø	1,631.8	815.9	166.7	19.0	9.5	6.9

achieved a completing time of 434 s per omnibus episode, whereas the execution time of *Task 4* is almost twice as long as shown in Table 4. This indicates that the object recognition based on the deletion of unusable and duplicates keyframes can be more quickly performed than on the whole data set. We recognize an affordable loss of around 18 % in accuracy in the number of retrieved instances between both tasks from 9.5 to 7.8 on average and a similar behavior in the standard deviations can be explained by high number of eliminated duplicates in addition to common human errors that usually occur by working on such complex tasks. The experience in object recognition reported by participants showed that this is task was perceived as a very tiring one, where the attention decreases without breaks. The participants report a focus on specific scenes or locations in order to retrieve the object more quickly. Some problems encountered by the participants are the search

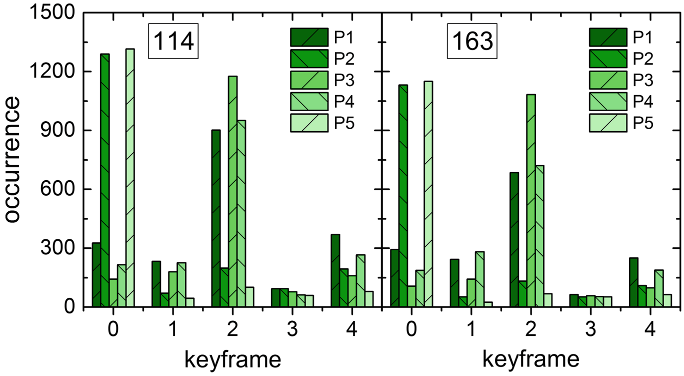


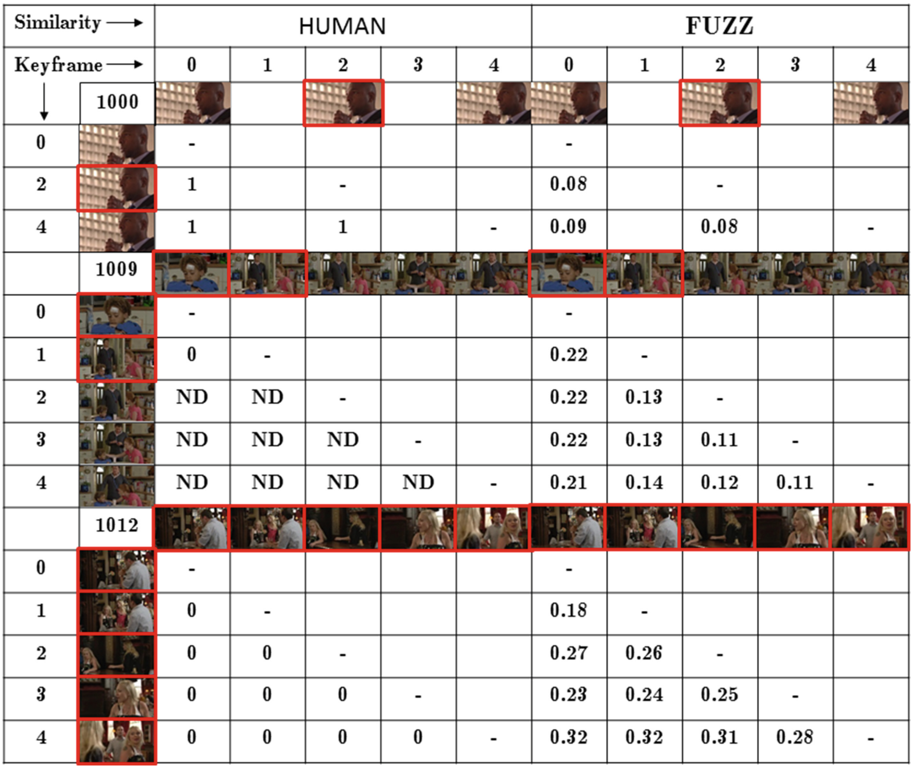
Fig. 1. Keyframe number selection statistics from *Task 2* of all five participants (P1–P5) for omnibus videos 114 and 163.

for small objects which took a long searching time compared to large objects. The poor quality of the images was given as one major reason.

4 Comparison of Intellectual Selections with Computational Similarity Measures

The combination of culled and remaining images of *Task 2* contains information about *human* categorization into quasi-similar and dissimilar, a binary classification represented by 1 and 0 in the following. It is used to assign both values for each image combination within a shot, represented as elements M_{ij} of a similarity matrix M which is shown in Fig. 2 for three typical shots from *video 114*. A drawback of our selection method is that for shots with 5 keyframes and 2, 3 or 4 selected images, we cannot assign 9, 7 or 4 human similarity values, exemplified in shot 1,009 of Fig. 2 by using the label "ND" (not determined). In shots with three keyframes and two selections, two ND fields remain, while selecting 1 or 3 images results in 3 similarity values of type 1 or 0, respectively. Furthermore, the comparison matrix is symmetric ($\mathrm{M} = \mathrm{M}^\mathrm{T}$) since the order of comparing two images does not play a role ($\mathrm{M}_{ij} = \mathrm{M}_{ji}$). The respective values were omitted together with diagonal elements (self comparison) M_{ii} for clarity in Fig. 2. For shots with 3 or 5 keyframes, we eventually extract a maximum of 3 or 10 values. With the given number of shots and keyframes, the raw data of *video 114 and 163* allow a maximum of 5,998 + 5,937 = 11,935 image comparisons giving a maximum of 59,675 comparisons of human and computational similarity. Due to the selection effect in *Task 1* and the above mentioned "ND" cases, the expected values will be lower, dominated by the information loss of the "ND" problem. The derivation of the computational similarity values using the FUZZ metric from ImageMagick[1] will be described in the following. Therefore

[1] http://www.imagemagick.org, 02-29-2016.



Similarity →		HUMAN					FUZZ				
Keyframe →		0	1	2	3	4	0	1	2	3	4
↓	1000										
0		-					-				
2		1		-			0.08		-		
4		1		1		-	0.09		0.08		-
	1009										
0		-					-				
1		0	-				0.22	-			
2		ND	ND	-			0.22	0.13	-		
3		ND	ND	ND	-		0.22	0.13	0.11	-	
4		ND	ND	ND	ND	-	0.21	0.14	0.12	0.11	-
	1012										
0		-					-				
1		0	-				0.18	-			
2		0	0	-			0.27	0.26	-		
3		0	0	0	-		0.23	0.24	0.25	-	
4		0	0	0	0	-	0.32	0.32	0.31	0.28	-

Fig. 2. Similarities as determined by participants (HUMAN) compared to a typical computational similarity metric (FUZZ) for 3 representative shots of *video 114* with either 3 or 5 keyframes are illustrated for shot 1,000; 1,009 and 1,012. Red boxes mark the corresponding choices of P1. Fields with “ND” refer to HUMAN similarity that could not be determined automatically in *Task 2*. (Color figure online)

the chosen evaluation criterion is the FUZZ metric, which calculates the difference between two given images pixel by pixel, adding up the squared distortions and normalizing the total of it. Formally:

$$FUZZ = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(c_i - \bar{c}_i)^2} \tag{1}$$

Given two images c and $\bar{c}$ of size N pixel, let c_i and $\bar{c}_i$ denote the respective ones in comparison. Those are calculated separately for each color channel and eventually averaged. From the similarity matrices presented in Fig. 2, the human and computational equivalent elements are combined to a scatter plot shown in Fig. 3 on the top left. The anti-correlation of the 23 selected data points clearly supports the hypothesis that FUZZ metric and human similarity follow a common trend. Low FUZZ values represent high similarity as the definition of FUZZ relies on the difference between two images, see Eq. 1. Note that for shot 1,009, the ND values have manually been determined as follows: $M_{2,0} = M_{3,0} = M_{4,0} = 0$ and $M_{2,1} = M_{3,1} = M_{4,1} = M_{3,2} = M_{4,2} = M_{4,3} = 1$. In conclusion the 3 example shots imply a strong consistence of human and computational similarity. Expanding this concept to the data to all 5 participants and all shots/keyframes selected in *Task 2*, for *videos 114 and 163*, we obtained 2,962 and 1,937 FUZZ values categorized by humans with type 0, as well as 18,884 and 21,313 type 1 rated FUZZ values whose probability density distributions are shown in Fig. 3, bottom. The dominance of 1 indicates the existence of a majority of shots with 3 or 5 similar keyframes. However, the individual as well as the sum distributions of *videos 114 and 163* show merely the same trend. FUZZ metric values closer to 0 than 0.4 are distinctly more often selected by human beings as similar keyframes. Still, all distributions overlap considerably. Defining a simple FUZZ threshold value is inappropriate to automatically remove duplicate type images within the data set analyzed in this study.

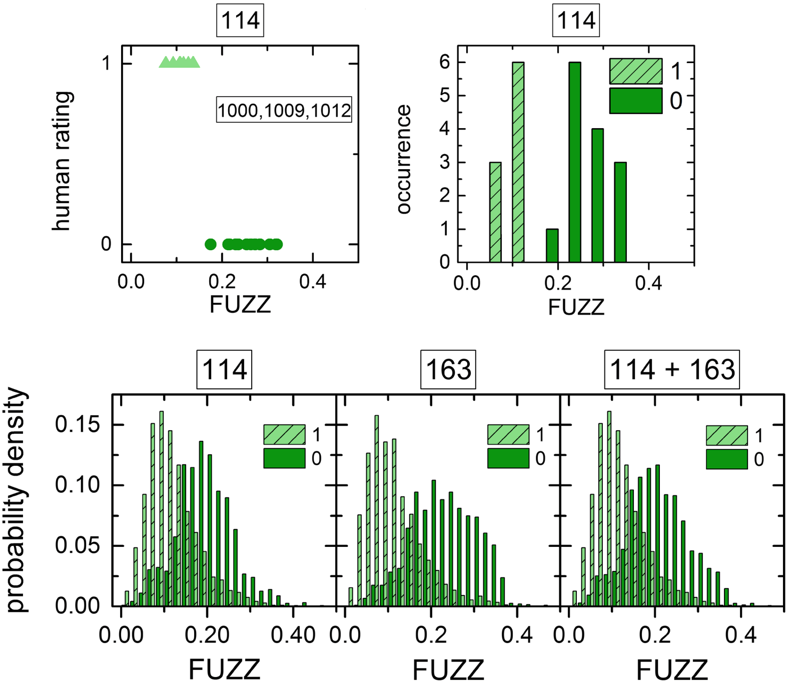


Fig. 3. Comparison of human and computational similarity metrics. Top: A non-overlapping anti-correlation is shown for the 3 exemplary shots of Fig. 2 with 23 data points. Bottom: Normalized histograms from the data of all participants for both videos, each including more than 20,000 HUMAN-FUZZ similarity pairs.

5 Summary and Future Work

In conclusion, we demonstrated that within a total experiment time of about 12 h and 5 participants more than 45,000 human-computer image similarity comparisons have been derived. The obtained overlapping histogram distributions (cmp. to Fig. 3) of human and computational similarities resemble those of other problems like face detection. Therefore, ensemble-based machine learning techniques like boosting are promising to further separate duplicates in an automated manner, reliably reducing the amount of data used in big data evaluation campaigns like TRECVid with a small loss of information. The binary annotation and classification tool [10] has proven beneficial over all 4 tasks in order to create statistically sound data sets within a reasonable period of time. Furthermore, 36 additional shots amongst five topics could be identified in both videos that were not contained in the ground-truth information provided by NIST. Future work focuses on the identification of the *not determined* values of the similarity matrices, a modification of *Task 2* will potentially increase the total number of human-computer image similarity pairs considerably.

Acknowledgments. This work was partially accomplished within the project *localizeIT* (funding code 03IPT608X) funded by the *Federal Ministry of Education and Research* (BMBF, Germany) in the program of *Entrepreneurial Regions InnoProfile-Transfer*. Programme material is copyrighted by the BBC.

References

1. Ritter, M.: Optimierung von Algorithmen zur Videoanalyse: Ein Analyseframework für die Anforderungen lokaler Fernsehsender. In: Wissenschaftliche Schriftenreihe Dissertionen der Medieninformatik (3), TU Chemnitz, 336 pp. (2014)
2. Smeaton, A.F., Over, P., Kraaij, W.: Evaluation campaigns and trecvid. In: ACM International Workshop on Multimedia Information Retrieval, pp. 321–330 (2006)
3. Alvi, M., Khan, M.U.G., Sadiq, M., Aslam, M.: University of Engineering & Technology, Lahore, The University of Sheffield at TRECVID, (2011) Observation of strains: Instance Search. In: TRECVID Workshop 2015, Gaithersburg, Maryland, 5 pp. (2015)
4. Feng, Y., Dong, Y., Wu, Y., Bai, H., Cen, S., Liu, B., Wang, K., Liu, Y.: BUPT & ORANGELABS (OrangeBJ) AT TRECVID 2014: INSTANCE SEARCH. In: TRECVID Workshop 2014, 10–12 November 2014, Orlando, Florida, USA, 9 pp. (2014)
5. Yao, L., Ye, M., Liu, D., Shao, R., Liu, T., Liu, J., Wang, Z., Liang, C.: WHU-NERCMS at TRECVID2015: Instance Search task. In: TRECVID Workshop (2015)
6. Ritter, M., Heinzig, M., Herms, R., Kahl, S., Richter, D., Manthey, R., Eibl, M.: Technische Universität Chemnitz at TRECVID Instance Search 2014. In: TRECVID Workshop 2014, 10–12 November 2014, Orlando, Florida, 8 pp. (2014)
7. Ritter, M., Rickert, M., Juturu Chenchu, L., Kahl, S., Robert, H., Hussein, H., Heinzig, M., Manthey, R., Bahr, G.S., Richter, D., Eibl, M.: Technische Universität Chemnitz at TRECVID Instance Search 2015. In: TRECVID Workshop (2015)

8. Ritter, M., Eibl, M.: An extensible tool for the annotation of videos using segmentation and tracking. In: Marcus, A. (ed.) HCII 2011 and DUXU 2011, Part I. LNCS, vol. 6769, pp. 295–304. Springer, Heidelberg (2011)
9. Storz, M., Ritter, M., Manthey, R., Lietz, H., Eibl, M.: Annotate. Train. Evaluate. A unified tool for the analysis and visualization of workflows in machine learning applied to object detection. In: Kurosu, M. (ed.) HCII/HCI 2013, Part V. LNCS, vol. 8008, pp. 196–205. Springer, Heidelberg (2013)
10. Ritter, M., Storz, M., Heinzig, M., Eibl, M.: Rapid model-driven annotation and evaluation for object detection in videos. In: Antona, M., Stephanidis, C. (eds.) UAHCI 2015 Part I. LNCS, vol. 9175, pp. 464–474. Springer, Heidelberg (2015)
11. Forsyth, D.A., Malik, J., Fleck, M.M., Greenspan, H., Leung, T., Belongie, S., Carson, C., Bregler, C.: Finding pictures of objects in large collections of images. In: International Workshop on Object Recognition for Computer Vision, pp. 335–360 (1996)
12. Over, P., Awad, G., Michel, M., Fiscus, J., Kraaij, W., Smeaton, A.F., Quenot, G., Ordelman, R.: TRECVID 2015-an overview of the goals, tasks, data, evaluation mechanisms and metrics. In: Proceedings of TRECVID 2015, NIST, USA (2015)