There is more than one way to get out of a car: Automatic Mode Finding for Action Recognition in the Wild

Olusegun Oshin*, Andrew Gilbert, and Richard Bowden

Centre for Vision, Speech and Signal Processing,
University of Surrey, Guildford, Surrey,
United Kingdom GU2 7XH
`{o.oshin,a.gilbert,r.bowden}@surrey.ac.uk`

Abstract. “Actions in the wild” is the term given to examples of human motion that are performed in natural settings, such as those harvested from movies [10] or the Internet [9]. State-of-the-art approaches in this domain are orders of magnitude lower than in more contrived settings. One of the primary reasons being the huge variability within each action class. We propose to tackle recognition in the wild by automatically breaking complex action categories into multiple modes/group, and training a separate classifier for each mode. This is achieved using RANSAC which identifies and separates the modes while rejecting outliers. We employ a novel reweighting scheme within the RANSAC procedure to iteratively reweight training examples, ensuring their inclusion in the final classification model. Our results demonstrate the validity of the approach, and for classes which exhibit multi-modality, we achieve in excess of double the performance over approaches that assume single modality.

1 Introduction

Human action recognition from video has gained significant attention in the field of Computer Vision. The ability to automatically recognise actions is important because of potential applications in video indexing and search, activity monitoring for surveillance, and assisted living purposes. The task is especially challenging due to variations in factors pertaining to video set-up and execution of the actions. These include illumination, scale, camera motion, viewpoint, background, occlusion, action length, subject appearance and style.

Approaches to action recognition attempt to learn generalisation over all class examples from training, making use of combinations of features that capture both shape and motion information. While this has resulted in excellent results for videos with limited variation, in natural settings, the variations in camera set-up and action execution are much more significant, as can be seen in Figure 1. It is,

* This work is supported by the EU FP7 Project Dicta-Sign (FP7/2007-2013) under grant agreement no 231135, and the EPSRC project Making Sense (EP/H023135/1).

therefore, unrealistic to assume that all aspects of variability can be modelled by a single classifier. This motivates our approach.

(a) *GetOutCar* action

(b) *HandShake* action

Fig. 1. Four examples of two actions of the Hollywood2 dataset, all showing the different modes of the same action.

The method presented in this paper tackles action recognition in complex natural videos by following a different approach. Instead of treating all examples of a semantic action category as one class, we automatically separate action categories into various *modes* or groups, thereby significantly simplifying the training and classification task. We achieve this by applying the tried and tested Random Sampling Consensus (RANSAC) algorithm [2] to training examples of actions, with a novel adaptation based on an iterative reweighting scheme inspired by boosting, and obtain impressive results. Whereas clustering merely groups class examples based on proximity within the input space, our approach groups the positive examples while attempting to exclude negative ones. This ensures less contamination within sub-categories compared to clustering. To our knowledge, our approach is the first to make use of the automatic separation of complex category examples into groups for action recognition. For classes where multi-modality is evident, we achieve a performance increase in excess of 100% over approaches assuming single modality.

The layout for the remainder of this paper is as follows: Section 2 discusses related research. In Section 3, we present our approach in detail. We describe

our experimental set-up in Section 4 and present recognition results in Section 5. Finally, Section 6 concludes the paper.

2 Related Work

There is a considerable body of work exploring the recognition of actions in video [11, 1, 6, 8, 4, 13]. While earlier action recognition methods were tested on simulated actions in simplified settings, more recent work has shifted focus to so-called Videos in the Wild, *e.g.* personal video collections available online, and movies. As a result of this increase in complexity, recent approaches attempt to model actions by making use of combinations of feature types. Laptev and Perez [7], distinguish between actions of Smoking and Drinking in movies, combining an optical flow-based classifier with a separately learned space-time classifier applied to a keyframe of the action. The works of [6] and [10] recognise a wider range of actions in movies using concatenated HoG and HoF descriptors in a bag-of-features model, with [10] including static appearance to learn contextual information. Han *et al.* [5] capture scene context by employing object detectors and introduce bag-of-detectors, encoding the structural relationships between object parts, whereas Ullah *et al.* [13] combine non-local cues of person detection, motion-based segmentation, static action detection, and object detection with local features. Liu *et al.* [8] also combine local motion and static features and recognise actions in videos obtained from the web and personal video collections.

In contrast to these multiple-feature approaches, our method makes use of one feature type. Then, instead of seeking to learn generalisation over all class examples, we argue that a single action can be split into subsets, which cover the variability of action, environment and viewpoint. For example, the action of *Getting Out of a Car* can be broken into sets of radically different actions depending on the placement of the camera with respect to the car and individual. We automatically discover these modes of action execution or video set-up, thereby simplifying the classification task. While extensive work exist on local classification methods for object category recognition [15], human pose estimation [14], etc [12], the assumption of multimodality has not so far been applied to action recognition. We employ RANSAC [2] for this grouping and introduce a reweighting scheme that increases the importance of difficult examples to ensure their inclusion in a mode.

3 Action Modes

The aim of this work is to automatically group training examples in natural action videos, where significant variations occur, into sub-categories for improved classification performance. The resulting sub-categories signify different modes of an action class, which when treated separately, allow for better modelling of training examples, as less variations exist within each mode.

To illustrate, Figure 2(a) shows positive and negative examples in a simple binary classification problem. It can be observed that, using a Gaussian classifier,

Fig. 2. Binary classification problem for classes with overlap due to large variability in the data

there exists a great deal of overlap between the classes. This is as a result of both positive and negative examples occupying regions within the classification space that make separation impossible with a single classifier.

While this is an obvious problem in classification that could be solved using a mixture model, for the task of action recognition in natural videos, this phenomenon is still observed, yet mostly ignored. Figure 1 shows six different modes of the action class *GetOutCar*, and four modes of the action category *HandShake*, respectively, taken from the Hollywood2 dataset [10]. It can be seen that, while the same action is being performed, all the examples appear radically different due to the differences in camera setup and in some cases, action execution. Despite these variations, the examples are given one semantic label, making automatic classification extremely difficult. We propose that, for cases such as this, it should be assumed that the data is multi-modal, and the use of single classifiers for such problems is unrealistic.

Our method is based on the notion that there exists more compact groupings within the classification space that when identified, reduces confusion between classes. Figure 2(b) shows examples of such groupings when applied to the binary classification problem. The grouping also enables the detection of outliers, which are noisy examples that may prove detrimental to the overall classifier performance, as can be observed by the single ungrouped positive example in Figure 2(b).

3.1 Automatic Grouping using Random Sampling Consensus

For a set, Φ of training examples belonging to a particular class C , we iteratively select a random subset, $\varphi \subset \Phi$ of the examples. We then train a binary classifier of the subset φ against all training examples from other classes. This forms the hypothesis stage. The resulting model is then evaluated on the remainder of the training example set, $\psi \subset \Phi$, where $\Phi = \varphi \cup \psi$. For each iteration, $t = \{1 \dots T\}$,

a consensus set is obtained, labelled Group ς_t , which is made up of φ and the correctly classified examples from ψ . This procedure identifies examples in the subset ψ where the mode is similar to examples in φ .

3.2 Sub-category Selection

After several iterations of this random training and evaluation, the sub-categories $\mathcal{S}_j, j = \{1 \dots J\}$ of the class C are selected from the groups $\varsigma_t, t = \{1 \dots T\}$, where T is the number of iterations, and J is the number of sub-categories. We apply AdaBoost [3] in the selection process in an attempt to ensure that all training examples are represented in at least one of the sub-categories. Each group, ς_t , is given a score which is the sum of weights $W_t(i)$ associated with each example i in the group. The process is initialised by assigning equal weights, $W_1(i) = \frac{1}{|\Phi|}$ to all training examples. Hence, in the first instance, we find the group that results in the highest number of correctly classified examples in subset ψ , labelled $\mathcal{S}_1$. This is the first sub-category.

For subsequent sub-categories, the weight of each example is given by

$$W_{t+1}(i) = \frac{W_t(i)}{Z_t} \exp(-\alpha_t y_i h_t(x_i)), \quad (1)$$

given that,

$$\alpha_t = \frac{1}{2} \ln\left(\frac{1 - \epsilon_t}{\epsilon_t}\right) \quad (2)$$

and the term $y_i h_t(x_i) = \{-1, +1\}$ denotes the absence or presence of a particular example in the previously selected sub-categories. Z_t is a normalisation constant, and ϵ_t is the error rate. This process is repeated until all examples are selected, or the maximum number of sub-categories is exceeded. In some cases outliers are discovered. Also, there is often overlap of examples between the sub-categories $\mathcal{S}_j$ as shown in 2(b).

During training, the sub-categories are trained separately against examples of other classes. Examples of class C that do not belong to the sub-category being trained are not included in the training. During classification, results of all sub-categories $\mathcal{S}_j$ are combined, with all true positives of $\mathcal{S}$ counted as belonging to one class C . The computational complexity of our approach is $O(T(\mathcal{C}_{hypo}(|\varphi|) + |\psi|\mathcal{C}_{test}) + J\mathcal{C}_{boost}(|\Phi|))$, where $\mathcal{C}_{hypo}$ and $\mathcal{C}_{test}$ are the costs of RANSAC hypothesis and test phases respectively, $\mathcal{C}_{boost}$ is the cost of the reweighting procedure, and $|\cdot|$ denotes cardinality.

4 Experimental Setup

We evaluate our method on the Hollywood2 Human Action dataset [10]. The dataset contains 12 action classes: AnswerPhone, DriveCar, Eat, FightPerson, GetOutCar, HandShake, HugPerson, Kiss, Run, SitDown, SitUp and StandUp. Obtained from 69 different movies, this dataset contains the most challenging

collection of actions, as a result of the variations in action execution and video set-up across the examples. The examples are split into 823 training and 884 test sequences, where training and test sequences are obtained from different movies.

For our experiments, we follow the experimental set-up of Laptev *et al.* [6]. We detect interest points using the spatio-temporal extension of the Harris detector and compute descriptors of the spatio-temporal neighbourhoods of the interest points, using the specified parameters. The descriptor used here is Histogram of Optical Flow (HoF). We cluster a subset of 100,000 interest points into 4000 visual words, using k-means with the Euclidean distance, and represent each video by a histogram of visual word occurrences.

As in [6], we make use of a non-linear support vector machine with a χ^2 kernel given by, $K(H_i, H_j) = \exp(-\frac{1}{2A} \sum_{n=1}^V \frac{(h_{in} - h_{jn})^2}{h_{in} + h_{jn}})$, where V is the vocabulary size, A is the mean distance between all training examples, and $H_i = \{h_{in}\}$ and $H_j = \{h_{jn}\}$ are histograms.

Performance is evaluated as suggested in [10]: Classification is treated as a number of one-vs-rest binary problems. The value of the classifier decision is used as a confidence score with which precision-recall curves are generated. The performance of each binary classifier is thus evaluated by the average precision. Overall performance is obtained by computing the mean Average Precision (mAP) over the binary problems.

For our RANSAC implementation, the size of the training subset, φ is chosen as one-fifth of the number of training examples, Φ . We set the number of RANSAC iterations $T = 500$, and train T *one-vs-rest* binary classifiers using φ_t against all other class examples. As detailed, reweighting is employed to select the N most inclusive groups.

Having trained using the more compact sub-categories, during testing, we obtain confidence scores from all sub-category binary classifiers for each test example. In order to obtain average precision values which combine results of multiple sub-categories within a class, we normalise the scores, such that the values are distributed over a range of $[0, 1]$, and make use of a single threshold across the multiple sub-category scores within that range. Precision-Recall curves which combine the results of the sub-categories are generated by varying this single threshold, and using the logical-OR operator across sub-categories, on the label given to each test example. In particular, for each increment of the threshold, positives, from which precision and recall values are obtained, are counted for the class if *any* one of its sub-category scores is above the threshold. Hence, a classification for a sub-category within a class is a classification for that class.

5 Results

Table 1 shows average precision obtained for each class using our method, compared with the results of Marszalek *et al.* [10]. The table shows average precision obtained using number of sub-categories $J = \{1...7\}$, and highlights the optimal

value of J for each class. The table also shows the improvement obtained over [10] by splitting examples into sub-categories.

Action	HoG/ HoF[10]	HoF + Mode Selection (Our Method)							
		Number of Sub-categories, J							Best (#Groups)
		1	2	3	4	5	6	7	
AnswerPhone	0.088	0.086	0.130	0.144	0.153	0.165	0.162	0.152	0.165 (5)
DriveCar	0.749	0.835	0.801	0.681	0.676	0.676	0.643	0.643	0.835 (1)
Eat	0.263	0.596	0.599	0.552	0.552	0.552	0.525	0.525	0.599 (2)
FightPerson	0.675	0.641	0.509	0.545	0.551	0.551	0.549	0.549	0.641 (1)
GetOutCar	0.090	0.103	0.132	0.156	0.172	0.184	0.223	0.238	0.238 (7)
HandShake	0.116	0.182	0.182	0.111	0.092	0.190	0.190	0.111	0.190 (5)
HugPerson	0.135	0.206	0.217	0.143	0.129	0.134	0.134	0.120	0.217 (2)
Kiss	0.496	0.328	0.263	0.239	0.253	0.263	0.101	0.091	0.328 (1)
Run	0.537	0.666	0.255	0.267	0.267	0.269	0.267	0.241	0.666 (1)
SitDown	0.316	0.428	0.292	0.309	0.310	0.239	0.255	0.254	0.428 (1)
SitUp	0.072	0.082	0.170	0.135	0.134	0.124	0.112	0.099	0.170 (2)
StandUp	0.350	0.409	0.342	0.351	0.295	0.324	0.353	0.353	0.409 (1)
Mean	0.324								0.407

Table 1. Average Precision on the Hollywood2 dataset.

It can be seen that while six of the classes appear to be more uni-modal in their execution or setup, the remaining six benefit from the discovery of additional modes, with the actions *Eat*, *HugPerson* and *SitUp* showing best results with two modes, and *HandShake* and *GetOutCar* giving best performance with 5 and 7 modes, respectively.

It should be noted that, where the number of sub-categories $J = 1$, the method simply reduces to outlier detection. In this case, examples not belonging to the largest consensus set are treated as noisy examples. As with categories which exhibit multi-modality, seeking generalisation over these examples may prove detrimental to the overall classifier performance. They are therefore discarded.

It can be observed that the improvements in performance are made for the worst performing classes without grouping. In the case of *AnswerPhone*, *GetOutCar* and *SitUp*, more than 100% improvement is observed. This shows that the low performance is due to the multi-modal nature of the examples in these classes, which is ignored without the grouping procedure. Discovering these modes and training them separately results in better performance. Conversely, breaking down of classes which performed well without grouping resulted in reduction in performance in most cases. This suggests that, in these cases, most of the actions are uni-modal. We obtain a mean average precision of 0.407 having discovered the modes of the actions, compared to 0.324 obtained in [10].

6 Conclusion

We present an approach to improving the recognition of actions in natural videos. We argue that treating all examples of a semantic action category as one class is often not optimal, and show that, in some cases, gains in performance can be achieved by identifying various modes of action execution or camera set-up. We make use of RANSAC for this grouping, but add a boosting-inspired reweighting procedure for the selection of optimal groups. Our results show that, for poorly performing classes, when different modes are trained separately, classification accuracy is improved. This is attributed to the learning of multiple classifiers on smaller, better-defined sub-categories within each of the classes. Our approach is generic, and can be used in conjunction with existing action recognition methods, and complex datasets. Future work will include finding the optimal number of modes for each action category.

References

1. Dollár, P., Rabaud, V., Cottrell, G., Belongie, S.: Behavior recognition via sparse spatio-temporal features. In: VS-PETS (2005)
2. Fischler, M.A., Bolles, R.C.: Ransac: A paradigm for model fitting with applications to image analysis and automated cartography. *Comms. of the ACM* 24(6), 381–395 (1981)
3. Freund, Y., Schapire, R.E.: A decision-theoretic generalization of on-line learning and an application to boosting. In: EuroCOLT. pp. 23–37 (1995)
4. Gilbert, A., Illingworth, J., Bowden, R.: Action recognition using mined hierarchical compound features. *PAMI 99(PrePrints)* (2010)
5. Han, D., Bo, L., Sminchisescu, C.: Selection and context for action recognition. In: ICCV. pp. 1–8 (2009)
6. Laptev, I., Marszalek, M., Schmid, C., Rozenfeld, B.: Learning realistic human actions from movies. In: CVPR. pp. 1–8 (2008)
7. Laptev, I., Perez, P.: Retrieving actions in movies. In: ICCV. pp. 1–8 (2007)
8. Liu, J., Luo, J., Shah, M.: Recognizing realistic actions from videos ”in the wild”. In: CVPR. pp. 1–8 (2009)
9. Liu, J., Shah, M.: Learning human actions via information maximization. In: CVPR. pp. 1–8 (2008)
10. Marszalek, M., Laptev, I., Schmid, C.: Actions in context. In: CVPR. pp. 1–8 (2009)
11. Schuldt, C., Laptev, I., Caputo, B.: Recognizing human actions: A local SVM approach. In: ICPR. pp. 32–36 (2004)
12. Szepannek, G., Schiffner, J., Wilson, J., Weihs, C.: Local modelling in classification. In: ICDM. pp. 153–164 (2008)
13. Ullah, M.M., Parizi, S., Laptev, I.: Improving bag-of-features action recognition with non-local cues. In: BMVC. pp. 95.1–11 (2010)
14. Urtasun, R., Darrell, T.: Sparse probabilistic regression for activity-independent human pose inference. In: CVPR (2008)
15. Zhang, H., Berg, A.C., Maire, M., Malik, J.: Svm-knn: Discriminative nearest neighbor classification for visual category recognition. In: CVPR. pp. 2126–2136 (2006)