# Relevance Filtering meets Active Learning: Improving Web-based Concept Detectors

Damian Borth
University of Kaiserslautern
D-67663 Kaiserslautern,
Germany
d_borth@cs.uni-kl.de

Adrian Ulges
German Research Center for
Artificial Intelligence (DFKI)
D-67663 Kaiserslautern,
Germany
adrian.ulges@dfki.de

Thomas M. Breuel
University of Kaiserslautern
D-67663 Kaiserslautern,
Germany
tmb@cs.uni-kl.de

## ABSTRACT

We address the challenge of training visual concept detectors on web video as available from portals such as YouTube. In contrast to high-quality but small manually acquired training sets, this setup permits us to scale up concept detection to very large training sets and concept vocabularies. On the downside, web tags are only *weak* indicators of concept presence, and web video training data contains lots of non-relevant content.

So far, there are two general strategies to overcome this *label noise* problem, both targeted at discarding non-relevant training content: (1) a manual refinement supported by *active learning* sample selection, (2) an automatic refinement using *relevance filtering*. In this paper, we present a highly efficient approach combining these two strategies in an interleaved setup: manually refined samples are directly used to improve relevance filtering, which again provides a good basis for the next active learning sample selection.

Our results demonstrate that the proposed combination – called *active relevance filtering* – outperforms both a purely automatic filtering and a manual one based on active learning. For example, by using 50 manual labels per concept, an improvement of 5% over an automatic filtering is achieved, and 6% over active learning. By annotating only 25% of weak positive samples in the training set, a performance comparable to training on ground truth labels is reached.

## Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Retrieval and Indexing

## General Terms

Algorithms, Measurement, Experimentation

## Keywords

Content-based Video Retrieval, Concept Detection

*MIR'10,* March 29–31, 2010, Philadelphia, Pennsylvania, USA.
Copyright 2010 ACM 978-1-60558-815-5/10/03 ...$10.00.

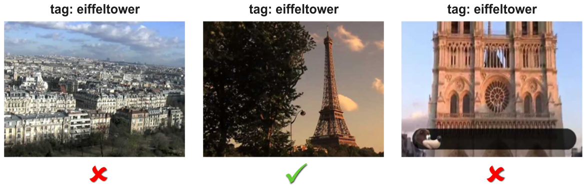

**Figure 1:** Sample frames from YouTube clips tagged with "eiffeltower". While some frames do show the concept (center), other content is *non-relevant*. This poses a challenge for concept detector training.

## 1. INTRODUCTION

As digital video has become an important source of information and entertainment to millions of users, databases grow larger and larger [28], and retrieval becomes a difficult challenge. This is particularly due to the *semantic gap* [22], the discrepancy between low-level features of a video signal on the one hand and the viewer's high-level interpretation of the video on the other.

To bridge this gap, *concept detection* has been proposed, which aims at automatically mining video collections for semantic concepts such as objects ("airplane"), scene types ("cityscape"), and activities taking place ("interview"). Concept detection has been studied intensively over the last years (for an overview, see [23]) and is a key building block of various video search prototypes [4, 30, 33]. However, the effort associated with acquiring training samples for many concepts causes a scalability problem: the size of concept vocabularies remains limited, and keeping track to dynamic changes of users' information needs (e.g. as new concepts of interest like "President Obama" emerge) is difficult.

This raises the question whether a manual acquisition of training material can be substituted with other information sources. One such source is *web video*, which is available at a large scale from portals like YouTube[1]. Web video content is usually enriched with user-generated *tags*, which indicate the presence of concepts in a clip. Utilizing this tag information as class labels, concept detection systems could automatically harvest training material from the web and thus perform a scalable and dynamic concept learning [10, 20, 25, 27].

Unfortunately, YouTube tags are coarse and unreliable and therefore difficult to utilize as label information. An

[1] www.youtube.com

example is given in Figure 1, which illustrates that not all YouTube material tagged with "eiffeltower" does in fact show the concept. This is due to several reasons: first, annotation behavior is subjective, and – though a concept may seem present to a specific user with certain knowledge and expectations – it may not be in general. Second, web video tags – which are usually given on a global scope – do not tell us *when* in a video the concept appears.

Consequently, training sets acquired from web video portals are noisy and contain only a certain amount of truly relevant material (typically, between 20% and 50%, as we estimated using manual sample annotations). Training concept detectors on such weakly labeled data must be expected to come with significant performance loss [10, 20].

One straightforward strategy to overcome this problem would be to manually refine the raw web-based training set and discard non-relevant content. While this has been demonstrated to improve the performance of the resulting concept detectors [20], it is very time-consuming and does not scale. To reduce manual annotation effort to some extent, **active learning** strategies have been proposed [1, 3]: instead of annotating the whole dataset, manual labels are only given for a subset of "most informative" samples. This has been demonstrated to achieve remarkable time savings when learning concepts from TV-based datasets [2]. In the context of web data filtering, however, active learning does not make optimal use of the given labels: these are employed to update the classifier, but remain neglected as a valuable clue for filtering noise in the training set. This raises the question if we can extend active learning for a better filtering of web-based training sets.

A second, alternative solution is to filter noise material automatically [9, 13, 26, 32]. This approach has been referred to as *relevance learning* [13] or **relevance filtering** [26]. Its core idea is to identify non-relevant content automatically based on its distribution in feature space and discard it during system training. However, such automatic relevance filtering systems do not reach the accuracy of a careful manual labeling. Therefore, it seems reasonable to assume that relevance filtering could benefit from a few manually provided labels – the question is how improvements can be achieved with minimal human intervention.

The key contribution of this paper is a novel combination of both approaches – active learning and relevance filtering – to a joint method, which we will refer to as **active relevance filtering** in the following. We propose an interleaved setup of active learning label refinement and automatic relevance filtering. This way, the web-based training set is refined both manually and automatically.

Using the proposed approach, we demonstrate that concept detectors trained on weakly labeled web material can be improved significantly with a minimum of human supervision. Also, our results show that the proposed active relevance filtering outperforms both a purely automatic noise removal and a standard manual refinement by active learning.

This paper is organized as follows: we first discuss related work in the context of visual learning from web data (Section 2). After this, the proposed active relevance filtering framework is introduced (Section 3) and evaluated in quantitative experiments on real-world web video data (Section 4). A discussion concludes the paper (Section 5).

## 2. RELATED WORK

Though visual learning from web content is clearly a challenging problem, this information source has been acknowledged as an attractive basis for training flexible and scalable visual recognition systems. Its exploitation is now an active area of research [8, 18, 20, 24, 26, 27, 34].

Similar to web video, web images as acquired from search engines or portals like Flickr contain significant amounts of non-relevant material. In case of Google Image Search, Fergus et al. [8] and Schroff et al. [18] reported a label precision between 18% and 77% for 7 object categories, and 39% over 18 categories respectively. Several approaches have been proposed to overcome this problem: one group of methods is targeted at a content-based refinement of raw web image sets [18, 24, 34]. Other methods closer to our work combine dataset refinement with model learning using topic models [8, 12] or a nearest neighbor analysis [13, 32].

In context of weakly labeled video content, Gargi and Yagnik [9] emphasized the additional problem that label information in videos may be coarse, which they refer to as the *label resolution problem*. Ulges et al. [26] presented a concept learning system that employs weakly labeled web video for concept detector training. They proposed an automatic kernel-based approach for relevance filtering, such that the system automatically learns relevance weights during detector training. We will adopt this approach and extend it in this paper.

When it comes to a manual acquisition of labeled training sets, active learning (see [19] for a survey) has been a successful approach in the context of visual concept learning [1, 3]. Its main goal is to select only a few samples for manual annotation, while the majority of the dataset remains unlabeled. The core of active learning is the selection of the "most informative" sample for the user to label. To do so, different *sample selection* have been proposed, like *relevance sampling* [16] – where samples are selected that are most likely to be relevant – or *uncertainty sampling* [11] – where samples closest to the current decision boundary are chosen.

Active learning has successfully been used in evaluation campaigns like TRECVID [2], where training examples for a concept of interest are accumulated from a completely unknown video database. This setup (which usually starts from very few reliable initial labels [1, 3, 5]) differs from the one studied in this paper, as we focus on a *refinement* of large and only partly non-relevant training sets. Despite this difference, however, an application of active learning in the context of visual learning from the web seems promising, and correspondingly we will adopt this approach in the following. Beyond this, we propose a novel combination of active learning sample selection with an automatic relevance filtering, which will be demonstrated to lead to even more robust concept detectors at less manual annotation cost.

## 3. APPROACH

In the following, a framework for visual concept learning from weakly labeled web video is described. The system is illustrated in Figure 2: to learn a concept like "basketball", training material is downloaded from online platforms. The core of the system – and the focus of this paper – is a filtering of this weakly labeled web content, which identifies non-relevant material and performs a concept detector training in parallel. This process is referred to as *relevance filter-*

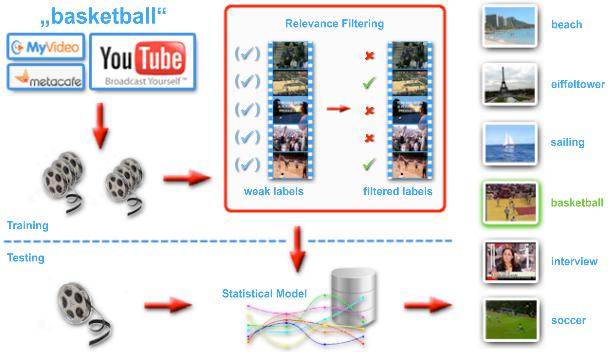

**Figure 2:** Concept learning from weakly labeled web video: material is downloaded from online platforms like YouTube, non-relevant content is filtered using *relevance filtering*, and a concept detector is trained, which can later be used to detect the learned concept in previously unseen videos.

*ing*, and is highlighted in a box in Figure 2. The procedure yields a statistical model – referred to as a *concept detector* – which can then be applied to find the concept of interest in previously unseen video material.

Relevance filtering can be performed by one of the following three strategies (as illustrated in detail in Figure 3):

1. an *automatic* relevance filtering, where non-relevant content is identified based on its distribution in feature space.

2. a *manual refinement* with the support of active learning, which selects the "most informative" samples for the user to label.

3. an *active relevance filtering*, which is the key contribution of this paper and combines the two previous strategies by alternatingly performing automatic relevance filtering and a manual label refinement

In this section, we will first introduce some basic notation and concepts (Section 3.1). After this, the two standard strategies will be addressed in detail, namely active learning (Section 3.2) and *automatic* relevance filtering (Section 3.3). Finally, we introduce the novel active relevance filtering approach (Section 3.4).

## 3.1 Basic Concepts

In the following, video content is represented by keyframes, each associated with a feature vector $x \in \mathbb{R}^d$. For each concept of interest, we formulate a binary classification problem: the presence of the target concept is denoted with a label $y$, such that $y = 1$ indicates concept presence and $y = -1$ concept absence. The goal of concept detection is – given a keyframe $x$ – to estimate the associated concept label $y$ (or its probability $P(y = 1|x)$, respectively).

For training, we assume a set of keyframes $x_1, ..., x_n$ to be given. Each of these is associated with a label $y_i \in \{-1, 1\}$ that indicates concept presence. In our setup of weakly labeled web videos, however, this true label is *latent* (i.e., not known), and we are only given a weak indicator of concept presence (in practice, this is a tag given to the corresponding web video clip). This information is denoted by a *weak label* $\tilde{y}_i \in \{-1, 1\}$, and forms the input to our concept learning procedure.

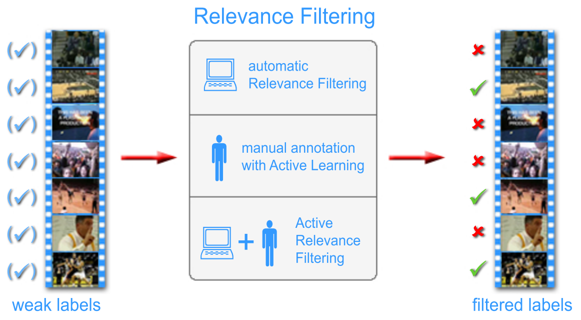
Relevance Filtering
automatic Relevance Filtering
manual annotation with Active Learning
Active Relevance Filtering
weak labels
filtered labels

**Figure 3:** Relevance filtering, as illustrated here for the concept "basketball", can be performed using three strategies: (1) an *automatic relevance filtering* [26], (2) a manual refinement with the help of *active learning* [1], or (3) a novel interleaved combination of automatic and manual filtering called *active relevance filtering*, which is the key contribution of this paper.

It should be kept in mind that the approaches discussed in the following – particularly, the proposed active relevance filtering – could be applied as a wrapper around a variety of statistical models. In this paper, we demonstrate this for *kernel densities* as a well-known standard approach that has successfully been used for concept detection before [31, 35]

**Baseline** We first introduce a simple supervised standard model that does not take label noise into account and will serve as a baseline in later experiments. This model uses two class-conditional distributions: $p^1$, which models positive keyframes (showing the target concept), and $p^0$ for negative frames (not showing the concept):

$$\begin{aligned} p^1(x) &= \frac{1}{Z_1} \cdot \sum_{i:\tilde{y}_i=1} K_h(x; x_i), \\ p^0(x) &= \frac{1}{Z_0} \cdot \sum_{i:\tilde{y}_i=-1} K_h(x; x_i). \end{aligned} \tag{1}$$

$Z_1$ and $Z_0$ are normalization factors. As a kernel function $K_h$, the well-known Epanechnikov kernel with Euclidean distance function and bandwidth $h$ is used [7, Ch. 4]:

$$K_h(x; x') = \frac{3}{4} \cdot \left(1 - \frac{||x - x'||^2}{h^2}\right) \cdot 1_{(||x-x'|| \leq h)}$$

By evaluating $p^1$ and $p^0$, the frame $x$ is scored using Bayes' rule (the class prior is assumed to be uniform):

$$P(y = 1|x) = \frac{p^1(x)}{p^1(x) + p^0(x)} \tag{2}$$

It is important to note that the approach – as introduced so far – does not take the unreliability of web-based training labels $\tilde{y}_i$ into account. Instead, these labels are treated just like in a fully supervised setup. Particularly, each positive sample ($\tilde{y}_i = 1$) – though it does not necessarily show the concept, as illustrated in Figure 1 – contributes to the density of positive samples $p^1$.

The key concern of this work, however, is to adapt concept training to the fact that user-generated labels on the web are inherently unreliable. In the following sections, we will discuss several approaches for dealing with label weakness. Our basic assumption will be that the given labels $\tilde{y}_i$ are only unreliable indicators of the true (but unknown) labels $y_i$ such that:

- If the weak label is negative ($\tilde{y}_i = -1$), the true label is negative as well ($y_i = -1$).

- If the weak label is positive ($\tilde{y}_i = 1$), the sample *may* belong to the positive class, but does not necessarily do so, i.e. the true label $y_i$ is unknown,

Briefly speaking, we assume that negative labels are reliable, but positive ones are not. This setup does not take false negatives ($\tilde{y}_i = -1$ and $y_i = 1$) into account, which is not strictly true (for example, a user could simply forget to tag a clip). According to observations we made on real-world web video, however, the fraction of these false negatives compared to truly negative content is negligible, and false positives pose a much more urgent problem.

## 3.2 Active Learning

One strategy to overcome label noise is to manually refine the raw web-based training set. In this context, active learning is a well-known effective approach. In this section, we discuss different active learning strategies for detector training that are targeted at achieving a refinement at minimal additional annotation cost. The goal is to select only the most important samples for inspection and therefore to improve concept detector performance up to the level of ground truth expert labels with only a few manual labels.

**Relevance Feedback as a Wrapper.**
In the following setup, a manual label refinement of selected samples is placed as a wrapper around a regular supervised learning method (here, the previously proposed kernel density learning from Equation (1)).

The procedure is illustrated in detail in Table 1: iteratively, concept detection is applied, obtaining class posterior probabilities $p^j = (p^j_1, ..., p^j_n)$ for all training samples, where $p^j_i \approx P(y_i = 1|x_i)$ (see Equation (2)). Based on these values, a keyframe $s^* \in \{i : \tilde{y}_i = 1\}$ is selected for manual annotation (we focus on *positive* weakly labeled keyframes because their labels are the unreliable ones). After a manual labeling of the selected sample $s^*$, we fix its label to either $-1$ or 1 depending on the received annotation result. Note, that in case of a positive feedback (i.e., $\tilde{y}_{s^*} = 1$), no change of the model will occur, whereas in case of negative feedback, the associated label turns to be $-1$ and the model will change in the next iteration, resulting in an improved concept detector. This retrained concept detector will then provide new posterior probabilities for the next iteration of active learning sample selection. When continuing further, this procedure acquires more and more expert labels, until finally the weakly labeled dataset turns into a strongly annotated one.

**Table 1: Active Learning**: Wrapped around concept detector training, active learning selects informative samples for refinement by a user. Once the sample is labeled, its label is fixed to either -1 or 1 and the system is re-trained.

```
1. for $j = 1, ., m$ do:
    • obtain class posteriors $p^j = (p^j_1, ..., p^j_n)$ from $p^1$ and $p^0$
    • select sample $s^*$ according to an active learning criterion $Q$:
          $s^* := \arg\max_{i:\tilde{y}_i=1} \; Q(p^j_i)$
    • get the true label $y_{s^*}$ from a human expert
    • fix the sample label:
          $\tilde{y}^{j+1,...,m}_{s^*} = \begin{cases} 1, & y_{s^*} = 1 \\ -1, & y_{s^*} = -1 \end{cases}$
      Once the true label is retrieved, the sample $s^*$ is excluded from sample selection.
```

**Active Learning Methods.**
Obviously, the quality of active learning heavily depends on the sample selection strategy $Q$ (see Table 1). In the literature, many criteria $Q$ have been proposed [19]. Here, we compare the most popular ones:

1. **random sampling**: samples are selected randomly (serves as a baseline).

2. **most relevant**: samples are selected which are most likely to be relevant and are therefore associated with the highest posterior [16]:

   $$Q_{REL}(p^j_i) := p^j_i$$

3. **uncertainty**: samples are selected for which the relevance filtering method is least confident, i.e. $p^j_i \approx 0.5$ [11]:

   $$Q_{UNC}(p^j_i) := 1 - |p^j_i - 0.5|$$

4. **density-weighted repulsion (DWR)**: Our last approach enhances "most relevant" sampling with an exploratory component. This is motivated by the assumption that the labels associated with clusters in feature space are homogeneous, and therefore the refinement of one sample within a cluster is sufficient of infer the remaining ones. This is realized by adding a repulsion term that enforces the query sample $x_i$ to be distant from previously labeled samples (which form a kernel density $p^+$):

   $$Q_{DWR}(p^j_i) := Q_{REL}(p^j_i) \cdot (p^+(x_i) + \epsilon)^{-\gamma},$$

   where the parameter $\gamma$ determines the strength of repulsion.

## 3.3 Automatic Relevance Filtering

While the active learning approaches introduced in the last section perform a refinement based on manual labels of selected samples, other systems have been introduced that replace this refinement with a fully automatic one. The basic idea of these *automatic relevance filtering* methods [26, 32] is that relevant content appears frequently and forms clusters in feature space, while non-relevant material comes as outliers that can be identified and relabeled.

In this section, we will discuss an automatic relevance filtering approach based on a *weighted kernel density model* [26, 31]. The class-conditional densities from Equation (1) are replaced with *weighted* kernel densities:

$$\begin{aligned} p^1_\beta(x) &= \frac{1}{Z'_1} \cdot \sum_{i=1}^{n} \beta_i \cdot K_h(x; x_i), \\ p^0_\beta(x) &= \frac{1}{Z'_0} \cdot \sum_{i=1}^{n} (1-\beta_i) \cdot K_h(x; x_i), \end{aligned} \tag{3}$$

where $Z'_1 = \sum_i \beta_i$ and $Z'_0 = n - Z'_1$ are normalization constants. Compared to the fully supervised setup from Equation (1), the key difference is that $p^1$ and $p^0$ are now parameterized by a vector $\beta = (\beta_1, ..., \beta_n)$. This vector consists of *relevance scores* $\beta_i := P(y_i = 1|\tilde{y}_i, x_i)$, meaning that each training sample is weighted according to its probability of being relevant: if a sample is likely to be relevant, it has a strong influence on the distribution of positive samples $p^1_\beta$ but low influence on $p^0_\beta$. This way, the uncertainty of label information is taken into account.

To compute the class-conditional densities $p^1_\beta$ and $p^0_\beta$, the vector of relevance scores $\beta$ must be inferred in system training, i.e. potentially relevant frames must be divided into actually relevant ones and non-relevant ones.

The relevance scores $\beta$ are estimated in a training procedure that – starting from a vector $\beta^0$ – iteratively updates the parameter vector $\beta^k$ to a new version $\beta^{k+1}$ by plugging it into the class-conditional densities $p^1_{\beta^k}$ and $p^0_{\beta^k}$ (Equation (3)). From these densities, new estimates of relevance scores can be obtained using Bayes' rule:

$$\begin{aligned} \beta_i^{k+1} &:= P(y_i = 1|x_i, \tilde{y}_i = 1) \\ &\approx \frac{P(y_i = 1|\tilde{y}_i = 1) \cdot p(x_i|y_i = 1)}{\sum_{y \in \{-1,1\}} P(y_i = y|\tilde{y}_i = 1) \cdot p(x_i|y_i = y)} \qquad (4) \\ &\approx \frac{\alpha \cdot p^1_{\beta^k}(x_i)}{\alpha \cdot p^1_{\beta^k}(x_i) + (1-\alpha) \cdot p^0_{\beta^k}(x_i)} \end{aligned}$$

$$(5)$$

This is repeated until convergence. Training is regulated by the relevance fraction $\alpha := P(y_i = 1|\tilde{y}_i = 1)$, which determines how many of the positively labeled samples do in fact show the target concept (if we choose $\alpha = 1$, the model degenerates to the supervised case as in Equation (1)). In the following, we assume a sufficiently good estimate of this parameter to be given.

Intuitively, this training procedure identifies regions in feature space where positively labeled frames concentrate and assigns high relevance scores to them, while outliers similar to negative content are given low relevance scores. The approach resembles the well-known Expectation Maximization (EM) algorithm [6], which maximizes the data likelihood in the presence of latent variables (here, the true concept labels $y_1, ..., y_n$). Also, a similar training procedure has been used by Wang et al. [31]. For more information on the approach, please refer to a previous publication [26].

## 3.4 Active Relevance Filtering

The relevance scores $\beta_1, ..., \beta_n$ in Section 3.3 captured the uncertainty of the given web-based label information. They have been fitted using an automatic training procedure, which has previously been shown to improve concept detection to some extent [26]. Yet, significant label uncertainty remains, which is why we propose to combine relevance filtering with active learning to enhance the system with a limited amount of manual feedback. This *active relevance filtering* is outlined in the following.

**Table 2: Active Relevance Filtering**: Wrapped around relevance filtering, active learning selects informative samples for refinement by a user. Once the sample is annotated, the system is re-trained and the remaining relevance scores are adapted.

```
1. for $j = 1, ..., m$ do:
    • **apply automatic relevance filtering, obtaining relevance scores** $\beta^j = (\beta^j_1, ..., \beta^j_n)$
    • **update the class-conditional densities** $p^0_\beta$ **and** $p^1_\beta$ **(Equation** (3)**)**
    • obtain class posteriors $p^j = (p^j_1, ..., p^j_n)$ from $p^1_\beta$ and $p^0_\beta$
    • select sample $s^*$ according to an *active learning* criterion $Q$:
          $s^* := \arg\max_{i: \tilde{y}_i = 1} \; Q(p^j_i)$
    • get the true label $y_{s^*}$
    • fix the sample label:
          $\tilde{y}^{j+1,...,m}_{s^*} = \begin{cases} 1, & y_{s^*} = 1 \\ -1, & y_{s^*} = -1 \end{cases}$
      Once the label is fixed, its relevance score is set to the true value, and the sample is excluded from further automatic relevance filtering.
```

We propose an iterative manual labeling of selected frames, which is alternated with a retraining of relevance scores $\beta$. This way, the previously introduced active learning mechanism is enhanced by an automatic relevance filtering step after concept detector training. Again, to reduce annotation effort, active learning strategies are used to select only the most informative samples for annotation.

The procedure is illustrated in Table 2 (modifications compared to the active learning procedure in Table 1 are highlighted in bold): in each iteration, an automatic relevance filtering is performed, from which the class-conditional densities are updated, obtaining class posteriors $p^j_1, ..., p^j_n$ for all training samples. Based on these posteriors, the most informative weakly labeled keyframes are selected for manual annotation (the same selection strategies as for active learning can be used, see Section 3.2). The received label information will now again serve as additional ground truth for the next iteration of automatic relevance filtering, providing improved relevance scores for the next iteration of sample selection. With more iterations of such combined relevance filtering and active learning, the procedure separates relevant content from non-relevant one more reliably.

Note that this approach *alternates* automatic and manual filtering: in contrast to a purely automatic filtering, the method uses an additional wrapper in which a human operator contributes more accurate labels than the purely automatic approach can estimate by itself. The key difference to active learning is that the labels are not only used to update the classifier, but also for further relevance filtering: each time a new label is given, it influences relevance scores on the training set and helps to filter non-relevant content more precisely. This provides an improved basis for the next active learning sample selection – alternatingly, automatic and manual refinement boost each other.

# 4. EXPERIMENTS

Experiments are performed on a dataset of real-world web video content downloaded from YouTube. For this, we select ten test concepts from the YouTube-22concepts[2] dataset, including objects ("cats", "eiffeltower"), locations ("beach", "desert"), and sports ("basketball", "golf"). For more details on the test concepts, please refer to our dataset definition[3]. For each concept, 100 video clips were downloaded by querying the YouTube API with an appropriate combination of keywords. Keyframes were extracted and manually assessed according to canonical concept definitions. For each concept, we sampled a training set of $1,000$ negative sample frames and 500 noisy positive frames. The label precision of these positive samples was set to 20% (which was validated to be a typical value for web video in previous annotation experiments). This means that the 500 positive samples contained only 100 true positives and 400 false positives (which were also sampled from YouTube clips tagged with the target concept, but were manually assessed to be non-relevant). To evaluate the concept detectors trained on this weakly labeled content, a test set of 500 positive and $1,500$ negative frames was sampled (it was made sure that training and test content was drawn from different clips).

As a feature representation of keyframes, we refer to the well-known *bag-of-visual-words* approach [21, 29]: a regular patch sampling was conducted at several scales, patches were described by SIFT [14], and finally clustered to a $2,000$-dimensional vocabulary using K-Means. After this, a PLSA dimensionality reduction [15] to 64 dimensions was applied for efficiency purposes.

We tested the relevance filtering system with a kernel bandwidth of $h = 0.0275$ (which was previously optimized using cross-validation). The parameter $\alpha$ of automatic relevance filtering (Equation (4)) was set to 20%. For DWR sampling a value of $\gamma = 0.1$ proved to work best. As a performance measure, *mean average precision* (MAP) is used. All results are averaged over all 10 test concepts and over 5 trials using different randomly sampled datasets.

Three main experiments were conducted to quantify the effects of different refinement strategies: first, we validate the impact of an automatic relevance filtering and demonstrate that this approach gives some improvements but does not reach the performance of a complete manual annotation (Section 4.1). After this, we evaluate a manual refinement using plain active learning (Section 4.2) and compare this with the novel active relevance filtering approach (Section 4.3).

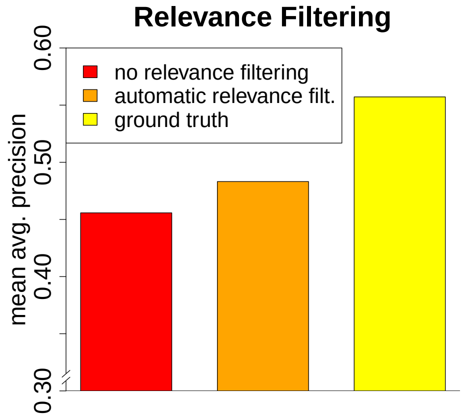

**Figure 4:** Results of Experiment 1, showing potential performance ranges for further refinement strategies. Though automatic relevance filtering provides some performance gain, its performance is far from the ground truth optimum. Note that ground truth label information is not given and should here only demonstrate the potential performance gain of a perfect relevance filtering.

**Table 3:** Detailed Results of Experiment 1. Average precision is displayed for each concept and each of the three runs.

| concept | no rel. filt. | auto. rel. filt. | ground truth |
|---|---|---|---|
| basketball | 0.570 | 0.606 | **0.651** |
| beach | 0.398 | 0.449 | **0.504** |
| cats | 0.320 | 0.333 | **0.388** |
| desert | 0.587 | 0.636 | **0.655** |
| eiffeltower | 0.425 | 0.421 | **0.526** |
| helicopter | 0.362 | 0.392 | **0.418** |
| sailing | 0.440 | 0.466 | **0.493** |
| soccer | 0.562 | 0.575 | **0.740** |
| swimming | 0.448 | 0.491 | **0.647** |
| tank | 0.441 | 0.457 | **0.543** |
| **MAP** | 0.455 | 0.482 | **0.557** |

## 4.1 Experiment 1: Weak Label Impact & Automatic Relevance Filtering

The first experiment evaluates several concept learning approaches when trained on weakly labeled web video material. We compare three systems: first, one that does not perform relevance filtering at all, which corresponds to a standard supervised system using plain kernel densities (this baseline is denoted with *no relevance filtering* and has been outlined in Section 3.1). Second, an automatic relevance filtering as outlined in Section 3.3, and third a control run using ground truth labels (note that such label information is not available in practice).

When comparing these three runs (Figure 4 and Table 3 for detailed concept-dependent results), we see that the system without relevance filtering performs worst, with a mean average precision (MAP) of 0.455. The automatic relevance filtering achieves a slight improvement (MAP: 0.482). However, a strong gap of 7% remains compared to the ground truth run (MAP: 0.557) – this indicates that we could improve concept learning from the web significantly if we were able to perform a more accurate filtering of non-relevant content. This motivates semi-automatic refinement strategies as evaluated in the next sections.

[2] https://madm.dfki.de/research/youtube-22concepts

[3] http://www.dfki.uni-kl.de/~ulges/MIR2010-testconcepts/

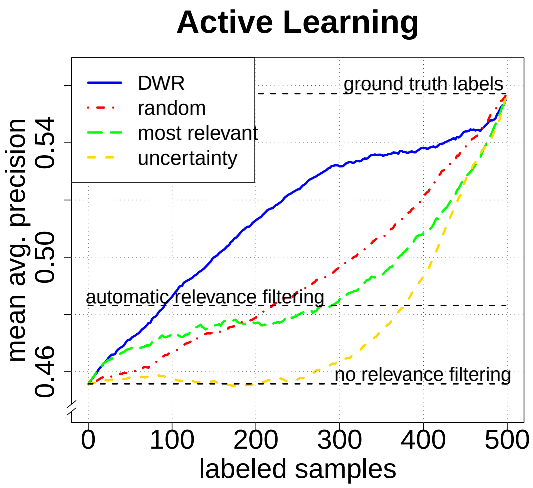

**Figure 5:** Results of active learning. The accuracy of the resulting concept detectors is plotted against the number of manually annotated training samples.

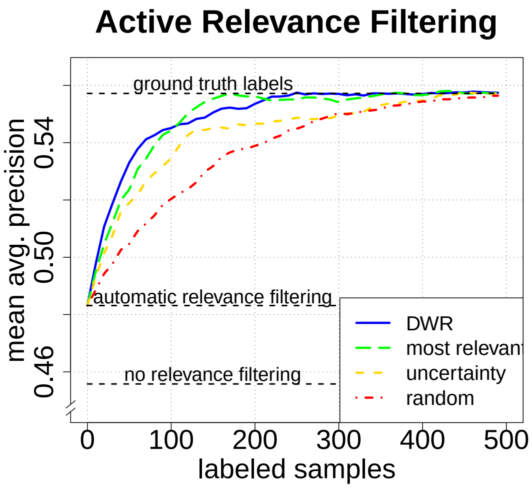

**Figure 6:** Results of active relevance filtering. Performance is plotted against the number of manually annotated training samples. It can be seen that – if using a proper sample selection – it is sufficient to annotate only $30-40$ weakly positive training samples to achieve a significant performance improvement.

## 4.2 Experiment 2: Active Learning

In the second experiment we quantify the performance of a manual refinement of web-based training sets using active learning. The results of this experiment are illustrated in Figure 5, where the performance of the trained concept detectors on the test set is plotted against the number of training samples annotated with active learning (different curves correspond to different sample selection strategies). To establish a relation to the last experiment, the three automatic runs (no relevance filtering, automatic relevance filtering, and ground truth labels) are plotted as dotted lines in Figure 5. We see that all sample selection methods start at an MAP of 0.45 (which equals the previously shown "no relevance filtering" system, as no automatic relevance filtering is done. However, as we collect more training labels manually, the quality of the training set (and with it the accuracy of the resulting detectors) improves. Sample selection stops when all weakly labeled samples are manually refinement i.e. after 500 annotations. Here, the MAP is the same for all selection methods and equals the "ground truth" run in Section 4.1 (which is not surprising, as the whole training set is now manually annotated).

When comparing the different sample selection methods, we see that different sampling strategies lead to a very different performance. Surprisingly, well-known samplings methods like *uncertainty sampling* are performing worse than a simple random sampling baseline. The best overall result is achieved by DWR sampling, which gives strong improvements over all other strategies. Yet, the improvements by active learning remain limited: even the best method requires a substantial amount of manual samples to give significant improvements over the automatic relevance filtering. To reach a performance close to a ground truth labeling, all methods require a manual annotation of wide parts of the training set.

## 4.3 Experiment 3: Active Relevance Filtering

In this experiment, the performance of the proposed active relevance filtering approach (a novel combination of a manual and automatic label refinement) is evaluated. Results of this experiment are illustrated in Figure 6. Just like in Figure 5, concept detector performance is plotted against the number of manual annotations used in training.

We first compare the different active learning strategies in Figure 6. It can be seen that all used sample selection methods outperform the random sampling baseline significantly. Systems based on *most relevant sampling* perform best, which can be explained by the fact that this approach helps to identify false positives that are "surprising" to the system and thus lead to strong model changes. For low numbers of annotations, the exploratory component of DWR leads to further improvements.

Overall, it can be seen that active relevance filtering — if combined with the right sample selection strategy — is highly efficient, giving strong improvements of concept learning even for very low numbers of manual annotations. For example, with as few as 50 annotations, a performance increase of 5% is achieved compared to automatic relevance filtering. When continuing with annotation, we can see that concept detection performance converges to the ground truth case at $125-150$ iterations (which corresponds to only $25-30$% of the positive weakly labeled training set and 10% of the whole training set).

Figure 8 provides a visual impression of active relevance filtering performance. Here, the top 20 test set classification results are shown for the three concepts "basketball", "tank" and "eiffeltower". For each concept a separate result list is displayed for a) non relevance filtering, b) automatic relevance filtering and c) active relevance filtering (50th iteration of DWR). The border of each keyframe is colored according to its true label (green=concept present; red=concept absent). Comparing the different lists, we can see that significantly better results can be achieved for c) compared to b), which itself shows improvements over a). Note that particularly for such challenging concepts as "eif-

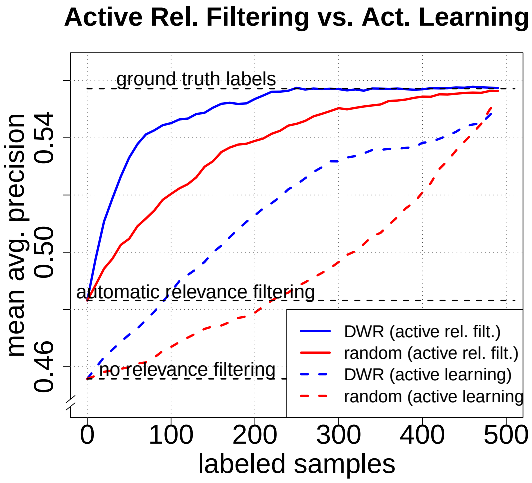

**Figure 7:** Comparing active learning and active relevance filtering, using random sample selection and DWR. The proposed active relevance filtering leads to better concept detectors at lower annotation cost.

feltower" – for which automatic relevance filtering is difficult due to lots of non-relevant YouTube content showing views *from* the tower but not the tower itself – active relevance filtering improves classification results significantly.

Finally, we compare the proposed active relevance filtering with pure active learning as discussed in Section 4.2. Again, we plot detection performance against the number of manual annotations (Figure 7). We plot the best systems for active relevance filtering and active learning (Section 4.2 vs. Section 4.3), namely the DWR-based runs, and also (as a baseline) random sampling. The results clearly indicate that active relevance filtering significantly outperforms a pure active learning. We see for both sample selection strategies that active relevance filtering starts with a higher MAP as it utilizes automatic relevance filtering. Also, system performance of active relevance filtering improves quicker than for pure active learning, which can be explained by the fact that active relevance filtering makes better use of user feedback: if a manual sample is provided, the additional relevance filtering mechanism propagates this label over neighbor samples. For example, after refining only 50 samples manually, active relevance filtering clearly outperforms pure active learning, resulting in an absolute improvement of 7%.

Concluding, active relevance filtering – particularly if combined with appropriate sample selection strategies – can improve concept learning on the difficult domain of web video content better than both an automatic relevance filtering and a manual label refinement using standard active learning techniques.

## 5. DISCUSSION

In this paper, we have addressed the challenge of learning visual concepts from web video, which offers a scalable alternative to the conventional manual acquisition of concept detection training data. On the downside, the tags coming with web video are only weak indicators of concept presence, and web-based training sets come with significant amounts of non-relevant content. To achieve robustness with respect to this *label noise*, we have combined *relevance filtering* – which discards non-relevant content automatically – and active learning – which is targeted at an efficient manual refinement. The resulting approach – called *active relevance filtering* – performs a highly efficient learning using a few manually labeled samples.

In quantitative experiments with real-world web content downloaded from YouTube, we have demonstrated that active relevance filtering improves concept learning significantly. Particularly, we outperform both a purely automatic refinement and standard active learning, reaching a performance comparable to ground truth training by refining only $25 - 30\%$ of weak positive labels in the training set.

Regarding future directions along this line of research, the next step is to integrate active relevance filtering with other statistical learning methods. In this paper, we have used a simple generative standard approach (namely, kernel densities). It remains to be investigated whether active relevance filtering could be used as a wrapper around other machine learning methods in a similar fashion, including generative ones (e.g., Gaussian mixture models, histograms) as well as discriminative ones (e.g., SVMs [17]).

## 6. ACKNOWLEDGMENTS

This work was supported by the German Research Foundation (DFG), project MOONVID (BR 2517/1-1).

## 7. REFERENCES

[1] S. Ayache and G. Quenot. Evaluation of active learning strategies for video indexing. *Signal Processing: Image Communication*, 22(7-8):692–704, 2007.

[2] S. Ayache and G. Quenot. TRECVID 2007 Collaborative Annotation using Active Learning. In *Proc. TRECVID Workshop*, November 2007.

[3] S. Ayache and G. Quenot. Video Corpus Annotation using Active Learning. In *Proc. Europ. Conf. on Information Retrieval*, pages 187–198, March 2008.

[4] M. Campbell, A. Haubold, M. Liu, A. Natsev, J. Smith, J. Tesic, L. Xie, R. Yan, and J. Yang. IBM Research TRECVID-2007 Video Retrieval System. In *Proc. TRECVID Workshop*, November 2007.

[5] M. Chen, M. Christel, A. Hauptmann, and H. Wactlar. Putting Active Learning into Multimedia Applications: Dynamic Definition and Refinement of Concept Classifiers. In *Proc. Int. Conf. on Multimedia*, pages 902–911, November 2005.

[6] A. Dempster, N. Laird, and D. Rubin. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1):1–38, 1977.

[7] R. Duda, P. Hart, and D. Stork. *Pattern Classification (2nd Edition)*. Wiley-Interscience, 2000.

[8] R. Fergus, L. Fei-Fei, P. Perona, and A. Zisserman. Learning Object Categories from Google's Image Search. *Computer Vision*, 2:1816–1823, 2005.

[9] U. Gargi and J. Yagnik. Solving the Label Resolution Problem in Supervised Video Content Classification. In *Proc. Int. Conf. on Multimedia Retrieval*, pages 276–282, October 2008.

[10] L. Kennedy, S.-F. Chang, and I. Kozintsev. To Search or to Label?: Predicting the Performance of

Search-based Automatic Image Classifiers. In *Int. Workshop Multimedia Information Retrieval*, pages 249–258, October 2006.

[11] D. Lewis and W. Gale. A Sequential Algorithm for Training Text Classifiers. In *Proc. Int. Conf. Research and Development in Information Retrieval*, pages 3–12, July 1994.

[12] L.-J. Li, G. Wang, and L. Fei-Fei. OPTIMOL: automatic Object Picture collecTion via Incremental MOdel Learning. In *Proc. Int. Conf. Computer Vision and Pattern Recognition*, pages 57–64, June 2007.

[13] X. Li, C. Snoek, and M. Worring. Learning Tag Relevance by Neighbor Voting for Social Image Retrieval. In *Proc. Int. Conf. on Multimedia Information Retrieval*, pages 180–187, October 2008.

[14] D. Lowe. Object Recognition from Local Scale-Invariant Features. In *Int. Conf. Computer Vision*, pages 1150–1157, September 1999.

[15] P. Quelhas, F. Monay, J. Odobez, D. Gatica-Perez, and T. Tuytelaars. A Thousand Words in a Scene. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 29(9):1575–1589, 2007.

[16] G. Salton and C. Buckley. Improving Retrieval Performance by Relevance Feedback. *Journal of the American Society for Information Science*, 41(4):288–297, 1990.

[17] B. Schölkopf and A. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, Cambridge, MA, USA, 2001.

[18] F. Schroff, A. Criminisi, and A. Zisserman. Harvesting Image Databases from the Web. In *Proc. Int. Conf. Computer Vision*, pages 1–8, October 2007.

[19] B. Settles. Active Learning Literature Survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2009.

[20] A. Setz and C. Snoek. Can Social Tagged Images Aid Concept-Based Video Search? In *Proc. Int. Conf. on Multimedia and Expo*, pages 1460–1463, 2009.

[21] J. Sivic and A. Zisserman. Video Google: Efficient Visual Search of Videos. In *Toward Category-Level Object Recognition*, pages 127–144. Springer-Verlag New York, Inc., 2006.

[22] A. Smeulders, M. Worring, S. Santini, and A. Gupta R. Jain. Content-Based Image Retrieval at the End of the Early Years. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 22(12):1349–1380, 2000.

[23] C. Snoek and M. Worring. Concept-based Video Retrieval. *Foundations and Trends in Information Retrieval*, 4(2):215–322, 2009.

[24] Y. Sun, S. Shimada, Y. Taniguchi, and A. Kojima. A Novel Region-based Approach to Visual Concept Modeling using Web Images. In *Int. Conf. Multimedia*, pages 635–638, October 2008.

[25] A. Ulges, M. Koch, C. Schulze, and T. Breuel. Learning TRECVID'08 High-level Features from YouTube$^{\text{TM}}$. In *Proc. TRECVID Workshop*, November 2008.

[26] A. Ulges, C. Schulze, D. Keysers, and T. Breuel. Identifying Relevant Frames in Weakly Labeled Videos for Training Concept Detectors. In *Proc. Int. Conf. Image and Video Retrieval*, pages 9–16, July 2008.

[27] A. Ulges, C. Schulze, M. Koch, and T. Breuel. Learning Automatic Concept Detectors from Online Video. *Comp. Vis. Img. Underst.*, 2009.

[28] YouTube Serves up 100 Million Videos a Day Online. in USA Today (Garnnett Company, Inc.); available from http://www.usatoday.com/tech/news/2006-07-16-youtube-views_x.htm (retrieved: Sep'08), July 2006.

[29] K. van de Sande, T. Gevers, and C. Snoek. A Comparison of Color Features for Visual Concept Classification. In *Proc. Int. Conf. Image and Video Retrieval*, pages 141–150, July 2008.

[30] D. Wang, X. Liu, L. Luo, J. Li, and B. Zhang. Video Diver: Generic Video Indexing with Diverse Features. In *Proc. Int. Workshop Multimedia Information Retrieval*, pages 61–70, September 2007.

[31] M. Wang, X.-S. Hua, Y. Song, X. Yuan, S. Li, and H.-J. Zhang. Automatic Video Annotation by Semi-supervised Learning with Kernel Density Estimation. In *Proc. Int. Conf. on Multimedia*, pages 967–976, October 2006.

[32] K. Wnuk and S. Soatto. Filtering Internet Image Search Results Towards Keyword Based Category Recognition. In *Proc. Int. Conf. Computer Vision and Pattern Recognition*, pages 1–8, June 2008.

[33] A. Yanagawa, S.-F. Chang, L. Kennedy, and W. Hsu. Columbia University's Baseline Detectors for 374 LSCOM Semantic Visual Concepts. Technical report, Columbia University, 2007.

[34] K. Yanai and K. Barnard. Probabilistic Web Image Gathering. In *Int. Workshop on Multimedia Inf. Retrieval*, pages 57–64, November 2005.

[35] A. Yavlinsky, E. Schofield, and S. Rüger. Automated Image Annotation using Global Features and Robust Nonparametric Density Estimation. In *Proc. Int. Conf. Image and Video Retrieval*, pages 507–517, July 2005.

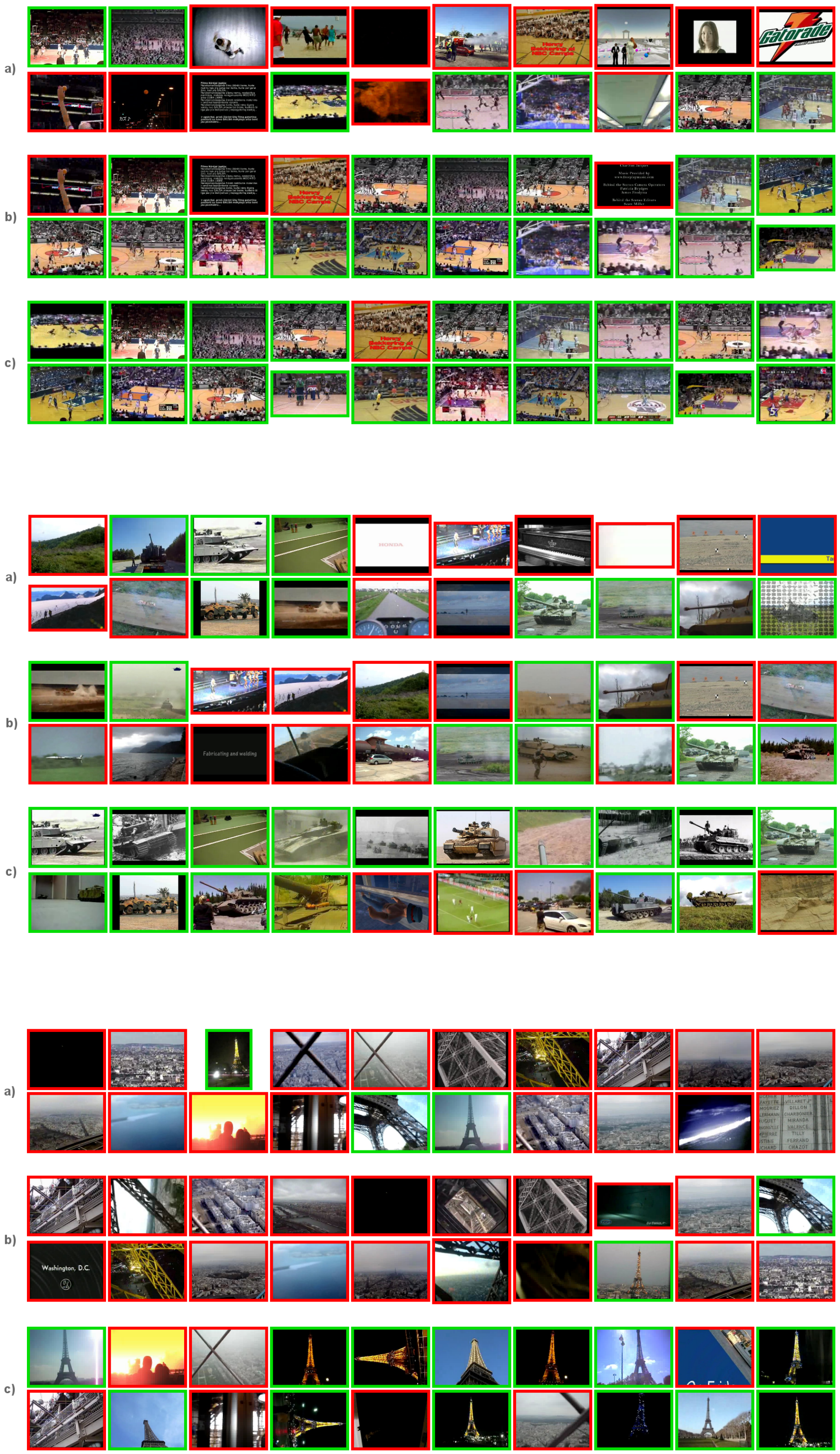

**Figure 8:** Results for the three concepts “basketball” (top), “tank” (center), and “eiffeltower” (bottom). The top 20 results are displayed for a) no relevance filtering, b) automatic relevance filtering and c) active relevance filtering (50th iteration of DWR). Each keyframe is colored according to its true label (green=concept present; red=concept absent). Results are improved significantly in c) compared to b) which itself gives improvements over a). Overall, active relevance filtering improves classification results significantly.