Action Recognition using Randomised Ferns

Olusegun Oshin Andrew Gilbert John Illingworth Richard Bowden
Centre for Vision, Speech and Signal Processing
University of Surrey
Guildford, Surrey
United Kingdom GU2 7XH
{o.oshin,a.gilbert,j.illingworth,r.bowden}@surrey.ac.uk

Abstract

This paper presents a generic method for recognising and localising human actions in video based solely on the distribution of interest points. The use of local interest points has shown promising results in both object and action recognition. While previous methods classify actions based on the appearance and/or motion of these points, we hypothesise that the distribution of interest points alone contains the majority of the discriminatory information. Motivated by its recent success in rapidly detecting 2D interest points, the semi-naive Bayesian classification method of Randomised Ferns is employed. Given a set of interest points within the boundaries of an action, the generic classifier learns the spatial and temporal distributions of those interest points. This is done efficiently by comparing sums of responses of interest points detected within randomly positioned spatio-temporal blocks within the action boundaries. We present results on the largest and most popular human action dataset [20] using a number of interest point detectors, and demostrate that the distribution of interest points alone can perform as well as approaches that rely upon the appearance of the interest points.

1. Introduction

In this paper, we address the problem of efficiently recognising and localising human actions in video. Various factors need to be accounted for in order to create a robust action classification system. These include camera motion, background clutter, occlusion, scale, illumination, appearance and intra-class variations. In order to recognise a particular action, such methods are required to determine, extract and encode characteristics which distinguish one action from another. These characteristics can vary from local spatio-temporal interest points to global optical flow representations.

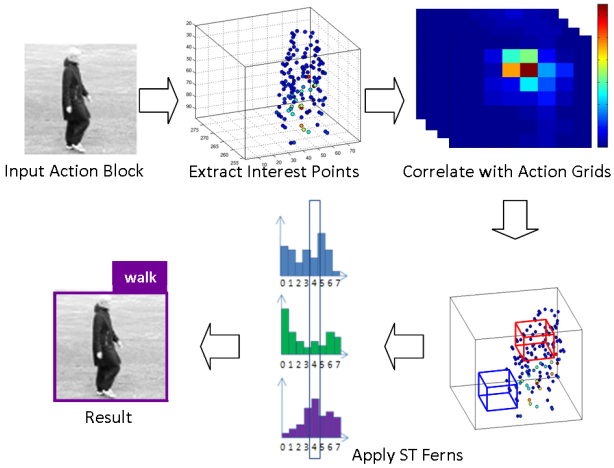

Figure 1. Main components of the action recognition framework.

Actions in video can be represented as a sparse set of spatio-temporal interest points, and the works of [2, 14, 19, 20], have shown that these compact, high information representations, are sufficient for categorisation. Existing methods categorise interest points based on their appearance, using descriptors that best separate a class of interest points from another, *e.g.*, intensity gradients [2, 10] or optical flow [3]. These methods require an additional phase in which the content of the interest points are examined.

In this paper, we propose a method that makes use of interest points in a different way, allowing for a very efficient method of classifying actions. We make exclusive use of the spatio-temporal distribution of interest points to categorise actions, and we extend the Randomised Ferns classifier, proposed by Ozuysal [18] for rapid keypoint recognition in object recognition, and recently used in spatio-temporal interest point Recogntion [17]. Our approach provides a generic framework for classifying actions, that is not dependent on the interest point detection method used.

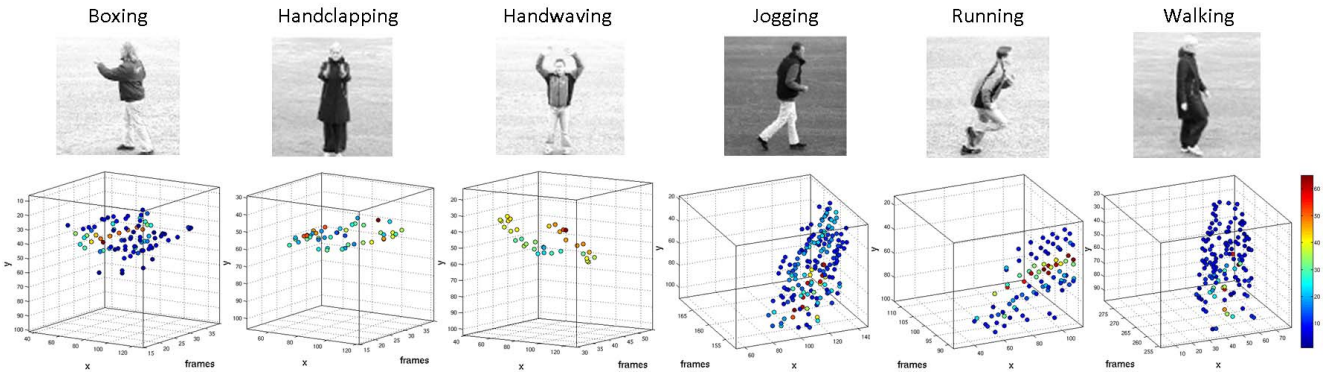

Figure 2. Volumetric representations of detected interest points for examples of the six KTH actions. The top row shows groundtruthed actions and the bottom row shows the corresponding distributions of interest points within groundtruthed Action Cuboids. Response strengths are depicted by colour. It can be seen that the distribution of interest points and their response strengths differ across actions.

Figure 1 shows the main components of our generic action recognition framework.

The key contributions of this paper are, a study of the exclusive use of the distribution of interest points for action recognition; the efficient encoding of the spatial and temporal distribution of interest points; and the extension of Randomised Ferns to action recognition. Our novel action classifier is tested on the KTH human action dataset, which, as a regularly cited test set, provides the perfect test bed with which to test our hypothesis. We demonstrate results using spatio-temporal interest points proposed by Dollar *et al.* [2], Laptev and Lindeberg [10]; and a 2D Harris Corner detector [6] applied to the video in (x, y), (x, t) and (y, t) as proposed by Gilbert *et al.* [5]. Figure 2 shows volumetric representations of six actions in terms of their interest points only. These interest points have been extensively used in action recognition methods [5, 14, 20], which rely heavily upon the appearance of interest points. We postulate that actions can be sufficiently described solely by the spatio-temporal distribution of interest points, and we present comparable results to the state-of-the-art in action recognition, supporting our hypothesis.

The layout for the remainder of this paper is as follows: Section 2 discusses related research. In Section 3, we give an overview of Randomised Ferns and its application in interest point recognition. In Section 4, we present our approach in detail. We describe our experimental setup and present recognition results in Section 5. Finally, Section 6 concludes the paper.

2. Related Work

Actions in video can be described in a number of ways, and there is a considerable body of work exploring these representations. Many action recognition methods make use of a set of sparse local interest points to describe actions, and have demonstrated remarkable performance. Examples of such methods include Laptev and Lindeberg's [10] extension of Harris 2D corners [6], to include corners in time. Laptev obtain a sparse set of spatio-temporal corners, which provide for a condensed representation of actions. Similarly, Oikonomopoulos *et al.* [16] extend the salient feature detection of Kadir and Brady [7]. In contrast to measuring saliency within a circular neighbourhood of pixels, they measure spatio-temporal saliency within a spherical neighbourhood of pixels in a video. Also, Scovanner *et al.* [21] generalise Lowe's SIFT descriptor [13] and create a 3D equivalent by obtaining 3D gradient magnitudes and orientations for each pixel, before construct weighted histograms and descriptors from these histograms as done in [13].

Dollar *et al.* [2] argue that the direct generalisation of spatial interest points to the spatio-temporal domain does not provide for an optimal representation and neglects important information, resulting in very sparse interest points. Dollar applied separable linear filters, which involves convolving the video with a 2D Gaussian smoothing kernel along the spatial dimensions, and applying a pair of 1D Gabor filters along the temporal dimension.

For the above methods, interest points are designed in such a way as to provide invariance to a number of possible transformations, as dictated by the authors. These approaches make several strong assumptions about the actions and interest points. Recent work by Gilbert *et al.* [5] and Uemura *et al.* [22] deviate from this paradigm. They extract large numbers of low level interest points per frame and build transformation invariant features without loss of information. For example, Gilbert build high level compound features from an over-complete set of simple 2D corners, using data mining. Uemura extract many features of various types along with their motion vectors, and encode them in multiple vocabulary trees.

Some methods use existing interest point detectors, but propose novel methods of describing interest points and classifying actions. Schuldt *et al.* [20] use the interest point detector of [10] and compute jet descriptors of spatio-temporal neighbourhoods, applying Support Vector Machine (SVM) classifiers to the descriptors. Dollar [2] *et al.* also apply SVM classifiers to detected interest points. The interest point detector of Dollar is employed by [12, 14, 15, 19, 23]. Niebles *et al.* [14] utilise a bag of visual words technique with probabilistic latent semantic analysis. Wong *et al.* [23] extends the probabilistic model to capture both semantic and structural information by including an implicit shape model. Liu and Shah [12] use the Maximisation of Mutual Information to discover an optimal number of visual word clusters, and capture structural information by exploring the correlation of the clusters.

Lepetit and Fua [11] extend the randomised trees classifier [1] and apply it to matching of interest points in images. To increase the speed of randomised trees, Ozuysal *et al.* [18] proposed the Fern classifier. Recently, Oshin *et al.* [17] applied the Fern classifier to spatio-temporal interest point detection. Ferns are non-hierarchical structures that have been shown to achieve excellent classification results while reducing computational overhead. The Fern classifier is of particular relevance to our work.

3. Randomised Ferns

Ferns are non-hierarchical classification structures, as shown in figure 3. Each Fern consists of a set of ordered binary tests, and returns probabilities of a patch belonging to each of the classes learnt during training. Ozuysal [18] used the simple binary test of pixel intensity comparisons, with the result f_j being 1 if the intensity at point j_1 is greater than that of point j_2, and 0 otherwise, given that $j = \{1 \dots S\}$ and S is the number of binary tests in a Fern, called nodes. The binary values $\{f_1, f_2, \dots, f_S\}$, returned from the ordered tests are combined and converted to decimal. Hence, a Fern with S nodes will return a decimal value between 0 and $2^S - 1$. For multiple patches that belong to the same class, the output of a Fern for that class can be modelled with a histogram, with each training patch incrementing the value of a bin in the histogram. More Ferns can be created by generating new nodes and obtaining distributions for all classes within the Ferns. Independence is assumed between Ferns.

During classification, the same set of ordered tests are performed on a test patch and a binary code is obtained, which when converted to decimal is used to select a bin in the class histograms to look up. The selected bin gives the likelihood of that patch belonging to each of the classes. The class with the maximum likelihood is chosen as the most probable class. For multiple Ferns, the class with the maximum product of class likelihoods across the Ferns

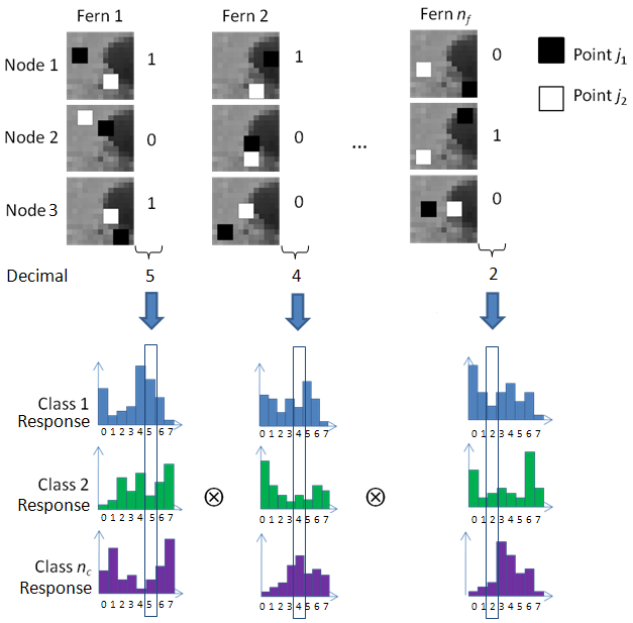

Figure 3. A set of n_f ferns, each containing S nodes. A comparison between pixels at points j_1 and j_2 for each node results in a binary digit. During training, these bits are to combined and populate a class histogram. During classification, they are combined to select class likelihoods from the histograms.

is selected, assuming independence between the Ferns. Performance-memory trade-offs can be made by changing the number of Ferns, allowing for a flexible implementation.

4. Learning for Action Recogntion

The aim of this work is to classify and localise human actions in video by encoding the distribution of spatio-temporal interest points of the actions. One of the key contributions of this paper is the extension of Randomised Ferns as applied in both object recognition [18] (described above), and spatio-temporal interest point recognition [17], to Action Recognition. In this section, we present this extension and describe our approach for achieving reliable action recognition and localisation.

In the approaches of [18] and [17], Ferns are employed to encode the appearance of interest point neighbourhoods. However, we wish to encode the *distribution* of spatio-temporal interest points for action recognition.

Randomised Ferns are applied to a spatio-temporal block around the action, called an *Action Cuboid*. We redefined a node as a comparison between *sums of interest point strengths* within selected regions (*Subcuboids*) within the parent *Action cuboid*. Figure 4 depicts nodes of the action classifier within an action cuboid. The spatial extent of the subcuboids are scaled by a factor σ of the action

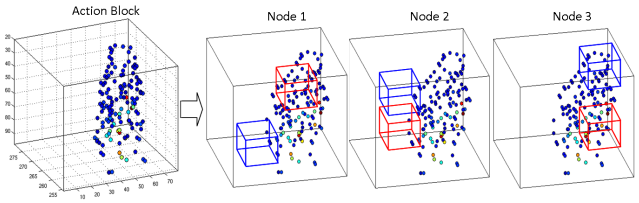

Figure 4. Three spatio-temporal nodes within an action cuboid, capturing the distribution of interest point responses. We have defined a node as a comparison between the sum of interest point response strengths within the two randomly positioned *subcuboids*, illustrated by the red and blue cuboids within the action cuboid.

cuboid, and the subcuboid thickness is a scale τ of the temporal extent of the action cuboid. Similar to the previous approaches [11, 17, 18], positions of subcuboids within an action cuboid can be chosen randomly or by using a greedy algorithm that attempts to maximise the information gain.

For an action cuboid i_{XYT} with dimensions X, Y, T taken from a video I, and subcuboids positioned within it at points x, y, t with spatial and temporal extents $\frac{X}{\sigma}, \frac{Y}{\sigma}, \frac{T}{\tau}$, the result of our node test f_j is given by,

$$f_j = \begin{cases} 1 & s(x_1, y_1, t_1, \frac{X}{\sigma}, \frac{Y}{\sigma}, \frac{T}{\tau}) < s(x_2, y_2, t_2, \frac{X}{\sigma}, \frac{Y}{\sigma}, \frac{T}{\tau}); \\ 0 & \text{otherwise.} \end{cases}$$

where s is the sum of interest point response strengths within the subcuboid, given by

$$s(x, y, t, \frac{X}{\sigma}, \frac{Y}{\sigma}, \frac{T}{\tau}) = \sum_{x'=x}^{x+\frac{X}{\sigma}} \sum_{y'=y}^{y+\frac{Y}{\sigma}} \sum_{t'=t}^{t+\frac{T}{\tau}} \varphi(x', y', t').$$

φ is the representation of the action cuboid in terms of detected interest points only, and is given by

$$\varphi(x, y, t) = \begin{cases} \Re(I_{xyt}) & \text{if } \Re(I_{xyt}) > T; \\ 0 & \text{otherwise.} \end{cases}$$

where $\Re$ is the strength of the response function of the interest point detector, applied to the video at point x, y, t, and T is the threshold above which interest points are detected.

We extract interest points from each training and test video and retain only the positions and response strengths of the interest points. All other information, including the interest points themselves and the video, are discarded. We then create a volumetric representation of these interest points, as shown in figure 2, then obtain an integral volume representation of the video based on the interest point responses. This is done to efficiently retrieve the sums of interest point responses and is similar to the integral video representation of Ke *et al.* [8].

4.1. Training

For the purpose of training, we groundtruth the actions in training videos, spatially capturing the entire person and the action being performed. The groundtruth box is adjusted to achieve a constant aspect ratio. Also, a temporal depth is selected that is large enough to capture at least one cycle of the action. To obtain robustness to noise, we generate additional positive examples by randomly offsetting groundtruth boxes in all directions. We then detect interest points in the video and train a spatio-temporal Ferns classifier on the groundtruthed action cuboids, comparing the sum of interest point responses. We also train an additional *no-action* class using negative data taken from the background. The no-action class overlaps with parts of actions, to achieve robustness against misclassifications due to a partial view of the action.

After obtaining Fern histograms for all classes, we combine the distributions to obtain a binary classifier for each class, such that, for each class considered during classification, a likelihood ratio $\frac{p(thisClass)}{p(allOtherClasses)}$ can be obtained. We can then select the class that has the highest likelihood ratio across all Ferns.

4.2. Classification

Given a novel video sequence, we first detect interest points of the action in the video. The scale of the action determines the types of interest points detected, therefore the detector is applied to the video at a number of scales, yielding different sets of interest points for each scale. We obtain an integral volume representation of the interest points, as previously explained, and sweep a scanning volume over the entire sequence, classifying the region within the volume. The scanning volume is applied at various scales to achieve invariance to scale. It follows that the scale of the scanning volume determines the set of interest points on which it is applied.

Given a new action cuboid to classify (from the scanning volume), we apply the Fern classifier, adapting the size of the subcuboids to the size of the action cuboid with predefined σ and τ. Each node compares the sums of interest point responses within two randomly positioned subcuboids. The classification of the action cuboid continues as previously detailed, with its class determined by combining results of these node tests and choosing the class with the highest likelihood ratio across the Ferns.

We use the vote of each frame to determine the action performed in the video. However, instead of simply using the vote of each region in a frame to determine the action in the frame, we apply multivariate Linear Discriminant Analysis (LDA) to strengthen the frame classification. The distribution of the classifications on a frame is used as a feature for the discriminant analysis, *i.e.*, the votes for all classes within a frame forms a feature vector $\mathbf{x}$ for that frame. We obtain a discrimant function,

$$d_i = \boldsymbol{\mu}_i \mathbf{C}^{-1} \mathbf{x}^T - \frac{1}{2} \boldsymbol{\mu}_i \mathbf{C}^{-1} \boldsymbol{\mu}_i^T + ln(\mathbf{p}_i)$$

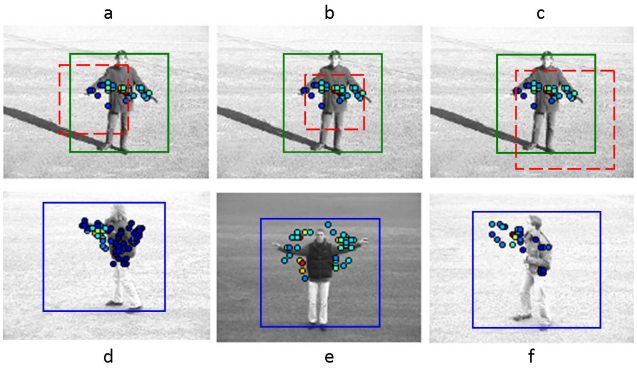

Figure 5. The effects of detection at various positions and scales on the same action. a, b and c (Top row) show the same frame of a *Handclapping* action with detected interest points. The solid green box in a, b and c indicates the ideal scanning volume position and scale; In a and b, the dashed lines show the scanning volume applied at wrong scales. In c, the dashed line indicate a scanning volume applied at the right scale, but not centred on the action. d, e and f (Bottom row) show other actions with similar distributions within their action cuboids, to the erroneous scanning volumes in a, b and c respectively. This results in confusion.

for each class i. The parameters $\boldsymbol{\mu}$, $\mathbf{C}$ and $\mathbf{p}$ are obtained prior to testing, from training or validation data, where $\boldsymbol{\mu}_i$ is the mean of features in class i, $\mathbf{C}$ is the between-class covariance matrix, and $\mathbf{p}_i$ is the prior probability vector, given by $\mathbf{p}_i = \frac{n_i}{N}$, where n is the number of samples from class i, and N is the total number of samples used in training. The frame is assigned the class with maximum d_i.

4.3. Localisation

Using the scanning volume on a test sequence, each region is classified as one of the available actions or *no-action*. Regions that contain no interest points have a high likelihood of being classified as no-action, while actual regions and scales of actions are expected to be correctly classified. We however expect to obtain false detections *around* the actual action region, and at significantly different scales, similar to the observation made in [8]. This is as a result of the scanning volume being applied on only a part of action or at the wrong scale. Figure 5 shows examples of similar distributions of interest points resulting from poor localisation. At these points, the distribution of interest points are similar to that of other actions, resulting in misclassification.

Due to the nature of Ferns, the likelihood ratio of detections cannot be used as a measure of confidence, as erroneous detections are made with confidences as high as accurate detections. It is possible to include erroneous data in training to improve robustness, however, the amount of data needs to be significant enough and the error consistent to be modelled as a characteristic of the action. On the other hand, the error must not be so significant as to alter the definition of the action, or influence the correct detection of other actions.

Figure 6. Sample of localisation results obtained. The Bhattacharyya coefficient is calculated between the scanning volume and average action grids obtained from training actions. Points with coefficients below a certain threshold (the dark regions of the images) are not classified.

To handle this problem, we make use of *Action grids*. During training, we split the groundtruthed action cuboid into an equally spaced $8 \times 8 \times 4$ grid, and sum the interest point responses within each grid cell. For each class, we obtain the average grid over all groundtruth cuboids used in training and normalise it. During testing, we perform the same operation on scanning window regions before classifying. We then calculate a Bhattacharyya Coefficient, β, to measure the correlation between the average class grids obtained from training, a, and a test grid, b:

$$\beta_{a,b} = \sum_{x'}^{\chi} \sum_{y'}^{\psi} \sum_{t'}^{\omega} \sqrt{a(x',y',t') \times b(x',y',t')},$$

where χ, ψ and ω are the extents of the grid in the spatial and temporal directions. Since both grids are normalised, the coefficient ranges from 0 to 1, where 1 indicates perfect correlation. We observed that applying the scanning window at the wrong scale or region returns a low coefficient for a particular action, and a high coefficient is obtained at the actual region and scale. We apply a threshold for the grid correlation, below which, the classifier is not applied, hence increasing the efficiency of the system. Figure 6 shows results of localisation.

We further exploit the value of the grid correlation at each region, and use it as a confidence measure for classification. Since the Bhattacharyya coefficient is calculated between the test grid at that region and the average grids of all classes, the grid is given several correlation values - one for each class. The confidence of the classification at that region is thus the value of the correlation between the test grid and the average grid of the selected class.

5. Experimental Setup

Interest points can be detected using a number of methods. These include methods proposed by Dollar *et al.* [2], Laptev and Lindeberg [10] and Gilbert *et al.* [5]. Our ac-

tion classifier is tested on the KTH human action dataset as outlined by Schuldt *et al.* [20].

5.1. Interest Point Detectors

Dollar *et al.* apply a response function of the form $\Re = (I * g * h_{ev})^2 + (I * g * h_{od})^2$ to the video, where $g(x, y : \sigma)$ is the 2D Gaussian kernel applied along the spatial dimensions of the video, and h_{ev} and h_{od} are a quadrature pair of 1D Gabor filters applied in the temporal dimension. The detector responds best to complex motions made by regions that are distinguishable spatially, including spatio-temporal corners as defined by [10], but not to pure translational motion or motions involving areas that are not distinct in space. Local maxima of the response function $\Re$ are selected as interest points.

The corner detector developed by Harris and Stephens [6] has been used in numerous applications involving object matching and tracking. The method involves detecting locations in an image where pixel intensities have significant local variations, obtained by applying a corner/edge response function and selecting its maxima. Gilbert *et al.* [5] extract 2D Harris corners in (x, y), (x, t) and (y, t), obtaining an over complete set of interest points. While Gilbert assembles compound features from these 2D corners before use in recognition, we make use of the corners in their ungrouped form. Also, the 2D corners used in our experiments do not have response strengths, hence they were all given an equal value.

Laptev and Lindeberg extend the Harris corner detector to the spatio-temporal domain by requiring that image values in space-time have significant variations in the spatial and temporal dimensions. They compute a windowed 3×3 second moment matrix composed of first order spatial and temporal derivatives, averaged with a Gaussian weighting function. Interest points are then detected by searching for regions that have significant eigenvalues of the matrix.

For the interest point detectors of Laptev and Dollar, we choose one threshold for all actions such that the actions can generate a sufficient number of interest points while minimising noisy detections.

5.2. Action Dataset

The KTH dataset contains video sequences of 25 persons each performing six actions: boxing, handclapping, handwaving, jogging, running and walking, in four different scenarios. The videos are taken over static uniform backgrounds with some camera motion, and the scenarios are outdoor, outdoor with scale variations, outdoor with appearance variations, and indoors. Figure 7 shows examples of the actions and the scenarios in which they are performed. The dataset is split into training, validation and test subsets.

For the KTH human action dataset, we follow the training/validation/test split of Schuldt *et al.* [20], and train our classifier with 8 persons, adjust parameters with 8 persons, and perform testing on the remaining 9 unseen persons. For the Boxing, Handclapping and Handwaving actions of the dataset, scale variation is obtained by using the zoom function of the camera. However, for dynamic actions (Jogging, Running and Walking), variation in scale is obtained by the person moving diagonally towards or away from the camera. (See Row S2 in figure 7). This can result in significant intra-class variation. To avoid this, we create 3 separate subclasses for the scale variation videos of dynamic actions. During classification, we add results of these sub-classes to their corresponding classes.

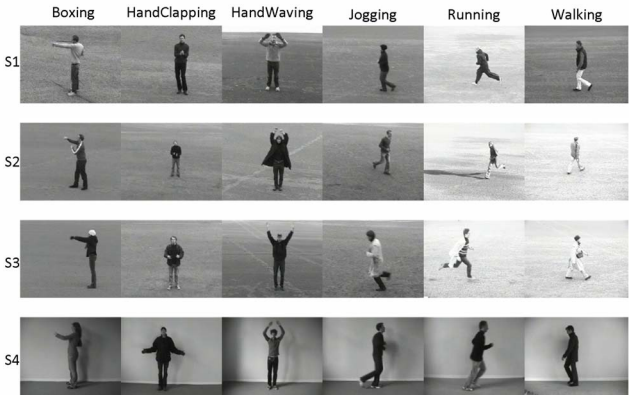

Figure 7. Examples of the KTH dataset actions. Columns represent each of the six actions, and rows show the Scenarios, where S1-S4 are outdoor, outdoor with scale change, outdoor with different clothes, and indoors scenarios respectively.

The spatial and temporal extents of an action determines the type and size of interest points detected. Actions in the KTH dataset can vary from 50 to 110 pixels in height. This means that interest points have to be detected at various scales to obtain consistent detections between scales. We select the appropriate scale in training using groundtruth information.

5.3. Results

For our experiments, we chose an aspect ratio of 1:1 for the action cuboid in (x, y). Videos in the dataset run at 25 fps and we observed that the period for most of the actions in the videos is less than 1 second. Hence, to be certain of capturing at least one cycle of the action, we set the depth of the action cuboid to 30 frames. We performed experiments to find the optimal size of subcuboids. Figure 8 shows a plot of average accuracy against the spatial scale σ. We chose a spatial scale $\sigma = 5$ and temporal scale $\tau = 3$. Hence, the size of the subcuboids is $\frac{X}{5}, \frac{Y}{5}, \frac{T}{3}$, for an action cuboid of size XYT. For the classifier, we chose 50 ferns, each with 5 nodes.

Figure 9 shows the confusion matrix obtained for our

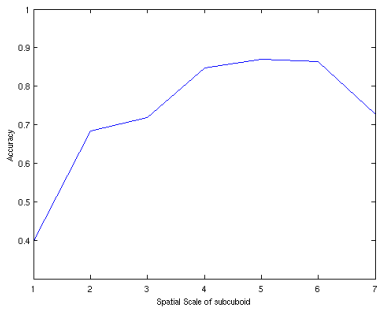

Figure 8. Plot of average accuracy results against the spatial scale of subcuboids, σ.

Randomised Ferns classifier using Dollar interest points. We obtain an average classification accuracy of 89.1% using the Training/Validation/Test split defined in [20]. Most of the confusion is observed between static actions or the dynamic actions, where the distribution of interest points are similar, and the most confusion was obtained between the Jogging and Walking actions. Table 1 shows a comparison of average accuracy obtained on the KTH human action dataset. It can be seen that our result is less than 2% lower than that of Fathi and Mori [4], who obtained the best results using the same training method.

Figure 10 shows the confusion matrix using the spatio-temporal corner detector of Laptev and Lindeberg [10]. It can be observed that there is a decrease in performance using these interest points. This decrease is due to the sparseness of detected interest points. The number of interest points extracted is not sufficient to allow for distinguishing between actions.

Figure 11 shows results obtained using the Harris 2D corners extracted from the video in the (x, y), (x, t) and (y, t) directions. In contrast to Laptev's interest points, while the Harris 2D corners are abundant, (typically 400 corners per frame) a large number of the detections are noisy. It should also be noted that response strengths were not available for the 2D corners used in this experiment, so all interest points, including noisy ones, were given equal weight. This result goes some way to show the benefit of using the response strength of interest points.

6. Conclusion

We presented a generic method for recognising and localising actions in video using local interest points. Our method makes exclusive use of the spatio-temporal distribution of the interest points in the description of actions, and we have shown that this description is sufficient for the purpose of action recognition, without using the appearance of the interest points. We generalised the semi-naive Bayesian classifier called Randomised Ferns, previously used in interest points recognition, and applied it to action recognition. The classifier learns interest point dis-

	Boxing	Handclapping	Handwaving	Jogging	Running	Walking
Boxing	0.86	0.08				0.06
Handclapping		0.97	0.03			
Handwaving	0.03	0.08	0.89			
Jogging				0.78		0.22
Running				0.05	0.92	0.03
Walking				0.08		0.92

Figure 9. Confusion Matrix for KTH Actions using *Dollar* interest points. Average detection accuracy is 89.10%.

	Boxing	Handclapping	Handwaving	Jogging	Running	Walking
Boxing	0.86	0.08	0.06			
Handclapping	0.17	0.78	0.06			
Handwaving		0.11	0.89			
Jogging		0.06	0.03	0.72	0.08	0.11
Running				0.05	0.92	0.03
Walking	0.06		0.03	0.19		0.72

Figure 10. Confusion Matrix for KTH Actions using *Laptev* interest points. Average detection accuracy using these interest points is 81.5%

Method	Training Method	Accuracy
Kim *et al.* [9]	LOOCV	95.33%
Wong *et al.*[23]	LOOCV	91.60%
Fathi and Mori [4]	Splits	90.50%
Gilbert *et al.* [5]	Splits	89.92%
Nowozin *et al.* [15]	Splits	87.04%
Niebles *et al.* [14]	LOOCV	81.50%
Dollar *et al.* [2]	LOOCV	81.20%
Schuldt *et al.* [20]	Splits	71.72%
Ke *et al.* [8]	Splits	62.97%
Our Method: Dollar	Splits	89.10%
Our Method: Laptev	Splits	81.50%
Our Method: 2D	Splits	73.80%

Table 1. Comparison of average recognition accuracy reported on the KTH human action dataset. LOOCV indicates Leave-One-Out Cross validation training method, and Splits indicates the Training/Validation/Test split as defined in [20].

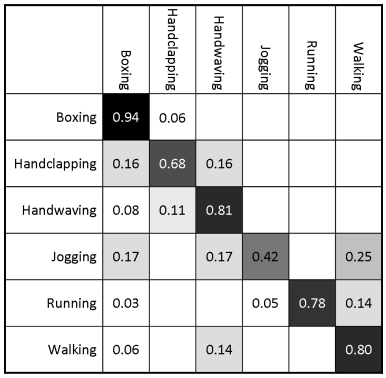

Figure 11. Confusion Matrix for KTH Actions using Harris *2D Corners*. Average detection accuracy is 73.80%.

tribution by comparing sums of interest point responses in randomly placed spatio-temporal subregions within action cuboids. We present results comparable to state-of-the-art on the largest available human action dataset using three interest point detectors. Our results show that, after interest points are detected, their composition is not vital to the description of the action, though denser, less noisy interest points are more desirable, achieving higher performance. In the future, we will examine perfomance gain from including the appearance of the interest points, if any. We will also investigate the classification of natural actions in video.

7. Acknowledgements

This work is supported by the EU FP6 Project, URUS, and the FP7 Project, DictaSign.

References

[1] Y. Amit and D. Geman. Shape quantization and recognition with randomized trees. *Neural Computation*, 9(7):1545–1588, 1997.

[2] P. Dollár, V. Rabaud, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. In *Proc. IEEE Intl. Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance (VS-PETS '05)*, pages 65–72, 2005.

[3] A. Efros, A. Berg, G. Mori, and J. Malik. Recognizing action at a distance. In *Proc. Intl. Conf. on Computer Vision (ICCV '03)*, pages 726–733, 2003.

[4] A. Fathi and G. Mori. Action recognition by learning mid-level motion features. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR '08)*, pages 1–8, 2008.

[5] A. Gilbert, J. Illingworth, and R. Bowden. Scale invariant action recognition using compound features mined from dense spatio-temporal corners. In *Proc. European Conf. on Computer Vision (ECCV '08)*, pages 222–233, 2008.

[6] C. Harris and M. Stephens. A combined corner and edge detection. In *Proc. Alvey Vision Conf.*, pages 147–151, 1988.

[7] T. Kadir and M. Brady. Scale saliency: a novel approach to salient feature and scale selection. In *Proc. Intl. Conf. on Visual Information Engineering (VIE '03)*, pages 25–28, 2003.

[8] Y. Ke, R. Sukthankar, and M. Hebert. Efficient visual event detection using volumetric features. In *Proc. IEEE Intl. Conf. on Computer Vision (ICCV '05)*, volume 1, pages 166–173, 2005.

[9] T. Kim, S. Wong, and S. Cipolla. Tensor canonical correlation analysis for action classification. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR '07)*, pages 1–8, 2007.

[10] I. Laptev and T. Lindeberg. Space-time interest points. In *Proc. IEEE Intl. Conf. on Computer Vision (ICCV '03)*, volume 2, pages 432–439, 2003.

[11] V. Lepetit and P. Fua. Keypoint recognition using randomized trees. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 28(9):1465–1479, 2006.

[12] J. Liu and M. Shah. Learning human actions via information maximization. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR '08)*, pages 1–8, 2008.

[13] D. G. Lowe. Object recognition from local scale-invariant features. In *Proc. IEEE Intl. Conf. on Computer Vision (ICCV '99)*, pages 1150–1157, 1999.

[14] J. Niebles, H. Wang, and L. Fei-Fei. Unsupervised learning of human action categories using spatial-temporal words. *Intl. Journal of Computer Vision*, 79(3):299–318, 2008.

[15] S. Nowozin, G. Bakir, and K. Tsuda. Discriminative subsequence mining for action classification. In *Proc. IEEE Intl. Conf. on Computer Vision (ICCV '07)*, pages 1–8, 2007.

[16] A. Oikonomopoulos, I. Patras, and M. Pantic. Spatiotemporal salient points for visual recognition of human actions. *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, 36(3):710–719, 2006.

[17] O. T. Oshin, A. Gilbert, J. Illingworth, and R. Bowden. Spatio-temporal feature recogntion using randomised ferns. In *Proc. ECCV Intl. Workshop on Machine Learning for Vision-based Motion Analysis (MLVMA '08)*, 2008.

[18] M. Ozuysal, P. Fua, and V. Lepetit. Fast keypoint recognition in ten lines of code. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR '07)*, pages 1–8, 2007.

[19] S. Savarese, A. DelPozo, J. Niebles, and L. Fei-Fei. Spatial-temporal correlatons for unsupervised action classification. In *Proc. IEEE Workshop on Motion and Video Computing (WMVC '08)*, pages 1–8, 2008.

[20] C. Schuldt, I. Laptev, and B. Caputo. Recognizing human actions: A local SVM approach. In *Proc. Intl. Conf. on Pattern Recognition (ICPR '04)*, pages 32–36, 2004.

[21] P. Scovanner, S. Ali, and M. Shah. A 3-dimensional sift descriptor and its application to action recognition. In *Proc. Intl. Conf. on Multimedia*, pages 357–360, 2007.

[22] H. Uemura, S. Ishikawa, and K. Mikolajczyk. Feature tracking and motion compensation for action recognition. In *Proc. British Machine Vision Conference (BMVC '08)*, 2008.

[23] S. Wong and R. Cipolla. Extracting spatiotemporal interest points using global information. In *Proc. IEEE Intl. Conf. on Computer Vision (ICCV '07)*, pages 1–8, 2007.