Abstract

In this paper we describe our TRECVID 2011 video retrieval experiments. The MediaMill team participated in two tasks: semantic indexing, and multimedia event detection. The starting point for the MediaMill detection approach is our top-performing bag-of-words system of TRECVID 2010, which uses multiple color SIFT descriptors, sparse codebooks with spatial pyramids, and kernel-based machine learning. All supported by GPU-optimized algorithms, approximated histogram intersection kernels, and multi-frame video processing. This year our experiments focus on 1) the soft assignment of descriptors with the use of difference coding and 2) the utility of 1,346 concept detectors for event retrieval. The 2011 edition of the TRECVID benchmark has again been a fruitful participation for the MediaMill team, resulting in the runner-up ranking for concept detection in the semantic indexing task.

1 Introduction

Robust video retrieval is highly relevant in a world that is adapting swiftly to visual communication. Online services like YouTube and Vimeo show that video is no longer the domain of broadcast television only. Video has become the medium of choice for many people communicating via the Internet. Most commercial video search engines provide access to video based on text, as this is still the easiest way for a user to describe an information need. The indices of these search engines are based on the filename, surrounding text, social tagging, closed captions, or a speech transcript. This results in disappointing retrieval performance when the visual content is not mentioned, or properly reflected in the associated text. In addition, when the videos originate from non-English speaking countries, such as China, or the Netherlands, querying the content becomes much harder as robust automatic speech recognition results and their accurate machine translations are difficult to achieve.

To cater for robust video retrieval, the promising solutions from literature are mostly concept-based [21], where detectors are related to objects, like a flag, scenes, like a beach, and people, like female human face closeup. Any one of those brings an understanding of the current content. The elements in such a lexicon of concept detectors offer users a semantic entry to video by allowing them to query on presence or absence of visual content elements. Last year we presented the MediaMill 2010 semantic video search engine [18], which made our robust concept detection system more efficient [11, 24, 26]. We have recently shown that progress in visual concept search has doubled in just 3 years [17]. Surprisingly, the progress even holds for cross-domain visual search engines, albeit with a loss in performance compared to within-domain search engines. This year our experiments focus on 1) the soft assignment of descriptors with the use of difference coding and 2) the utility of 1,346 concept detectors for event retrieval. Taken together, the MediaMill 2011 semantic video search engine provides users with robust semantic access to Internet video collections.

The remainder of the paper is organized as follows. We first define our bag-of-words foundation in Section 2. Then we highlight our detection approaches for concepts in Section 3. We summarize our efforts in the multimedia event detection task in Section 4.

2 Bag-of-Words Foundation

Our TRECVID 2011 concept and event detection approach builds on previous editions of the MediaMill semantic video search engine [18–20], which draws inspiration from the bag-of-words approach propagated by Schmid and her associates [7, 12, 31], as well as recent advances in keypoint-based color features [25], codebook representations [27, 29], and efficient algorithmic refinements of the bag-of-words approach [11, 24], a GPU implementation [26], and compute clusters. In our description of the bag-of-words scheme, we follow the video data as it flows through the computational process, as summarized in Figure 1, and detailed per component next.

2.1 Spatio-Temporal Sampling

The visual appearance of a semantic concept in video has a strong dependency on the spatio-temporal viewpoint un-
der which it is recorded. Salient point methods [23] introduce robustness against viewpoint changes by selecting points, which can be recovered under different perspectives. Another solution is to simply use many points, which is achieved by dense sampling. Appearance variations caused by temporal effects are addressed by analyzing video beyond the key frame level. By taking more frames into account during analysis, it becomes possible to recognize concepts that are visible during the shot, but not necessarily in a single key frame.

Temporal multi-frame selection In [19, 20, 22] we demonstrated that a concept detection method that considers more video content obtains higher performance over key frame-based methods. We attribute this to the fact that the content of a shot changes due to object motion, camera motion, and imperfect shot segmentation results. Therefore, we employ a multi-frame sampling strategy. To be precise, we sample up to 6 additional i-frames distributed around the (middle) key frame of each shot.

Harris-Laplace point detector In order to determine salient points, Harris-Laplace relies on a Harris corner detector. By applying it on multiple scales, it is possible to select the characteristic scale of a local corner using the Laplacian operator [23]. Hence, for each corner, the Harris-Laplace detector selects a scale-invariant point if the local image structure under a Laplacian operator has a stable maximum.

Dense point detector For concepts with many homogeneous areas, like scenes, corners are often rare. Hence, for these concepts relying on a Harris-Laplace detector can be suboptimal. To counter the shortcoming of Harris-Laplace, random and dense sampling strategies have been proposed [5, 6]. We employ dense sampling, which samples an image grid in a uniform fashion using a fixed pixel interval between regions. In our experiments we use an interval distance of 6 pixels and sample at multiple scales.

Spatial pyramid weighting Both Harris-Laplace and dense sampling give an equal weight to all keypoints, irrespective of their spatial location in the image frame. In order to overcome this limitation, Lazebnik et al. [7] suggest to repeatedly sample fixed subregions of an image, *e.g.*, 1x1, 2x2, 4x4, *etc.*, and to aggregate the different resolutions
2.2 Visual Descriptors

In the previous section, we addressed the dependency of the visual appearance of semantic concepts in a video on the spatio-temporal viewpoint under which they are recorded. However, the lighting conditions during filming also play an important role. Burghouts and Geusebroek [2] analyzed the properties of color features under classes of illumination and viewing changes, such as viewpoint changes, light intensity changes, light direction changes, and light color changes. Van de Sande et al. [25] analyzed the properties of color features under classes of illumination changes within the diagonal model of illumination change, and specifically for data sets as considered within TRECVID.

SIFT The SIFT feature proposed by Lowe [10] describes the local shape of a region using edge orientation histograms. The gradient of an image is shift-invariant: taking the derivative cancels out offsets [25]. Under light intensity changes, i.e., a scaling of the intensity channel, the gradient direction and the relative gradient magnitude remain the same. Because the SIFT feature is normalized, the gradient magnitude changes have no effect on the final feature. To compute SIFT features, we use the version described by Lowe [10].

OpponentSIFT OpponentSIFT describes all the channels in the opponent color space using SIFT features. The information in the $O_3$ channel is equal to the intensity information, while the other channels describe the color information in the image. The feature normalization, as effective in SIFT, cancels out any local changes in light intensity.

RGB-SIFT For the RGB-SIFT, the SIFT feature is computed for each RGB channel independently. Due to the normalizations performed within SIFT, it is equal to transformed color SIFT [25]. The feature is scale-invariant, shift-invariant, and invariant to light color changes and shift.

We compute the SIFT [10] and ColorSIFT [25] features around salient points obtained from the Harris-Laplace detector and dense sampling. For all visual features we employ a spatial pyramid of 1x1 and 1x3 regions.

2.3 Word Projection

To avoid using all visual features in an image, while incorporating translation invariance and a robustness to noise, we follow the well known codebook approach, see e.g., [6, 8, 16, 27, 29]. First, we assign visual features to discrete codewords predefined in a codebook. Then, we use the frequency distribution of the codewords as a compact feature vector representing an image frame. By using a vectorized GPU implementation [26], our codebook transform process is an order of magnitude faster for the most expensive feature compared to the standard implementation. Two important variables in the codebook representation are codebook construction and codeword assignment. Based on previous experiments, balancing accuracy and performance, we employ codebook construction using $k$-means clustering in combination with hard codeword assignment and a maximum of 4,096 codewords.

In is well known that the traditional hard-assignment may be improved by using soft-assignment through kernel codebooks [29]. A kernel codebook uses a kernel function to smooth the hard-assignment of image features to codewords by assign descriptors to multiple clusters, weighted by their distance to the center. Recently, many improved soft assignment approaches have been proposed [13, 32]. In [13] Peronnin et al. train a Gaussian Mixture Model, where each codebook element has its own sigma one per dimension. They do not store the assignment, but the differences in all descriptor dimensions. Super Vector Coding by Zhou et al. [32] also counts the dimension-wise difference of a descriptor to a visual word. While these methods propose many new components and algorithms, we consider the difference coding their main contribution. We employ difference coding also.

Kernel library Each of the possible sampling methods from Section 2.1 coupled with each visual feature extraction method from Section 2.2, a clustering method, and an assignment approach results in a separate visual codebook. An example is a codebook based on dense sampling of RGB-SIFT features in combination with $k$-means clustering and hard assignment. We collect all possible codebook combinations in a (visual) kernel library. By using a GPU implementation [26], this kernel library can be computed efficiently. Naturally, the codebooks can be combined using various configurations. Depending on the kernel-based learning scheme used, we simply employ equal weights in our experiments or learn the optimal weight using cross-validation.

3 Detecting Concepts in Video

We perceive concept detection in video as a combined computer vision and machine learning problem. Given an n-dimensional visual feature vector $x_i$, part of a shot $i$ [14], the aim is to obtain a measure, which indicates whether semantic concept $\omega_j$ is present in shot $i$. We may choose from various audiovisual feature extraction methods to obtain $x_i$, and from a variety of supervised machine learning approaches to learn the relation between $\omega_j$ and $x_i$. The supervised machine learning process is composed of two phases: training
and testing. In the first phase, the optimal configuration of features is learned from the training data. In the second phase, the classifier assigns a probability $p(\omega_j|x_i)$ to each input feature vector for each semantic concept.

Learning robust concept detectors from visual features is typically achieved by kernel-based learning methods. Similar to previous years, we rely predominantly on the support vector machine framework [30] for supervised learning of semantic concepts. Here we use the LIBSVM implementation [3] with probabilistic output [9, 15]. In order to handle imbalance in the number of positive versus negative training examples, we fix the weights of the positive and negative class by estimation from the class priors on training data. While the $\chi^2$ kernel function usually performs better than other kernels [31], it is computationally demanding when classifying multiple frames per shot. Therefore, we use the Histogram Intersection kernel and its efficient approximation as suggested by Maji et al. [11]. For difference coded bag-of-words we use a linear kernel [13, 32].

In general, we obtain good parameter settings for a support vector machine, by using an iterative search on both $C$ and kernel function $K(\cdot)$ on cross validation data [28]. From all parameters $q$ we select the combination that yields the best average precision performance, yielding $q^*$. We measure performance of all parameter combinations and select the combination that yields the best performance. We use a 3-fold cross validation to prevent over-fitting of parameters. Rather than using regular cross-validation for support vector machine parameter optimization, we employ an episode-constrained cross-validation method, as this method is known to yield a less biased estimate of concept detection performance [28].

The result of the parameter search over $q$ is the improved model $p(\omega_j|x_i, q^*)$, contracted to $p^*(\omega_j|x_i)$, which we use to fuse and to rank concept detection results.

### 3.1 Submitted Concept Detection Results

**Run: Michelangelo** The Michelangelo run is our baseline. It is based on multiple (visual) kernel libraries using both hard-assignment and difference coding on SIFT, OpponentSIFT, and RGB-SIFT descriptors, which have been applied on a single keyframe per shot. Fusion is performed using an AVG rule combination. This run achieved a mean infAP of 0.150.

**Run: Donatello** The Donatello run is a multi-frame version of the baseline. Here we have classified up to 6 additional i-frames per shot in combination with a MAX rule, before averaging the hard-assigned version with the difference coding version. This run achieved a mean infAP of 0.168, with the overall highest infAP for 4 concepts: charts, female human face closeup, mountain, and scene text.

**Run: Raphael** The Raphael run is similar in spirit to our best performing run of last year. It is based on multiple
(visual) kernel libraries using hard-assigned SIFT, OpponentSIFT, and RGB-SIFT descriptors, which have been applied spatio-temporally with up to 10 additional i-frames per shot in combination with a MAX rule combination. This run achieved a mean infAP of 0.170, with the overall highest infAP for 4 concepts: beach, car, demonstration, and flowers.

Run: Leonardo The Leonardo run is similar to the Donatello run, with the only exception that 10 additional i-frames per shot are classified. This run achieved a mean infAP of 0.172, with the overall highest infAP for 7 concepts: car, demonstration, flowers, hand, flags, speaking to camera, and table.

3.2 1,346 Concept Detectors

In addition to the 346 concept detectors from the TRECVID SIN task, we have also employed our Raphael run setting on the entire concept set of the ImageNet Large Scale Visual Recognition Challenge 2011 [4], containing 1,000 object categories. All 1,346 detectors are included in the 2011 MediaMill semantic video search engine.

4 Detecting Events in Video

We participated in the multimedia event detection task using a visual-only approach founded on the same bag-of-words used for concept detection. We will detail the approach and the results in the final notebook paper [1].

5 Conclusion

TRECVID continues to be a rewarding experience in gaining insight in the difficult problem of semantic video retrieval. The 2011 edition has again been a successful participation for the MediaMill team resulting in runner-up ranking for concept detection and a first exploration of the challenging problem of event detection.

Acknowledgments

The authors are grateful to NIST and the TRECVID coordinators for the benchmark organization effort. This research is sponsored by the STW SEARCHER project, the Beeld-Canon project, FES Commit, and the IARPA ALADDIN via Department of Interior National Business Center contract number D11PC20067. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoI/NBC, or the U.S. Government.

References


