Episode-Constrained Cross-Validation in Video Concept Retrieval

Jan C. van Gemert*, Cor J. Veenman, and Jan-Mark Geusebroek *Member, IEEE*

***Abstract*—Whereas video tells a narrative by a composition of shots, current video retrieval methods focus mainly on single shots. In retrieval performance estimation, similar shots in a narrative may result in performance over-estimation. We propose an episode-based version of cross-validation leading up to 14% classification improvement over shot based cross-validation.**

I. Introduction

MACHINE learning techniques have proven to be a valuable addition to the repertoire of a multimedia researcher. Applications of machine learning techniques in multimedia are found in semantic video labeling [1], video shot detection [2], audio classification [3], scene recognition [4], sports analysis [5], and in many other areas. Moreover, multimedia researchers have contributed to specifically designed classifiers for multimedia analysis [6], [7].

Several machine learning techniques rely on accurate performance estimation [8]. The estimated performance may be used in finding the best parameters of a classification model and helps when deciding between different features. Thus, accurate performance estimation influences the quality of the machine learning method.

The central issue addressed in this paper is the following: *How is classification performance estimation affected by the narrative structure in multimedia data?* Much multimedia data is narrative in nature. For example, popular music has a verse and a chorus, multimedia presentations have slides designed with a message in mind, and shots in video data may be part of a storyline. Such narratives typically build a story by repeating similar elements. In separating narrative data in a test and training set, these highly similar elements may easily end up in both the test and the training set. Hence, commonly used classifier performance estimation techniques need special care when applied to multimedia classification.

In this paper we exploit the narrative structure present in multimedia data to achieve accurate classification performance estimation. We show that more accurate performance estimation increases the final classification performance. Furthermore, we investigate how unbiased performance indicators can be constructed, resulting in unbiased and accurate estimation of classification performance in a narrative. As an instantiation of narrative multimedia data we will focus on semantic concept detectors in video. However, the described techniques readily apply to other types of data that share a narrative structure.

The idea of exploiting the narrative structure in video is not novel [1], [9], [10], [11], [12], though using narrative units for unbiased classification performance estimation is novel to the best of our knowledge. Our earlier work [13] also noted the influence of narrative structure on classification performance estimation. This current paper, however, provides a more in-depth analysis of this earlier work while also presenting a new unbiased performance indicator for narrative data.

The organization of this paper is as follows. The next section revisits standard classifier evaluation techniques. Then, section III introduces an evaluation technique that respects narrative structure in video concept retrieval. This narrative structure introduces unbalanced data, which is discussed in section IV. Section V presents the experimental setup followed by the results in section VI and the conclusions in section VII.

II. Classifier Performance Evaluation

Correct classification error estimation not only provides a quantitative assessment of the classifier, it also influences classifier performance. Classifier performance depends on the quality of the classifier model, which in its turn relies on the input features and classifier parameters. These classifier parameters and features are typically tuned by maximizing the estimated performance over various input features and parameter settings. For example in a semantic video concept retrieval task, Snoek et al. [1] use the estimated classifier performance to select the best low level features. Furthermore, they find the best parameters for a Support Vector Machine (SVM) by maximizing the estimated classifier performance. In their framework, inaccurate classifier performance estimation might result in choosing the wrong features, or in sub-optimal parameter settings. Hence, classifier performance estimation affects the selected classifier model, and thus the quality of the tuned classifier.

Estimating classification performance is typically done by training a classifier on one set, and testing the classifier on an independent hold-out set. Thus, a straightforward approach to classifier performance estimation is keeping a random sample of the available data in an unseen hold-out set. This hold-out set should be as large as possible, to accurately represent the class variation that may be expected. However, keeping a large part of the data from the training set gives the classifier less data to train on. Hence, a balance between the size of the training set and the size of the hold-out set must be struck.

In contrast to a single hold-out set, the cross-validation method rotates the hold-out set over all available data. Cross-validation randomly splits the available data in X folds,

* Corresponding Author.

All authors are with the Intelligent Systems Lab Amsterdam, Informatics Institute, University of Amsterdam, 1098 SJ Amsterdam, The Netherlands (e-mail: jvgemert@uva.nl; c.j.veenman@uva.nl; mark@science.uva.nl).

C.J. Veenman is also affiliated with the Netherlands Forensic Institute.

EDICS: 4-KEEP, Indexing, Searching, Retrieving, Query, and Archiving Databases.

Manuscript received December 21, 2007.

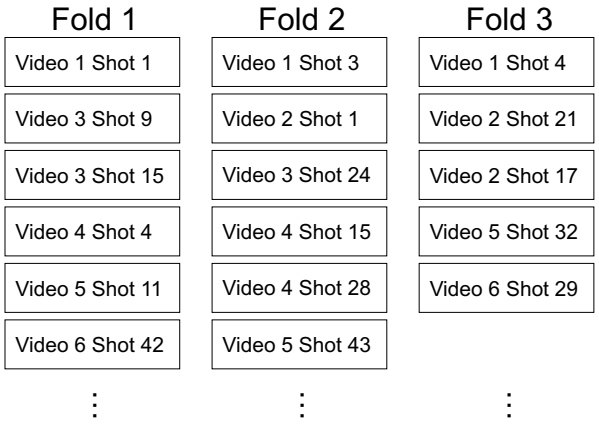

Fig. 1. An example of partitioning a video set by using shot based 3-fold cross-validation.

where each of these X folds is once used as a hold-out set. The performance estimates on all rotating hold-out folds are averaged, yielding an estimate of the classifier performance. The cross-validation procedure may be repeated R times, to minimize the effect of the random partitioning. An example of cross-validation for a set of shots in a video is shown in figure 1. The advantage of using cross-validation is the combination of a large training set with several hold-out sets. Therefore, cross-validation is the standard procedure for classification performance estimation [8].

III. Cross-Validation in Video Classification

Machine learning is heavily used in semantic video indexing [7], [1]. The aim of semantic video indexing is retrieving all relevant shots in a dataset to a given semantic concept. Some examples of semantic concepts are *Airplane, Car, Computer Screen, Bill Clinton, Military Vehicle, Sports*. Machine learning techniques, and specifically classifiers, are commonly used to rank a list of shots according to their probability of being relevant to a semantic concept. These machine-indexed semantic concepts provide a user with automated tools to browse, explore, and find relevant shots in a large collection of video. With growing digital video collections, there is a need for automatic concept detection systems, providing instant access to digital collections. Therefore, machine learning techniques are vital to automatic video indexing.

For semantic video concept indexing, a video is typically represented as a set of single shots [14], [1]. However, a video document is the end result of an authoring process [1], where shots are used to convey a message. For example, a topic in news video, may consist of several similar shots, as shown in figure 2. This temporal co-occurrence of similar shots in a topic may be exploited for video indexing [10], [11], [12]. Nevertheless, the video indexing task is oriented towards single shots, whereas a semantic concept might span several shots.

The granularity difference between the indexing task that focuses on single shots, and semantic concepts that may span several shots requires special care in estimating retrieval performance. Consider figure 2, and note the high similarity

Fig. 2. An example of narrative structure in video: four consecutive shots showing an interview with the former Lebanese President Mr. Lahoud.

between shot 250 and shot 252. The similarity between these two shots can be expected, since they are part of the same narrative structure. However, the retrieval task focuses on single shots, and does not take this semantic relation between shots into account. Therefore, the common practice [14], [1] of estimating retrieval performance by cross-validation on shots is biased. Cross-validation on shots will mix shots in a single topic to different folds while randomly partitioning the data. Thus, shots that belong to the same semantic concept will be present both in the training set and in the rotating hold-out set. This leaking of near-identical information creates a dependency between the training set and the hold-out set, which will manifest in too optimistic estimates for retrieval performance. Moreover, if cross-validation is used for classifier parameter tuning, the parameters will be biased towards near-duplicate data and might consequently fail to find the best parameters for true independent hold-out data. Therefore, the narrative structure of video data should be taken into account when estimating retrieval performance.

In order to preserve the narrative relation between shots in a semantic concept, we propose an episode-constrained version of cross-validation. In contrast to a shot based partitioning of the video data, an episode-constrained partitioning aims to keep shots together if they are part of the same episode. In the context of a semantic concept retrieval task, an episode ideally consists of all constituent shots of the concept at hand. However, video story segmentation is an unsolved problem [9], [15]. Therefore, we resolve to using whole videos as atomic episodes. With videos as atomic elements, all shots in a video are kept together, preventing the leaking of near-identical information to the hold-out set. Whereas the traditional method randomly distributes shots, our method randomly distributes videos. An example of episode-constrained cross-validation

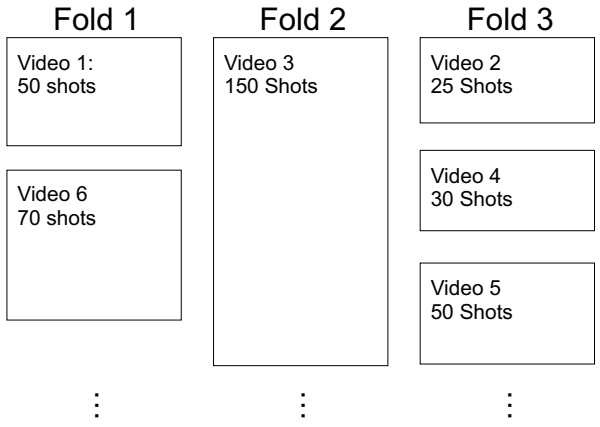

Fig. 3. An example of a partitioning a video set by using episode-constrained 3-fold cross-validation.

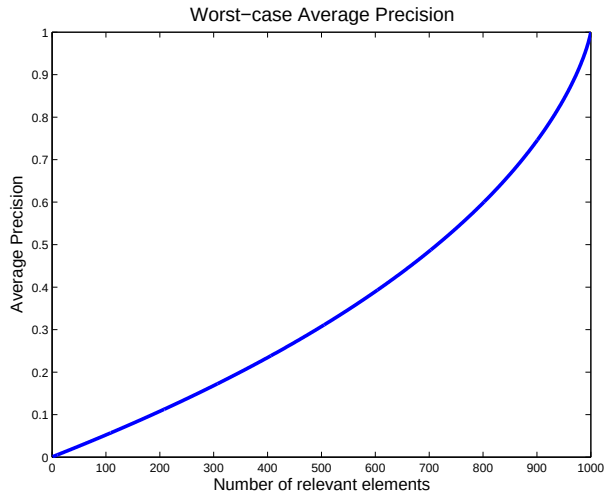

Fig. 4. Average precision for a worst-case retrieved list of 1000 elements, where all relevant items are found at the bottom of the list. Note that the worst-case average precision score increases with the number of relevant items.

for a video set is shown in figure 3. The episode-constrained version of cross-validation creates truly independent hold-out data, and will yield more accurate performance estimates of video concept classification.

IV. Performance Estimation Between Unbalanced Sets

In semantic video retrieval, the performance measure of choice is average precision [14], [16], [1]. For a ranked list of elements, average precision denotes the area under the precision recall graph. Let $L_k = \{s_1, s_2, \ldots, s_k\}$ be the top k ranked elements from the retrieved results set L, and let R denote the set of all relevant items, then average precision (AP) is defined as

$$\mathrm{AP}(L) \quad = \quad \frac{1}{|R|} \sum_{k=1}^{|L|} \frac{|L_k \cap R|}{k} I_R(s_k) \quad \text{, for } |R| > 0, \text{ (1)}$$

where $|\cdot|$ denotes set cardinality and the indicator function $I_R(s_k) = 1$ if $s_k \in R$ and 0 otherwise. Average precision places a high emphasis on the top of the retrieved results list. The bottom of the results list is weighted less heavy and retrieval system benchmarks often truncate after a couple of thousand results. This practical approach to truncation and the high emphasis on the top retrieval results may explain the popularity of average precision in the video retrieval community.

Average precision describes the shape of the retrieved results list. However, average precision does not take the a-priori probability of relevant elements into account. Hence, average precision is not normalized for the number of relevant elements, and will give high scores when there are many relevant elements. Consider a worst-case retrieval system, that consistently places all relevant elements R at the bottom of the retrieved result list L. When the cardinality of L is fixed, $|L| = c$, the worst-case average precision (WAP) depends only on the number of relevant elements $|R|$, reducing equation 1 to

$$\mathrm{WAP}(|R|) \quad = \quad \frac{1}{|R|} \sum_{k=1}^{|R|} \frac{k}{(|L| - |R|) + k}, \text{for } |R| > 0. \text{ (2)}$$

Figure 4 illustrates the worst-case average precision for an increasing number of relevant elements. Note that a growing number of relevant elements results in an increasing a-priori average precision score. Thus, average precision scores are hard to compare between sets with a varying number of relevant elements because the average precision score is biased towards high-frequency relevant elements.

Given average precision as the performance measure for semantic video retrieval, it stands to reason to adopt average precision as the performance measure in cross-validation. In episode-constrained cross-validation, however, shots are kept together to prevent leaking of similar shots to a rotating test set. These atomic sets of shots hamper an equal distribution of the relevant shots over the cross-validation folds. For example, one news episode may contain several shots of a popular sports event, whereas other episodes may contain none. Hence, episode-constrained cross-validation yields an unbalanced distribution of relevant elements over the folds. Since the estimated performances on the folds are averaged to give a final cross-validation performance estimate, the folds that are randomly endowed with a high number of relevant-item episodes will dominate the cross-validation performance estimation. The effects of this will manifest itself in the classifier model selection that fit best to the fold that has the most relevant elements. Thus, in general, and for episode-constrained cross-validation in particular, an alternative to average precision is required that normalizes for unbalanced folds.

A performance measure for cross-validation should optimize average precision and allow equal weights when averaging cross-validation folds. Hence, this performance measure should scale between a fixed minimum and maximum, say 0 and 1, where 0 should represent the case where all relevant elements are retrieved at the bottom of the list, and 1 should indicate that all relevant elements are found at the top of the list. This normalization between 0 and 1 remedies the bias of average precision towards a high number of relevant elements. Besides normalization, the performance measure should guarantee that it optimizes the original average precision score. Any alternative to average precision as a performance measure should follow these criteria.

Several alternatives to average precision may be found in the literature. In classifier evaluation it is common to use receiver operating characteristic (ROC) curves for representing classification performance [8]. The ROC-curve shows the variation between the ratio of correctly classified positive elements and the incorrectly classified negative elements. As an alternative to average precision, the area under the ROC curve (AUC) may be maximized [6]. Maximization of the AUC optimizes the pairwise probability of retrieving a relevant element over a non-relevant element [17]. The AUC has the required property that an AUC value of 1 indicates perfect retrieval, and 0 denotes worst-case retrieval. However, optimizing the AUC

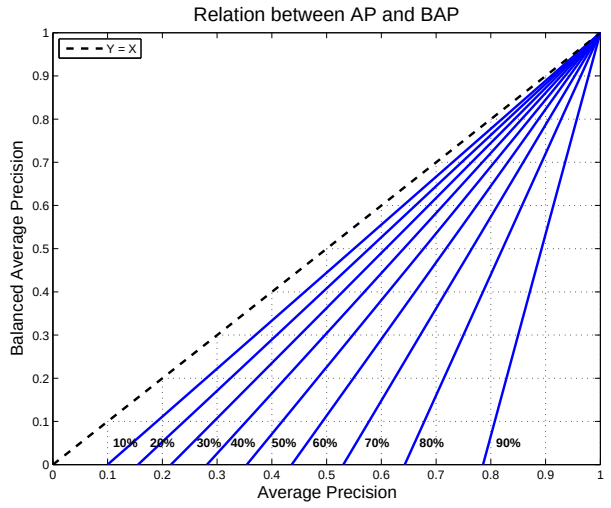

Fig. 5. The relation between average precision and balanced average precision. The solid blue lines represent different ratios of the number of positive elements compared to the number of negative elements in a retrieved list. In this example, the ratios range from 10% to 90% positive elements. The dashed line $y = x$ is included as an AP self-reference.

does not guarantee to optimize average precision [18]. Other performance measures like R-precision [19], normalized average rank [20], normalized average precision [16], inferred average precision [21] or interpolated precision [22] may optimize average precision, however they do not scale between a fixed minimum and maximum. To the best of our knowledge, no performance measure exists that satisfies our demands. Hence, for a retrieved results set L, we propose an unbiased version of average precision which we name balanced average precision (BAP),

$$\mathrm{BAP}(L) \quad = \quad \frac{\mathrm{AP}(L) - \mathrm{WAP}(L)}{1 - \mathrm{WAP}(L)} \quad \text{, for WAP} < 1, \quad (3)$$

where AP and WAP refer to average precision and worst-case average precision in equations 1 and 2 respectively. The balanced average precision merely rescales the average precision where the worst possible result is set at 0, and the best possible results set at 1. Since balanced average precision is a monotone rescaling of average precision, optimizing this measure also optimizes the original average precision. Hence, balanced average precision allows a normalized comparison between sets with an unbalanced number of relevant elements, while maintaining all properties of average precision.

In figure 5 we show the relation between average precision (AP) and balanced average precision (BAP). The figure illustrates the BAP score corresponding to a given AP value for various ratios between positive and negative elements in a list. For a fixed AP score, an increasing positive ratio yields a substantially smaller BAP score (vertical lines). Note that a larger difference between positive ratios yields a larger difference between BAP scores. For example, at an AP value of 0.8, the difference in BAP scores between the ratios of 80% and 90% is 0.35, whereas the difference between the ratios 80% and 90% is 0.13. For cross-validation, therefore, BAP will have more impact for folds with large differences between their positive elements ratios. What is more, the inequality between varying positive ratios increases, as the BAP score decreases (horizontal lines). For example, the difference between the ratio lines of 80% and 90% for a BAP value of 0.6 is 0.05, whereas this difference is 0.12 for a BAP score of 0.1. Hence, the effect of BAP becomes more pronounced for low classifier performance, *i.e.*, with hard problems. We deem multimedia indexing a hard problem. Moreover, episode-constrained cross-validation increased the inequality between folds. Hence, we argue for using BAP for parameter estimation in multimedia classification.

V. Experimental Setup

We compare the episode-constrained version of cross-validation with the shot based version of cross-validation on a large corpus of news video: the Challenge Problem [23]. The Challenge Problem provides a benchmark framework for video indexing. The framework consists of visual features, text features, classifier models, a ground truth, and classification results for 101 semantic concepts[1] on 85 hours of international broadcast news data, from the TRECVID 2005/2006 benchmark [14]. The advantage of using the challenge framework is that the framework provides a standard set of features to the TRECVID data. Furthermore, the framework is well suited for our experiment, since there are a large number or shots, *i.e.* close to $45,000$, and an abundance of semantic concepts.

The Challenge data comes with a training set consisting of the first 70% of the video data, and a hold-out set containing the last 30% of the data. We use the training set for training both a k-nearest neighbor classifier (kNN) and a support vector machine classifier with an rbf-kernel [8]. We opted for the k-nearest neighbor classifier because of its simplicity, its generally decent performance, and the fact that it has a single tunable parameter. We included the SVM because it is a popular classifier which performs well on this data [23]. The features we use are the visual features [24] that are provided with the Challenge framework.

VI. Results

The focus of the experiment is on comparing episode-constrained cross-validation versus shot based cross-validation. To this end, we use both cross-validation methods to randomly partition the data in 10 folds. These 10 folds are subsequently used to estimate the best value for k for a kNN classifier, where $k \in \{1, 2, 3, 4, 5\}$. For the SVM classifier we preset the slack parameter C per class to the inverse of the class frequency and logarithmically tune the rbf-kernel size γ, where $\gamma \in \{1, 3.16, 10, 31.6, 100\}$. To evaluate the results, we computed the classification scores for all k and γ parameters on the hold-out set. The estimates and true hold-out average precision scores for the of the SVM and kNN classifier are displayed in figures 6 and 7 respectively.

[1]We did not evaluate the concept *baseball*, since all the examples in the training set of this concept are found in a single video.

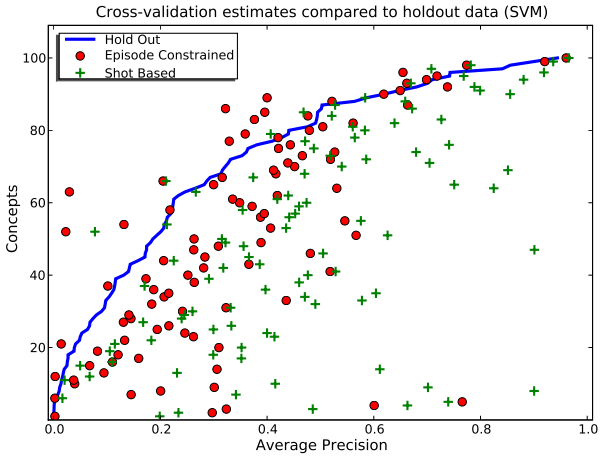

Fig. 6. Performance estimates of episode-constrained and shot based cross-validation compared to the true hold-out performance for the SVM classifier.

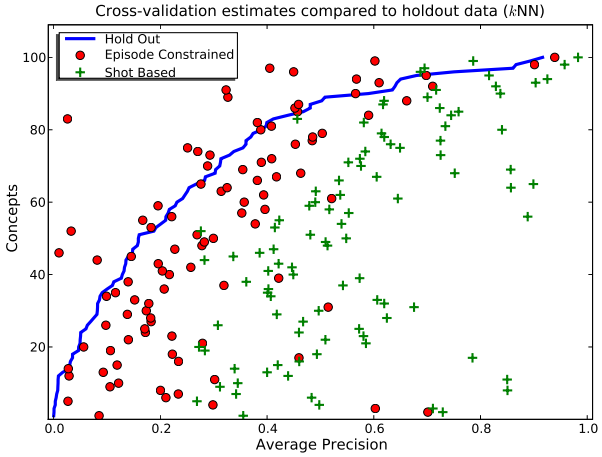

Fig. 7. Performance estimates of episode-constrained and shot based cross-validation compared to the true hold-out performance for the kNN classifier.

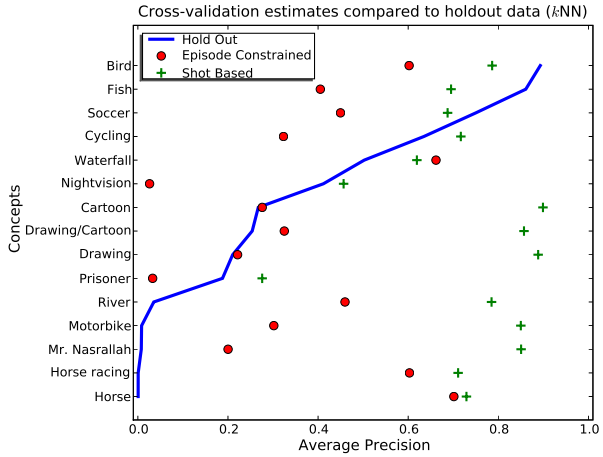

Fig. 8. Performance estimates of episode-constrained and shot based cross-validation compared to the true hold-out performance for some selected concepts with the kNN classifier.

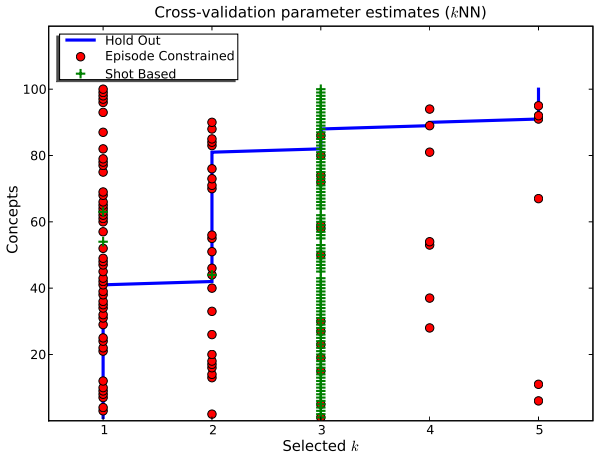

Fig. 9. Selected parameters of episode-constrained and shot based cross-validation compared to the true hold-out performance for the kNN classifier.

A. Evaluating Episode Constrained Cross-Validation

The results in figures 6 and 7 clearly show the over-estimation of the average precision scores by the shot based cross-validation method. This over-estimation is more evident for the kNN classifier than for the SVM classifier. For the SVM classifier the episode-constrained estimation is closer to the true hold-out performance for 81 concepts, and in case of the kNN classifier this holds for 93 concepts. We show a more detailed figure for the kNN classifier in figure 8. In this figure we show concepts with a large difference between their scores on hold-out or between the scores of the two cross-validation methods. Furthermore, we show the 7 concepts where shot based cross-validation gives a closer estimate to the hold-out performance than episode-constrained cross-validation. These 7 concepts either have very few examples or consist of shots that have near-copies in the hold-out set. Concepts with few examples (*i.e. Prisoner*) cannot be distributed completely over the 10 folds when videos are kept together. These concepts yield zero-scores for some folds, which in turn leads to a low fold average. The near copies in the hold-out set are due to commercials (*Bird, Fish, Cycling, Waterfall*) or due to little appearance variation (*Soccer, Nightvision*). Other concepts score significantly lower on the hold-out set because they have too little appearance overlap between the examples in the train set and the hold-out set (River, Motorbike, Mr. Nasrallah, Horse racing, Horse). The remaining concepts (*Cartoon, Drawing/Cartoon, Drawing*) are made up of highly repetitive shots within a video and therefore benefit most from episode-constrained cross-validation as can be seen by its accurate performance estimation compared to shot based cross-validation.

In figure 9 we show the estimated classifier parameters and the best parameters on the hold-out set. For space considerations we only show the kNN classifier, since it gives the best results. The first thing that is striking about the estimated parameters in figure 9 is the discrepancy between methods in selecting the best classifier parameter. The shot based cross-validation method for kNN selects $k = 3$ for 97 out of 100 concepts, whereas episode-constrained cross-validation correlates better with the best parameter of the hold-out set. The parameter estimates influence the final classification performance, and we summarize this in table I. In this table

	Shot Based		Episode-Constrained	
	kNN	SVM	kNN	SVM
Training set	0.573	0.474	0.310	0.345
Hold-out set	0.187	0.201	0.213	0.210

TABLE I
THE MEAN PERFORMANCE IN AP OVER ALL CONCEPTS USING THE ESTIMATED PARAMETERS AS SELECTED BY EACH METHOD.

we present the mean performance in average precision over all concepts, for both cross-validation methods and for both classifiers. We show the estimated results on training data, and the results on hold-out data where we tune the classifier parameter by selecting the maximum performance according to the cross-validation method at hand.

In analyzing table I, we focus on two points: 1) the accuracy in estimating classifier performance and 2) the final classification performance. Starting with point 1, we consider the difference between the estimated performance on training data and the reported performance on hold-out data. For shot based cross-validation there is considerable difference between the estimated performance on training data and the performance on hold-out data. Specifically, the difference is 0.386 for the kNN classifier, and 0.273 for the SVM classifier. In contrast, for episode-constrained cross-validation the difference between training data and hold-out data is only 0.097 for the kNN, and 0.135 for SVM. This clearly shows that the estimated performance of the episode-constrained cross-validation is more accurate than the performance estimate based on shots. Continuing with the issue of final classification performance, we compare the performance on hold-out data for both methods. An analysis of the hold-out results per concept shows that episode-constrained cross-validation yields equal or better results for 85 concept with kNN and for 79 concepts for SVM. Averaged over all concepts, the episode-constrained method outperforms the shot based method by 14% for kNN, and 5% for SVM, as shown in table I. The smaller improvement in the case of the SVM is due to a large performance increase when near-duplicates are present in the hold-out set. Since near-duplicates are very similar, the SVM with its parameters tuned by shot based cross-validation is very well tuned to these duplicates. The large performance increase for near-duplicates leads to a disproportional increase in the average value over all concepts. The near-duplicates mostly consist of commercials: *Bird* (+0.09), *Waterfall* (+0.10), *NightVision* (+0.19), *SwimmingPool* (+0.09), *Beach* (+0.06). Nevertheless, for the SVM the performance for 79 out of 100 concepts improves by using episode-constrained cross-validation. Therefore these results show that performance estimation with episode-constrained cross-validation is considerably more accurate than using shot based cross-validation, and that this improvement in performance estimation directly translates to an improvement in final classification performance.

B. The Influence of Balanced Average Precision

Here, we evaluate the assumptions and motivation of using balanced average precision. Balanced average precision allows a fair comparison between collections with an unbalanced number of relevant elements. We assumed that unbalanced collections are more likely to occur with episode constrained cross-validation, since atomic sets of shots hamper an equal distribution of relevant elements over the cross-validation folds. In order to test this hypothesis, we compare the spread of the relevant elements over the folds for both episode-constrained cross-validation and the traditional shot based cross-validation. Specifically, we compute the standard deviation of the number of relevant elements per fold, averaged over all concepts. Both methods of cross-validation have on average 135.21 relevant elements per fold, where the average standard deviation of the shot based and episode-constrained cross-validation method is 0.40 and 30.51 respectively. The difference between both standard deviations clearly shows that episode-constrained cross-validation creates significantly more unbalanced folds than shot based cross-validation. Hence, the motivation of using balanced average precision with episode-constrained cross-validation is sound.

The unbalanced folds in episode-constrained cross-validation necessitate the use of balanced average precision. However, the difference between balanced average precision (BAP) and traditional average precision (AP) may not necessarily prove significant. We evaluate this significance on the Challenge Problem. We employ episode-constrained cross-

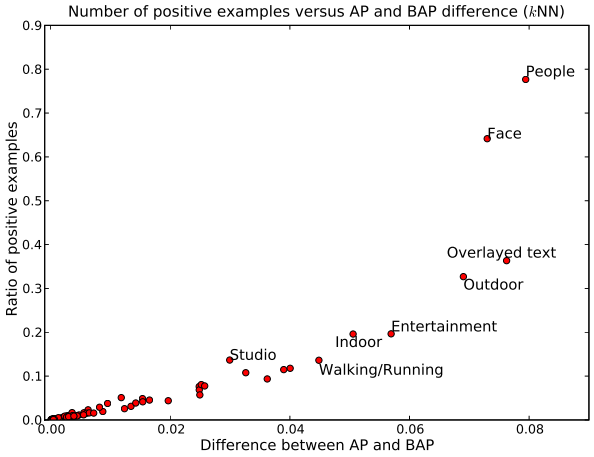

Fig. 10. The difference per concept between estimated average precision (AP) and balanced average precision (BAP) on training data.

Fold	AP	BAP	% Relevant shots
1	0.919	0.863	63
2	0.917	0.864	60
3	0.900	0.824	65
4	0.911	0.841	66
5	0.867	0.779	61
6	0.906	0.830	66
7	0.873	0.785	62
8	0.892	0.820	61
9	0.896	0.810	67
10	0.930	0.866	70

TABLE II
AVERAGE PRECISION (AP) BALANCED AVERAGE PRECISION (BAP) SCORES AND THE PERCENTAGE OF RELEVANT SHOTS IN EACH FOLD FOR THE CONCEPT *Face*.

validation for classifier parameter selection and compare the scores of average precision versus balanced average precision. The results on the Challenge Problem show no difference in parameter selection for both the kNN as the SVM classifier. Hence, for this dataset there is no difference between average precision and balanced average precision. In figure 10 we show the difference between AP and BAP compared to the ratio of positive examples for a concept. As illustrated in figure 10, there are 89 out of 100 concepts with less than 10% positive examples. Such concepts with relatively few positive examples are less affected by unbalanced data. As we have shown in figure 5, the benefit of BAP comes into its own with larger number of positive examples. As an example, the scores on the cross-validation folds for the concept *Face* are given in table II. This table shows that the over-estimation bias in average precision does occur, however not often enough. For example, when comparing the scores for fold 1 and fold 2, the AP in fold 1 is higher than the AP in fold 2, whereas the BAP for fold 1 is lower than the BAP for fold 2. The same holds for fold 8 and 9. Therefore, despite that there is no difference between AP and BAP for parameter selection on this dataset, the unbalanced data does have a biased effect on average precision. Thus, when using episode-constrained cross-validation balanced average precision is preferred over average precision.

VII. Conclusions

In this paper, we compare two methods of cross-validation for estimating classification performance for semantic concept detection in video. The traditional method of cross-validation is based on shots, whereas we propose a method based on episodes. An episode-constrained method for cross-validation prevents the leaking of similar shots to the rotating hold-out set. We use a whole video as an episode. However, video story segmentation [9], [15] seems a likely alternative to obtain natural episodes. Since episode-constrained cross-validation tends to produce sets with an unbalanced number of relevant items, we introduce balanced average precision. Balanced average precision is an unbiased alternative to average precision. In contrast to average precision, balanced average precision normalizes for the number of relevant items and is therefore a theoretically better choice when dealing with sets that contain an unbalanced number of relevant elements. Experimental results show that the bias of average precision for unbalanced data does occur. However, in our dataset, balanced average precision performs equal to average precision because of the low ratio of positive examples in this dataset. Further experimental evaluation show that the episode-constrained method yields a more accurate estimate of the classifier performance than the shot based method. Moreover, when cross-validation is used for parameter optimization, the episode-constrained method is better able to estimate the optimal classifier parameters, resulting in higher performance on validation data compared to the traditional shot based cross-validation.

VIII. Acknowledgments

This research is sponsored by the BSIK MultimediaN project.

References

[1] C. G. M. Snoek, M. Worring, J. M. Geusebroek, D. C. Koelma, F. J. Seinstra, and A. W. M. Smeulders, "The semantic pathfinder: Using an authoring metaphor for generic multimedia indexing," *TPAMI*, vol. 28, no. 10, 2006.

[2] Y. Qi, A. Hauptmann, and T. Liu, "Supervised classification for video shot segmentation," in *ICME*, July 2003.

[3] L. Lu, H. Jiang, and H. Zhang, "A robust audio classification and segmentation method," in *ACM Multimedia*, 2001.

[4] J. C. van Gemert, J. M. Geusebroek, C. J. Veenman, and A. W. M. Smeulders, "Kernel codebooks for scene categorization," in *European Conference on Computer Vision*, October 2008.

[5] L. Y. Duan, M. Xu, X. D. Yu, and Q. Tian, "A unified framework for semantic shot classification in sports video," *Trans. on Multimedia*, vol. 7, no. 6, 2005.

[6] S. Gao and Q. Sun, "Improving semantic concept detection through optimizing ranking function," *Trans. on Multimedia*, vol. 9, no. 7, 2007.

[7] M. R. Naphade and T. S. Huang, "A probabilistic framework for semantic video indexing, filtering, and retrieval." *Trans. on Multimedia*, vol. 3, no. 1, 2001.

[8] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. Wiley-Interscience Publication, 2000.

[9] A. Hanjalic, R. L. Lagendijk, and J. Biemond, "Automated high-level movie segmentation for advanced video-retrieval systems," *Trans. on Circuits and Systems for Video Technology*, vol. 9, no. 4, 1999.

[10] B. Huurnink and M. de Rijke, "Exploiting redundancy in cross-channel video retrieval," in *ACM Multimedia-MIR*, 2007.

[11] J. Yang, M. Y. Chen, and A. G. Hauptmann, "Finding person x: Correlating names with visual appearances," in *CIVR*, 2004.

[12] J. Yang and A. G. Hauptmann, "Exploring temporal consistency for video analysis and retrieval," in *ACM Multimedia-MIR*, 2006.

[13] J. C. van Gemert, C. G. M. Snoek, C. J. Veenman, and A. W. M. Smeulders, "The influence of cross-validation on video classification performance," in *ACM Multimedia*, 2006.

[14] NIST, "TRECVID Video Retrieval Evaluation," 2001–2007. [Online]. Available: www-nlpir.nist.gov/projects/trecvid/

[15] J. Vendrig and M. Worring, "Systematic evaluation of logical story unit segmentation," *Trans. on Multimedia*, vol. 4, no. 4, 2002.

[16] M. Rautiainen, T. Seppänen, and T. Ojala, "On the significance of cluster-temporal browsing for generic video retrieval: a statistical analysis," in *ACM Multimedia*, 2006.

[17] C. Cortes and M. Mohri, "Auc optimization vs. error rate minimization," in *NIPS*, 2004.

[18] J. D. and M. Goadrich, "The relationship between precision-recall and roc curves," in *ICML*, 2006.

[19] J. A. Aslam, E. Yilmaz, and V. Pavlu, "A geometric interpretation of r-precision and its correlation with average precision," in *SIGIR*, 2005.

[20] H. Müller, W. Müller, D. M. Squire, S. Marchand-Maillet, and T. Pun, "Performance evaluation in content-based image retrieval: overview and proposals," *Pattern Recogn. Lett.*, vol. 22, no. 5, 2001.

[21] E. Yilmaz and J. A. Aslam, "Estimating average precision with incomplete and imperfect judgments," in *CIKM*, 2006.

[22] V. Raghavan, P. Bollmann, and G. S. Jung, "A critical investigation of recall and precision as measures of retrieval system performance," *Trans. Inf. Syst.*, vol. 7, no. 3, pp. 205–229, 1989.

[23] C. G. M. Snoek, M. Worring, J. C. van Gemert, J. M. Geusebroek, and A. W. M. Smeulders, "The challenge problem for automated detection of 101 semantic concepts in multimedia," in *ACM Multimedia*, 2006.

[24] J. C. van Gemert, J. M. Geusebroek, C. J. Veenman, C. G. M. Snoek, and A. W. M. Smeulders, "Robust scene categorization by learning image statistics in context," in *CVPR-SLAM*, 2006.