



Improving video event retrieval by user feedback

Maaïke de Boer^{1,2}  · Geert Pingen^{1,3} ·
Douwe Knook^{1,4} · Klammer Schutte¹ · Wessel Kraaij^{5,6}

Received: 14 October 2016 / Revised: 21 April 2017 / Accepted: 2 May 2017 /

Published online: 12 May 2017

© The Author(s) 2017. This article is an open access publication

Abstract In content based video retrieval videos are often indexed with semantic labels (*concepts*) using pre-trained classifiers. These pre-trained classifiers (*concept detectors*), are not perfect, and thus the labels are noisy. Additionally, the amount of pre-trained classifiers is limited. Often automatic methods cannot represent the query adequately in terms of the concepts available. This problem is also apparent in the retrieval of events, such as *bike trick* or *birthday party*. Our solution is to obtain user feedback. This user feedback can be provided on two levels: *concept level* and *video level*. We introduce the method Adaptive Relevance Feedback (*ARF*) on video level feedback. *ARF* is based on the classical Rocchio relevance feedback method from Information Retrieval. Furthermore, we explore

✉ Maaïke de Boer
maaïke.deboer@tno.nl

Geert Pingen
geert.pingen@tno.nl

Douwe Knook
douwe.knook@gmail.com

Klammer Schutte
klammer.schutte@tno.nl

Wessel Kraaij
w.kraaij@liacs.leidenuniv.nl

¹ TNO, Oude Waalsdorperweg 63, 2597 AK The Hague, Netherlands

² Radboud University, Toernooiveld 200, 6525 EC Nijmegen, Netherlands

³ University of Twente, Drienerlolaan 5, 7522 NB Enschede, Netherlands

⁴ University of Amsterdam, Science Park 904, 1098 XH Amsterdam, Netherlands

⁵ TNO, Anna van Buerenplein 1, 2595DA The Hague, Netherlands

⁶ Leiden University, Niels Bohrweg 1, 2333 Leiden, Netherlands

methods on concept level feedback, such as the re-weighting and Query Point Modification (QPM) methods as well as a method that changes the semantic space the concepts are represented in. Methods on both concept level and video level are evaluated on the international benchmark TRECVID Multimedia Event Detection (MED) and compared to state of the art methods. Results show that relevance feedback on both concept and video level improves performance compared to using no relevance feedback; relevance feedback on video level obtains higher performance compared to relevance feedback on concept level; our proposed ARF method on video level outperforms a state of the art k-NN method, all methods on concept level and even manually selected concepts.

Keywords Video event retrieval · Relevance feedback · Information retrieval · Semantic space · Rocchio

1 Introduction

Current video search systems, such as Youtube [4], mostly rely on the keywords typed with the uploaded videos. In the field of content-based video retrieval, systems retrieve videos using the content of the video within keyframes of the video. Typically *concept detectors* are trained to index videos with the concepts present. One of the constraining factors in concept-based video retrieval systems is the limited amount of concepts a system can be trained to detect. While current state-of-the-art systems are able to detect an increasingly large amount of concepts (*i.e.* thousands), this amount still falls far behind the near infinite amount of possible (textual) queries general-purpose heterogeneous video search systems need to be able to handle [2]. One of the challenging areas within the concept-based video retrieval is that of event retrieval. Events can be defined as complex queries that consist of a multitude of concepts, such as objects, actions and scenes. One example of an event query is *Attempting a bike trick*. This query can be represented by more general concepts such as *bike trick*, *attempt* and *flipping bike*. Creating an automatic representation of a query can, however, include non-relevant or less representative concept detectors and, thus, decrease retrieval performance. Furthermore, the meaning of a concept is different in different contexts, and therefore the quality of a concept detector might differ in the context in which it is applied.

One approach to improve performance when less or non-relevant detectors are selected is the use of *relevance feedback*. With relevance feedback the (estimated) behaviour of the user with the system is used to improve the system. This method is well accepted and commonly used in text retrieval. In video retrieval the trend is to either use click behaviour or to use pseudo-relevance feedback [20, 40, 65], in which we assume that the first x videos are relevant. In this paper, we focus on explicit user feedback, both on the retrieved videos and on selected concepts that represent a query. We compare which relevance feedback level can provide the highest performance gain. Furthermore, we propose a novel method on video level. Our Adaptive Relevance Feedback (ARF) is inspired by the Rocchio algorithm [41], that is often applied in the field of text retrieval. Whereas state of the art relevance feedback algorithms on video level use the annotated videos to create a novel model based on nearest neighbour or SVM type of algorithms [12, 14], we use the videos to approximate the proper weights of the selected concepts in our query representation. The advantage of changing the weights is that this method is able to benefit from just a few positive and negative annotations, compared to newly trained models.

We compare the results of our ARF algorithm on the MEDTRAIN set of the TRECVID benchmark [38] against traditional relevance feedback approaches, such as approaches on concept level such as re-weighting and QPM, and a k-NN based method. Results show that

1. relevance feedback on both concept and video level improves performance compared to using no relevance feedback
2. relevance feedback on video level obtains higher performance compared to relevance feedback on concept level
3. our proposed ARF method on video level outperforms a state of the art k-NN method, all methods on concept level and even manual selected concepts.

2 Related work

In the related work, we focus on state of the art methods in video event retrieval and relevance feedback methods.

2.1 Video event retrieval

In the past decades, image classification has progressed from using handcrafted features on a few images to deep learning methods applied on large image datasets. This progress has lead to the testbed named TRECVID Multimedia Event Detection in which the aim is to obtain a deeper understanding of a video than only object or action recognition [38]. This deeper understanding is obtained by searching for high-level events, defined as ‘long-term spatially and temporally dynamic object interactions’ [21]. Examples of the high-level events are social events (*birthday party*) and procedural events (*making a sandwich*) [21]. The TRECVID MED tasks contains a supervised classification task, in which 100 or 10 training examples of the event are given, and a ‘zero-example’ task, in which only a textual description of the event is provided [38].

A common strategy in video event retrieval is to extract features. The first type of features are static features. These static features can be obtained from the images / frames of which the video consists. Often a video is chopped into keyframes, which are the most determinant images in a sequence of frames within the video. Examples of static features are SIFT [31], SURF [1] and LBP [37] features. Currently, one of the layers of a pre-trained deep neural network is used as the static feature vector [53]. The static features can be represented in a Bag of Words approach [23] and be aggregated over the video using an average or max pooling strategy [53].

Besides the static features in the image, the dynamic or motion features are used in video event retrieval. Examples of such features are motion SIFT [5], STIP [28], dense trajectories [9] and improved dense trajectories [56]. These features are often described in HOG [8], HOG3D [26], HOF [54] or MBH descriptors [55]. Recently, the dynamic features are often encoded into vectors with a fixed dimensionality using Fisher vectors [47] or VLAD encoding [17].

The described features are used to train concept detectors. In supervised video event retrieval, machine learning methods, such as SVMs, Bayesian Classifiers and Random Forests [21, 30] were commonly used, but deep learning techniques have become the state of the art [22, 48]. In ‘zero-example’ video retrieval, which is the case of our interest, concept detectors are trained on large image datasets, such as ImageNet [11], Places [64], FCVID [22] and/or Sports [25]. The concept detectors are applied to the test dataset and the

query is represented as a set of related concepts. This representation can be obtained using knowledge bases, such as Wikipedia [3] or EventNet [61], a semantic embedding, such as VideoStory [16] or word2vec [33], or a manual mapping [63]. Often, a linear weighted sum is used to score the videos on relevance to the query [19, 63].

2.2 Relevance feedback

The use of relevance feedback stems from the dynamic nature of information seeking [44]: information needs can be continuously changing and be unique to each user. Relevance feedback can be done in different ways: *implicit*, *explicit* and *blind/pseudo*. In implicit relevance feedback, implicit information, such as user clicks or dwell time, is used. The advantage of this method is that you do not have to bother the user, but the inference of the results is much harder. Because we focus on a subdomain of costumers in which we expect less queries, we expect that implicit feedback will gain less compared to explicit user feedback. In explicit relevance feedback, the user explicitly indicates if a certain item is relevant or not relevant. This can be done using a binary scale or a gradual scale. The advantage of this method is that you have a clear indication of the relevance and a higher performance, but the disadvantage is that you have to bother the user. This user might not have time or motivation to give such feedback. In blind or pseudo relevance feedback, the manual user part is automated. In this automation, we assume that the first k ranked items are relevant. This assumption is not without a risk, because in the case of rare events or new query domains, bad retrieval systems or ambiguous queries this assumption might not hold. Human relevance feedback (implicit and explicit) has been known to provide major improvements in precision for information retrieval system. Dalton et al. [10] have shown that - in the domain of video retrieval - pseudo-relevance feedback can increase Mean Average Precision (MAP) up to 25%, whereas with human judgments this number can grow up to 55% [10]. Of course the effectiveness of pseudo relevance feedback critically depends on the assumption that the collection contains at least a reasonable number of relevant results and that the first retrieval pass is able to pick up a good fraction of those in the top k . It is clear that relevance feedback, when applied correctly, can help the user in better finding results.

According to Mironică et al. [34], relevance feedback can be incorporate in three ways. The first way is to change the query points, i.e. Query Point Modification (QPM). One of the most well-known and applied relevance feedback algorithms in this category that has its origins in text retrieval is the Rocchio algorithm [41]. The Rocchio algorithm works on a vector space model in which the query drifts away from the negatively annotated documents and converges to the positively annotated documents. The Rocchio algorithm is effective in relevance feedback, fast to use and easy to implement. The disadvantages of the method are that parameters have to be tuned and it cannot handle multimodal classes properly.

The second way is to change the feature representation, i.e. re-weighting. Often a document is represented as a vector with a real-valued component (e.g. TFIDF weight [46]) for each term. The terms used to match the query with are re-weighted according to the relevance feedback [19, 24, 32, 42, 43, 52]. Another strategy is to change the Fisher representation [34] based on the relevance feedback.

The third way is to use classification, which include navigation-pattern and cluster-based approaches. These approaches are explained by Zhou et al. [65] and Patil et al. [40]. The positive and negative images are used to train a classifier. Examples are classification trees, such as Random Forests [58], and boosting techniques, such as AdaBoost [62]. Other methods include decision trees, SVM's, or multi-instant approaches [7]. A disadvantage of those methods that they need sufficient annotations to work properly. Often the system will

actively select the documents that achieve the maximal information gain [51]. Some vector space models use k-nearest neighbour methods, such as in the studies by Gia et al. [14] and Deselaers et al. [12]. K-NN based methods are shown to be effective, are non-parametric, but run time is slower and it can be very inaccurate when the training set is small.

SVM's are often used [50, 59, 60], but according to Wang et al. [57] SVM-based RF approaches have two major drawbacks: 1) multiple feedback interactions are necessary because of the poor adaptability, flexibility and robustness of the original visual features; 2) positive and negative samples are treated equally, whereas the positive and negative examples provided by the relevance feedback often have distinctive properties, such as that the positive examples are close to each other whereas negative examples are arbitrarily distributed. Within the pseudo relevance feedback, this second point is taken by Jiang et al. [18–20], who use an unsupervised learning approach in which the 'easy' samples are used to learn first and then the 'harder' examples are iteratively added. Regarding SVM's, Xu et al. [59] show that SVM-based methods can work with incrementally refining the user query through relevance feedback. Yang et al. [60] introduce a learning-to-rerank framework in combination with an adapted reranking SVM algorithm. Tao et al. [50] improves on the SVM-based methods using orthogonal complement component analysis (OCCA).

3 Video event retrieval system

Our Video Event Retrieval System is inspired by state-of-the-art video event retrieval systems without training examples [19, 63]. The pipeline of our system is shown in Fig. 1. In our system a user can enter a textual query (*Event Query*) into the search engine. This query is represented by a combination of concepts in the module *Query Interpretation* using the *word2vec* model and the *ConceptBank*. This combination of concepts is propagated back to the user to obtain *relevance feedback* on concept level and the *top n concepts* are used as an OR query in the *Scoring+Ranking* module. This module retrieves the videos in the database, sums the evidence from individual concepts and ranks the results in descending order of estimated relevance. These results are presented back to the user and the user can provide *relevance feedback* on video level. These modules are explained in more depth in the next subsections.

3.1 Query interpretation

The Event Query is translated to a system query (video concept representation) using a word2vec model, which is commonly used in video retrieval [13, 20, 36, 49]. A word2vec model uses a shallow neural network that is trained on a huge dataset, such as Wikipedia, Gigawords, Google News or Twitter, to create semantic word embeddings. The Word2Vec models operate on the hypothesis that words with similar meanings occur in similar contexts [15], resulting in a good performance on associations, such as *king – man + woman = queen*. We use a model that is pre-trained on Google News¹. The embedding of each word is expressed in a 300-dimensional feature vector. This model is used because it shows better results compared to the other pre-trained word2vec models, such as the Wikipedia models. We do not re-train the network, because this did not increase performance in our experiments. Using the word2vec model, we calculate the distance between the event query and

¹<https://code.google.com/archive/p/word2vec/>

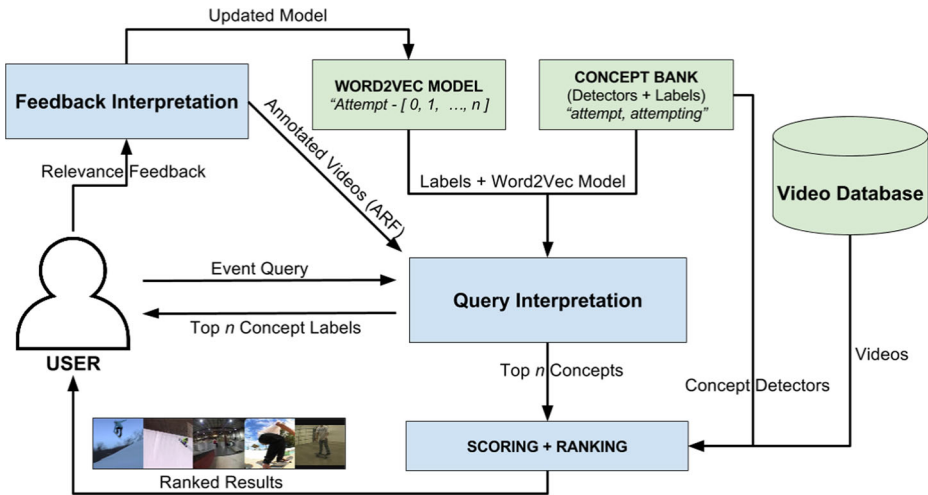


Fig. 1 Pipeline of our Semantic Video Event Search System

each of the concepts that can be detected. The concepts that can be detected are obtained from the ConceptBank (explained in the next subsection). In the word2vec model we calculate the vector of the event query by mean pooling the vectors representing the words in the event query, without using the vectors of stopwords, such as 'a'. If a word in the event query is not in the vocabulary of the word2vec model, we discard this word as well. As shown by Lev et al. [29], mean pooling is a simple pooling method that performs well. The words in the labels of the concepts in our ConceptBank are mean pooled as well and compared to the vector representing the event query. We follow the suggestion to use the cosine similarity, which is a robust similarity measure in this semantic space, to calculate the distance between the event query and each of the concepts that can be detected independently, as explained in (1):

$$\vec{q2c} = L \cdot \frac{\vec{q}}{\|\vec{q}\|}, \quad (1)$$

where L is a matrix with in each row the normalized word2vec vector for the label of detector d ($\frac{l_d}{\|l_d\|}$) and $\frac{\vec{q}}{\|\vec{q}\|}$ is the normalized word2vec vector for the event query.

In our experiments, we create a sparse vector $\vec{q2c}$, because we only keep the values of the top n detectors with the highest similarity measure, and set the other values to zero. This choice is based on initial experiments that show that using all concepts decreases performance. These concepts are used for 1) relevance feedback and 2) scoring.

3.1.1 ConceptBank

The ConceptBank contains labels and detectors that are trained on different datasets using Deep Convolutional Neural Networks (DCNN). We use the eighth layer of the DCNN network trained on the ILSVRC-2012 [11], which is a common strategy in this field [20, 49, 63]. We finetune the architecture on the data in the dataset for SIN [38], Places [64] and TRECVID MED [38] to obtain more concepts (2048) than the 1000 objects used in the

ILSVRC-2012. For finetuning, we use the original network, chop off the last layer (FC8) and create a new FC8 layer with the appropriate amount of nodes, which is amount of concept detectors to be trained. We train the network using the positive and negative examples provided by the dataset. The nodes are labeled using the labels of the positive examples/classes. The concepts from the TRECVID MED are manually annotated on the Research set, comparable to Natarajan et al. [35] and Zhang et al. [63]. We purposely did not use higher level concept detectors, such as those available in the FCVID [22] or Sports [25] dataset, to obtain more interesting experiments using relevance feedback. We, therefore, not aim at highest possible initial ranking, but at a gain with the use of relevance feedback. We believe this is applicable to real world cases, because relevant high level concepts are not always present.

3.2 Scoring and ranking

For the scoring, we use the video scores of the top n concept detectors, obtained from the Query Interpretation module, from our database. The pre-trained concept detectors are applied on each of the videos in our database. Because the network is trained on images, we extract 1 keyframe per 2 seconds uniformly from a video. We use max pooling over these keyframes to obtain a concept detector score per video. Furthermore, we use the average concept detector scores on a background set to normalize the detector scores on the videos in our database.

The scoring function is defined as:

$$s_v = \vec{q2c} \cdot (\vec{cd}_v - \vec{cd}_b), \quad (2)$$

where $\vec{q2c}$ is the query representation in concept space, \vec{cd}_v is the vector of concept detector scores on video v and \vec{cd}_b is the vector of concept detector scores on the background set (average value). In the experiments, the background set is the BACKGROUND set of TRECVID MED, that contains 5000 videos. The videos are returned to the user in descending order of their overall score s_v .

3.3 Feedback interpretation - adaptive relevance feedback (ARF)

Feedback can be obtained on concept level and on video level. We propose an algorithm on video level for explicit relevance feedback, but implementations on concept level are available in our system as well (explained in the experiments).

Our *Adaptive Relevance Feedback* algorithm (ARF) is inspired by the Rocchio algorithm [41]. Different from traditional algorithms on video level [7, 39, 65], we use relevance feedback to update the weights for our concept detectors instead of training a new model based on (few) annotations. We choose to update the weights to make our algorithm more robust to few or noisy annotations. Our perceived system will have limited and unbalanced annotations, because our ‘zero-example’ case is difficult and will have many negative examples on top of the list. With a limited amount of positive examples and a possibly larger amount of negative examples, we still aim to improve retrieval performance based on these annotations. In k-NN methods, noisy annotations can have a high impact on ranking performance. By taking into account the initial concept detector cosine distance to the query, the proposed algorithm is more robust to this type of relevance feedback.

In text retrieval, the scoring function using a vector space model would be:

$$s_D = \vec{Q}' \cdot \vec{D}', \quad (3)$$

where \vec{Q}' is the normalized vector of words in the query with TFIDF [46] as value $\left(\frac{\vec{Q}}{\|\vec{Q}\|}\right)$ and \vec{D}' is vector of words in the document with TFIDF as value $\left(\frac{\vec{Q}}{\|\vec{Q}\|}\right)$.

The Rocchio algorithm is defined as [41]:

$$\vec{Q}_m = (a \cdot \vec{Q}_0) + \left(b \cdot \frac{1}{|D_r|} \cdot \sum_{\vec{D}_j \in D_r} \vec{D}_j \right) - \left(c \cdot \frac{1}{|D_{nr}|} \cdot \sum_{\vec{D}_k \in D_{nr}} \vec{D}_k \right), \quad (4)$$

where \vec{Q}_m is the modified query vector, \vec{Q}_0 is the original query vector, D_r is the set of relevant documents, D_{nr} is the set of non-relevant documents, D_j is a document in the relevant document set, D_k is a document in the non-relevant document set, and a , b and c are parameters.

Translating the Rocchio algorithm to the video retrieval domain, we use videos instead of documents and concepts instead of words. Similar to the Rocchio algorithm, we can change the original query vector $q2c$ using the relevant and non-relevant videos, using:

$$\vec{q2c}_m = (a \cdot \vec{q2c}_0) + \left(b \cdot \frac{1}{|R|} \cdot \sum_{vr \in R} \vec{cd}_{vr} - \vec{cd}_b \right) - \left(c \cdot \frac{1}{|NR|} \cdot \sum_{vn \in NR} \vec{cd}_{vn} - \vec{cd}_b \right), \quad (5)$$

where $\vec{q2c}_m$ is the modified query vector, $\vec{q2c}_0$ is the original query vector, R is the set of relevant videos, NR is the set of non-relevant videos, $\vec{cd}_{v(r/n)}$ is the vector of concept detector scores on video v and \vec{cd}_b is the vector of concept detector scores on the background set (average value) and a , b and c are Rocchio weighting parameters. Similar to the sparse vector used in (1), we only adjust the values of those detectors that are non-zero, i.e. the initial top n detectors. This is based on preliminary experiments.

The adjusted query vector, $\vec{q2c}_m$, is used the scoring function (2), where we substitute the original query vector $\vec{q2c}_0$ for the adjusted query vector $\vec{q2c}_m$. This results in new scores, s'_v , for each video v , which is used to create an updated ranked list of videos.

4 Experiments

In our experiments, we evaluate our proposed methods in an international video retrieval benchmark and compare performance to state of the art.

4.1 Experimental set-up

We use the MEDTRAIN data set from the TRECVID Multimedia Event Detection (MED) benchmark [38]. This data set contains 5594 videos of user-generated content. The MEDTEST set is often used in other papers to report performance on, but the MEDTRAIN contains relevance judgments for forty events (i.e. queries), whereas MEDTEST contains judgments for only twenty events. Although we purposely did not use higher level concept detector datasets to obtain our concepts, some concepts caused a (near-) perfect performance

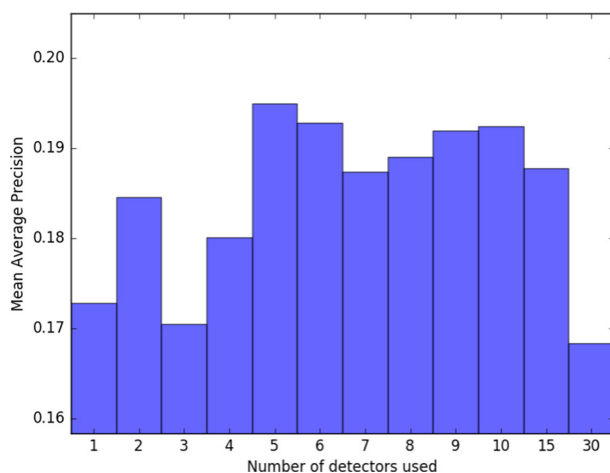


Fig. 2 MAP per amount of concept detectors over all events

because of a direct match between an event and the concept. We, therefore, excluded eight of the forty events ². These events are not interesting for the user feedback experiments.

The number of concepts n for ARF is chosen to be 30. Our baseline experiments showed highest performance for $n = 5$ as shown in Fig. 2, but our experiments showed that a higher performance gain can be achieved by using more concepts. Furthermore, the parameter a and b are set to 1.0 and c is set to 0.5, which is in line with text-information retrieval [41]. Visualizations of these results can be found in Fig. 3. All values for b and c from 0.0 to 2.0 with step size of 0.1 are tested on the 32 events in the MEDTRAIN dataset.

4.1.1 Evaluation

Mean Average Precision (MAP) [38], which is the official performance measure in the TRECVID MED task, is used to measure performance. With relevance feedback on video level, the positively annotated videos will remain on the top of the list and, thus, increase MAP. It is, however, also interesting to know whether the algorithm is able to retrieve new relevant videos. This is why we introduce a MAP variant. MAP^* calculates MAP disregarding the videos that have been viewed already by the user. We assume that a user has viewed all videos up to the last annotated video.

Additionally, we calculate robustness of our proposed method compared to the best state of the art method on that level by the robustness index (RI) [45] and the concept level methods against the initial ranking using:

$$RI = \frac{|Z_P| - |Z_N|}{|Z|}, \quad (6)$$

where $Z_P - Z_N$ is the amount of queries in which the first method has higher performance compared to the second method, and $|Z|$ is the total number of queries.

²excluded events are *Wedding ceremony*; *Birthday party*; *Making a sandwich*; *Hiking*; *Dog show*; *Town hall meeting*; *Beekeeping*; *Tuning a musical instrument*

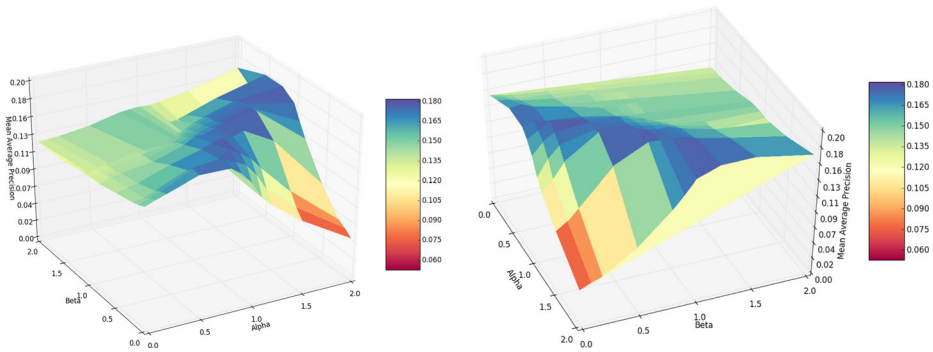


Fig. 3 MAP^* relative to b and c values

4.1.2 User interface

To provide the user with a quick and efficient way of viewing the concepts and the videos in the experiment, we designed a User Interface (*UI*). A screenshot of the UI is presented in Fig. 4. For the videos, we aim to show a small subset of the keyframes instead of the whole video. For each video, the 5 keyframes with the highest scores over the top n detectors are selected. The number 5 is based on state of the art in current search engines, such as Bing³. A single frame is shown initially for each video in a container, under which we presented the relevance selection tools. When a user moves the mouse over the container, a new frame appears based on the relative position of the mouse in the container. This means that the first frame would be visible when the user was hovering in the first 20% of the container, the second frame when the mouse position was detected in the next 20%, and so on. This enabled our users to get a quick overview of the relevant parts of the video, without having to spend minutes watching each video. For the feedback on the concepts, the videos were not shown and a list of the top 15 concept detectors was shown. This is further explained in the next section.

4.2 Relevance feedback on concepts

Fifteen participants (12 male; 3 female; $\mu_{age} = 24.87$; $\sigma_{age} = 3.739$) were asked to volunteer in providing relevance feedback. The majority of the participants were non-native but fluent English speakers and an education level of Bachelors or higher. The participants were presented with a list of the 32 events on several pieces of paper with the top 15 concepts (in English) per event as provided by the initial system. They were asked to evaluate these concepts and provide relevance judgments by marking the non-relevant concepts for each of the events. On average, participants marked 6.2 out of 15 concept detectors as non-relevant ($\sigma = 1.494$). The average number of detectors marked as non-relevant differed greatly per event (minimum 0.5 to maximum 11.7) and per user (minimum 3.7 to maximum 8.7). A Fleiss' Kappa test was performed to determine user agreement in the flagging

³www.bing.com/videos

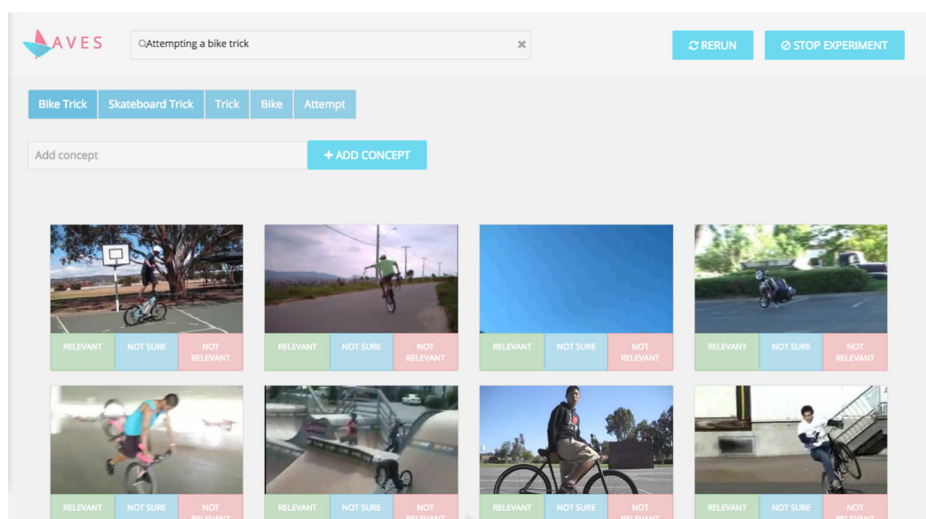


Fig. 4 Screenshot of our User Interface for the event ‘Attempting a bike trick’

of non-relevant concepts, which resulted in $\kappa = 0.514$. According to the Landis and Koch scale [27], this indicates a *moderate agreement* among users.

4.3 Relevance feedback on videos

For the relevance feedback on videos, a group of ten male participants ($\mu_{age} = 26.3$, $\sigma_{age} = 1.567$) with mainly non-native but fluent English speakers and an education level of Bachelors or higher without dyslexia, colour-blindness, concentration problems, or RSI problems, voluntarily participated in an experiment. The task of the participants was to select relevant and non-relevant videos in our UI. 24 results were shown initially, and more could automatically be loaded by scrolling to the bottom of the page. The experiment consisted of two conditions, which correspond to the re-ranking results by ARF and the k-NN method named RS (next subsection). In each of the conditions, 16 queries, randomly assigned using a Latin rectangle [6], were presented to the user using our UI, after which they performed relevance feedback on the retrieved videos.

4.4 Baseline methods

We compare our ARF algorithm with several baselines, which are presented in the next subsections. The SVM-based methods are not included in this paper, because preliminary experiments showed that on average performance is poor due to limited amount of positive samples. Due to our use case, we cannot use the newer Fisher representation, because we work on the ‘zero example’ case without a fisher vector representation.

4.4.1 No feedback

The No Feedback method is the system without the relevance feedback module. The number of concepts n is chosen to be 5, based on the results reported in Fig. 2.

4.4.2 Manual

An expert familiar with the TRECVID MED events, the ConceptBank and dataset was asked to select a set of relevant concepts and their weights for each event. The number of selected concepts varies among the events.

4.4.3 Concept level - AlterWeights

As a re-weighting method, we alter the weights of the concept detectors following an approach inspired by the Rocchio algorithm [41]. The weights of the relevant detectors are increased, whereas the weights of the irrelevant detectors are decreased following (7). The values for γ and δ are the best values based on our experiments on the MEDTRAIN dataset, testing all values between 0.0 and 2.0 with a stepsize of 0.1. These parameters, thus, provide an upper bound performance. This method is different from ARF, because this method works on the relevance feedback on concept level and not on video level. The number of concepts n for all concept level based experiments is set to 15, because previous experiments showed that a higher amount of concepts in relevance feedback can achieve higher performance gain compared to using only the top 5 (often positive) concepts.

$$q2c_{d,m} = \begin{cases} q2c_{d,0} + \gamma \cdot q2c_{d,0}, & \text{if } d \text{ is relevant.} \\ q2c_{d,0} - \delta \cdot q2c_{d,0}, & \text{otherwise.} \end{cases}, \quad (7)$$

where $\gamma = 0.4$ and $\delta = 0.9$.

4.4.4 Concept level - QuerySpace

As a QPM method, we change the semantic space of the query using the Rocchio algorithm. Using the vector representations of both the relevant and non-relevant detectors provided by concept level relevance feedback, we update the initial query vector \vec{q}_0 that is used to calculate the cosine similarity with the available concepts $\vec{q}2c$ ((1) in Section 3.2) according to (8). Again, the values for ϵ and ζ are the best values based on our experiments on the MEDTRAIN to provide optimal performance, using all values between 0.0 and 2.0 with a step size of 0.1.

$$\vec{q}_m = \vec{q}_0 + \epsilon \cdot \left(\frac{1}{|C_r|} \sum_{dr \in C_r} \vec{v}_{dr} \right) - \zeta \cdot \left(\frac{1}{|C_{nr}|} \sum_{dn \in C_{nr}} \vec{v}_{dn} \right), \quad (8)$$

where \vec{q}_m is the modified query vector, C_r and C_{nr} are the set of relevant and non-relevant concept detectors, respectively and $\vec{v}_{d(r/n)}$ is the word2vec vector representation of detector d , $\epsilon = 0.6$ and $\zeta = 0.7$.

4.4.5 Concept level - DetectorSpace

Instead of changing the query space, we can also change the semantic space. We change the concept detector labels (l_d) by moving the vector of the relevant concepts toward the vector of the event query (\vec{q}), whereas we move the non-relevant concepts away from the event query with the following equation:

$$\vec{l}_{d,m} = \vec{l}_{d,0} + \eta \cdot \theta_d \cdot (\vec{q} - \vec{l}_{d,0}), \quad (9)$$

Table 1 MAP and Standard Deviation over all users and all events on MEDTRAIN dataset

	Method	MAP (μ)	Standard Deviation (σ)
Baseline	No Feedback	0.19	0.15
	Manual	0.23	0.18
Concept Level	AlterWeights	0.21	0.16
	QuerySpace	0.20	0.16
	DetectorSpace	0.19	0.15
Video Level	RS	0.20	0.17
	ARF	0.24	0.17

where $\vec{l}_{d,m}$ is the new label vector for detector d , $\vec{l}_{d,0}$ is the old vector of detector d and \vec{q} is the event query vector, $\eta = 0.1$, θ is described as:

$$\theta_d = \begin{cases} -1, & \text{if } d \in C_{nr} \\ 1, & \text{otherwise,} \end{cases} \quad (10)$$

where d is the detector, C_{nr} is the set of non-relevant concept detectors.

This new vector is used to calculate the $q2c$, which is used in the determination of the relevant concepts and the scoring function ((2) in Section 3.2). This method changes the concepts in the space and, therefore, this method can change performance on other events, whereas in the other methods the performance on only one query is improved. This method, however, introduces different results for different order of events. In our experiments, we choose the average performance over 2 runs of 32 events over all 15 users.

4.4.6 Video level - RS

The final baseline is a k-NN based relevance feedback algorithm named *Relevance Score* (RS). The RS algorithm is well-performing in image retrieval [12, 14] and the relevance score $relevance(v)$ of a video v calculated as

$$relevance(v) = \left(1 + \frac{dR(v)}{dNR(v)} \right)^{-1}, \quad (11)$$

where dR is the dissimilarity, measured as Euclidean distance, from the nearest video in relevant video set R , dNR is the dissimilarity from the nearest video in non-relevant video set NR . The video set is ordered such that the videos with the highest *relevance score* are listed first and MAP is calculated on this list.

Table 2 MAP* scores and standard deviations on video level on MEDTRAIN dataset

Algorithm	MAP* (μ)	σ
No Feedback	0.13	0.01
RS	0.11	0.02
ARF	0.15	0.02

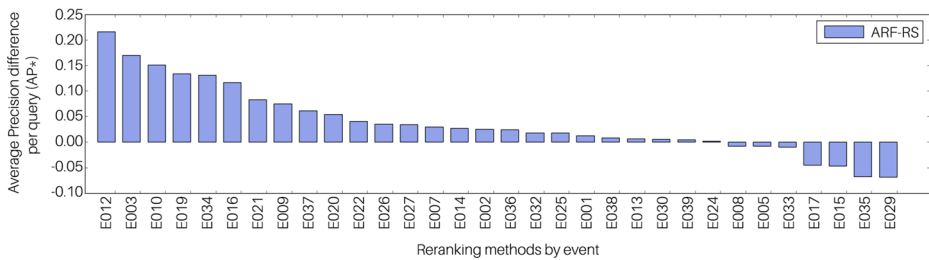


Fig. 5 Average precision difference (AP^*) per event

5 Results

5.1 MAP and MAP*

The MAP results on all methods are displayed in Table 1. The results show superior performance for our ARF method. All relevance feedback methods outperform the No Feedback run, except DetectorSpace.

This comparison is, however, not completely fair, because annotations on video level will keep the positively annotated videos on the top of the ranked list. One method to overcome this problem is to discard the videos which the users have already seen (MAP^*). We assume that all videos displayed before the last video are seen. The results in $\%MAP^*$ over all video level methods, including the initial method without these videos is presented in Table 2.

These results show that RS performs worse compared to No Feedback, because this method might move in the wrong direction when little positive examples are annotated. A Shapiro-Wilk test showed that the precision score distributions do not deviate significantly from a normal distribution at $p > 0.05$ ($p = 0.813$; $p = 0.947$; $p = 0.381$, for No Feedback, RS, and ARF respectively). A statistically significant difference between groups was determined by a one-way ANOVA ($F(2,27) = 18.972$, $p < 0.0005$). A post-hoc Tukey's HSD test was performed to verify intergroup differences. The means of all algorithms differed significantly at $p < 0.05$ ($p = 0.006$; $p = 0.01$; $p < 0.0005$, for No Feedback-RS, No Feedback-ARF, and RS-ARF, respectively).

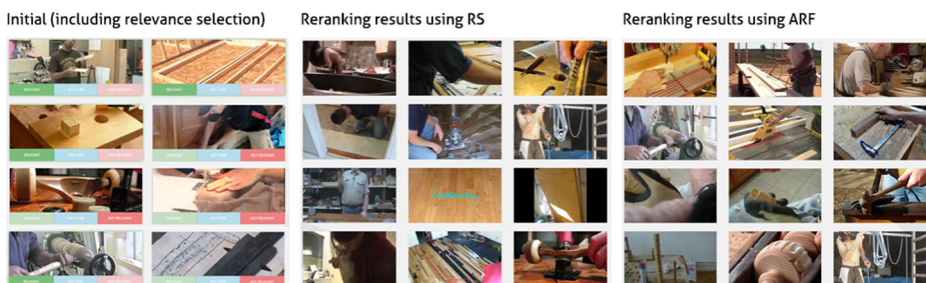


Fig. 6 Example of returned results for the query *Working on a woodworking project*. The initial result set on the left also shows relevance selection

Table 3 Comparison Concepts and weights for Attempting a board trick

No Feedback (0.19)		Manual (0.31)		AlterWeights (0.20)		ARF (0.25)	
<i>c</i>	<i>c_s</i>	<i>c</i>	<i>c_s</i>	<i>c</i>	<i>c_s</i>	<i>c</i>	<i>c_s</i>
attempt	0.65	skateboard trick	0.33	trick	0.88	board game	0.72
trick	0.63	surf	0.33	board2	0.81	skateboard trick	0.54
board1	0.58	snowboard	0.33	skateboard trick	0.76	board1	0.43
board2	0.58			board game	0.74	trick	0.38
skateboard trick	0.54			flipping	0.44	attempt	0.13

5.2 Robustness

The robustness index (RI) on concept level, compared to No Feedback, is $RI = 0.125$ for AlterWeights (better in 18 events), $RI = -0.375$ for QuerySpace (better in 9 events) and $RI = -0.0625$ for DetectorSpace (better in 15 events). Interestingly, QuerySpace has higher performance compared to DetectorSpace, although RI is lower. One reason is that in some events DetectorSpace has moved a concept in a wrong direction by which it is not able to retrieve that concept anymore, resulting in a lower MAP.

The RI on video level is calculated by comparing RS to ARF. The RI for ARF compared to RS is $RI = 0.6875$ (better in 27 events), and $RI = -0.6875$ for RS (better in 5 events). The bar plot is shown in Fig. 5. Compared to No Feedback ARF improves ranking in 23 of the events ($RI = 0.4375$) and RS in 12 of the events ($RI = -0.25$).

Giving an example of results of the methods, Fig. 6 shows the different results from the video level methods.

Table 3 shows the weights of the top 5 concepts for the baseline and the best method for the concept level and video level for the event *Attempting a board trick*. These results show that the manual annotator is able to capture all type of board tricks, such as skateboard, surfboard and snowboard tricks. AlterWeights does not have the general concept *attempt* or two board concepts as the No Feedback, but added the concepts *flipping* (highly relevant) and *board game* (semantically discussable relevant). ARF also has the concept *board game*, even on top of the list. This indicates that the detector has relevance for this event. The concept *attempt* is moved to the bottom of the list.

6 Conclusions and future work

Results show that relevance feedback on both concept and video level improves performance compared to using no relevance feedback; relevance feedback on video level obtains higher performance compared to relevance feedback on concept level; our proposed ARF method on video level outperforms a state of the art k-NN method, all methods on concept level and even manual selected concepts.

Our results are, however, bound to few events and few users. For the concept level method, we also use an indirect performance metric, because we obtain performance on video level. We, thus, do not take into account that relevant concepts can have bad performing detectors. We believe that these experiments clearly show that although concept level user feedback can improve performance upon the initial ranking, video level user feedback

is more valuable. One reason might be that this feedback can provide information on both the relevance of the concept semantically and the accuracy of the concept detector. In future work it might be interesting to investigate whether we can distinguish whether the concept detector is not accurate or whether the concept is not semantically related based on the video level feedback.

Additionally, we do not fully benefit from the insights in text retrieval. Whereas in text retrieval the result of the Rocchio algorithm are used in a cosine similarity, our results are used in a non-normalized scoring function. Because we score the videos per query and evaluate using a ranking, the normalization of the query does not produce different (ranking) results. Rocchio is used to change the query vector and, thus, using Rocchio on a non-normalized query vector does not hurt performance. Normalization on the concept detector scores, however, does decrease performance, because it is dominated by the many irrelevant concepts. A first step in normalization is established by using the background score, which might resemble a term frequency in text retrieval (although an aggregated normalized score over the different keyframes could be a better measure compared to our normalization after max pooling). The IDF part of TFIDF is not yet taken into account. In future work, it would be interesting to investigate how to properly normalize the concept detector scores to fully exploit the insights from text retrieval.

Acknowledgments We would like to thank the VIREO team from the City University of Hong Kong for the application of their concept detectors on the TRECVID MED 2014 data sets and the ERP Making Sense of Big Data (MSoBD) for their financial support. Furthermore, we would like to thank Thomas Mensink and Robin Aly for their feedback on the work of Douwe and Geert that initiated this journal paper.

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

References

1. Bay H, Ess A, Tuytelaars T, Van Gool L (2008) Speeded-up robust features (surf). *Comp Vision Image Underst* 110(3):346–359
2. de Boer M, Daniele L, Brandt P, Sappelli M (2015) Applying semantic reasoning in image retrieval. *Proceedings of the ALLDATA*
3. de Boer M, Schutte K, Kraaij W (2016) Knowledge based query expansion in complex multimedia event detection. *Multimedia Tools and Applications* 75(15):9025–9043
4. Burgess J, Green J (2013) *YouTube: Online video and participatory culture*, Wiley. ISBN-13: 978-0745644790
5. Chen MY, Hauptmann A (2009), *Mosift: Recognizing human actions in surveillance videos*
6. Cochran WG, Cox GM (1957) *Experimental designs*
7. Crucianu M, Ferecatu M, Boujemaa N (2004) Relevance feedback for image retrieval: a short survey. *Report of the DELOS2 European Network of Excellence (FP6)*
8. Dalal N, Triggs B (2005) Histograms of oriented gradients for human detection. In: *CVPR*, vol 1. IEEE, pp 886–893
9. Dalal N, Triggs B, Schmid C (2006) Human detection using oriented histograms of flow and appearance. In: *European conference on computer vision*. Springer, pp 428–441
10. Dalton J, Allan J, Mirajkar P (2013) Zero-shot video retrieval using content and concepts. In: *Proceedings of the 22nd International Conference on information & knowledge management*. ACM, pp 1857–1860
11. Deng J, Dong W, Socher R, Li LJ, Li K, Fei-Fei L (2009) Imagenet: a large-scale hierarchical image database. In: *CVPR*. IEEE, pp 248–255

12. Deselaers T, Paredes R, Vidal E, Ney H (2008) Learning weighted distances for relevance feedback in image retrieval. In: 19th International Conference on pattern recognition. IEEE, pp 1–4
13. Elhoseiny M, Liu J, Cheng H, Sawhney H, Elgammal A (2015) Zero-shot event detection by multimodal distributional semantic embedding of videos. Preprint arXiv:[1512.00818](#)
14. Gia G, Roli F et al (2004) Instance-based relevance feedback for image retrieval. In: Advances in neural information processing systems, pp 489–496
15. Goldberg Y, Levy O (2014) word2vec explained: Deriving mikolov et al.'s negative-sampling word-embedding method. Preprint arXiv:[1402.3722](#)
16. Habibian A, Mensink T, Snoek CG (2014) Videostory: a new multimedia embedding for few-example recognition and translation of events. In: Proceedings of International Conference on multimedia. ACM, pp 17–26
17. Jegou H, Perronnin F, Douze M, Sánchez J, Perez P, Schmid C (2012) Aggregating local image descriptors into compact codes. Trans on Pattern Analysis and Machine Intelligence 34(9):1704–1716
18. Jiang L, Meng D, Mitamura T, Hauptmann AG (2014) Easy samples first: Self-paced reranking for zero-example multimedia search. In: Proceedings of the ACM International Conference on multimedia. ACM, pp 547–556
19. Jiang L, Mitamura T, Yu SI, Hauptmann AG (2014) Zero-example event search using multimodal pseudo relevance feedback. In: Proceedings of International Conference on multimedia retrieval. ACM, p 297
20. Jiang L, Yu SI, Meng D, Mitamura T, Hauptmann AG (2015) Bridging the ultimate semantic gap: a semantic search engine for internet videos. In: ACM International Conference on multimedia retrieval, pp 27–34
21. Jiang YG, Bhattacharya S, Chang SF, Shah M (2013) High-level event recognition in unconstrained videos. Int J of Multimedia Information Retrieval 2(2):73–101
22. Jiang YG, Wu Z, Wang J, Xue X, Chang SF (2015) Exploiting feature and class relationships in video categorization with regularized deep neural networks. Preprint arXiv:[1502.07209](#)
23. Jiang YG, Yang J, Ngo CW, Hauptmann AG (2010) Representations of keypoint-based semantic concept detection: a comprehensive study. Trans on Multimedia 12(1):42–53
24. Kaliciak L, Song D, Wiratunga N, Pan J (2013) Combining visual and textual systems within the context of user feedback. In: Advances in multimedia modeling. Springer, pp 445–455
25. Karpathy A, Toderici G, Shetty S, Leung T, Sukthankar R, Fei-Fei L (2014) Large-scale video classification with convolutional neural networks. In: CVPR, pp 1725–1732
26. Klaser A, Marszałek M, Schmid C (2008) A spatio-temporal descriptor based on 3d-gradients. In: BMVC. British Machine Vision Association, pp 275–1
27. Landis JR, Koch GG (1977) The measurement of observer agreement for categorical data. biometrics
28. Lavtey I (2005) On space-time interest points. Int J of computer vision 64(2-3):107–123
29. Lev G, Klein B, Wolf L (2015) In defense of word embedding for generic text representation. In: Natural language processing and information systems. Springer, pp 35–50
30. Liu Y, Zhang D, Lu G, Ma WY (2007) A survey of content-based image retrieval with high-level semantics. Pattern Recogn 40(1):262–282
31. Lowe DG (2004) Distinctive image features from scale-invariant keypoints. Int J of computer vision 60(2):91–110
32. Lv Y, Zhai C (2010) Positional relevance model for pseudo-relevance feedback. In: Proceedings of the 33rd International ACM SIGIR conference on research and development in information retrieval. ACM, pp 579–586
33. Mikolov T, Sutskever I, Chen K, Corrado GS, Dean J (2013) Distributed representations of words and phrases and their compositionality. In: Advances in neural information processing systems, pp 3111–3119
34. Mironică I, Ionescu B, Uijlings J, Sebe N (2016) Fisher kernel temporal variation-based relevance feedback for video retrieval. Comput Vis Image Underst 143:38–51
35. Natarajan P, Natarajan P, Manohar V, Wu S, Tsakalidis S, Vitaladevuni SN, Zhuang X, Prasad R, Ye G, Liu D et al (2011) Bbn viser trecvid 2011 multimedia event detection system. In: NIST TRECVID Workshop
36. Norouzi M, Mikolov T, Bengio S, Singer Y, Shlens J, Frome A, Corrado GS, Dean J (2013) Zero-shot learning by convex combination of semantic embeddings. Preprint arXiv:[1312.5650](#)
37. Ojala T, Pietikainen M, Maenpaa T (2002) Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Transactions on pattern analysis and machine intelligence 24(7):971–987

38. Over P, Awad G, Michel M, Fiscus J, Sanders G, Kraaij W, Smeaton AF, Quenot G, Ordelman R (2015) TRECVID 2015 – an overview of the goals, tasks, data, evaluation mechanisms and metrics. In: Proceedings TRECVID 2015. NIST, USA, p 52
39. Patil PB, Kokare MB (2011) Relevance feedback in content based image retrieval: a review. *Journal of Applied Computer Science & Mathematics* 10(10):40–47
40. Patil S (2012) A comprehensive review of recent relevance feedback techniques in CBIR. *International Journal of Engineering Research & Technology (IJERT)* 1(6)
41. Rocchio JJ (1971) Relevance feedback in information retrieval
42. Rocha R, Saito PT, Bugatti PH (2015) A novel framework for content-based image retrieval through relevance feedback optimization. In: *Progress in pattern recognition, image analysis, computer vision, and applications*. Springer, pp 281–289
43. Rui Y, Huang TS, Ortega M, Mehrotra S (1998) Relevance feedback: a power tool for interactive content-based image retrieval. *Trans on circuits and systems for video technology* 8(5):644–655
44. Ruthven I, Lalmas M (2003) A survey on the use of relevance feedback for information access systems. *The Knowledge Engineering Review* 18(02):95–145
45. Sakai T, Manabe T, Koyama M (2005) Flexible pseudo-relevance feedback via selective sampling. *Trans on Asian Language Information Processing (TALIP)* 4(2):111–135
46. Salton G, Buckley C (1988) Term-weighting approaches in automatic text retrieval. *Inf Process Manag* 24(5):513–523
47. Sánchez J, Perronnin F, Mensink T, Verbeek J (2013) Image classification with the fisher vector: Theory and practice. *Int J comput vis* 105(3):222–245
48. Schmidhuber J (2015) Deep learning in neural networks: an overview. *Neural Netw* 61:85–117
49. Snoek CG, Cappallo S, Fontijne D, Julian D, Koelma DC, Mettes P, Sande K, Sarah A, Stokman H, Towal RB et al (2015) Qualcomm Research and University of Amsterdam at TRECVID 2015: Recognizing concepts, objects and events in video
50. Tao D, Tang X, Li X (2008) Which components are important for interactive image searching? *IEEE Transactions on Circuits and Systems for Video Technology* 18(1):3–11
51. Tong S, Chang E (2001) Support vector machine active learning for image retrieval. In: *Proceedings of the 9th ACM International Conference on multimedia*. ACM, pp 107–118
52. Tsai CF, Hu YH, Chen ZY (2015) Factors affecting rocchio-based pseudorelevance feedback in image retrieval. *Journal of the Association for Information Science and Technology* 66(1):40–57
53. Tzelepis C, Ma Z, Mezaris V, Ionescu B, Kompatsiaris I, Boato G, Sebe N, Yan S (2016) Event-based media processing and analysis: a survey of the literature. *Image Vis Comput* 53:3–19
54. Wang H, Kläser A, Schmid C, Liu CL (2011) Action recognition by dense trajectories. In: *CVPR. IEEE*, pp 3169–3176
55. Wang H, Kläser A., Schmid C, Liu CL (2013) Dense trajectories and motion boundary descriptors for action recognition. *Int J of computer vision* 103(1):60–79
56. Wang H, Schmid C (2013) Action recognition with improved trajectories. In: *Proceedings of the International Conference on computer vision*, pp 3551–3558
57. Wang XY, Liang LL, Li WY, Li DM, Yang HY (2016) A new SVM-based relevance feedback image retrieval using probabilistic feature and weighted kernel function. *J Vis Commun Image Represent* 38:256–275
58. Wu Y, Zhang A (2004) Interactive pattern analysis for relevance feedback in multimedia information retrieval. *Multimedia Systems* 10(1):41–55
59. Xu S, Li H, Chang X, Yu SI, Du X, Li X, Jiang L, Mao Z, Lan Z, Burger S et al (2015) Incremental multimodal query construction for video search. In: *Proceedings of the 5th ACM on International Conference on multimedia retrieval*. ACM, pp 675–678
60. Yang L, Hanjalic A (2010) Supervised reranking for web image search. In: *Proceedings of the International Conference on multimedia*. ACM, pp 183–192
61. Ye G, Li Y, Xu H, Liu D, Chang SF (2015) Eventnet: a large scale structured concept library for complex event detection in video. In: *Proceedings of the 23rd International Conference on multimedia*. ACM, pp 471–480
62. Yu J, Lu Y, Xu Y, Sebe N, Tian Q (2007) Integrating relevance feedback in boosting for content-based image retrieval. In: *ICASSP, vol 1. IEEE*, pp I–965
63. Zhang H, Lu YJ, de Boer M, ter Haar F, Qiu Z, Schutte K, Kraaij W, Ngo CW (2015) VIREO-TNO @ TRECVID 2015: Multimedia event detection. In: *Proceedings of TRECVID 2015*

64. Zhou B, Lapedriza A, Xiao J, Torralba A, Oliva A (2014) Learning deep features for scene recognition using places database. In: Advances in neural information processing systems, pp 487–495
65. Zhou XS, Huang TS (2003) Relevance feedback in image retrieval: a comprehensive review. *Multimedia systems* 8(6):536–544



Maaïke de Boer received her BS and MS degrees in Artificial Intelligence from the University of Utrecht, The Netherlands, in 2011 and 2013. Currently, she is a PhD student in the Department of Intelligent Imaging at TNO, The Hague, The Netherlands and the Institute for Computing and Information Sciences at the Radboud University, Nijmegen, The Netherlands. Her present research interests include multimedia event detection, semantic analysis and knowledge bases.



Geert Pingen obtained his BSc in Artificial Intelligence from the Radboud University of Nijmegen in 2013, and is currently finishing his MSc thesis in HMI on Machine Learning for Ground Cover and Hot Target Analysis in RGB and Satellite Imagery at the University of Twente, the Department of Intelligent Imaging at TNO, and the Faculty of Geo-Information Science and Earth Observation (ITC). He is presently working at the Department of Monitoring and Control Services at TNO. His professional interests include machine learning, deep neural networks, and image processing.



Douwe Knook received a BA in Media & Culture (New Media track) from the University of Amsterdam in 2015. In 2016 he received a MSC in Information Studies (Human Centered Multimedia track) from the same university. His contribution to the research presented in this paper was part of his master thesis.



Klammer Schutte received an MS degree in physics from the University of Amsterdam in 1989 and a PhD degree from the University of Twente, Enschede, The Netherlands, in 1994. He had a postdoctoral position with the Delft University of Technology's Pattern Recognition (now Quantitative Imaging) group. Since 1996, he has been employed by TNO, currently as lead research scientist of intelligent imaging. Within TNO he has actively led multiple projects in areas of signal and image processing. Recently, he has led many projects, including super-resolution reconstruction for both international industries and governments, resulting in super-resolution reconstruction based products in active service. His research interests include behavior recognition, pattern recognition, sensor fusion, image analysis, and image restoration.



Wessel Kraaij Prof. Wessel Kraaij is a principal scientist working for TNO, The Hague, the Netherlands since 1995 and PI of the Data Science research group. In 2008 he was appointed as full professor in ‘information filtering and aggregation’ at Radboud University Nijmegen. In 2016 he was appointed full professor ‘applied data analytics’ at Leiden University. He is lead scientist of a 8MEuro research programma on Making Sense of Big Data. He is currently coordinator of a large (7MEuro) Dutch national funded project in the area of aggregating heterogeneous sensor related to health and well-being data, with a focus on recognizing temporal patterns that help individuals to self-regulate their lifestyle. He is also active in the field of multimedia retrieval (co-coordinating the global NIST TRECVID multimedia retrieval benchmark since 2003). A recent initiative is the Prana Data project with Academic hospitals, patient organizations and SME’s, on the topic of privacy preserving data analysis from distributed data repositories. Wessel Kraaij published over 120 scholarly papers, received several best paper and 10 year achievement awards and is an editor for several journals.