iGroup: Weakly supervised image and video grouping

Andrew Gilbert, Richard Bowden
University of Surrey
Guildford, UK

{a.gilbert, r.bowden}@surrey.ac.uk

Abstract

We present a generic, efficient and iterative algorithm for interactively clustering classes of images and videos. The approach moves away from the use of large hand labelled training datasets, instead allowing the user to find natural groups of similar content based upon a handful of "seed" examples. Two efficient data mining tools originally developed for text analysis; min-Hash and APriori are used and extended to achieve both speed and scalability on large image and video datasets. Inspired by the Bag-of-Words (BoW) architecture, the idea of an image signature is introduced as a simple descriptor on which nearest neighbour classification can be performed. The image signature is then dynamically expanded to identify common features amongst samples of the same class. The iterative approach uses APriori to identify common and distinctive elements of a small set of labelled true and false positive signatures. These elements are then accentuated in the signature to increase similarity between examples and "pull" positive classes together. By repeating this process, the accuracy of similarity increases dramatically despite only a few training examples, only 10% of the labelled groundtruth is needed, compared to other approaches. It is tested on two image datasets including the caltech101 [9] dataset and on three state-of-the-art action recognition datasets. On the YouTube [18] video dataset the accuracy increases from 72% to 97% using only 44 labelled examples from a dataset of over 1200 videos. The approach is both scalable and efficient, with an iteration on the full YouTube dataset taking around 1 minute on a standard desktop machine.

1. Introduction

There is a large and ever growing quantity of media available on-line and on personal computers and therefore an increasing demand to meaningfully sort and group media efficiently, without the need for large scale groundtruth labelling or meta annotations. Ideally, users would weakly supervise the learning process by picking a few images or

videos that belong to the same class or group. An automatic approach would then extract and learn rules that can generalise and cluster the remaining unseen media.

In order to achieve this, we aim to move away from hard supervised learning that is often proposed [19, 10, 23, 26, 12]. These approaches rely on groundtruthing large quantities of training data, previously, approaches have been proposed to use a "single example", so called "one shot learning" [24, 20], where only one example is used in training. However, these approaches are sensitive to the training examples ability to generalise and techniques are generally applied to simple staged datasets.

The similarity measure employed in this paper extends the min-Hash algorithm that was originally designed to identify the similarity between text in documents [3] by efficiently computing the distances between high dimensional representations. Similarly, the association rule data mining technique used (known as *APriori* [1]), was originally designed to identify co-occurrences from large text files. Rather than using groundtruthed training data, that can be increasingly hard to produce as datasets get larger and more complex, we rely on an iterative process to select far fewer training examples. Using *APriori*, the distinctive and discriminative elements of these selected examples are identified, and accentuated across the dataset by dynamically augmenting the representation with new compound visual words to form an image signature.

In the paper we propose and explain the idea of the image signatures in section 3 and extend the min-Hash algorithm for video similarity through the use of visual words in section 4. Furthermore, an iterative process is presented, that in section 5 uses *APriori* data mining to identify common elements of the signatures, that are then accentuated in section 6 to further increase accuracy. Extensive results of the approach on image and video datasets are shown in section 7.

2. Related Work

Within computer vision there has been a number of approaches that utilise data mining's ability to work efficiently

and with large amounts of images or video.

Within the image domain, Quack *et al* [22] applied *Association rule* data mining to supervised object recognition by mining spatially grouped SIFT descriptors. This made use of the algorithms ability to efficiently compute the distinctive spatial combinations of the SIFT features. Chum [5] proposes a min-Hash based approach to detect rare co-occurrences of data in groups of image in a similar fashion to that of APriori data mining.

Within the temporal domain, Gilbert *et al* [10] argue that with many other action recognition approaches [15, 16, 8], the features used are engineered to fire sparsely, to ensure that the overall problem is tractable. However, they suggest that this can sacrifice recognition accuracy as it cannot be assumed that the optimum features in terms of class discrimination are obtained from this approach. In contrast, they take an over complete set of Harris corners [13], group them spatially and temporally and mine out the optimal feature combinations, to be used to classify the video sequences.

Chum *et al* [6] demonstrated the ability of min-Hash to efficiently identify near duplicate images within datasets. This was extended further by Chum *et al* [7] to an efficient fast method to approximate the histogram intersection of images to improve near duplicate image detection. We bring many of the tools and ideas together to iteratively cluster data regardless of source or representation, based on the idea of an image signature

3. Input Data Signature

An image signature is a frequency histogram of a set of discrete symbols similar in nature to the popular Bag of Words (BoW) model [17, 25]. In fact we demonstrate how a BoW histogram can be used as the initial stage of the signature in section 7. However there are two key differences from a classical BoW: Firstly the signature is increased in size to accentuate symbols and combinations of the symbols that discriminate or provide similarity between examples of classes. Secondly the signature can be based on any feature and we demonstrate signatures based on both BoW representations and other classifier responses for both image and video media

Figure 1 gives an overview of the approach in this paper. First the feature classifiers are applied to the input sample to form the initial signature. The visual words of the signature could come from unsupervised clustering of training data, as is commonplace for BoW approaches, or as the output of a set of classifiers. We will demonstrate both approaches in the results section. This signature is then converted from a weighted histogram into the min-Hash representation to facilitate high speed similarity measures between the input samples. The min-Hash process provides a signature-wise similarity, which the user can utilise to select M signatures

Figure 1. An overview of the proposed approach

that are true positive and a single false positive for a particular class. These selected signatures are then mined to identify the distinctive combinations of words. The rules of mining are then converted into new compound visual words and appended to all the signatures, which will have the effect of pulling the signatures from the positive examples closer together. This process is then iterated, allowing a user to iteratively cluster data.

4. Similarity of signatures

Min-Hash was originally developed for near-duplicate detection of text [3] and was more recently adapted for the near duplicate detection of images [7]. It is a randomised hashing approach, where the computation is proportional only to the number of input samples rather than the size of the vocabulary. This means that it is ideally suited for use as the similarity measure for image signatures which can be of high and increasing dimensionality.

4.1. The min-Hash algorithm

The distance measure between two input samples is computed as the similarity of sets S_1 and S_2 , which is the ratio of the number of elements in the intersection, over the union of the two sets.

$$\text{sim}(S_1, S_2) = \frac{|S_1 \cap S_2|}{|S_1 \cup S_2|} \quad (1)$$

min-Hash is able to estimate $\text{sim}(S_1, S_2)$ without performing an exhaustive naive element by element comparison of S_1 and S_2 . In order to estimate $\text{sim}(S_1, S_2)$, N random permutations, π , of the vocabulary, ν , are formed. These are used to form min-Hashes of the input samples. A min-Hash is defined as $\min \pi(S_i)$, i.e. the minimum element position of the π permutation that is present in the set S_i . For example, given the visual word vocabulary of $\nu = A, B, C, D, E, F$ and the three sets $\{A, B, C\}$, $\{B, C, D\}$ and $\{A, E, F\}$, with 4 random permutations, N , the resultant min Hashes are shown in Table 1. The similarity, $\text{sim}(S_1, S_2)$ is estimated as $l_{(1,2)}/N$, where

$$l_{(1,2)} = \sum (\min \pi(S_1) = \min \pi(S_2)) \quad (2)$$

Table 1. Example of the 4 random permutations and resultant min-Hashes

Permutations						min-Hash		
						ABC	BCD	AEF
A	B	E	F	C	D	1	1	1
B	C	E	A	F	D	1	1	3
E	A	B	A	D	F	2	3	1
F	A	E	D	B	C	2	4	1

For the example in table 1, the estimated similarity between sets $\{A, B, C\}$ and $\{B, C, D\}$ will be 0.5 as they share 2 min-Hashes, which is the same as the exhaustive (naive) similarity, while the sets $\{A, B, C\}$, and $\{A, E, F\}$ share 1 min-Hash, with an estimated similarity of 0.25, compared to the exhaustive value of 0.2.

The false positive rate can be further reduced by grouping the min-Hash results into “sketches”. Where $K(S_1)$ is the sketch $\{\min \pi_1(S_1), \dots, \min \pi_n(S_n)\}$ consisting of n Hashes. The grouping process will only estimate the similarity for input sample pairs that have at least m (set as 1) identical sketches out of the h sketches $(K_1, \dots, K_h)$. For example, taking Table 1, with a sketch size of $n = 2$, the sets $\{A, B, C\}$, $\{B, C, D\}$ and $\{A, E, F\}$ would be represented by the sketches $((1, 1), (2, 2)), ((1, 1), (3, 4)), ((1, 3), (1, 1))$ respectively. The only sketch match would be $\{A, B, C\}$, $\{B, C, D\}$ in the first sketch (1, 1), with no other pair considered similar.

4.2. Histogram weighting approximation

The original min-Hash algorithm [6] is designed for a set of uniformly weighted symbols, therefore to be able to represent the frequency of symbols, the algorithm is adapted.

For a visual vocabulary containing $|X|$ visual words or features, e.g. $X = \{A, B, C\}$, t_i is the frequency response of the feature e.g., $t_1 = \{3, 0, 2\}$ $t_2 = \{2, 1, 0\}$. In order to convert the frequency based image signature into a min-Hash set of uniform symbols, the symbols are duplicated based upon their frequency to produce $t_1 = \{A1, A2, A3, C1, C2\}$ $t_2 = \{A1, A2, B1\}$.

5. Expanding signatures through co-occurring discriminatory features

With challenging datasets including recognition of realistic video actions and the inherent noise and redundancy of any signature based upon a BoW approach, it is likely that min-Hash will produce falsely matching signatures that are unrelated. This is likely as there is often minimal inter (across) class variation, while lacking intra (within) class similarity. Therefore we propose a novel approach to “pull”

positive sets together. Association Rule data mining will be used to identify the compound visual words that are distinctive and descriptive to a selected subset of the true positives (positive transaction sets) when compared to the false positives (a negative transaction set). The new compound visual words are then added to all the image signatures and this, in-turn, will provide an increase in intra class similarity.

To identify the unique set elements, a version of association rule data mining called *APriori* [1] is used. This has been used to great success for both image [22] and video [11] recognition, as it is well suited and computationally efficient for large amounts of data. This is a brief introduction to the data mining *APriori* algorithm, but for a more detailed explanation see [11]. The algorithm, is designed to search databases and identify the set elements that co-occur most frequently within the positive sets with respect to the negative sets. The frequency of a set element is related to the support and confidence of an association rule. An association rule of the form $A \Rightarrow B$ is evaluated by looking at the relative frequency of its antecedent and consequent parts i.e. the set elements A and B . The support for a set element is the probability that a Transaction contains the set element. For A , this is calculated as the size of the set of all the transactions T , such that T is an element of the overall database of the transactions D and A is a subset of T , normalised by the size of D . The support of the rule $A \Rightarrow B$ is therefore

$$\text{sup}(A \Rightarrow B) = \frac{|\{T \mid T \in D, (A \cup B) \subseteq T\}|}{|D|} \quad (3)$$

and measures the statistical significance of the rule. The confidence of a rule is then calculated as

$$\begin{aligned} \text{conf}(A \Rightarrow B) &= \frac{\text{sup}(A \cup B)}{\text{sup}(A)} \\ &= \frac{|\{T \mid T \in D, (A \cup B) \subseteq T\}|}{|\{T \mid T \in D, A \subseteq T\}|} \end{aligned} \quad (4)$$

The support for the rule is the probability of the joint occurrence of A and B i.e. $P(A, B)$ while confidence is the conditional probability $P(B|A)$.

In addition to the set elements being frequent, they must also be discriminative with respect to the negative set. To achieve this, the algorithm is run on transactions from both the positive and negative sets. The transaction vectors of all examples are appended with a label, α , that identifies if the set is a positive or negative example. The results of data mining then include rules of the form $(A, B) \Rightarrow \alpha$ and an estimate of $P(\alpha|A, B)$ is given by the confidence of the rule. As the Transaction database contains both positive and negative training examples $P(\alpha|A, B)$ will be large only if (A, B) occurs frequently in the positive examples

but infrequently in the negative examples. If (A, B) occurs frequently in both positive and negative examples, then $P(\alpha|A, B)$ will remain small. The support threshold is the number of positive transactions over the total number of transactions, while the confidence threshold is 100, to ensure an association rule is only found if the elements are within all the positive sets and none of the negative sets.

6. Iterative signature learning

The mined association rules are employed to adjust the min-Hash representation of the image signatures. Each rule is taken in turn and if it contains the returned mined subset an additional min-Hash symbol element is added. This will accentuate common and distinctive elements of the positive rules from the mining and increase their overall similarity - in essence “pull” the positive signatures together. For example, using the image signatures from section 4.2, $t_1 = \{A1, A2, A3, C1, C2\}$ $t_2 = \{A1, A2, B1\}$. If the association rule returned from the mining was a subset of t_i , e.g. Ax where x is any number, the element $(A4)$ would be added to set t_1 and the element $(A3)$ would be added to set t_2 . The process would be repeated over all the input sets, however if the set does not contain the subset (Ax) , it would not be incremented. This increased weighting on the subset (A) would “pull” together sets that contain subset (A) over time improving accuracy. In addition, the mining can return association rules that contain multiple subsets that together are descriptive and distinctive. Using the same example, if the mining returns the rule $(A2, B1)$, it would not be appended to t_1 as the set does not contain any (B) elements. However, it would be appended to t_2 , making $t_2 = \{A1, A2, B1, AB1\}$ and then treated as a single element. This means that for the min-Hash permutations to match, both sets would have to contain the symbol $A2$ and $B1$, not just a subset. This has the ability to reduce the confusion between classes further.

The adjusted min-Hash representations of the input image signatures can then be used to recompute similarities based on the set overlap and the mining cycle is repeated.

7. Results

To illustrate the generalisation of the approach, and to evaluate the quality of the clustering and categorisation of the approach, both image and video clustering were investigated. We demonstrate iterative grouping on the image categorisation dataset of [21] using their histograms of colour and edges to form the initial signatures, on the caltech101 [9] dataset using the BoW features of [4], on the KTH dataset and the YouTube dataset using the compound corners features of [10], and on the Hollywood2 dataset using the BoW features of [19]. To evaluate grouping performance, each item is assigned to the class of its nearest

neighbour.

7.1. Images

Designed to demonstrate that the approach is not restricted to a single media type and to illustrate the approach, the images in two image datasets are grouped. First the Image Scene dataset used in [21] is tested. It consists of 100 images spread across 4 classes of landscape: city, jungle, mountain and winter, with examples shown in Figure 2

Figure 2. Image Dataset examples

In order to form the signatures for the images, an 11 bin colour histogram and a 42 bin edge orientated histogram were concatenated and histograms computed over each image. The image signature database had, $\nu = 432$, with $N = 1500$ min-Hash permutations, and a sketch size $n = 3$. For each query image, the distance to all other images in the dataset was computed using the min-Hash histogram similarity measure and a nearest neighbour assignment used to group all images. The class label of the resultant maximised image was compared to the query image, and a true positive occurred if both images had the same class label. This is visualised in Figure 3(a), which shows the correlation between the query image and the rest of the dataset. The white dot shows the closest image in the dataset. Self similarity along the diagonal is ignored but a strong block diagonal shows good class clustering. Class boundaries are indicated as green boxes.

It can be seen that initially there is a degree of confusion between the classes (Figure 3(a)) and this is reflected in the 65.5 % overall accuracy. To improve the accuracy, the iterative mining process is run by selecting M true positive examples from a class and 1 false positive. The *APriori* identifies the elements within the min-Hash sets that are distinctive and discriminative to the true positive examples with respect to the false positive example and all the image signatures are adjusted accordingly. Figure 4 shows that

Figure 3. (a) Initial confusion between query image and dataset, (b) Confusion after 7 iterations using $M = 1$ (Green lines indicate class boundaries)

Figure 4. Image Dataset Accuracy with respect to iteration level, and varying M

the accuracy increases with respect to iterations for different values of M . For example with $M = 3$ it takes 5 iterations to achieve an accuracy of 91%, therefore requiring a total of 20 labelled images. However with $M = 1$ it takes only 7 iterations, therefore only requiring 14 labelled images for the full dataset of 100 images. The resultant confusion matrix between images in the dataset is shown in Figure 3(b).

To provide a more challenging test and demonstrate flexibility to features, the commonly used benchmark dataset, *Caltech101* [9] was used for evaluation. The dataset consists of 101 object categories with between 31 to 800 images per category. The semantic image signature is then iteratively adapted using 15 training examples randomly selected from each class. Then these image signatures were tested on another 50 unseen "test" images from each class by performing a nearest neighbour assignment to the closest class and this process was repeated 10 times. The initial image signatures were formed from BoW histograms of standard SIFT descriptors with the dimension reduced to 30 as employed in [4]. The average results of the 10 runs are shown in table 2 where a 40% increase in performance is seen from the baseline signature, this compares well with accuracies achieved in classification by Cai [4].

Table 2. Accuracy of *Caltech101* dataset

Approach	Accuracy
Cai [4]	64.9%
Baseline min-Hash	21.54%
Sig min-Hash	59.7%

7.2. YouTube Video Dataset

The *YouTube* dataset [18] consists of eleven categories: *basketball shooting, cycling, diving, golf, horse riding, juggling, play swings, tennis swinging, trampolining, volleyball, and dog walking*. This real life dataset contains over 1200 videos, that exhibit large variations in camera motion, object appearance and pose, object scale, viewpoint, cluttered background and illumination conditions. Some examples of the dataset are shown in Figure 5.

Figure 5. *YouTube* dataset examples

7.2.1 Feature Classifier

To form the initial image signature, a histogram is computed using the features by Gilbert *et al* [10]. These consist of compound corner classifiers trained on the *KTH* dataset [23], with results on the *KTH* dataset also shown in the following section. While these features might be optimal for classification of the *KTH* dataset, it is not the case for other datasets and it is unlikely that they are optimal in any sense for the *YouTube* dataset. However, as they were constructed to discriminate between different types of motion, they should provide some class discrimination over other datasets including motions. We will rely on the image signature to differentiate between the classes. Other than the different features used in the initial image signature, the experimental process is the same as outlined for the image datasets.

For the video domain, the size of the initial signatures now increases to 3108, due to the additional feature classifiers used. The results over the whole *YouTube* dataset are shown in Figure 6(a), for the accuracy with respect to the number of iterations, with varying sizes of M . The size of the vocabulary is shown in Figure 6(b), while the speed for each iteration is shown in Figure 6(c)

For the *YouTube* dataset it can be seen that the accuracy increases from a basic min-Hash baseline of 72%, to 97%

(a) YouTube Dataset Accuracy with respect to iteration level, and varying M

(b) Signature size for YouTube, with varying M

(c) YouTube Time taken per iteration for varying M

Figure 6. The *YouTube* dataset results

correctly clustered and takes only 36 labelled groundtruth videos in 6 iterations using $M = 5$. This is a small amount, with other standard approaches using a leave one out training-test partitions requiring 1121 labelled videos for training. More importantly, as the accuracy increases, so does the complexity of the vocabulary. However, the average time for each iteration increases by only 1 second for $M = 5$. This indicates the important feature of min-Hash, that it is computationally affected only by the number of input samples or signatures and not the complexity of the actual signature.

For a comparison with more standard published approaches, we adopt the commonly used Leave-One-Out Cross-Validation. More specifically, for the *YouTube* dataset, adopting the settings given in [18], the dataset was divided into 25 subsets, out of which 24 subsets were used

for training and the remaining subset was used for testing. Where for each unseen test subset, the other subsets were used to adjust the signatures using $M = 5$, for 4 iterations. However as only 4 iterations were used only 24 videos needed to be groundtruthed unlike the 1121 videos used in the other approaches. The semantic clusters were iteratively built on the 24 training subsets, and then classified by performing a nearest neighbour assignment to the closest class. Table 3 shows the results for our signature min-Hash approach compared to other recently published results on the same dataset.

Table 3. Accuracy of *YouTube* dataset

Approach	Accuracy
Cinbis [14]	75.2%
Liu [18]	71.2%
Bregonzio [2]	63.1%
Baseline min-Hash	56.4%
Sig min-Hash	79.7%

The results compare well to other state of the art results, especially as the feature classifiers used to form the image signatures are built on the *KTH* dataset. In addition, this approach needs very little actual groundtruthed data, just 44 labelled images per class, so a total of 528 labelled examples in total.

7.3. *KTH* Dataset

The well known and popular *KTH* dataset [23] was also evaluated, it contains 6 different actions; *boxing*, *hand-waving*, *hand-clapping*, *jogging*, *running* and *walking*, examples of each action are shown in Figure 7.

(a) boxing (b) handclapping (c) handwaving

Figure 7. Examples from the *KTH* dataset

There are a total of 25 people performing each of the 6 actions, 4 times; giving 599 video sequences (1 sequence is corrupt). Each video contains 4 instances of the action totalling 2396 unique actions. We present results using training and testing data split as suggested by Schüldt, with 8 people for training, and 8 people testing. The image signatures were iteratively retrained using the 8 training people.

To construct the image signatures, the same features as used for the *YouTube* dataset are employed. These are provided by Gilbert *et al* [10] and consist of compound corner classifiers trained on the *KTH* dataset [23].

Table 4. Accuracy of *KTH* dataset

Approach	Accuracy
Schüldt [23]	71.71%
Laptev [16]	91.8%
Laptev [16]	91.8%
Wang [26]	92.1%
Gilbert [10]	95.7%
Baseline min-Hash	44.3%
Signature min-Hash APriori	91.2%

Table 4 shows the results, it can be seen that the initial baseline min-Hash value is very low at only 44.3% however after seven successive iterative selections, the accuracy is increased to 91.2%. The 91.2% compares very favourably with the other state-of-the-art, especially when it is noted that as $M = 5$, only 42 videos were required as labelled training data, instead of the normal 192 videos.

7.4. Hollywood2 Dataset

To provide additional challenge, the *Hollywood2* dataset [19] is evaluated. It builds upon [16] and consists of 12 action classes; *AnswerPhone*, *DriveCar*, *Eat*, *FightPerson*, *GetOutCar*, *HandShake*, *HugPerson*, *Kiss*, *Run*, *SitDown*, *SitUp*, *StandUp* with around 600,000 frames or 7 hours of video sequences split evenly between training and test datasets, of clips made from hollywood movies.

Figure 8. Examples from the *Hollywood2* dataset [19]

We obtain HoG/HoF descriptors using the interest point detection method of [15], and construct a visual dictionary using K-Means with $K = 4000$ visual words and train a SVM classifier. The classifier response is used as the input for the image signature. For the *Hollywood2* dataset, the clean train and test partitions proposed by Marszalek [19] were used. There are a total of 810 videos within the training subset spread over the 12 action classes, with 884 test sequences. Table 5 shows the baseline and final accuracy of the iterative min-Hash APriori method.

The final accuracy of 43.2% is nearly double the original baseline of 26%. This improvement is from using 8 iterative selections of 5 true positive classifications and a single negative classification in the training subset. This means that a total of 48 videos are actual labelled compared to the approaches by Gilbert [10] and Wang [26] requiring all of the 810 training videos.

Table 5. Accuracy of *Hollywood2* dataset

Approach	Accuracy
Marszalek [19]	35.5%
Han [12]	42.1%
Wang [26]	47.7%
Gilbert [10]	50.9%
Baseline min-Hash	26.9%
Signature min-Hash APriori	43.2%

7.5. Computational costs

The approach has been designed to be efficient, allowing for near real time operation. Table 6 shows how that as the dataset size increases so does the time taken, from 1 second per iteration on the Image Scene dataset, to 40 seconds for the *Hollywood2* dataset with 884 videos. However the

Table 6. Computational Time of datasets

Dataset	Dataset Size	Img Sig Size	Iter Time
Image Scene	100	53	1 sec
Caltech101	5050	2150	30 sec
YouTube	1200	3108	63 sec
KTH	768	1204	25 sec
Hollywood2	884	4503	45 sec

approach is ideally suited to video based problems as the computational cost is mainly proportional to the size of the dataset, not the dataset complexity.

8. Conclusion

While the iterative clustering procedure provides excellent results across a range of datasets and features using relatively little training data, it is important to note that in the case of Sec 7.3 and 7.4, while the grouping requires minimal data, the features themselves have been learnt over the entire training data. This serves to highlight that features that are learnt for classification, may not be the best features for grouping or clustering using naive distance metrics, but through the use of signatures, these features can be reweighed appropriately. We have presented an approach to efficiently cluster similar videos and images from large datasets. By selecting a small subset of the true positive and false positive results, we can pull additional true positive examples of the classes together by identifying the distinguishing components of the input samples signature. The use of a signature for each input sample allows for a very flexible generic approach to the input format. In future, the implemented feature classifiers; mined compound corners; HoG/HoF; edges and colour information, could be appended by the response of further feature classifiers, such

as SIFT. This additional information about the input sample would then be able to increase the performance but due to the min-Hash there would be minimal additional computational costs.

References

- [1] R. Agrawal and R. Srikant. "Fast Algorithms for Mining Association Rules in Large Databases". In *VLDB'94, Proceedings of 20th International Conference on Very Large Data Bases*, pages 487–499, 1994. 1, 3
- [2] M. Bregonzio, J. Li, S. Gong, and T. Xiang. Discriminative Topics Modelling for Action Feature Selection and Recognition. In *Proc of British Machine Vision Conference (BMVC'10)*, 2010. 6
- [3] A. Broder. On the Resemblance and Containment of Documents. In *Proc SEQs: Sequences'91*, 1998. 1, 2
- [4] H. Cai, K. Mikolajczyk, and J. Matas. Learning linear discriminant projections for dimensionality reduction of image descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2010. 4, 5
- [5] O. Chum and J. Matas. Unsupervised discovery of co-occurrence in sparse high dimensional data. In *Proc of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR'10)*, 2010. 2
- [6] O. Chum, J. Philbin, M. Isard, and A. Zisserman. Scalable Near Identical Image and Shot Detection. In *Proceedings of the 6th ACM international conference on Image and video retrieval, CIVR '07*, pages 549–556, New York, NY, USA, 2007. ACM. 2, 3
- [7] O. Chum, J. Philbin, and A. Zisserman. Near Duplicate Image Detection: min-Hash and tf-idf Weighting. In *Proc. of BMVA British Machine Vision Conference (BMVC'08)*, 2008. 2
- [8] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie. "Behavior Recognition via Sparse Spatio-temporal Features". *ICCCN '05: Proceedings of the 14th International Conference on Computer Communications and Networks*, pages 65–72, 2005. 2
- [9] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: an incremental bayesian approach tested on 101 object categories. In *Proc of IEEE CVPR 2004, Workshop on Generative-Model Based Vision.*, 2004. 1, 4, 5
- [10] A. Gilbert, J. Illingworth, and R. Bowden. "Scale Invariant Action Recognition Using Compound Features Mined from Dense Spatio-Temporal Corners". In *Proc. of International Conference on Computer Vision (ICCV'09)*, 1:222–233, 2009. 1, 2, 4, 5, 7
- [11] A. Gilbert, J. Illingworth, and R. Bowden. Action recognition using mined hierarchical compound features. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 883 – 897, 2011. 3
- [12] D. Han, L. Bo, and C. Sminchisescu. Selection and Context for Action Recognition. In *Proc. of International Conference on Computer Vision (ICCV'09)*, 1:1933–1940, 2009. 1, 7
- [13] C. Harris and M. Stephens. "A Combined Corner and Edge Detector". In *Proc. of Alvey Vision Conference*, pages 189–192, 1988. 2
- [14] N. Ikizler-Cinbis and S. Sclaroff. Object, Scene and Actions: Combining Multiple Features for Human Action Recognition. In *Proc. of European Conference on Computer Vision (ECCV'10)*, 2010. 6
- [15] I. Laptev and T. Lindeberg. "Space-time Interest Points". In *Proc. of IEEE International Conference on Computer Vision (ICCV'03)*, pages 432–439, 2003. 2, 7
- [16] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. "learning Realistic Human Actions from Movies". In *Proc. of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR'08)*, pages 1–8, 2008. 2, 7
- [17] F.-F. Li and P. Perona. A bayesian hierarchical model for learning natural scene categories. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 2 - Volume 02*, CVPR '05, pages 524–531, Washington, DC, USA, 2005. IEEE Computer Society. 2
- [18] J. Liu, J. Luo, and M. Shah. Recognizing Realistic Actions from Videos "in the Wild". In *Proc of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR'09)*, 2009. 1, 5, 6
- [19] M. Marszalek, I. Laptev, and C. Schmid. Actions in Context. In *Proc. of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR'09)*, 2009. 1, 4, 7
- [20] H. Ning, T. Han, D. Walther, M. Liu, and T. Huang. Hierarchical Space-time Models, Enabling Efficient Search for Human Actions. In *Proc of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR'09)*, pages 2568 – 2574, 2009. 1
- [21] E. Ong and R. Bowden. Learning Distances for Arbitrary Visual Features. In *Proc of British Machine Vision Conference (BMVC'06)*, 2006. 4
- [22] T. Quack, V. Ferrari, B. Leibe, and L. Gool. "Efficient Mining of Frequent and Distinctive Feature Configurations". In *Proc. of IEEE International Conference on Computer Vision (ICCV'07)*, 2007. 2, 3
- [23] C. Schuldt, I. Laptev, and B. Caputo. Recognizing Human Actions: a Local SVM Approach. In *Proc. of International Conference on Pattern Recognition (ICPR'04)*, III:32–36, 2004. 1, 5, 6, 7
- [24] E. Shechtman and M. Irani. Space-time Behavior-based Correlation -or- How to Tell if two Underlying Motion Fields are Similar without Computing them. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29:2045 – 2056, 2007. 1
- [25] J. Sivic, B. C. Russell, A. A. Efros, A. Zisserman, and W. T. Freeman. Discovering object categories in image collections. In *Proc of IEEE International Conference on Computer Vision (ICCV'05)*, 2005. 2
- [26] H. Wang, M. Ullah, A. Klaser, I. Laptev, and C. Schmid. Evaluation of Local Spatio-Temporal Features for Action Recognition. In *Proc. of BMVA British Machine Vision Conference (BMVC'09)*, 2009. 1, 7