Leveraging from group classification for video concept detection

Usman Niaz and Bernard Merialdo
EURECOM
Campus SophiaTech,
450 Route des Chappes, 06410 Biot FRANCE
{niaz@eurecom.fr, merialdo@eurecom.fr}

Abstract

The performance of a content based retrieval system is limited mainly because of the unavailability of sufficient annotated examples, descriptor noise and the semantic gap that is the representation difference between the high level concept and the low level feature. Finding the optimal parameters of the learner for each concept adds to the difficulty of this task. We argue that grouping certain concepts together can affect the performance of the learning task. We explore the similarity between different semantic concepts and group associated concepts together to learn a few more classifiers improving the performance of video concept detection. It is further investigated if grouping of concepts in that clever way exploiting the similarity is better or if random grouping does the job. We also compare with the RAKEL framework for video concept detection. With experimentation on the TRECVID 2010 dataset and show that the clever group based classifiers outperforms random grouping of concepts and the multi-label RAKEL algorithm. We further analyze the improvements different grouping techniques bring when fused with individual concept learning for video concept detection.

1. Introduction

Understanding the content in videos is a problem of growing interest with the explosive increase in the amount of video content being generated and uploaded over the internet. Though a very hard problem due to the uncountable variations a concept could be presented in a video the results reported from scientific community are reassuring and encourage further research.

Typical image and video datasets are multi-label in nature in that the classes are not mutually exclusive. The visual content is very rich and usually comprises multiple objects or concepts, in a broader sense, in a single image or a video frame. A city skyline picture, for example, contains many objects and any video is typically tagged with more than one semantic concepts.

Based on such commonalities we argue that different concept categories can complement each other for concept detection performance if they are considered together for training. Considering them together means that if they are very similar they should be trained together to augment training resources and simultaneously they should be classified against each other in order to highlight their differences. An example could be to merge the examples of *Car* and *Road* together to train them as a strong *group* classifier and at the same time arrange for at least one of those to fall into another *group* so that they can be distinguished against each other for the cases when the car is in the garage for example or the road is empty or is crowded by busses. The idea may seem somewhat similar to learning visual attributes [2, 1, 4] where e.g. all the examples of *Bus*, *Car*, *Bicycle* etc. are trained together to learn the attribute *Wheel* but we highlight the differences in the literature review in detail.

Learning all possible label combinations is an insurmountable task and practically the label sets that actually exists are very sparse [10, 11]. This sparseness thus helps greatly to reduce the classifiers to be learned. Methods to find label sets to be learned can be divided into data or label dependent and data independent approaches[10]. Data independent approaches randomly select label sets like RAKEL [11] while we propose a data dependent approach that exploits visual correlation between labels or concepts to find good label sets.

Our goal is to learn from multiple labels and minimize the number of multi-label classifiers. We try to find intelligently the sets of labels to be trained together for learning, to minimize the number of multi-label classifiers or *group* classifiers as we call them. We propose to use the visual similarity between concepts to partition the label space into multiple overlapping groups and then learn classifiers for those groups. Thus we achieve multi-label classification through group based classification. The groups learned are

effectively binary classifiers that combine annotations from different concepts and learn a 1-vs-all classifier on the new set of annotations. Individual concept labels are then inferred from the multi-label group predictions. The labels predicted are always the same and are defined at the time of making groups. The number of concepts belonging to one group is not similar.

We explore a quick way of grouping visually close concepts together which outperforms a method of randomly grouping labels and state of the art multi-label classifier *RAKEL* for different group sizes. We surpass significantly the concept detection performance over the baseline with fusing information from as little as 10 group classifiers for a total of 50 concepts on the TRECVID 2010 dataset.

2 Related Work

[2, 1] introduced attributes that describe visual objects. Attributes can be physical for example visual parts like *leg* and *wheel*, or descriptive like *blue colored* and *striped* or a property that some object might have and other do not. In image and video retrieval a concept or more specifically an object is composed of a set of attributes.

Lampert et al. [4] present two methods for attribute based classification and perform learning for disjoint training and test datasets for object detection. Use of attributes in large scale video retrieval or classification is rare, however [14] describes a video concept sparsely from a set of around 6000 weak attributes such as classifiers scores on low level visual features and classemes, image distance to some randomly selected images based on the visual features and some discriminative attributes. Weak attributes for a query concept are found through a semi-supervised graphical model using correlation between the concepts and the attributes labels.

Group based classification is different from attributes in that a concept is defined by a set of multiple attributes while a group contains multiple concepts. Looking at it in another way an attribute also contains many concepts, or more specifically it is present in many concepts, but the number of attributes to be learned is far greater than the expected number of groups to be learned. Furthermore each concept is identified by a unique combination of the groups. We are trying to improve video concept detection performance with using a small number of classifiers to be learned. Definition of attributes is very specific to the type of dataset and the task at hand. The number of object specific attributes increases rapidly with the addition of more diverse content. Also the attributes are usually named in advance but the groups of concepts or labels are not fixed or pre-defined.

Closely related to attributes is hashing which tries to distinguish examples by assigning binary representations to individual images. Spherical hashing [3] divides the feature space into hyperspheres such that each image is assigned a binary code. Rastegari et al. [6] combine attribute learning with generating binary codes and jointly learn all the attributes (binary codes) together for all the training data.

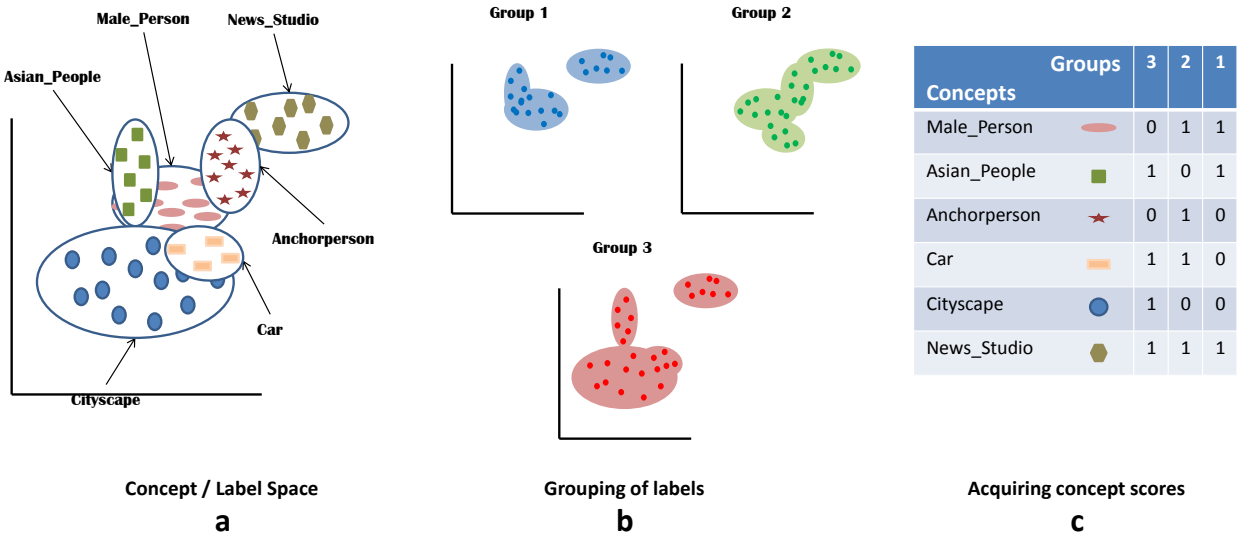
In hashing techniques two examples of the same concept have very similar binary codes [3, 6], or two (visually) close examples have very similar binary codes. Contrarily in our grouping approach codes are assigned at concept level. Thus two examples of the same concept have the exact same binary codes as they belong to the similar set of groups while two concepts that are closely related to each other may have very similar binary code.

RAKEL is a multi-label classification algorithm that works on Label Powersets (LP) and considers each distinct combination of labels that exist in the training set as a different class value of a single-label classification task [11]. Subsets of labels of fixed size k are generated randomly and single label classifiers are learned for all the label combinations in the powerset of this subset. These single label classifiers may take into account label correlations if adequate number of examples are present. In the end each LP classifier predicts values for the k labels.

3 Group Based Classification

Similarity of concepts can be judged in many different ways, including web semantics information, ontology rules or relationships between concepts provided with some video databases like for example TRECVID 2010 [9]. Example of such rules are *Anchorperson* is *Adult* and *News Studio* contains *Anchorperson*. To quantitatively express the similarity of concepts, intersection of common annotations can be used to find a similarity index between two concepts. Also feature vectors belonging to shots containing concepts can be used to find distance between two concepts. The criterion for clever grouping of concepts we do uses feature vectors for finding similarity between concepts to group them together.

After the concepts are grouped together into different groups, each concept is assigned to a number of different groups. The idea is that each concept is uniquely identified by a combination of outputs of certain groups. Thus if the concepts *Car* and *Road* appear in the same group, one of them should belong to at least one of the other groups to differentiate it from the other. In other words each concept is represented by a *unique* bit string with length equal to the number of total groups. Each bit of the bit string represents a group and the value of the bit is 1 if the concept belongs to the specified group. Figure 1 shows our intuition of the group based classification scheme. Ideally for C concepts $\log_2 C$ group classifiers are enough for learning as each concept can be identified with a unique bit string.



Concepts \ Groups	3	2	1
Male_Person	0	1	1
Asian_People	1	0	1
Anchorperson	0	1	0
Car	1	1	0
Cityscape	1	0	0
News_Studio	1	1	1

Figure 1. Grouping of concepts into non mutually exclusive partitions: a) A simplified space is shown with some labeled (and multi-labeled examples. b) Grouping of concepts. Concepts may appear in multiple groups. c) Each concept is a unique bit string and no group is empty.

3.1 Clever Grouping

We use average feature vector for each concept to find the alikeness between concepts. Average feature vectors for each concept $c \in C$ are obtained by averaging all the feature vectors for images containing the concept from our training set. The feature vectors are Bag of Words histograms [8] as described later in the experiments section. The clever grouping criterion we use is based only on visual similarity between concepts i.e. we use the similarity or the inverse of the distance between the average features as the closeness.

Suppose we want to generate D groups based on this criterion. First we consider the scenario where where the number of groups is less than the number of concepts $D < C$. The C average vectors are first clustered into D centers using k-means clustering with random initial centers. After clustering we generate a list for each average feature vector containing top n closest centers, like soft assignment [12]. The list is sorted with the closest center at the first rank and so on. D groups are generated corresponding to the D cluster centers. Each concept is always assigned to the closest group. Next the concept is assigned randomly to the next closest center with decreasing probability. The decrease in probability is proportional to the increase in the distance to the next closest center. Clustering is done so that the outliers (concepts whose average features are far from others) are considered for grouping as well when $D < C$.

The case where the number of groups is greater than the number of concepts, i.e. $D > C$, we drop the clustering mechanism. To create the first group we start with randomly selecting an average feature vector of a concept and drawing a number n randomly. Then n closest concepts with minimum distance to the selected concept are assigned to the first group along with the randomly selected concept making the first group. This process is repeated until D groups are generated. The n closest concepts are first sorted with the concept with the minimum distance ranked at first position and are assigned to the group with decreasing probability. This decrease in the probability of assignment is proportional to the increase in distance of the selected concept to the next closest concept.

The concepts are assigned sequentially that is the bit strings are generated sequentially. In case of a conflict between the assignment of two concepts, i.e. when two bit string are exactly the same, the two concerned bit strings are regenerated until the conflict is resolved. In this way each concept is uniquely identified by a different combination of groups. The case where $D > C$ there is rarely a conflict but if there is, the grouping is regenerated. The maximum value of n is fixed as 12 in our experiments with $C = 50$.

We then determine the labels of the members of the groups. All the examples belonging to those labels become part of that group. More specifically two examples that are visually very close do not necessarily end up in the same group unless their respective average feature vectors are close. Unlike attributes these groups are complex enti-

ties or complex attributes as they combine annotations from many objects (concepts).

We compare clever grouping of concepts with random grouping where for each group n concepts are selected and then bit strings are acquired for each concept. The process is repeated if there is a conflict between the bit strings of any two concepts. For both grouping criteria an example is considered positive if any of the participating labels (concepts) is positive for that example. Each group is then trained in a 1-vs-all fashion giving scores for each test frame $s(f|g)$ which is the score of the test frame f for the group g. Concept score is then calculated on the normalized groups scores as:

$$s_g(f|c) = \sum_{c \in g} s(f|g)$$

giving the score of the frame f for concept c. We do not subtract the scores of negative groups i.e. the groups which do not contain c as we found experimentally that using this information worsens the results.

4 Experiments and Results

We present here experiments carried out on the TRECVID 2010 datasets.

4.1 Experimental Setup

Dataset and Features: We have used the TRECVID 2010 IACC [9] dataset containing 11644 internet videos. This comprehensive dataset is divided into the training part with 3200 videos of 200 hours with a total of 119,685 keyframes. Rest of the approximately 8000 videos of 200 hours containing 146,788 keyframes are used for testing purposes. For testing the performance of various multi-label or group classification based systems on video concept detection we have used the list of 50 concepts from the TRECVID 2011 Light Semantic Indexing task.

We use 128 dimensional SIFT features [5] to describe local patches extracted using a Dense grid of points on the video keyframes [7]. The points on the grid are distanced 8-pixels apart. All the SIFT descriptors from the training set are then used to build a 500 word visual dictionary using k-means. For classification we have used linear SVM to learn from a suitable feature map (homogeneous kernel map) built by the histogram intersection kernel [13].

We have used Average Precision (AP) to measure concept detection performance as used in TRECVID semantic indexing benchmark [9]. For the overall performance we use Mean Average Precision (MAP) of 50 concepts.

RAKEL: For RAKEL we fix the value of $k = 3$ which is also known to give the best results [11]. We adapt the RAKEL algorithm for Average Precision in that we generate score for each shot so that a sorted list can be generated for each concept. Each LP classifier is a multi-label classifier that gives classification scores for the k concepts. We call a k-labelset a group with k concepts and each group predicts scores for each of the k concepts. In the end to obtain the score for each concept, scores from all the LP (group) classifiers, of which that concept is a part of, are added and normalized.

Late Fusion: We have also used late fusion to combine the baseline results with the 3 group based approaches. Weighted linear fusion is used in order to obtain a single output score for each concept $s(f|c)$ that is used to rank the video frame f for the concept c:

$$s(f|c) = w_s s_s(f|c) + w_g s_g(f|c)$$

where $s_s(f|c)$ and $s_g(f|c)$ are the concept scores acquired from the single label classification (baseline) and the group based approach respectively. The scores are rescaled according to one another using min-max normalization. The weights w_s and w_g are optimized over the development set.

As all the three approaches include some randomness we have repeated each experiment 5 times. We show the mean and standard deviation for the scores.

4.2 Results

The Semantic indexing results for various approaches on TRECVID 2010 test dataset for the 50 concepts of TRECVID 2011 light semantic indexing task are presented in figure 2 containing: i) single label classification which is also the baseline, ii) random grouping, iii) fusion of baseline with random grouping, iv) clever grouping, v) fusion of baseline with clever grouping, vi) RAKEL, vii) fusion of baseline with RAKEL. For baseline each concept is treated as a separate label.

Performance of various grouping approaches increases almost linearly with the increase in the number of groups with random grouping of concepts performing better than RAKEL for almost every group size. Intelligent grouping significantly outperforms the other two techniques and approaches single label classification performance for training around 80 intelligently formed groups. Further increasing the number of groups increases marginally the performance over the baseline.

Figure 2 also present the results of fusing various group based techniques with the baseline. Fusion with RAKEL improves concept detection performance over baseline and increases linearly with the increase in the number of groups. The best performance is acquired using 100 groups with 10% overall increase in the indexing performance over the baseline. Clever grouping when fused with single label

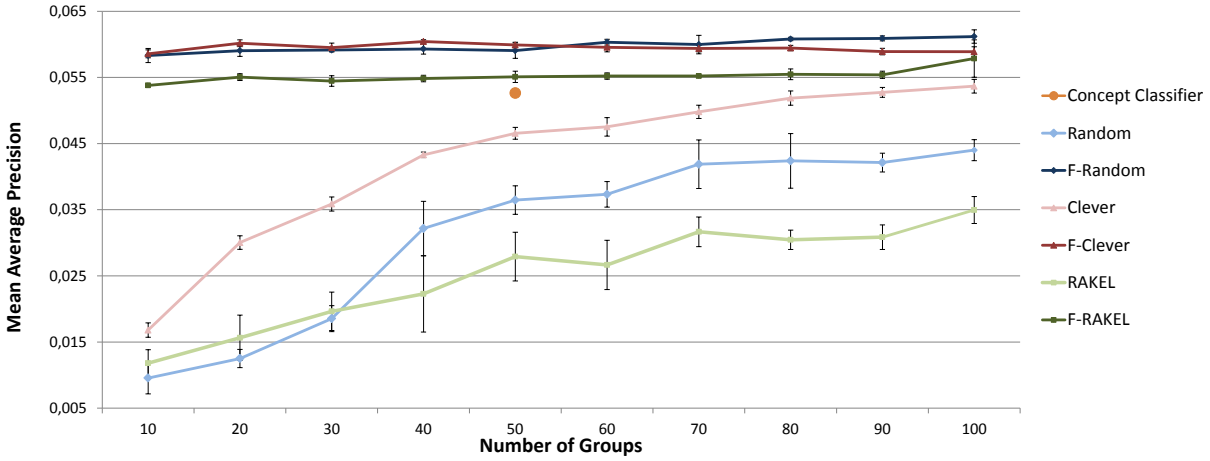


Figure 2. Mean Average Precision for 50 concepts for different grouping criteria and their fusion with the baseline.

classification provided best improvement for a very few number fo groups trained. This is observable in the upper left part of the figure 2 as fusing concept scores from only 10 clever groups with the baseline improves MAP score from 0.0527 to 0.0592 with 12% increase in performance. Using 20 intelligent groups the improvement is 15%. The fusion performance is good with clever grouping till the number of groups is 40 and after that the MAP decreases linearly, when the number of groups equals or is greater than the number of concepts. With random grouping the trend is somewhat similar to RAKEL as the fusion performance increases with the increase in the number of groups. Although the MAP here is better than that of RAKEL for every grouping. Fusion results with random grouping lingers close to the fusion results with intelligent grouping. The performance is slightly inferior to intelligent grouping for up to 50 groups and then outperforms intelligent grouping as the number of groups increases further. Best MAP score is 0.0612 which is a 16% increase over the baseline observed with 100 random groups used (upper right corner).

Analysis of Fusion Weights: Although intelligent grouping outperforms random grouping for concept detection for all group sizes this does not stay the same when the group based scores are fused with single label classification scores. To further analyze this we look closer at the fusion and investigate the fusion weights assigned to the concepts for different number of groups. Figure 3 plots the evolution of the fusion weights for the three multi-label approaches for different group sizes derived from the training data. The figure shows the average of the weights assigned to each concept score derived from the group based approaches compared to concept scores from individual concept learning in the linear fusion. The fusion weights increase with the number of groups for the three approaches as more groups means better classification at concept level except when there is overfitting or when the group based approach performs worse than the baseline for a certain concept. For clever grouping the average weight quickly reaches the level where both the group based and the single label approaches contribute almost equally to the final score. Thus when the number of groups equals 50 both approaches contribute exactly equally for the final performance and as the number of groups increases more weight is assigned to clever grouping on average. In figure 2 we see that the clever grouping approaches single label classification with the increase in the number of groups. Thus adding more and more groups with concepts grouped

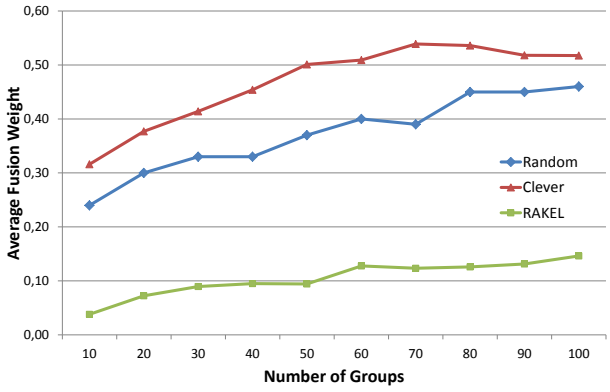


Figure 3. Evolution of the fusion weights with the increase in the number of groups.

on visual similarity will always converge to single label concept wise classification and fusing those two together will only results in marginal improvement.

Note that this is not the case for random grouping technique as the maximum weight assigned on average is around 46%. In other words random grouping contributes around 46% to the fusion even with 100 groups. Thus as the performance of random grouping is always inferior to baseline in figure 2, fusing it with baseline always brings in complimentary information and improves performance with the increase in number of groups. However clever grouping does bring more useful information in fusion than random grouping if number of groups is kept inferior to the number of concepts. Thus we are able to improve 12% and 15% over the baseline with only 10 and 20 new classifiers trained respectively in addition to 50 single-label classifiers.

One explanation for the not so good performance of RAKEL is that the label set in the TRECVID dataset we used is very sparse i.e. only a few combinations of labels are possible. In the RAKEL mechanism each group has $k = 3$ and thus up to 8 single label classifiers are trained for each group. For good classification results the number of positive examples for each combination of labels should be adequate [11]. We have found that in the Label Powersets the number of examples for the label set $\{0, 0, 0\}$ where all the 3 concepts are negative dominates the number of examples for other label sets. This complicates things as the label sets like $\{1, 0, 0\}$ or $\{1, 0, 1\}$, where one concept is truly distinguished against others have very few positive examples for training. The final score for each concept in the end is thus dominated by the negative score of the classifier trained on examples from the label set $\{0, 0, 0\}$. From the TRECVID 2010 training data and our setting of the RAKEL algorithm we have on average 6188 positive training examples for the label set $\{0, 0, 0\}$ for every LP classifier compared to only 168 positive examples for other label sets. Thus an LP classifier lacks the examples for the truly discriminative label combinations owing to the relatively poor performance.

5 Conclusions

We have devised a quick way of grouping concepts together based on their visual similarity to train them together for concept detection in internet videos. The group making criterion is intuitive, very fast and each concept is represented by a unique combination of groups.

With the introduction of a little useful even random information we are able to improve concept detection performance on the TRECVID 2010 datasets for the list of 50 concepts. We improve 12% and 15% over the baseline with only 10 and 20 new group classifiers formed on the clever criterion. Using random grouping we further improve but at a cost of training a total of twice as many classifiers as compared to the intelligent criterion.

We feel that this group based classification can help ultimately reduce the number of classifiers to be trained if effective combination techniques can be found. So far for the grouping of concepts we have only used visual similarity while there are other options that may be fruitful to explore. Grouping only on visual similarity results in overfitting as the number of groups increases. We would also like to add diverse information in the group or create groups from negative information. This may be acheived with using mutual information principles on negative and positive annotations for concepts. This information is inherently provided with TRECVID style multi-label datasets.

For RAKEL the complication lies in finding the adequate number of examples for each possible label set in the TRECVID dataset. We feel that there is a need to find a better way to combine and train single label classifiers for making one multi-label LP classifier.

References

[1] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth. Describing objects by their attributes. In *CVPR*, 2009.

[2] V. Ferrari and A. Zisserman. Learning visual attributes. In *Advances in Neural Information Processing Systems*, Dec. 2007.

[3] J.-P. Heo, Y. Lee, J. He, S.-F. Chang, and S.-E. Yoon. Spherical hashing. In *CVPR*, pages 2957–2964, 2012.

[4] C. H. Lampert, H. Nickisch, and S. Harmeling. Learning to detect unseen object classes by betweenclass attribute transfer. In *CVPR*, 2009.

[5] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, November 2004.

[6] M. Rastegari, A. Farhadi, and D. Forsyth. Attribute discovery via predictable discriminative binary codes. In *ECCV*, pages 876–889, 2012.

[7] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman. Labelme: A database and web-based tool for image annotation. *Int. J. Comput. Vision*, 77:157–173, May 2008.

[8] J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. In *ICCV*, 2003.

[9] A. F. Smeaton, P. Over, and W. Kraaij. Evaluation campaigns and trecvid. In *MIR*, 2006.

[10] G. Tsoumakas and I. Katakis. Multi-label classification: An overview. *Int J Data Warehousing and Mining*, 2007:1–13, 2007.

[11] G. Tsoumakas, I. Katakis, and I. P. Vlahavas. Random k-labelsets for multilabel classification. *IEEE Trans. Knowl. Data Eng.*, 23(7):1079–1089, 2011.

[12] J. C. van Gemert, C. J. Veenman, A. W. M. Smeulders, and J. M. Geusebroek. Visual word ambiguity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(7):1271–1283, 2010.

[13] A. Vedaldi and B. Fulkerson. VLFeat: An open and portable library of computer vision algorithms, 2008.

[14] F. Yu, R. Ji, M.-H. Tsai, G. Ye, and S.-F. Chang. Weak attributes for large-scale image retrieval. In *CVPR*, 2012.