Shuffled ImageNet Banks for Video Event Detection and Search

Mettes, P.; Koelma, D.C.; Snoek, C.G.M.

DOI
10.1145/3377875

Publication date
2020

Document Version
Final published version

Published in
ACM Transactions on Multimedia Computing Communications and Applications

License
Article 25fa Dutch Copyright Act (https://www.openaccess.nl/en/in-the-netherlands/you-share-we-take-care)

Citation for published version (APA):
Shuffled ImageNet Banks for Video Event Detection and Search

PASCAL METTES, DENNIS C. KOELMA, and CEES G. M. SNOEK, University of Amsterdam, the Netherlands

This article aims for the detection and search of events in videos, where video examples are either scarce or even absent during training. To enable such event detection and search, ImageNet concept banks have shown to be effective. Rather than employing the standard concept bank of 1,000 ImageNet classes, we leverage the full 21,841-class dataset. We identify two problems with using the full dataset: (i) there is an imbalance between the number of examples per concept, and (ii) not all concepts are equally relevant for events. In this article, we propose to balance large-scale image hierarchies for pre-training. We shuffle concepts based on bottom-up and top-down operations to overcome the problems of example imbalance and concept relevance. Using this strategy, we arrive at the shuffled ImageNet bank, a concept bank with an order of magnitude more concepts compared to standard ImageNet banks. Compared to standard ImageNet pre-training, our shuffles result in more discriminative representations to train event models from the limited video event examples. For event search, the broad range of concepts enable a closer match between textual queries of events and concept detections in videos. Experimentally, we show the benefit of the proposed bank for event detection and event search, with state-of-the-art performance for both tasks on the challenging TRECVID Multimedia Event Detection and Ad-Hoc Video Search benchmarks.

CCS Concepts: • Computing methodologies → Activity recognition and understanding;

Additional Key Words and Phrases: Event detection, event search, concepts, shuffle

ACM Reference format:

1 INTRODUCTION

This article strives to detect and search events such as Tuning a musical instrument, Renovating a home, and Giving directions in videos. Such events are difficult to recognize, since they are composed of high-level activities [72], are semantically diverse [2], and generally have few training examples [38]. To tackle these issues, the literature on event detection and search has initially sought to represent videos through low-level visual and audial features. Prototypical features include local visual features [48], local motion features [59, 64], and short-term audio features [13]. While such features result in discriminative video representations, several works have indicated
that the representations require many training examples [5, 18, 38] and lack semantic interpretation [33, 39]. In this article, we therefore represent videos by concept classification scores, which are pre-trained on images. Such concept banks have shown to be effective for event detection [18, 37, 54] and search [21, 63, 66], especially when training examples are scarce [37, 39].

In concept-based event detection and search, there are two leading approaches to represent videos. The first employs a bank of 1,000 concepts, pre-trained on a subset of the ImageNet dataset defined by the Large Scale Visual Recognition Challenge [52]. This concept bank is known to aid event detection [6, 26, 69] but does not fully leverage the ImageNet dataset, as only eight percent of the available examples are used. The second obtains a large-scale concept bank from the web [17, 18, 54]. While this results in a bank with a wide variety of concepts to detect in event videos, concepts are generally noisier than the concepts from ImageNet, as the annotations are not curated. In this article, we focus on event detection and search on the complete ImageNet dataset with 21,841 classes. We seek to obtain a concept bank that is both large in size and rich in variety, for discriminative event detection and search.

To get the most out of the complete ImageNet dataset, we identify two problems. First, there is a vast imbalance in the number of training examples between concepts. For example, the concept Yorkshire Terrier has over 3,000 training examples, while more than 250 concepts have only a single training example (e.g., Ichyostega, an amphibian found in Greenland). As such, directly pre-training on all ImageNet concepts is not possible. Second, not all concepts are equally relevant for the purpose of event detection (e.g., prolonge knot, a knot in the rope to drag a gun carriage) [65]. While their training examples can be useful, we ideally want to merge highly specific concepts to maintain a general concept bank [51].

The main contribution of this work is the shuffled ImageNet bank, a large-scale concept bank with an order of magnitude more concepts than the standard ImageNet bank. To obtain this bank, we propose a shuffling strategy that enables us to leverage the full ImageNet hierarchy for event detection and search. Concepts are shuffled and combined by exploiting the hierarchical ontological connections among concepts. We propose a bottom-up and top-down shuffle of the ImageNet hierarchy. For the bottom-up shuffle, we start from the complete concept hierarchy, and we define four operations to create a more balanced and general hierarchy. The top-down shuffle performs a breadth-first search on the ImageNet hierarchy for concept selection. Both approaches exploit the wide variety of training examples in ImageNet for pre-training, while maintaining a balanced and useful concept bank, resulting in video representations with rich semantics for event detection.

Beyond event detection with training examples, we show how to perform event search, where event clips need to be retrieved from only textual queries. This task was popularized in TRECVID a decade ago, where leading approaches sought to detect event-specific concepts in videos from multi-modal data [21, 55, 63, 66]. The task was recently reintroduced as ad-hoc video search by TRECVID [1], where short video clips need to be retrieved from text queries. We show how the use of a shuffled ImageNet bank naturally generalizes to event search, by matching textual event queries to the rich set of concept detections in videos from the concept bank.

Experimentally, we find that the shuffled ImageNet bank results in a better event detection compared to the standard ImageNet bank, which is attributed to the increased diversity in concepts for the video frame representations. For event search, the bank provides a larger set of concepts to match semantically with event queries, leading to improved results. Overall, our approach obtains state-of-the-art results on two challenging benchmarks of TRECVID, namely, Multimedia Event Detection (MED) and Ad-hoc Video Search (AVS).

This article builds upon our conference paper at the International Conference on Multimedia Retrieval [40]. Here, we provide three extensions: (i) a generalization of shuffled ImageNet banks to event search from textual queries, (ii) shuffled ImageNet banks learned atop the latest generation
of convolutional networks, which will be made publicly available of further research, and (iii) new ablation studies, analyses, and benchmark comparisons for event detection and search.

2 RELATED WORK

2.1 Video Event Detection

Concept-based event detection thrives on concept banks that are both diverse and of high quantity [15, 20]. The leading approach is to transfer knowledge from large-scale concept sources to event videos. Typically, curated datasets such as ImageNet are used to train concept detectors [6, 26, 69]. While ImageNet provides a wealth of concepts and visual examples with curated annotations, current event detection works employ a standard subset of 1,000 categories from the Large Scale Visual Recognition Challenge [52], which amounts to roughly eight percent of the whole ImageNet collection. Our proposed shuffled ImageNet bank contains many more concepts than the LSVRC bank, resulting in richer concept representations for event detection.

Several works have investigated improving event representations by merging and filtering concepts after detection in videos. Mazloom et al. [37] learn to focus on the most representative concepts in videos while training event detection models. Similar strategies have been employed by Chang et al. [4] and Yang and Shah [71]. Habibian et al. [19] jointly train event detectors and learn to merge concepts. Chang et al. [7] trim concepts that are not deemed relevant or reliable based on video sources with provided descriptions. Yiang et al. [26] re-score concept detections by examining the hierarchical relations among concepts. A common rationale in these works is that not all concepts are equally relevant for events and might be over-specific. In this work, we share this rationale and incorporate it by shuffling the 21,841-class hierarchy of ImageNet into a balanced, diverse, and representative concept bank prior to learning event detection models. This alleviates the need for expensive event training and inference, reducing computation while maintaining high performing concept representations.

Several works have investigated deep temporal modelling for event detection. Temporal modelling is performed for example using recurrent networks on video frames [73, 78, 79] or using memory networks [80]. Here, we focus on temporally orderless encodings of event videos, which have shown to be state-of-the-art for event detection [69, 76]. We show the potential of pre-training on large-scale ImageNet shuffles based on such encodings. Li et al. [31] leverage noisy web concepts to shot relevance in event videos, while Zhang et al. [77] and Yu et al. [74] learn which semantics are relevant using auxiliary textual or ontological knowledge about events. We also focus on determining desirable concepts for representing event videos, but without the need for expensive shot-level optimization or auxiliary event information.

2.2 Video Event Search

Different from event detection, event search focuses on ranking event videos without training examples. Only a user query is provided. The problem is typically tackled in two steps: detecting a rich set of concepts in videos and matching detected concepts to the textual input query. To map detected concepts to queries, a common solution is to perform ontology reasoning, which aims to select concepts that minimize a linguistic distance between the set of concept detectors and the individual query words [24, 47, 55, 66]. This is performed on large-scale linguistic ontologies such as WordNet [44]. For the concept detection, a wide range of works have advocated the importance of using large banks. Concept detector banks of multiple sizes have been proposed, e.g., based on 101 concepts [56], 311 concepts [63], 374 concepts [27, 28, 70], and 1,000 concepts [46]. The consensus seems to be that more concepts allow for a better selection of relevant concepts with respect to the input query. By extrapolation, it has been projected that event search continues to
benefit when increasing concept detector banks into the thousands [21]. In this work, we take this projection to heart and investigate event search using many thousands of concept detectors pre-trained on large-scale concept data. To map concepts to queries, we focus on a sparse selection of relevant concepts, but perform the concept to query mapping using learned word embeddings [36], rather than rule-based ontology systems.

2.3 Concept Hierarchies

For event detection, Ye et al. [72] propose EventNet, a framework to gather event-specific concepts from known event hierarchies. A hierarchy of 500 events is discovered from knowledge bases such as WikiHow and this event hierarchy is used to search for concepts on the web. The goal of our work is different in that we aim to arrive at a general bank of balanced concepts. We therefore focus on hierarchical relations among concept classes, rather than events.

For ImageNet specifically, Vreeswijk et al. [62] have previously shown that images from concepts at different layers of the hierarchy are visually distinct. General concepts benefit from including linked concepts from lower in the hierarchy. Building upon this work, we propose shuffling strategies that merge over-specific and under-represented concepts to more general concepts, resulting in concept banks that are better suited for event detection.

Girard et al. [16] show that for transfer learning of concepts, incorporating information from the most specific concepts in the ImageNet hierarchy benefits the performance for other image-based problems. We do so too when we shuffle the ImageNet hierarchy by focusing on avoiding the problems of class imbalance and over-specific concepts for event detection and search in videos.

For pre-training on ImageNet, a common pre-processing strategy is to perform data augmentation, e.g., through randomized cropping [22], warping [67], or deformations [50]. While data augmentation results in more training examples, it does not increase the variety of concepts, since no new classes are created. Ordonez et al. [49] perform a pre-processing of ImageNet concepts from leaf nodes to basic-level concepts. While we also pre-process the ImageNet hierarchy by moving up concepts, we do so to obtain a rich concept bank for event detection and search, rather than investigate the importance of basic-level objects for image classification.

3 OBTAINING SHUFFLED IMAGENET BANKS

We present two ways to shuffle the complete ImageNet concept hierarchy. To account for unbalanced and over-specific concepts, we exploit that concepts have different levels of granularity and are connected in a graph hierarchy. For ImageNet, this hierarchy is provided by WordNet [44]. The hierarchy allows us to alter concept class assignment and selection for balanced pre-training, which in turn leads to richer concept banks. More formally, let \( G = (C, E) \) denote a graph, where \( C \) denotes the set of concepts in the image datasets and \( E \) denotes the set of directed edges between concepts. Edge \( E_{ij} \) denotes a directed edge from concept \( i \) to \( j \), with \( i \) the parent concept of \( j \). We investigate shuffled reorganizations of this graph in both a bottom-up and a top-down manner. An illustrative overview of both approaches is shown in Figure 1 and discussed next.

3.1 Bottom-up Shuffle

For the bottom-up shuffle, we start from the full concept hierarchy, i.e., we start from the full graph \( G \) given by the large-scale image dataset. To perform the shuffle, we introduce four operations. The operations rebalance the graph hierarchy to become better suited for pre-training.

\textbf{Operation A (Roll)} For a graph \( G \) and concepts \( i, j, k \), the examples of all three concepts are merged to concept \( i \), iff \( \text{deg}(k) = 1, E_{j,k} \in C, \text{deg}(j) = 2, \) and \( E_{i,j} \in C. \)
Fig. 1. Schematic overview of the bottom-up and top-down shuffle of the concept hierarchy. The top-down shuffle selects concepts through a breadth-first search from the most general to the most specific concepts. The bottom-up shuffle consists of four operations to rebalance the concept hierarchy. Both approaches result in a large set of ImageNet concepts, which will be used to learn concept banks for event detection and search.

Fig. 2. Two visual examples of the four bottom-up shuffle operations (roll, bind, promote, subsample) on the ImageNet concept hierarchy.

In the definition of the above Roll operation, $deg(i)$ denotes the degree of concept $i$, i.e., the number of edges connected to $i$. The Roll operation can be seen as a condition to perform multi-level edge contraction on graph $G$. The idea behind the operation is to merge multi-level single child-parent connections. In large-scale image datasets, this occurs when a concept has one specific subcategory, which in turn has one specific subcategory. The motivation behind merging these occurrences into the most general concept is twofold: (i) there is little semantic difference between the child and parent concepts and (ii) a single child of a parent is more likely to be over-specific for event detection and search in videos. This holds especially for instances where single child-parent concepts occur consecutively. Therefore, we treat the examples within the concepts as one category.

Figure 2(a) shows two examples of the Roll operation on the ImageNet concept hierarchy. In both cases, there is a link between three concepts in a single child-parent fashion. Because of the minute
differences between the concepts and the lack of other children in these subgraphs, we merge the concepts into their grandparent concept, respectively, Mamba and Corn chip in the examples.

**Operation B1 (Bind)** For a graph $G$ and concepts $i, j_0, \ldots, j_Z$, the examples of all concepts are merged to concept $i$, iff $Z \geq 2, \forall z=1, \ldots, Z \text{deg}(j_z) = 1, \forall z=1, \ldots, Z E_{i,j_z} \in C$, and $\sum_{z=1}^{Z} c(j_z) \leq \tau_b$.

In the Bind operation, $c(j_z)$ denotes the number of examples for concept $j_z$. The operation specifies the scenario where a concept has multiple child concepts, which in turn have no child concepts. In other words, all child concepts do not have more specific subcategories. More importantly, the child concepts need to be scarce in the number of examples, given by a constraint $\tau_b$ on the total number of examples in the child concepts. The Bind operation tackles the occurrence of a set of small and coherent child concepts by merging them with their common parent concept. Individually, the child concepts might not have enough examples, but combined they form a consistent set with enough examples for proper pre-training.

Figure 2(b) shows two examples of the Bind operation on the ImageNet concept hierarchy. Both for the Balloon and the Tongueless frog concepts, the child concepts are scarce in the number of examples while these concepts do not have any other children. As such, we move the examples of the child concepts to the parent concepts.

**Operation B2 (Promote)** For a graph $G$ and concepts $i, j$, the examples of concept $i$ are merged to concept $j$, if $\text{deg}(i) = 1, E_{i,j} \in C$, and $c(i) \leq \tau_p$.

The Promote operation differs from the first two operations in that it is a unary concept operation. Akin to the Roll operation, it is a condition for performing edge contraction on graph $G$. The main difference is that in the Promote operation, the condition is strictly based on the properties of the individual concept $i$. The idea behind this operation is to simply promote those concepts that are individually not represented with enough examples to warrant a separate category. By moving such concepts up in the hierarchy to the parent concept, the examples are retained for pre-training, while dealing with the problem of the original concept being under-represented. We note that this operation is similar to the Bind operation, but on individual concepts, rather than sets of two or more concepts. This operation is performed strictly after the first two operations as an attempt to further move up trailing concepts in the hierarchy.

Figure 2(c) shows two examples of the Promote operation on the ImageNet hierarchy. Concepts Triclinium and New Yorker have only few examples and are therefore merged with their parent concept.

**Operation C (Subsample)** For a graph $G$ and concept $i$, the examples of concept $i$ are randomly subsampled during pre-training if $c(i) \geq \tau_s$.

The Subsample operation finally deals with the opposite problems of the first three operations. We note that the shuffle after the first three operations is finished and this operation does not further change graph $G$. The Subsample operation instead focuses on dealing with over-represented concepts. The subsampling is again for balancing purposes. If all examples on an over-represented concept are used during pre-training, then the network might overfit to this concept. Figure 2(d) shows two examples of the Subsample operation on the ImageNet hierarchy. Concepts such as Keyboard and Butterfly are popular concepts to photograph and will therefore have many examples. We subsample examples randomly during pre-training to avoid overfitting to such concepts.

While the bottom-up shuffle consists of four operations, they do not play an equal role. Roll serves as pre-processing, since the ImageNet hierarchy contains many irrelevant child-parent strings. Subsample serves as post-processing; without it, networks do not converge. Although the ImageNet hierarchy is a large graph, the operations have to be conducted only once offline before...
pre-training. The complete shuffle takes less than one hour on a single core, a small fraction of the total pre-training time.

3.2 Top-down Shuffle
An alternative and complementary shuffle is to start from the top of the hierarchy, rather than the bottom. Instead of working our way up from the most specific concepts, we start from the head node. Here, we investigate a breadth-first search approach. Let $\tau$ denote a pre-specified threshold stating the minimum amount of images required for a class to be used. Then, starting from the top layer of the hierarchy, we move down in the hierarchy and keep adding classes with at least $\tau$ images until we reach a desired amount of classes.

Procedurally, once we are at layer $l$, we list all ImageNet classes in layer $l + 1$ for which edges exist to the concepts in layer $l$. The retrieved concepts in layer $l + 1$ are then sorted by their number of examples to ensure that we select the most represented concepts first in case of a strict threshold $\tau$. Then, we iterate over the sorted list and select concepts using $\tau$. The whole process is in turn repeated for layer $l + 2$. With a breadth-first search as the top-down shuffle, we ensure that a balanced concept bank is obtained for event detection and search.

3.3 Concept Bank Pre-training
After performing either the bottom-up or top-down shuffle, we end up with a set of concept categories $C_R$, where each concept can contain examples from multiple concepts in the case of the bottom-up shuffle. Using the concepts in $C_R$, we pre-train a convolutional network in a standardized fashion. Let $f_{\text{CNN}} : \mathbb{R}^{256 \times 256 \times 3} \rightarrow \mathbb{R}^{C_R}$ denote an arbitrary convolutional network that maps an input image to a $C_R$ manifold. We then compute a probability distribution over the concepts using the softmax function. For training example $i$, let $y_i \in C_R$ denote the concept label and let $p_i$ denote the probability distribution over the concepts. The loss for example $i$ is then computed using cross-entropy as

$$L(i) = \sum_{j=1}^{C_R} [[y_i = j]] \cdot \log(p_{ij}),$$

where $[[\cdot]]$ denotes the indicator function and $p_{ij}$ denotes the output probability of being class $j$ for training example $i$. The incurred loss is backpropagated through the network during pre-training to update the model parameters. After pre-training, the network is used to obtain semantic video representations for event detection and search. For a pre-trained convolutional network, the feature extractor is defined as $\phi_L : \mathbb{R}^{256 \times 256 \times 3} \rightarrow \mathbb{R}^{D_L}$, where $L$ denotes the layer in the network and $D_L$ denotes the dimensionality of the activations at the $L$th layer.

4 SHUFFLED IMAGENET BANKS FOR DETECTING AND SEARCHING EVENTS
Shuffling the ImageNet concept hierarchy, followed by pre-training, results in a deep network that can be used to extract semantic representations on individual video frames. Here, we explain how to aggregate these frame representations into video representations to perform event detection from limited amounts of training examples and event search from text queries. We outline how to aggregate our obtained frame representations into video representations by building on established pooling for detection [29, 39, 45, 48, 57] and text matching for event search [23].

4.1 Event Detection
For detection of an event $E$, we are given a training set $\{V_i, Y_i\}_{i=1}^N$, consisting of $N$ examples, where $V_i$ denotes the $i$th video and $Y_i \in \{0, 1\}$ states the binary label with respect to $E$. For video $V_i$, we first sample frames uniformly, i.e., $V_i = \{F_{ij}\}_{j=1}^{|V_i|}$, where $|V_i|$ denotes the number of sampled frames
in video $i$. For each frame $F_{ij}$, we employ our network to yield a concept bank representation, denoted as $\phi_L(F_{ij}) \in \mathbb{R}^{D_L}$, where again $D_L$ denotes the feature dimensionality and $L$ denotes the used layer in the network for feature extraction. To perform event detection, we need to aggregate the frame representations into fixed-sized video concept bank. In this work, we employ pooling [29, 37, 39] and codebook approaches [57, 69].

**Average pooling.** Following References [29, 39], we perform pooling by taking the average value of each feature activation over the frame in a video. For video $V_i$, this yields the following concept bank:

$$\psi_{\text{avg}}(V_i) = \frac{1}{|V_i|} \sum_{j=1}^{|V_i|} \phi_L(F_{ij}).$$

This results in a fixed-sized representation of dimension $D_L$ for each video.

**Codebook encoding.** For a codebook encoding, we employ Fisher Vectors [53], using video frames as local features to aggregate over a video [45, 48, 57]. Where Fisher Vectors employ Gaussian Mixture Models to generate a codebook with anisotropic means, we found that more robust estimates can be made by separately estimating the means and variances of the codebook clusters. This is due to the small sample size in the training set to estimate the codebook. We compute the codebook means through $k$-means clustering, after which the variance is computed as the isotropic variances of the frame features assigned to each codebook mean. Given the estimated means and variances, we obtain a $(2D_LB)$-dimensional concept bank per video, where $D_L$ denotes the dimensionality of the frame representations and $B$ denotes the number of codebook elements. Following Reference [53], we perform a normalization by the number of samples, as well as $\ell_2$-normalization and power normalization.

Given a fixed-sized representation for each video using either pooling or codebook encodings, we perform event training with a non-linear SVM [3] on the concept bank and the binary labels of the training videos. During testing, a fixed-sized concept bank is computed for each test video in similar fashion. The videos are ranked by their SVM confidence score.

### 4.2 Event Search

Where event detection deals with scoring videos using models learned from training examples, event search deals with scoring videos given just a textual description of the event. We follow the setup for event search as specified by the TRECVID Ad-hoc Video Search benchmark, where we score video shots, rather than whole videos [1]. For event $E$, let $E_T = \{E_{(i)}^T\}_{i=1}^T$ denote its textual description consisting of $T$ words, such as *A crowd demonstrating in a city street at night*. To score video shots with respect to query $T$, we employ a three-step approach.

The first step in event search is to obtain semantic scores for video frames. For video $V$, we sample frames uniformly and feed each sampled frame to the pre-trained network. We use the concept probabilities at the softmax layer as the frame representation. This results in a score matrix $V_S \in \mathbb{R}^{C_R \times |V|}$ for $|V|$ sampled video frames and $C_R$ concepts in our bank.

In the second step, we aggregate frame-level scores into shot-level scores. Since video shots can be of arbitrary length, we found that simply aggregating only the frames within the shot itself is suboptimal. Therefore, we first aggregate video frames into overlapping windows of seven sampled frames. For each concept $c \in C_R$, we perform the aggregation by computing the $p$th percentile filter score. The percentile filter score is computed by sorting the concepts scores within the window and taking the score at the $p$th percentile. This approach over the maximum concept score over a video is more robust, as it disregards single outliers. The result is a single score per concept per window.
The third step revolves around scoring shots. To this end, we first compute a semantic match between query \( E_T \) and each concept \( c \in C_R \) using word embeddings \([43]\). Word embeddings provide a feature vector for each word, which can be used to compute similarities. To obtain the word embeddings \( y_{E_T} \) for query \( E_T \), we first split the query into individual words. For each word, we use a pre-trained word2vec model to obtain a 500-dimensional representation. This representation is averaged over the words in the query \([23, 25, 41]\). The same procedure is performed for each concept \( c \) to obtain \( y_c \). The semantic match is as such given as

\[
s_{w2v}(E_T, c) = \cos(y_{E_T}, y_c) .
\]

After computing all the similarities, we keep the \( K \) most similar concepts. Let \( E_{C_R} \) denote the selected concepts for event \( E \). Then, we score each window \( W \) in video \( V \) as follows:

\[
s(V, E_T, E_{C_R}) = \sum_{c \in E_{C_R}} s_{w2v}(E_T, c) \cdot V_W^{(c)} ,
\]

where \( V_W^{(c)} \) denotes the percentile score for concept \( c \) in window \( W \) of video \( V \). Finally, for a video shot, we take the maximum score across all windows that overlap with the shot as the final score for event \( E \).

5 EXPERIMENTAL SETUP

5.1 Datasets

TRECVID MED 2013. For the event detection evaluation, we employ the 2013 TRECVID Multimedia Event Detection (MED) dataset. This dataset consists of over 27,000 test videos and 20 events, such as Birthday party, Parade, and Dog show. We evaluate on the 10ex and 100ex settings, where, respectively, 10 and 100 positive video examples are given per event during training. Furthermore, a generic background set of over 5,000 videos guaranteed to not contain any positive video is provided as negative examples for each event.

TRECVID MED 2014. We also evaluate event detection on the 2014 TRECVID Multimedia Event Detection dataset. This dataset also consists of over 27,000 test videos and 20 events. The events are partially taken from the MED 2013 dataset and partially contain new events, such as Beekeeping, Felling a tree, and Tuning a musical instrument. The training is akin to the MED 2013 dataset, with 10 and 100 positive examples per event and a generic background set of over 5,000 videos as the negative examples.

TRECVID AVS 2016. For the event search evaluation, we employ the 2016 TRECVID Ad-hoc Video Search (AVS) dataset. This dataset consists of 4,593 videos, with 600 hours of footage in total. A set of 30 event queries are provided, such as A person playing drums indoors, Military personnel interacting with protesters, and People shopping. Each video is split into shots, which are provided by TRECVID, and the aim is to return the shot most relevant for each query.

5.2 Implementation Details

Networks. We leverage three well-known convolutional architectures, namely, GoogLeNet from Szegedy et al. \([58]\), ResNet from He et al. \([22]\), and ResNeXt from Xie et al. \([68]\). To pre-train all three networks, we utilize the MxNet library \([9]\). We pre-train for 40 epochs using SGD, with a learning rate of 0.01, a momentum of 0.9. Learning rate decay is incorporated by dividing it by 10 every 10 epochs. A batch size of 270 is used. The pre-trained models are publicly available at https://github.com/psmmettes/shuffled-imagenet-bank.

Frame and video concept banks. For event detection, we extract features from individual video frames. For each video, we sample two frames per second. Sampled frames are fed to the pre-trained network, where we use either the activations at the second fully connected layer or at
the softmax layer. For the codebook encoding, we use eight clusters. After performing the frame pooling over the video, we employ $\ell_1$-normalization.

**Event detection training and testing.** For each event, we train a non-linear SVM classifier with a Histogram-Intersection Kernel on the pooled video representations. The $C$ parameter is fixed to 100 through all experiments. During testing, we use the distances to the separating hyperplane as the score for each test video.

**Event search testing.** For each event query, we use the three most semantically similar concepts in our ImageNet hierarchy to compute the event score. We found that using more concepts did not result in better search performance. We sample one frame per second, and we use the second-highest score for each concept in each window as the percentile score. For the word embedding, we employ word2vec [43], pre-trained on Wikipedia text.

6 EXPERIMENTAL RESULTS

6.1 Event Detection

In the first experiment, we evaluate the effectiveness of the shuffled ImageNet bank for event detection. For each event, 10 or 100 training videos are provided as positive examples. We start with four ablation studies to evaluate the most prominent elements of our approach. The ablation studies are performed on the workings of our approach for event detection. Unless stated otherwise, we employ average pooling as the video representation.

**Ablation study 1: Evaluating shuffles.** In the first ablation study, we evaluate the performance of our shuffle strategies and parameters. For bottom-up shuffle, we evaluate three different networks using a combined set of 4,437 concepts ($\tau_b = 7,000$, $\tau_p = 1,250$), 8,201 concepts ($\tau_b = 7,000$, $\tau_p = 500$), and 12,988 concepts ($\tau_b = 3,000$, $\tau_p = 200$). We furthermore perform top-down shuffle with 4,000 concepts. All approaches are compared to the baseline, which applies pre-training on the standardized 1,000 concepts from the Large Scale Visual Recognition Challenge [52].

Table 1 provides the event detection results on TRECVID MED 2013, both using 10 and 100 training examples per event. All models are pre-trained using the GoogLeNet architecture [58]. For all approaches, we evaluate the performance using frame representations at the fully connected layer and softmax layer. We observe that all variants of our bottom-up shuffle obtain favourable event detection results compared to pre-training on the standardized 1,000 concepts from the Large Scale Visual Recognition Challenge [52]. This result shows that using the full ImageNet hierarchy based on reorganized pre-training is preferred over standard pre-training on a smaller part of ImageNet, as is done, e.g., in References [5, 26, 68]. We find that retaining more concepts in the shuffle is beneficial when using features at the softmax layer, and we recommend using the 12,988 concept set. When using features at the fully connected layer, retaining 4,437 concepts is preferred. For both 10 and 100 training examples per event, fully connected features obtain slightly higher scores. We also compare our shuffles to a baseline that uses all ImageNet concepts with at least 50 examples. We find that this baseline scores lower for both the 10 and 100 example scenarios. This indicates that merely using many concepts is not optimal, they need to be combined and shuffled. Another downside of using all available concepts is that the ImageNet hierarchy is ignored, resulting in many over-specific and irrelevant concepts. For the top-down shuffle, we find that using more concepts benefit the event detection results, although the effect diminishes from 2,000 to 4,000 concepts. We recommend using 4,000 concepts for the top-down strategy.

When merging the bottom-up shuffle with 12,988 concepts and the top-down shuffle with 4,000 concepts on TRECVID MED 2013 in the 100ex setting using GoogLeNet FC features, the mAP improves to 0.465 compared to 0.448 (bottom-up) and 0.438 (top-down). The independent training of the shuffles results in complementary representations for event detection. The computational
Table 1. Ablation Study 1 for Event Detection on TRECVID MED 2013, Where 10ex and 100ex Denote the Number of Examples per Event for Training

<table>
<thead>
<tr>
<th>(nr. concepts)</th>
<th>( \tau_b )</th>
<th>( \tau_p )</th>
<th>10ex FC</th>
<th>softmax</th>
<th>100ex FC</th>
<th>softmax</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Baselines</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1,000 (LSVRC [52])</td>
<td>n/a</td>
<td>n/a</td>
<td>0.262</td>
<td>0.190</td>
<td>0.401</td>
<td>0.341</td>
</tr>
<tr>
<td>18,856 (all classes †)</td>
<td>n/a</td>
<td>n/a</td>
<td>0.288</td>
<td>0.280</td>
<td>0.436</td>
<td>0.434</td>
</tr>
<tr>
<td><strong>Bottom-up shuffle</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4,437</td>
<td>7,000</td>
<td>1,250</td>
<td>0.292</td>
<td>0.270</td>
<td><strong>0.448</strong></td>
<td>0.410</td>
</tr>
<tr>
<td>8,201</td>
<td>7,000</td>
<td>500</td>
<td>0.293</td>
<td>0.288</td>
<td>0.435</td>
<td>0.431</td>
</tr>
<tr>
<td>12,988</td>
<td>3,000</td>
<td>200</td>
<td>0.290</td>
<td>0.286</td>
<td>0.424</td>
<td>0.441</td>
</tr>
<tr>
<td><strong>Top-down shuffle</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1,000</td>
<td>n/a</td>
<td>n/a</td>
<td>0.263</td>
<td>0.225</td>
<td>0.394</td>
<td>0.355</td>
</tr>
<tr>
<td>2,000</td>
<td>n/a</td>
<td>n/a</td>
<td>0.290</td>
<td>0.246</td>
<td>0.414</td>
<td>0.397</td>
</tr>
<tr>
<td>4,000</td>
<td>n/a</td>
<td>n/a</td>
<td><strong>0.300</strong></td>
<td>0.266</td>
<td>0.438</td>
<td>0.428</td>
</tr>
</tbody>
</table>

FC (fully connected) and softmax denote the network layers used for frame representation. † denotes all classes with at least 50 examples. Shuffling always improves the performance compared to the baseline that pre-trains on the 1,000 standard concepts of LSVRC [52]. We recommend to use features from the fully connected layer. No single shuffle performs best across both tasks; for 10ex, top-down shuffle into 4,000 concepts works best, while for 100ex, bottom-up shuffle into 4,437 concepts works best.

Overhead between different shuffles is negligible during inference, since all but the very last layer have the same size. Hence, once a network is pre-trained, computation is no factor in selecting a specific shuffle. The pre-training cost is higher for the shuffles with more concepts; the bottom-up shuffles with 8,201 and 4,437 concepts are roughly 1.3 and 1.7 times faster to train than the shuffle with 12,988 concepts. Overall, we conclude that the shuffled ImageNet bank has a positive effect for event detection using both 10 and 100 training examples per event.

To analyze why the shuffled ImageNet bank is effective for event detection, we have compared our concept distributions with 12,988 concepts to the standard subset of 1,000 classes [52]. For both concept banks, we have investigated their topic distribution, where the topic is given by the high-level entity of each concept based on the WordNet hierarchy. For each concept, we assign it to the high-level entity in the third level of the WordNet hierarchy. The log-normalized distributions for our concept set and the standard concept set are shown in Figure 3. We find that our bank enables us to pre-train not only on a larger set of concepts, but more importantly, on a more balanced and wider variety of topics. We will therefore opt for the shuffled ImageNet bank over standard ImageNet bank in further ablation studies.

**Ablation study 2: Evaluating networks.** In the second ablation study, we investigate the effect of different convolutional network architectures on the event detection performance. We employ the 12,988 concept bank from the bottom-up shuffle for the experiment, and we investigate three networks: GoogLeNet [58], ResNet [22], and ResNeXt [68]. In Table 2, we show the event detection results on TRECVID MED 2014, again using 10 and 100 examples for the fully connected and softmax frame representations. Across all settings, we find that ResNet outperforms GoogLeNet, while ResNeXt in turn outperforms ResNet. The performance relation ResNeXt > ResNet > GoogLeNet also holds for concept classification in images [68]. This result implies that the better a network is at recognizing concepts, the more useful the network is for event detection. We therefore employ ResNeXt as the network of choice throughout the rest of the experiments, unless stated otherwise.

---

**Table 2. Ablation Study 2 for Event Detection on TRECVID MED 2014**

<table>
<thead>
<tr>
<th>(nr. concepts)</th>
<th>( \tau_b )</th>
<th>( \tau_p )</th>
<th>10ex FC</th>
<th>softmax</th>
<th>100ex FC</th>
<th>softmax</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Baselines</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1,000 (LSVRC [52])</td>
<td>n/a</td>
<td>n/a</td>
<td>0.262</td>
<td>0.190</td>
<td>0.401</td>
<td>0.341</td>
</tr>
<tr>
<td>18,856 (all classes †)</td>
<td>n/a</td>
<td>n/a</td>
<td>0.288</td>
<td>0.280</td>
<td>0.436</td>
<td>0.434</td>
</tr>
<tr>
<td>4,437</td>
<td>7,000</td>
<td>1,250</td>
<td>0.292</td>
<td>0.270</td>
<td><strong>0.448</strong></td>
<td>0.410</td>
</tr>
<tr>
<td>8,201</td>
<td>7,000</td>
<td>500</td>
<td>0.293</td>
<td>0.288</td>
<td>0.435</td>
<td>0.431</td>
</tr>
<tr>
<td>12,988</td>
<td>3,000</td>
<td>200</td>
<td>0.290</td>
<td>0.286</td>
<td>0.424</td>
<td>0.441</td>
</tr>
<tr>
<td><strong>Top-down shuffle</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1,000</td>
<td>n/a</td>
<td>n/a</td>
<td>0.263</td>
<td>0.225</td>
<td>0.394</td>
<td>0.355</td>
</tr>
<tr>
<td>2,000</td>
<td>n/a</td>
<td>n/a</td>
<td>0.290</td>
<td>0.246</td>
<td>0.414</td>
<td>0.397</td>
</tr>
<tr>
<td>4,000</td>
<td>n/a</td>
<td>n/a</td>
<td><strong>0.300</strong></td>
<td>0.266</td>
<td>0.438</td>
<td>0.428</td>
</tr>
</tbody>
</table>

FC (fully connected) and softmax denote the network layers used for frame representation. † denotes all classes with at least 50 examples. Shuffling always improves the performance compared to the baseline that pre-trains on the 1,000 standard concepts of LSVRC [52]. We recommend to use features from the fully connected layer. No single shuffle performs best across both tasks; for 10ex, top-down shuffle into 4,000 concepts works best, while for 100ex, bottom-up shuffle into 4,437 concepts works best.

---

**Ablation study 2: Evaluating networks.** In the second ablation study, we investigate the effect of different convolutional network architectures on the event detection performance. We employ the 12,988 concept bank from the bottom-up shuffle for the experiment, and we investigate three networks: GoogLeNet [58], ResNet [22], and ResNeXt [68]. In Table 2, we show the event detection results on TRECVID MED 2014, again using 10 and 100 examples for the fully connected and softmax frame representations. Across all settings, we find that ResNet outperforms GoogLeNet, while ResNeXt in turn outperforms ResNet. The performance relation ResNeXt > ResNet > GoogLeNet also holds for concept classification in images [68]. This result implies that the better a network is at recognizing concepts, the more useful the network is for event detection. We therefore employ ResNeXt as the network of choice throughout the rest of the experiments, unless stated otherwise.
Fig. 3. Log-normalized topic distributions of the standard 1,000 ImageNet concepts of Reference [52] and our set of 12,988 ImageNet concepts. The topic of a concept is defined as the high-level entity of each concept given by the WordNet hierarchy (third hierarchy level). The inner circle shows three high-level concept entities, with the closest event match in the outer circle. Our concepts cover a wider variety and a more balanced set of topics than the standard 1,000 concepts.

Table 2. Ablation Study 2 for Event Detection on TRECVID MED 2014

<table>
<thead>
<tr>
<th></th>
<th>GoogLeNet</th>
<th>ResNet</th>
<th>ResNeXt</th>
</tr>
</thead>
<tbody>
<tr>
<td>10 examples per event</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FC</td>
<td>0.256</td>
<td>0.311</td>
<td>0.324</td>
</tr>
<tr>
<td>Softmax</td>
<td>0.254</td>
<td>0.293</td>
<td>0.286</td>
</tr>
<tr>
<td>100 examples per event</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FC</td>
<td>0.355</td>
<td>0.401</td>
<td>0.422</td>
</tr>
<tr>
<td>Softmax</td>
<td>0.359</td>
<td>0.395</td>
<td>0.408</td>
</tr>
</tbody>
</table>

ResNeXt outperforms ResNet, which in turn outperforms GoogLeNet. This is in line with the order for image classification [68], indicating that better concept classification results in better event detection.

We furthermore employ frame representations at the fully connected layer for our further event detection experiments, since these outperform softmax representations slightly.

**Ablation study 3: Evaluating encodings.** So far, we have used average pooling as the video concept bank. In the third ablation study, we evaluate the effect of employing codebook encoding. Using the bottom-up shuffle with 12,988 concepts and features at the fully connected layer, average pooling obtains an mAP of 0.324 (10 examples per event) and 0.422 (100 examples per event). When using the codebook encoding with five clusters, the performance improves to 0.342 (10 examples per event) and 0.462 (100 examples per event). While somewhat more effective for event detection, the codebook encoding comes at both additional computational overhead and with a video representation that is five times larger in size. As such, we recommend codebook encoding when computational and storage complexity is sufficient and average pooling otherwise.
Table 3. Ablation Study 4 for Event Detection on TRECVID MED 2014

<table>
<thead>
<tr>
<th>Modality</th>
<th>10ex</th>
<th>100ex</th>
</tr>
</thead>
<tbody>
<tr>
<td>Individual modalities</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Audio (MFCC) [13]</td>
<td>0.068</td>
<td>0.101</td>
</tr>
<tr>
<td>Motion (Dense Trajectories) [64]</td>
<td>0.128</td>
<td>0.261</td>
</tr>
<tr>
<td>This article (avg)</td>
<td>0.324</td>
<td>0.422</td>
</tr>
<tr>
<td>Fusion</td>
<td></td>
<td></td>
</tr>
<tr>
<td>This article (avg) + audio</td>
<td>0.323</td>
<td>0.480</td>
</tr>
<tr>
<td>This article (avg) + motion</td>
<td>0.306</td>
<td>0.429</td>
</tr>
<tr>
<td>This article (avg) + audio + motion</td>
<td>0.309</td>
<td>0.459</td>
</tr>
</tbody>
</table>

Our approach outperforms the audio and motion modalities. A fusion with the audio modality further improves results in the 100ex setting.

As additional comparison, we employ a recurrent network on our frame representations [10, 73]. We use an LSTM layer on 20 input frames with 512 features, followed by two fully connected layers with 512 and 256 features, using ReLU activations and trained with binary cross-entropy [73]. We train for 150 epochs with a learning rate of 0.01. We obtain an mAP of 0.22 for 10 examples. While outperforming features from other modalities, this baseline is not as effective as average and codebook-based pooling with the same features, and we will therefore not adopt it.

**Ablation study 4: Comparing modalities.** In the fourth ablation study, we compare to other multi-modal video representations. In Table 3, we compare to video representations based on audio and motion information. The results show that the shuffled ImageNet bank is preferred over audio and motion representations. We have also performed a multi-modal fusion. We find that for the 100ex setting, adding audio information improves the performance of our bank. This does not hold, however, for the 10ex setting. A fusion with motion is not beneficial for event detection. We conclude that the shuffled ImageNet bank improves over other common modalities and fusion with audio information further boosts the results.

To gain insight into when a fusion with other modalities is beneficial, we show the effect of a fusion with motion and audio per event in Figure 4. We find that adding motion helps for instructional event types, such as Tuning a musical instrument or Felling a tree. This benefit is counteracted by events such as Tailgating and Town hall meeting, which do not have a strong motion component. Adding audio improves the scores for 16 of the 20 events, especially music-related events. Audio is less informative for events regarding groups of people, such as Wedding shower and Tailgating. We conclude that a multimodal fusion can be beneficial, but depends on the event. We recommend a fusion with audio as it tends to improve the results.

**Qualitative analysis.** To better understand how our approach works and when it succeeds or fails, we perform a qualitative analysis by inspecting the top ranked videos for four events in TRECVID MED 2014, as shown in Figure 5. For Dog show in Figure 5(a), all top ranked videos are correct. This is because the event triggers on detections of dogs, which are highly present in ImageNet. For Attempting a bike trick (Figure 5(b)) and Fixing a musical instrument (Figure 5(c)), we obtain mixed results. Videos from Attempting a bike trick trigger on concepts such as bikes. This is also reflected in failure cases, which include person biking without performing tricks. For Fixing a musical instrument, failure cases stem from confusion with highly similar events (e.g., instrument tuning) or different events in overlapping settings (e.g., cleaning with an instrument present). These two events show that shuffled ImageNet banks, while providing a wider variety...
Fig. 4. Per event improvements on TRECVID MED 2014 for a multimodal fusion with motion and audio, respectively. Fusion is beneficial, but depends on the event. For motion, there is an equal amount of events that are downgraded by the fusion, while for audio, the fusion is more helpful in general. We recommend audio for multimodal fusion.

of concepts, struggle to differentiate different actions performed with similar concepts. Last, for Giving directions to a location in Figure 5(d), we observe that the top ranked videos are nearly all incorrect. We attribute this to the fact that this event depends heavily on semantic understanding from speech. This is also an indicator for the jump in performance when incorporating audio in the fourth ablation study. We conclude that shuffled ImageNet banks are able to rank event videos correctly when they are associated with unique and consistent concepts. To further improve the event detection performance, better understanding of motion and semantic understanding from speech seem key.

6.2 Event Search

Next, we evaluate the effect of the shuffled ImageNet bank for event search from text queries on the TRECVID Ad-hoc Video Search (AVS) 2016 dataset. We first provide an ablation study on the type of shuffle and the used network. Afterwards, we perform an analysis of the role of concepts in event search and the relation between concepts and event queries. We report the (mean) inferred Average Precision score [1].

Ablation study. Figure 6 shows the results for three shuffled ImageNet bank variants, each trained on ResNet and ResNeXt. For all settings, we use the three most semantically similar objects per event. We find that for event search, including more concepts in the bank slightly improves the results. Contrary to event detection, we find that using ResNeXt does not lead to increased performance over ResNet. In fact, the mean Inferred Average Precision is a bit lower for ResNeXt across all variants. Next to standard inferred AP, which uses the top 1.000 shots (iAP 0.065), we also report inferred AP for the top 100 (iAP 0.085), top 50 (iAP 0.117), and top 10 (iAP 0.154) shots. At the top of the list, scores are higher, which indicates that the highest scoring shots correlate more with their corresponding event relevance. We conclude that using bottom-up shuffling with many concepts is slightly preferred for event search, while more sophisticated models do not lead to better results.
Fig. 5. Qualitative analysis of the top five ranked videos for four events in TRECVID MED 2014 trained using 10 training examples per event. For Dog show, the top ranked videos are all correct, triggered by detections of dogs. For Attempting a bike trick and Fixing an instrument, mistakes among the top ranked videos are examples where the correct concepts are detected, but different actions are performed. For events without consistent settings and concepts, such as Giving directions to a location, our approach struggles to rank correct videos among the top. Here, a multimodal approach including audio is more advantageous.

Fig. 6. Evaluation of three shuffled ImageNet banks, each trained on two networks, for event search on the 2016 TRECVID Ad-hoc Video Search task. We find that including more concepts in the bank is slightly preferred for event search. Contrary to event detection, deeper networks do not result in better event search results. The reason behind this observation is investigated in Figure 7 and Table 4.
Fig. 7. Relating concept quality versus event search quality on AVS 2016. The linear relation plotted in gray shows that there is almost no relation between the two variables. Knowing the quality of the concept in the bank provides little information about the quality of the event search.

Table 4. Qualitative Examples of Event Queries and Selected Concepts, Compared to an Oracle Selection of Concepts

<table>
<thead>
<tr>
<th>A person playing drums indoors</th>
<th>A person playing drums indoors</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Selected top concepts</strong></td>
<td><strong>Selected top concepts</strong></td>
</tr>
<tr>
<td>guitarist, guitar player, outdoor game, drum, drumfish</td>
<td>helmet (protective), helmet (armor), pith hat, pith helmet</td>
</tr>
<tr>
<td>Percussionist, cymbal, drummer, drum, membraphone</td>
<td><strong>Oracle top concepts</strong></td>
</tr>
<tr>
<td><strong>Oracle concepts</strong></td>
<td>hockey skate, hockey stick, ice hockey, hockey</td>
</tr>
</tbody>
</table>

The examples show two open issues in event search from concepts: Left: concept ambiguity. Selecting different but semantically similar concepts can lead to different results. Right: the domain gap between of concepts depicted in images and videos. While the selected concepts match well with the event query, the oracle concepts obtain high event search scores, because the context of these concepts match better with the context of the event query.

To better understand why deeper networks do not lead to better event search, we investigated the role of concept quality and their link to the event queries. In Figure 7, we plot the link between the recognition quality of the concepts selected by each event query and the quality of the event search. Here, concept quality is quantified as the mean Average Precision of the concepts on the validation set of ImageNet. The gray line states the linear relation between the two observed entities. The figure shows that on the AVS 2016 dataset, we find little correlation between concept quality and event search performance. This indicates that better concept networks do not directly improve event search, even though event queries are strongly related to semantics, which are supposed to be carried out by the networks. A possible reason is that the use of word embeddings result in a suboptimal match between queries and relevant concepts, resulting in diminishing effects of the networks themselves.

**Concept analysis.** To understand which factors do have a large impact on the event search, we highlight two problems with the top concept selection for two event queries in Table 4. The table compares the concepts selected by our approach to an oracle concept selection. Oracle selection is performed by selecting those concepts that in hindsight would have given the highest event
search performance. For each event in the oracle selection, we perform a greedy optimization by ranking the concepts based on their individual performance on event search and selecting the top concepts. For the event query *A person playing drums indoors*, we observe the problem of ambiguity in the concept selection. While our top concepts are seemingly relevant for the event, the oracle concepts hardly overlap with our top concepts. Contrarily, the oracle concepts are other concepts with similar semantic relevance to the event. This example indicates that semantically similar concepts can lead to different event detection results. For the event query *A person wearing a helmet*, we observe another problem. While our concepts match the event query well, the actual best concepts to select are not directly related to the event. This discrepancy is a result of a domain gap between concepts in images and videos. Helmet categories in ImageNet are clear full-scale pictures of helmets only, while in videos, helmets are small objects worn by people. Concepts such as *Ice hockey*, however, generally have image examples of people wearing helmets, but in different contexts. These concepts are therefore highly relevant for our event query, even though the concepts are not from the same semantic scene. The performed analyses state that while concept quality is not a major factor for event search, concept ambiguity and the domain gap of concepts between images and videos are important aspects to consider for future progress.

### 6.3 State-of-the-Art Comparison

For the comparison to the state-of-the-art in event detection and search, we consider two benchmark tasks. For event detection, we provide a comparison on the 2015, 2016, and 2017 evaluations of the TRECVID Multimedia Event Detection benchmark, using 10 examples per pre-specified event. For event search, we provide a comparison on the 2017 evaluation of the TRECVID Ad-hoc Video Search benchmark. We report the Inferred Average Precision.

In Figures 8(a)–8(c), we provide the comparison on TRECVID MED 2015, 2016, and 2017. We note that the state-of-the-art comparison is not an apples-to-apples comparison, as each approach utilizes different features, modalities, and pre-training strategies for optimal results. The comparison serves to show the overall effectiveness of the shuffled ImageNet bank for event detection and search. The figure shows that the bank outperforms the results of all other participants for all three years. The results can even be improved by a fusion with other modalities, akin to Table 3, although the improvements are modest. In Table 5, we compare to other papers in event detection.
Table 5. Comparative Evaluation to the State-of-the-Art in Event Detection on TRECVID MED 2014

<table>
<thead>
<tr>
<th>Modality</th>
<th>10ex</th>
<th>100ex</th>
</tr>
</thead>
<tbody>
<tr>
<td>Visual</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Motion</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Audio</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Wang et al. [64] ✓ 0.128 0.261
Lan et al. [30] ✓ 0.149 0.290
Mazloom et al. [38] ✓ 0.166 -
Mettes et al. [42] ✓ - 0.276
Tran et al. [60] from [12] ✓ 0.205 0.314
Yu et al. [75] ✓ ✓ ✓ 0.233 0.326
Gan et al. [14] ✓ - 0.344
Chang et al. [6] ✓ - 0.344
Ma et al. [34] ✓ ✓ 0.238 0.387
Xu et al. [69] ✓ 0.245 0.368
Chang et al. [7] ✓ ✓ 0.258 -
Chang et al. [8] ✓ ✓ 0.272 0.381
Ma et al. [35] ✓ ✓ 0.272 0.382
Zha et al. [76] ✓ ✓ - 0.387
Fan et al. [12] ✓ ✓ 0.296 0.415
This article ✓ ✓ ✓ 0.324 0.422
This article ✓ ✓ ✓ 0.323 0.480

Results are reported for the settings with 10 and 100 training examples per event using mean Average Precision. The shuffled ImageNet bank results in state-of-the-art event detection performance, with further improvements possible when adding audio.

In Figure 8(d), we provide a comparison on TRECVID AVS 2017. The shuffled ImageNet bank yields top performance for event search, outperforming other participants, while a fusion works best. We conclude that when searching for events from text queries, the shuffled ImagNet bank is also preferred.

7 CONCLUSIONS

This work investigates the detection and search of events in videos using concept banks pre-trained on ImageNet. Rather than starting from the standard 1,000 ImageNet concepts, we use the full hierarchy containing 21,841 concepts. To deal with class imbalance and over-specific concepts, we propose a top-down and bottom-up shuffle. This results in the shuffled ImageNet bank; concept representations with an order of magnitude more concepts. Where this article focuses on the effect of the shuffled ImageNet bank for event detection and search, the concept bank is of interest in more research problems. The previous iteration of the bank [40] has already found applications in video captioning [11], visualizing image collections [61], and detecting violence in videos [32] amongst others. We hope that by making the shuffled ImageNet banks of this article publicly available, a broad range of multimedia problems can benefit from the concept bank representations.

REFERENCES


