Data mining for Action Recognition

Andrew Gilbert Richard Bowden

Centre for Vision Speech and Signal Processing (CVSSP), University of Surrey, Guildford, GU2 7XH, UK

Abstract. In recent years, dense trajectories have shown to be an efficient representation for action recognition and have achieved state-of-the-art results on a variety of increasingly difficult datasets. However, while the features have greatly improved the recognition scores, the training process and machine learning used hasn't in general deviated from the object recognition based SVM approach. This is despite the increase in quantity and complexity of the features used. This paper improves the performance of action recognition through two data mining techniques, APriori association rule mining and Contrast Set Mining. These techniques are ideally suited to action recognition and in particular, dense trajectory features as they can utilise the large amounts of data, to identify far shorter discriminative subsets of features called rules. Experimental results on one of the most challenging datasets, Hollywood2 outperforms the current state-of-the-art.

1 Introduction

Action recognition has been a popular area of research within the computer vision and machine learning communities for a number of years. This is partly due to the huge number of applications that would benefit, given the ability to automatically recognise actions within natural videos. Driving this research has often been the ease of dataset availability, from the earliest Weizmann [1] and KTH [2] datasets, to the current state of the art, the more realistic and difficult HMDB51 [3] and Hollywood2 [4] datasets. These later datasets pose significant challenges to action recognition, for example, background clutter, fast irregular motion, occlusion and viewpoint changes. The identification of an action class is related to many other unsolved high-level visual problems, such as human pose estimation, interaction with objects, and scene context. Furthermore, determining the temporal extent of an action is much more subjective than for a static object and the size of video datasets are considerably higher than those consisting of static images.

Initially, to solve the action recognition problem, the image recognition framework was generalised to videos. This included the extension of many classical image features; 3D-SIFT [5], extended SURF [6] and HOG3D [7], Space Time Interest Points (STIPs) [8], and more recently dense trajectories [9]. Similarly the classification pipelines applied to single frame image recognition were and still are applied to action recognition. This means the extensive use of SVMs [2],

boosting [10] and Multiple Instance Learning (MIL) [11]. While these approaches can provide excellent results for object recognition, it might not be optimal to directly transfer into the temporal domain, for action recognition, without compromise.

We propose standard learning approaches by data mining techniques which are especially suited for use with densely sampled features. We argue that due to the fact that these dense features are over complete compared to the final solution, mining can efficiently identify the small subset of distinctive and descriptive features that provide the greatest description of the data. In this work we propose the use of Contrast Set Mining as it is able to provide improved results over APriori association rule mining with a lower computational cost.

The rest of the paper is organized as follows. In section 2, we introduce related work in action recognition and data mining. While in Section 3, we detail the APriori and Contrast Set Mining methods. The experimental setup and evaluation protocols are explained in section 4 and experimental results in section 5.

2 Related Work

Related to this work, there are two main areas of relevant work within the computer vision community; action recognition and data mining. The action recognition field is an active area, including research on the features used [5, 7, 6] and methods for spatially and temporal encoding features [12–15]. As the datasets have become more realistic, additional modelling of the videos has become a recent important area of research. For example using Optical flow, Uemura [16] estimates the dominant camera motion, while Park [17] performs simple optical flow based camera stabilisation to remove both the camera and object motion. Similarly Wang [9] uses the optical flow in conjunction with SURF features to compensate for the camera motion. While Hoai [18] performed segmentation on the actions to increase accuracy before classification. The context of the video can also provide information [19, 4], learning relationships between objects in the scene and the scene itself to provide additional cues. In addition, the encoding of the features has been improved by moving away from the standard bag of words towards fisher vector encoding as employed by Oneata [20]. In our work, we concentrate on the learning method, instead of the often used SVM [2] or MIL [11] frameworks, we investigate a data mining based learning technique.

Data mining is a feature selection process increasingly used in computer vision, as the efficiency benefits with increasingly large amounts of data become more marked, where the aim is to generate a higher level super set of features. Yuan [21] mined visual features to generate a high level visual lexicon for object recognition while work by Newozin [22] learnt a temporal based sequential representation of the features to encode the temporal order of features for action recognition. Within image recognition, the spatial encoding of SIFT features by Quack [13] was be learnt through APriori data mining, while the hierarchical encoding of simple corner features were mined by Gilbert [12] to perform action

recognition. More recent work on negative mining to find the *non* frequently occurring rules in images [23] has shown promise in learning the differences between classes. Finally, the work by Wang [24] uses a form of APriori data mining for action recognition to efficiently evaluate their motion features called phrases, leveraging APriori's ability to efficiently mine the large feature space. This paper continues the research into data mining techniques by proposing contrast set learning for action recognition.

3 Data mining

In order to provide scalable solutions to learning from large datasets we propose to adapt text mining techniques. APriori data mining and Contrast Set Mining ignore noise or infrequent features and instead identify frequently reoccurring unique and discriminative subsets of the data. Term Frequency Inverse Document Frequency (TF-IDF) [25] is another popular numerical statistic providing a measure of the importance of a feature (word) to a document or class compared to the rest of the data. However, contrast set mining and APriori have efficient methods to generate the key feature rules, especially contrast set mining which is designed to identify rules that provide the maximum class separation.

3.1 APriori Association Rules Mining

One of the most popular data mining approaches, originally proposed by Agrawal and SriKant [26] is APriori, its aim is to find frequently occurring sets of features or items in the form of association rules. An association rule is a relationship between a number of items that frequently occur within the data. For example, a discovered rule might be, given the items A, B and C, people who buy the items A and B are very likely to purchase item C at the same time. This can then be written as an association rule in the form $\{A, B\} \Rightarrow C$. To assess the quality of a possible association rule, two measures are computed, the support and confidence.

Support Given transaction T, which consists of a number of encoded features or items, a, a database of all the transactions, DB, the complete feature or item vocabulary is I, where $a \subset I$. The support $s(a)$ of a specific set of items, measures the statistical significance of the proposed rule. The support is defined in equation 1

$$s(a) = \frac{|\{T \mid T \in DB, a \subseteq T\}|}{|DB|} \tag{1}$$

A frequently occurring set of items is defined as a set for which $s(a) \geq \sigma > 0$, for a user defined σ, or support threshold. The support threshold σ, is used to filter the large set of transaction vectors to remove the insignificant rules that rarely occur.

Finding sets of items that frequently occur is not trivial because of its combinatorial explosion. It is characterized as a level-wise complete search algorithm

using anti-monotonicity of the set of items, i.e. if a set of items is not frequent, any of its supersets will also never be frequent. In order to discover the rules of frequent items, APriori first scans the database of all transactions and searches for frequent sets of items of size $k = 1$ by accumulating the count for each item and collecting those that satisfy the minimum support requirement. Then given the set of frequent items of size k and the possible rules r_k, it iterates on the following five steps and extracts all the frequent sets of items.

1. Increment k by 1.
2. Generate r_k candidates of frequent sets of items of size k, from the frequent set of items of size r_{k-1}.
3. Compute the support for each candidate of the frequent set of items of size k.
4. Remove the candidates that do not satisfy the minimum support requirements.
5. Repeat steps until no further possible candidates exist of size k.

While the support can be used to find frequent set of items, these features could occur across multiple classes or concepts and therefore would not provide discriminative information against other classes. To solve this, the evaluation of a possible rules is extended to measure a confidence of the rule.

Confidence Measure The confidence is key for identifying the discriminative sets of items that occur in a single class, providing a measure of how discriminative a rule is. The confidence of a frequent item a with respect to a class label α is equivalent to $P(\alpha|a)$. $P(\alpha|a)$ will be large only if a occurs frequently in transactions containing the specific class label α but infrequently in the other class labels. If a occurs frequently in multiple concepts, then $P(\alpha|a)$ will remain small as the denominator in the conditional probability will be large. It is defined as

$$K(a \Rightarrow \alpha) = \frac{s(a \cap \alpha)}{s(a)} \tag{2}$$

Through the association rule generation process outlined above, each rule is measured with respect to both a minimum support and confidence threshold. The confidence threshold γ is set high at 70%, to ensure rules related to a single class are discriminative with respect to others.

3.2 Contrast Set Mining

APriori achieves excellent performance, however, in situations with many frequent sets of item, large sets of items, or very low minimum support, it suffers from the cost of generating a huge number of candidate sets and scanning the database repeatedly to check candidate items. The simple pruning strategy means that in the worst case it may be necessary to generate 2^{20} candidate items to obtain frequent sets of items of size $k = 20$. To reduce this figure, higher support thresholds can be used, but, this limits the search space of the approach

possibly missing significant rules. Instead, Contrast Set Mining [27, 28] is based on sub sampling the transactions multiple times. It aims to identify the meaningful differences between separate classes by reverse engineering the key predictors that identify each class. To illustrate the key principle behind Contrast Set Mining, the example in Table 1 shows 4 people's supermarket transactions. The

Burger	Chips	Foie Gras	Wine	Purpose
1	1	0	0	Family Meal
1	1	0	0	Family Meal
0	0	1	1	Anniversary
1	1	0	0	Family Meal

Table 1. Supermarket Purchases

goal would be to learn that people who bought burgers and chips were *having* a family meal. APriori Association rule mining from the section above would learn that people who buy burgers and chips are *likely* to have a family meal. However Contrast Set Mining would identify that the main *difference* or *contrast* between people shopping for a family meal, compared to an anniversary, is that people buying for a family meal buy burgers and chips and *don't* buy Foie Gras and Wine. This distinction is useful for larger datasets or ones with low inter class variation, as it focuses on the discriminate information and not just the frequent. Therefore instead of modelling all the data, it identifies rules that can provide the most impact or change on the dataset and this generally results in a simple set of rules for each class that are both distinctive and descriptive. To measure the quality of possible rules, two concepts; *lift* and *support* are examined.

Lift The *lift* of a rule is a measure of how the class distribution of the training data shifts in response to the use of the rule compared to the baseline class distribution. It will seek the smallest set of rules that induce the largest shifts. Given a set of transactions, T, which contain a number items a. The transactions will be labelled with a specific class α from the training data $\{T_1, T_2, ...\} \longrightarrow \alpha_1$ etc., where $C = \{\alpha_1, \alpha_2, ..., \alpha_A\}$. Within the transaction database, the frequency of the transactions attributed to a specific class is given by $\{F_1, F_2, ..., F_C\}$ and is used as a normalisation factor. The overall aim is to identify the short subsets of items or rules r that provide the greatest improvement in the overall class distribution. To achieve this, the frequency of the rule r occurring within each class α is computed f_α^r. Ideally the rule will have a high frequency of occurrence in the positive class and low occurrence elsewhere and this will provide maximal lift. The lift is defined as

$$lift(r) = \sum_{\alpha=0}^{C} \frac{f_\alpha^r}{U_\alpha F_\alpha} \tag{3}$$

where C denotes the set of class labels, and U_α is a class weight, where

$$U_\alpha = \begin{cases} 1 & \alpha = Positive \\ 0.1 & otherwise \end{cases} \tag{4}$$

If the lift is greater than 1, the rule is improving the input or baseline distribution of the classes, i.e. the rule is more frequent in the positive class and less so in the negative classes. Ideally the rule will have a large lift, and this can achieved by making it more specific. However, the more specific the rule is, the greater the amount of the data it filters out. This can cause over fitting, causing the unwelcome property of the rule only occurring in a very small selection of the positive examples. Therefore an additional measure related to the frequency of the rule within the data is computed.

Minimum Best Support It is problematic to rely on the lift of a rule alone, incorrect or noisy items within the data, may results in an overfitted rule set. Such an overfitted rule may have a high lift score, but will not accurately reflect the positive data. In order to avoid overfitting, the approach uses a threshold called *minimum best support*, to reject rules below the minimum support. This is the percentage of transactions supporting the rule, it is the ratio of the frequency of occurrence of the rule within the positive class, with respect to the frequency of the occurrence of the rule in the rest of the dataset as shown in equation 5

$$support(r) = \frac{f^r_{\alpha_p}}{\sum_{i=0}^{C} f^r_i} where\ i \neq \alpha_p \tag{5}$$

where α_p is the positive class label.

The lift and support assesses the effectiveness of a rule, but the possible candidate rules need to be generated. A naive approach would test all possible subset item combinations in a similar fashion to the APriori rule generation. However, this is wasteful and increasingly infeasible as the complexity of data increases. Therefore we use a weighted random sampling strategy to combine the best single rules together and reduce training time.

Rule formation To learn the class rules, initially a random subset of individual features are selected, and the lift is calculated for each. These lift scores are then sorted and converted into a cumulative probability distribution as shown in figure 1. Contrast Set Mining then randomly selects K concatenations of the features, up to a maximum rule size M, generally $M = 5$ to reduce the unnecessary formation of rules that will have too low support. The use of the Cumulative Probability distribution to weight the rule formation, means that single features with a low lift are unlikely to be selected, as if the rule has a low lift it will be ignored. The K concatenations of the rules are then scored and ranked with respect to their lift and support and the process is repeated with a new random subset of individual features. If there are no changes in the top X rules after a defined number of rounds, it terminates and returns the top X

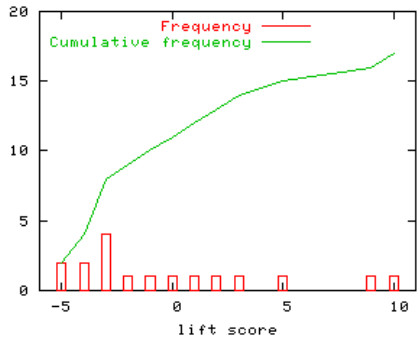

Fig. 1. Cumulative Probability distribution of single attribute lifts

rules. Contrast Set Mining is run independently for each class of the training data, to produce a set of rules for each class $M(\alpha) = \{m(\alpha)_1, m(\alpha)_2, ...m(\alpha)_A\}$.

4 Experimental Setup

In this section, we introduce the features and dataset used and the training process.

4.1 Features and Dataset

For this work we test the approach on what is generally considered to be one of the most challenging but well supported action recognition datasets, Hollywood2 [4]. Hollywood2 was collected from 69 different Hollywood movies and includes 12 action classes. It contains 1,707 videos split into a training set (823 videos) and a test set (884 videos). Importantly the training and test videos come from different movies. To measure the performance, mean average precision (mAP) over all classes, as in [4] is used, with examples of the videos shown in Figure 2.

The features extracted are based on dense trajectory features [9], the feature points on each frame are tracked by median filtering a dense optical flow field. To avoid drift, the trajectories are limited to 15 frames. In addition, feature trajectories that are static are ignored as they provide no motion information. For each trajectory, we compute and concatenate several descriptors; the Trajectory, HOG, HOF and MBH. The Trajectory descriptor is a concatenation of normalized displacement vectors. While the other descriptors are computed in the space-time volume aligned with the trajectory. HOG captures the static appearance information and is based on the orientation of image gradients. While both HOF and MBH measure motion information, and are based on optical flow. HOF directly quantizes the orientation of flow vectors, while MBH splits the optical flow into horizontal and vertical components, and quantizes the derivatives

Fig. 2. Examples from the *Hollywood2* dataset [4]

of each component. The final dimensions of the descriptors are 30 for Trajectory, 96 for HOG, 108 for HOF and 192 for MBH, giving a base feature size of 426. We then train a 4000 element codebook using 100,000 randomly sampled feature descriptors with k-means.

4.2 Training

Given the 4000 element codebook, all the trajectories detected within a given video are assigned to their closest neighbour and this results in a frequency count of specific codebook detections for a video. These frequency counts are then symbolised to allow the application of mining.

Given a feature or item vocabulary containing $|\mathbf{I}|$ items or features, where $\mathbf{I} = \{A, B, C\}$, and T_i is a transaction vector of the codebook frequency response of the input, i, with two example input transactions, $\mathbf{T}_1 = \{3, 0, 1\}$ $\mathbf{T}_2 = \{1, 3, 2\}$.

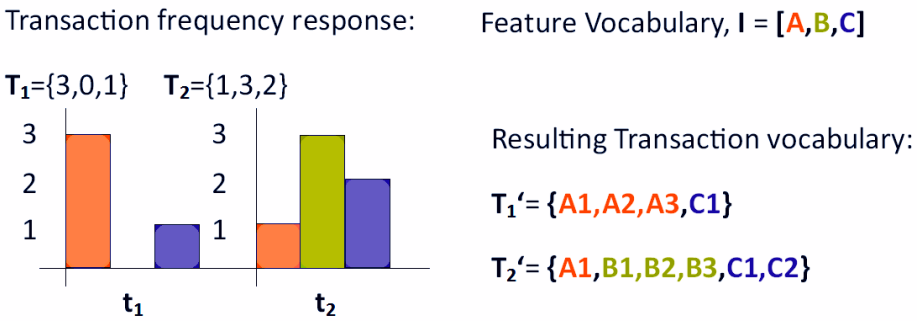

Fig. 3. The symbolisation of the codebook detections

As shown in Figure 3, in order to convert the feature frequency response into unique symbols for data mining, the frequency of each element in $\mathbf{T}_i$ is used to form the same number of new but unique symbols as the value of the frequency. Therefore, in the example above, the transactions become, $\mathbf{T}_1^{'} = \{A1, A2, A3, C1\}$ $\mathbf{T}_2^{'} = \{A1, B1, B2, B3, C1, C2\}$.

In order to classify test videos, the rules for each class need to be mined through the two data mining techniques we propose. For each video, the codebook frequency response is symbolised into a transaction vector, and appended with the relevant class label α, and this is repeated for each video, to form a database of transactions, to be used as the input to the data mining. Generally there is between 10,500 and 20,100 unique items in each transaction each representing a video sequence, many of these items repeating both inter and intra class. Then, for each class the database is mined with respect to the class, to produce a set of rules for each class $M(\alpha) = \{m(\alpha)_1, m(\alpha)_2, ...m(\alpha)_A\}$.

4.3 Classification

Both data mining techniques generally produce concise rules for each class label, therefore the top rules can be formatted into class specific lookup tables for classification. To classify, the codebook response of a test video is found and symbolised to form a test transaction and each rule in the lookup table is compared to the transaction. The response score of the classifier R for a test transaction T_i with respect to a specific class label α is given by

$$R(T_i, \alpha) = \frac{1}{A} \sum_{j=0}^{M(\alpha)_A} \frac{1}{|M(\alpha)_j|} m(T_i, M(\alpha)_j) \tag{6}$$

where

$$m(T_i, M(\alpha)_j) = \begin{cases} 1 & T_i \in M(\alpha)_j \\ 0 & otherwise \end{cases} \tag{7}$$

This response score is computed over all class labels, and the maximum response taken as the classification label.

5 Experimental Results

In the results section, initially we compare the stability of the user specified support threshold used in the mining techniques, and computational cost. This is followed by a comparison of the approaches to the current state of the art.

5.1 Stability of Thresholds

Within both APriori and Contrast Set Mining, the support value is used to filter out rules that don't represent the data class. Table 2 shows how the mAP

Table 2. Comparison of the performance of APriori and Contrast Set Mining with varied support thresholds

	APriori		Contrast Sets	
Support Value	mAP (%)	Train Time (mins)	mAP (%)	Train Time (mins)
0.01	65.1	940	65.2	120
0.05	65.1	495	65.2	125
0.1	60.1	475	65.4	129
0.2	39.7	260	65.5	127
0.3	22.1	190	62.4	135

and training computation time on the Hollywood2 dataset varies for a range of specified support values.

It can be seen that in general, the performance for the APriori is dependent on the support value threshold specified, this is because, as the support value is increased more of the data is filtered out and, the quality of the rules is reduced. This is why the training time decreases from over 15 hours to 3 hours between the support value of 0.01 and 0.3. In comparison, in Contrast Set Mining both the performance and training time is more constant. This is due to the weighted random sampling technique used to form the rules, which means it is far less dependent on the value of the minimum support.

5.2 Evaluation of Action Learning Framework

In order to compare the use of the APriori and Contrast Set Mining with other current state-of-the-art approaches, the standard train and test subsets of the Hollywood2 dataset as provided by [4] was used. For the APriori mining, the support value was set as $\sigma = 0.1$, as this provide the highest results, when using the training data. While for the Contrast Set Mining, $\sigma = 0.2$. The results are presented in Table 3.

Table 3. Comparison of the Active learning on the of Hollywood2 dataset

Approach	mAP (%)
Mathe [29]	61.0
Jain [30]	62.5
Wang Baseline [24]	60.1
Wang [24]	64.3
APriori Association Rule Mining	65.1
Contrast Set Mining	65.4

The results show that the use of a data mining technique to classify action recognition is able to improve on current Sate-of-the-art by around 1 % compared

to the most recent results reported in the literature [24]. An illustration of the compact nature of the mined rules can be seen in Figure 4, it shows the location of the matched Contrast Set Mining features for successfully classified video.

The baseline result by Wang [9], is interesting as it is using the same features and a standard SVM classifier to give a perfromance of 60.1%, but with additional processing to remove camera motion and to learn a human detector was able to boost their baseline performance by around 4% to 64.3%. These processing techniques could be added to our approach and therefore further improve the performance. The Contrast Set Mining is able to exceed the high performance of the APriori, but with a significant reduced training time, taking only 2 hours to train, compared to the 8 for the APriori. Furthermore as shown previously in table 2, the performance of the Contrast set Mining is not affected by the change in a support value of the learnt rules. An additional feature related to the rules, that is unusual compared to many other state-of-the-art approaches in the field of action recognition, is the size of the rules that are learnt. Typically the mined rules are short, the rules mined using APriori had a median length of 7 items, while in the case of Contrast Sets Mining they are at most a combination of 5 individual codebook elements. They have been identified within the mining process to provide the greatest contrast against the other classes. The compact learnt model allows for fast test time operation as well, as at run time, both data mining approaches are fast, requiring around 15 minutes to classify all 884 test videos, excluding the dense trajectory feature extraction.

Figure 4, shows where discriminative features fire for each class. The end coordinate of the trajectories are shown. They are generally sparse, and the mining has identified the features of the highest contrast with respect to other classes. In summary, what the rules capture are the combinations of features important to a class. If you were to treat each rule as a single classifier, this would form a very weak classifier that always fires for a certain combination of visual words. However, they often fire at points within the video sequence that are most indicative of the action, for example on the hand shake itself for the action *Hand Shake*, or the car door for the action *Get out of Car*. This high localization of the features, could be extended in future to segment the action within a longer video sequence.

6 Conclusions

This paper is able to improve on the standard dense trajectory features and SVM learning pipeline, through the inclusion of an improved training technique. We demonstrate that through using Contrast Set Mining, performance can be significantly improved on the state-of-the-art. The use of a weighted randomly sampling strategy allows for a reduction in training time and a stabilisation of the user defined minimum support thresholds. An evaluation on the current state-of-the-art action recognition dataset, Hollywood2, demonstrates the approaches effectiveness.

Fig. 4. Successfully classified feature locations in Hollwyood2 videos

Acknowledgement: This work was supported by the EPSRC grant "Learning to Recognise Dynamic Visual Content from Broadcast Footage" (EP/I011811/1).

References

1. Blank, M., Gorelick, L., Shechtman, E., Irani, M., Basri, R.: Actions as space-time shapes. In: ICCV'05. (2005) 1395–1402
2. Schuldt, C., Laptev, I., Caputo, B.: Recognizing Human Actions: a Local SVM Approach. In ICPR'04 (2004) 32–36
3. Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., Serre, T.: HMDB: a large video database for human motion recognition. In: ICCV'11. (2011)
4. Marszalek, M., Laptev, I., Schmid, C.: Actions in Context. In: CVPR'09 (2009)
5. Scovanner, P., Ali, S., Shah, M.: "A 3-dimensional Sift Descriptor and its Application to Action Recognition". In: Proc. of MULTIMEDIA '07. (2007) 357–360
6. Willems, G., Tuytelaars, T., Gool, L.: An Efficient Dense and Scale-Invariant Spatio-Temporal Interest Point Detector. In: ECCV'08 (2008) 650–663
7. Klaser, A., Marszalek, M., Schmid, C.: A Spatio-Temporal Descriptor based on 3D Gradients. In: BMVC'08 (2008)
8. Laptev, I., Lindeberg, T.: "Space-time Interest Points". In: ICCV'03 (2003) 432–439
9. Wang, H., Kläser, A., Schmid, C., Liu, C.L.: Dense trajectories and motion boundary descriptors for action recognition. In: International Journal of Computer Vision. Volume 103., Springer (2013) 60–79
10. Viola, P., Jones, M.: "Rapid Object Detection using a Boosted Cascade of Simple Features". In: CVPR'01 (2001) 511–518
11. Maron, O., Lozano-Pérez, T.: A framework for multiple-instance learning. In: Advances in neural information processing systems. (1998) 570–576
12. Gilbert, A., Illingworth, J., Bowden, R.: Action recognition usingmined hierarchical compound features. in: IEEE Transactions on Pattern Analysis and Machine Intelligence (2011) 883 – 897
13. Quack, T., Ferrari, V., Leibe, B., Gool, L.: "Efficient Mining of Frequent and Distinctive Feature Configurations". In: ICCV'07 (2007)
14. Laptev, I., Marszalek, M., Schmid, C., Rozenfeld, B.: "learning Realistic Human Actions from Movies". In: CVPR'08 (2008) 1–8
15. Niebles, J.C., Wang, H., Fei-Fei, L.: Unsupervised learning of human action categories using spatial-temporal words. In: International journal of computer vision **79** (2008) 299–318
16. Uemura, H., Ishikawa, S., Mikolajczyk, K.: "Feature Tracking and Motion Compensation for Action Recognition". In: BMVC'08 (2008)
17. Park, D., Zitnick, C.L., Ramanan, D., Dollár, P.: Exploring weak stabilization for motion feature extraction. In: CVPR'13 (2013) 2882–2889
18. Hoai, M., Lan, Z.Z., De la Torre, F.: Joint segmentation and classification of human actions in video. In: CVPR 2011 (2011) 3265–3272
19. Han, D., Bo, L., Sminchisescu, C.: Selection and Context for Action Recognition. In ICCV'09 (2009) 1933–1940
20. Oneata, D., Verbeek, J., Schmid, C.: Action and event recognition with fisher vectors on a compact feature set. In: ICCV 2013 (2013) 1817–1824
21. Yuan, J., Wu, Y., Yang, M.: Discovery of collocation patterns: from visual words to visual phrases. In: CVPR'07 (2007) 1–8

22. Nowozin, S., Bakir, G., Tsuda, K.: Discriminative subsequence mining for action classification. In: ICCV'07 (2007) 1–8
23. Siva, P., Russell, C., Xiang, T.: In defence of negative mining for annotating weakly labelled data. In: ECCV'12 (2012) 594–608
24. Wang, L., Qiao, Y., Tang, X.: Mining motion atoms and phrases for complex action recognition. In: ICCV'13 (2013) 2680–2687
25. Sparck Jones, K.: A statistical interpretation of term specificity and its application in retrieval. In Journal of Documentation **28** (1972)
26. Agrawal, R., Srikant, R.: "Fast Algorithms for Mining Association Rules in Large Databases". In: VLDB'94, Proceedings of 20th International Conference on Very Large Data Bases. (1994) 487–499
27. Menzies, T., Hu, Y.: Data mining for very busy people. In: Computer. Volume 36. (2003) 22–29
28. Bay, S.D., Pazzani, M.J.: Detecting change in categorical data: Mining contrast sets. In: KDD. (1999) 302–306
29. Mathe, S., Sminchisescu, C.: Dynamic eye movement datasets and learnt saliency models for visual action recognition. In: ECCV'12 (2012) 842–856
30. Jain, M., Jégou, H., Bouthemy, P.: Better exploiting motion for better action recognition. In: CVPR'13 (2013) 2555–2562