Natural action recognition using invariant 3D motion encoding

Simon Hadfield, Karel Lebeda, and Richard Bowden

Centre for Vision, Speech and Signal Processing, University of Surrey, UK
{S.Hadfield,K.Lebeda,R.Bowden}@surrey.ac.uk

Abstract. We investigate the recognition of actions "in the wild" using 3D motion information. The lack of control over (and knowledge of) the camera configuration, exacerbates this already challenging task, by introducing systematic projective inconsistencies between 3D motion fields, hugely increasing intra-class variance. By introducing a robust, sequence based, stereo calibration technique, we reduce these inconsistencies from fully projective to a simple similarity transform. We then introduce motion encoding techniques which provide the necessary scale invariance, along with additional invariances to changes in camera viewpoint.
On the recent Hollywood 3D natural action recognition dataset, we show improvements of 40 % over previous state-of-the-art techniques based on implicit motion encoding. We also demonstrate that our robust sequence calibration simplifies the task of recognising actions, leading to recognition rates 2.5 times those for the same technique without calibration. In addition, the sequence calibrations are made available.

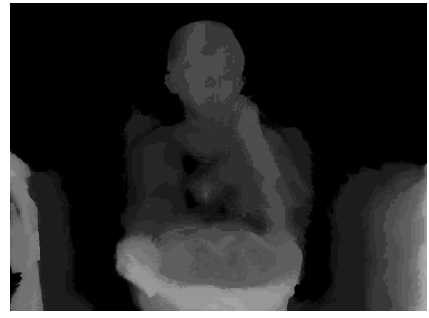
Keywords: Action recognition, in the wild, 3D motion, scene flow, invariant encoding, stereo sequence calibration

1 Introduction

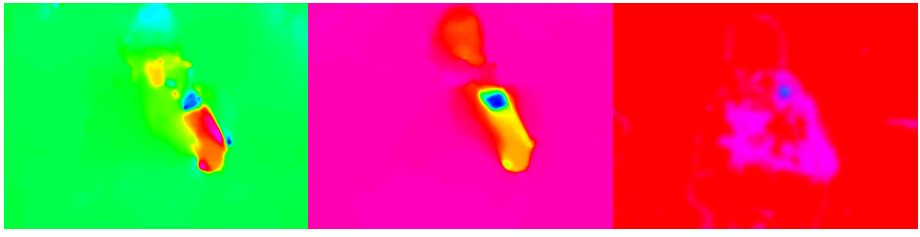
In recent years, the field of action recognition has been exploring techniques for effectively exploiting the wealth of 3D video data which has recently become available. However, the area of natural or "in the wild" action recognition using 3D data offers it's own unique, and so far unaddressed, challenges. Attempting to make use of 3D data from disparate sources with unknown calibration, adds additional layers of variations into a field which is already typified by huge intra-class variance and limited training data. This is especially obvious in recent datasets such as Hollywood-3D [9] which contains a wide variety of 3D data, but provides no calibration information. This severely limits the amount of 3D information which can be extracted, forcing authors to resort on projected 2D motion fields [9] or implicit depth encodings [12]. In this paper we propose not only the use of true 3D motion fields (see Figures 1 and 2) as a descriptor for recognising action categories, but also techniques for reliably extracting robust, invariant and comparable descriptors from uncontrolled 3D data.

Motion information has long been one of the primary tools to distinguish actions. Interest points in natural videos are detected based on temporal gradients [14,32] and

(a) *Eat* Left video



(b) *Eat* Disparity



(c) *Eat* World X velocity (d) *Eat* World Y velocity (e) *Eat* World Z velocity

Fig. 1: An example *Eat* action taken from the Hollywood 3D dataset. The appearance and disparity (top row) are provided. Also shown is a 3D motion field for the sequence. Note that motion is concentrated on the arm and head, which move towards each other.

motion fields are directly encoded to describe videos [23,21,15]. This is perhaps unsurprising, as it can be argued that motions are what define an action. In recent years, 3D structural data has seen increasing use in action recognition, however 3D motions remain conspicuously unexploited. This is primarily due to the difficulty in obtaining such data. Although the Kinect directly provides 3D structural information at every frame, the motion fields which warp from one structure to the next are unknown, and estimating them is still a topic of ongoing research [20,26,8,1].

In addition to the difficulty in extracting such data, there has recently been a rapid increase in the potential sources of 3D data. This includes consumer devices like the Kinect, 3D broadcast footage from television networks & film studios, and even upcoming mobile devices like Google's Project Tango. This variety of input domains further emphasises the need for invariant encoding, in order to fully exploit the diverse set of input data.

2 Related Work

In action recognition, it has long been standard practice to employ interest point detectors [14] to focus attention on salient regions of the scene during learning. This serves to suppress irrelevant background information and reduce the computational complexity

(a) *Drive* Left video (b) *Drive* Disparity

(c) *Drive* World X velocity (d) *Drive* World Y velocity (e) *Drive* World Z velocity

Fig. 2: An example *Drive* action taken from the Hollywood 3D dataset. The appearance and disparity (top row) are provided. Also shown is the 3D motion field for the sequence. The primary motion occurs on the foreground regions of the car, with secondary x and y motion on the passengers.

of many algorithms [6,18,15]. The use of 3D data has removed the need for this step in much recent work recognising actions in constrained environments, due to the simplicity of segmenting the actor (for example by using the Kinect's user mask) [18,3,7]. This enables complex "volumetric" descriptions of the actors body over time [33,31,30,22]. However, for "in the wild" action recognition this is not the case as it generally remains impossible to segment the actor reliably, due to noisy 3D data, cluttered environments, and scenes containing multiple people. As such, it is still common to use interest point detectors as a kind of "soft user mask". In this paper, we use the depth-aware spatio-temporal corner detectors of [9] for this purpose.

Once salient parts of the sequence have been detected, various local feature descriptors are generally extracted from these regions. Local features which have proved effective in the past include gradient based appearance information [27,16], 2D motion information [21,4] and spatio-temporal extensions to SIFT and SURF descriptors [28][32]. For "in the wild" action recognition, the use of the Hollywood-3D dataset has prompted the investigation of local features based on 3D information. However, previous work has been limited by the lack of consistent calibration information. As such, authors have been forced to rely on the recognition system learning to generalise across variations arising from miscalibration [12,9]. It is here that one of the major contributions of this paper lies, overcoming this limitation and making it possible for a new and powerful type of local information to be extracted to encode actions, based on 3D motion fields.

The final stage of "in the wild" action recognition, is often for the collection of local features to be encoded into a single holistic description of the sequence, often via a Bag-of-Words approach with a codebook of exemplar features. This approach is analogous to the highly successful Bag-of-Words techniques for object recognition, but with an additional temporal dimension. The Bag-of-Words approach to sequence encoding is generally performed by accumulating occurrences spatially and temporally across the entire sequence. This provides invariance to a range of important deformations, such as spatial and temporal translation, stretching and reflection. This is invaluable for generalisation, but it also leads to much of the relational information being discarded, such as the spatial configuration and temporal ordering of features. Laptev *et al*. attempt to mitigate this by splitting the spatio-temporal volume into sub-blocks, creating a descriptor for each sub-block, and concatenating them to create the sequence descriptor [15]. Sapienza *et al*. follow a similar vein, encoding individual sub-sequences, however rather than concatenating to create a single descriptor, they employ Multiple Instance Learning (MIL) [25]. This accounts for some parts of the sequence being irrelevant, for example before and after the action. In this paper we propose a number of novel encoding schemes, incorporating invariances particularly suited to our 3D motion features, such as scale and viewpoint invariance.

The remainder of this paper is structured as follows. Section 3 describes the robust auto-calibration technique proposed for use with varied footage. We then explain how this calibration allows comparable 3D motion information to be extracted from varied sequences, and propose invariant encoding schemes in Section 4, allowing us to make the best use of the varied training sequences. Finally, in Section 5 the proposed techniques are evaluated on a recent dataset for 3D action recognition "in the wild" and compared against the existing state of the art results.

3 Stereo Sequence Auto-calibration

To extract comparable 3D motion information from multi-view sequences, we need some form of calibration between the views. This is particularly an issue for "in the wild" action recognition, where the camera models, and layout, change between sequences. Without accounting for these differences, extracted 3D information varies greatly from sequence to sequence. This introduces a huge amount of artificial variation to the action classes, making classification even more challenging. To mitigate this issue, we introduce an approach for stereo auto-calibration of video pairs.

The first step towards calibrating a pair of video sequences I^l and I^r, each of which consists of n frames ($\boldsymbol{I}^l_{1\ldots n}$ and $\boldsymbol{I}^r_{1\ldots n}$), is to detect a set of candidate correspondences. In this paper, sets (S^l and S^r) of SIFT [19] points $\boldsymbol{s} = (x, y, \tau)$ are extracted, where,

$$S = \{\boldsymbol{s} : \text{SIFT}\left(\boldsymbol{I}_\tau\left(x, y\right)\right) > \lambda_s\}\,. \tag{1}$$

based on the threshold λ_s. Each SIFT point $\boldsymbol{s}_i$ has an associated SIFT descriptor $\boldsymbol{f}_i$. Correspondences between point detections are calculated subject to the condition that their descriptors are closer than a threshold λ_f, and that they occur at the same frame in both sequences,

$$C = \left\{(\boldsymbol{s}^l_i, \boldsymbol{s}^r_j) : |\boldsymbol{f}^l_i - \boldsymbol{f}^r_j| < \lambda_f \text{ and } \tau^l_i = \tau^r_j\right\}. \tag{2}$$

Given this set of cross sequence correspondences, the epipolar geometry of the scene is estimated using 7-point RANSAC with Local Optimisation [17]. The fundamental matrix is estimated by

$$\boldsymbol{F} = \arg\min_{\boldsymbol{F}'} \sum \epsilon_s \left(\boldsymbol{s}_i^{l\top}, \boldsymbol{s}_i^r | \boldsymbol{F}' \right), \tag{3}$$

where ϵ_s is the Sampson error (linearised approximation to projection error). In this work ϵ_s also applies a truncated quadratic cost function (as in MSAC), which provides an approximation to the maximum likelihood estimate [29].

Given the estimated F we can also extract the set of inlier correspondences,

$$\hat{C} = \left\{ (\boldsymbol{s}_i^l, \boldsymbol{s}_i^r) : \left| \boldsymbol{s}_i^{l\top} \boldsymbol{F} \boldsymbol{s}_i^r \right| < \lambda_r \right\}, \tag{4}$$

which obey the epipolar constraints estimated. For the experiments in this paper, the detection, matching and inlier thresholds (λ_s, λ_f and λ_r respectively) use the default values suggested by their respective authors.

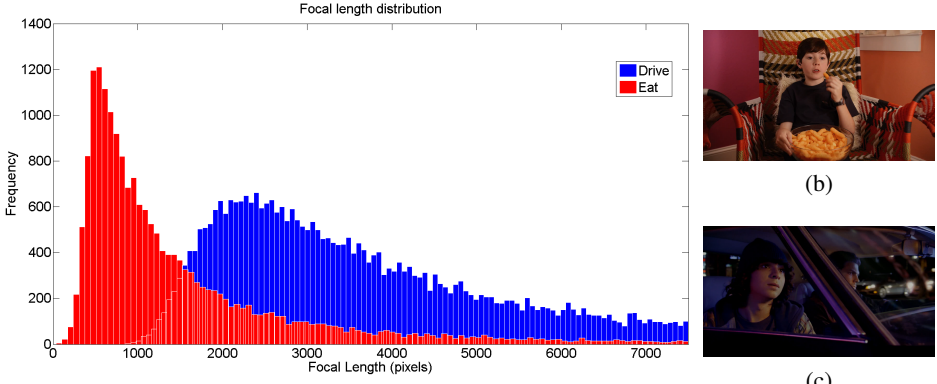
3.1 Full 3D sequence calibration

Estimating the epipolar geometry between the sequences is only the first step to consistent 3D calibration. Next the focal length (and hence the Essential matrix $\boldsymbol{E}$) must be estimated. This is feasible, subject to the assumption of square pixels, and that focal length is consistent between the two sequences (this assumption is reasonable, as stereo capture rigs generally utilise the same type of camera for both views). This can then be combined with constraints on the rank of $\boldsymbol{F}$, and the trace of $\boldsymbol{E}$, to construct a Polynomial Eigenvalue Problem (PEP) which may be efficiently solved [13]. As with the estimation of $\boldsymbol{F}$, this is solved in a RANSAC framework, with the inliers to the epipolar geometry $\hat{C}$ used as input.

Unfortunately, in general 3D footage it is very common for cameras to be in a near-parallel configuration. This adversely affects the stability of the PEP, which (although deterministic) may become sensitive to changes in the input correspondences $\hat{C}$. In other words, for a given $\hat{C}$ a particular $\boldsymbol{E}$ is estimated consistently. However, adding or removing a small number of points from $\hat{C}$ can in some cases lead to significant differences in the estimated $\boldsymbol{E}$. Luckily, the offline nature of the auto-calibration system, coupled with efficient PEP solvers, mean the process can be repeated a number of times. Each iteration finds a slightly different $\boldsymbol{F}$ and $\hat{C}$ which in turn leads to a different $\boldsymbol{E}$.

Figure 3 shows the distribution of focal lengths estimated over 1000 repetitions, for two sequence pairs with different levels of zoom. The distribution of focal lengths arising due to the near-parallel camera configuration, follows the log-normal distribution which should be expected from a multiplicative entity such as the focal length. As such, we can achieve robustness to near-parallel cameras, by taking the mode of this distribution, for each sequence pair. In our experiments we use 100 calibration repetitions to model this distribution, which takes a few minutes in our single thread Matlab implementation.

Finally, given our robust estimate of $\boldsymbol{E}$ it is possible to estimate the projections matrices $\boldsymbol{P}^l$ and $\boldsymbol{P}^r$ for the cameras [11]. This leads to 4 possible solutions as shown



(a) (b) (c)

Fig. 3: Distribution of estimated focal lengths over 20000 repetitions, on the 2 different sequences pairs shown in B and C (wide-angle, close-up *Eat* shot, and extreme zoom *Drive* shot).

in Figure 4. We select the solution that maximizes the number of corresponding point pairs $\hat{C}$ intersecting in front of the cameras,

$$\boldsymbol{P}^l, \boldsymbol{P}^r = \underset{\boldsymbol{P'}^l, \boldsymbol{P'}^r}{\arg\max} \sum_{(\boldsymbol{s}_i^l, \boldsymbol{s}_i^r) \in \hat{C}} \text{sign}(d_l) + \text{sign}(d_r), \tag{5}$$

where d_l and d_r are the distances along the rays defined by homogeneous points $\bar{\boldsymbol{s}}_i^l$, $\bar{\boldsymbol{s}}_i^r$ and $\mathcal{D}$ is the 3D position of the rays intersection,

$$d_l \bar{\boldsymbol{s}}_i^l = \boldsymbol{P'}^l \mathcal{D} \qquad \text{and} \qquad d_r \bar{\boldsymbol{s}}_i^r = \boldsymbol{P'}^r \mathcal{D}. \tag{6}$$

The proposed approach to stereo sequence calibration has some limitations. Firstly, lens distortion is not included in the model. This is acceptable for a wide range of footage from Kinect devices and broadcast sources, which generally exhibit little distortion, however this may be an issue for upcoming 3D mobile devices. Secondly, in order to exploit correspondences over entire sequences, a consistent focal length is assumed (i.e. no zooming). In theory the technique could be extended by collecting correspondences within a sliding window, and estimating a time varying focal length. However, to obtain a sufficient number of correspondences within the window, it becomes necessary to reduce robustness by allowing weaker matches. Finally, the reconstructions achieved by our calibration technique, are only consistent with each other up to a similarity transform (reconstructions using a generic calibration are consistent up to a homography). The removal of projective distortions does greatly reduce the variability in the data, but the remaining scale ambiguity still must be addressed during encoding.

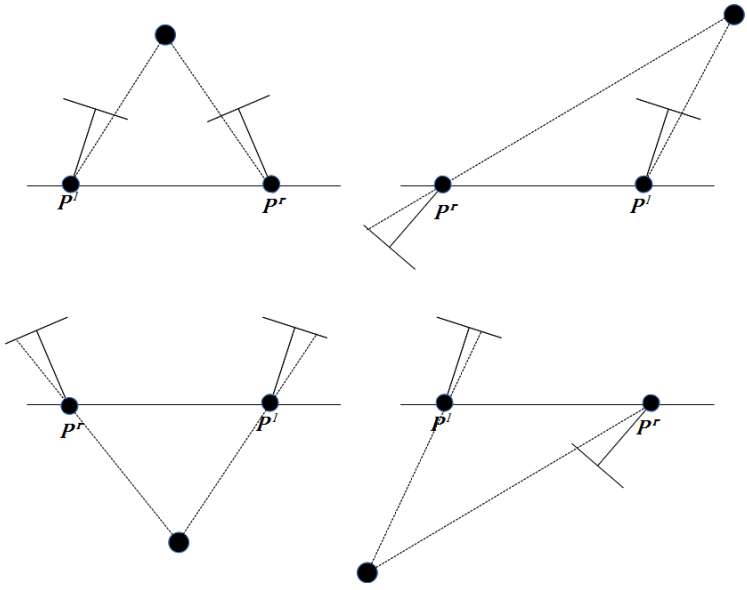


Fig. 4: The 4 possible solutions for stereo camera projection matrices, with a given $\boldsymbol{E}$ matrix. Note that only one solution leads to the 3D point being in front of both cameras [11].

4 Invariant Motion Encoding

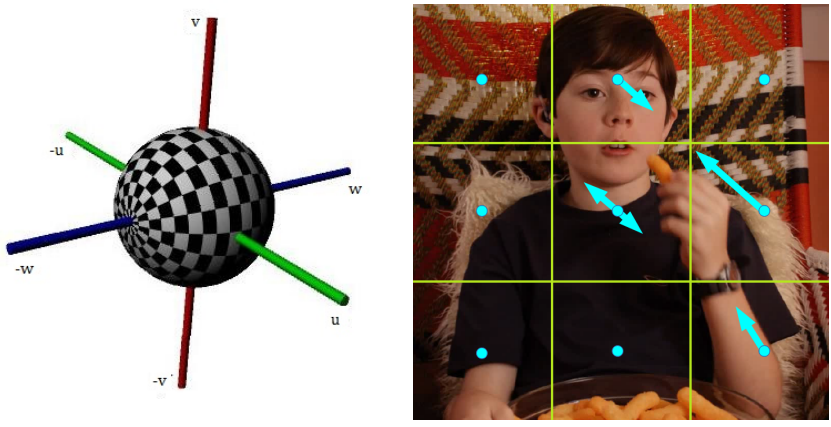
The estimated calibration can be used in conjunction with an efficient 3D motion estimation scheme such as [10]. This will estimate the "scene flow" (optical flows 3D counterpart) resulting in 3D structure and velocity (u, v, w) estimates at every point in the scene.

Given these dense flow fields, we can extract a local 3D motion descriptor around each of the spatio-temporal interest points within a sequence. We use a spherical coordinate system

$$\phi = \arctan\left(\frac{v}{u}\right) \qquad \text{and} \qquad \psi = \arctan\left(\frac{w}{v}\right), \tag{7}$$

to describe the 3D orientation of flow vectors. Note that ϕ refers to the "in plane" orientation (from the viewpoint of the left camera) *i.e.* when ϕ is $0°$, the motion is toward to the top of the image, when ϕ is $90°$ the motion is toward the right of the image, etc. In contrast ψ refers to the "out of plane" orientation, *i.e.* how much the motion is angled away from, or towards, the camera.

We encode the distribution of 3D orientations in a region around each interest point, capturing the nature of local motion field using a spherical histogram H as shown in Figure 5. This is similar to the approach used for shape context [2], but in the velocity domain. The contribution of each flow vector to the histogram is weighted based on the magnitude of the flow vector. Although this histogramming discards much of the



(a) Spherical orientation histogram (b) An encoded motion field

Fig. 5: (a) The spherical orientation histogram. Different orientation bins are illustrated with alternating white and black squares. The ϕ orientation relates to rotation around the w axis (which points away from the camera). This leads to movement between the cells of one concentric rings in the histogram. The ψ orientation relates to rotation around the u axis, i.e. moving between concentric rings. (b) a scene divided into a 3 by 3 grid of subregions, with the motion of each subregion aggregated (for clarity aggregated motions are shown in 2D).

spatial information, some general attributes are maintained by separating the region into several neighbouring blocks, and encoding each of them independently as $H_{1\ldots n}$. These sub-region spherical histograms are then combined to form the overall descriptor H. It should be noted that placing histogram bins at regular angular intervals in this way leads to the bins covering unequal areas of the sphere's surface. An exaggerated version of this effect can be seen in Figure 5a, although in practice fewer bins are used and the difference is less pronounced. In the future regular or semi-regular sphere tessellations could be considered to remove this effect [24].

At this stage we introduce our first layer of invariance. By normalising the local descriptors, we are able to resolve the scale ambiguity which remained in our auto-calibration of Section 3. As mentioned previously, our motion fields are only consistent up to a similarity transform. However, the normalised spherical histograms,

$$\bar{H} = \frac{H}{|H|} \tag{8}$$

are consistent up to a 3D rotation, making these 3D motion descriptors much more comparable between camera configurations, and thus suitable for "in the wild" recognition. In addition to this, the normalised features provide invariance to the speed at which actions are performed, as only the shape and not the value of the motion field is encoded. This is again very import for "in the wild" recognition, with many different actors, each of whom have their own action style.

4.1 Rotational Invariance

Next we look at including viewpoint invariance in our 3D motion features (i.e. removing the final 3D rotation ambiguity, and making the descriptors completely consistent). This is one of the biggest challenges for "in the wild" action recognition. The the same action viewed from different angles looks completely different. However, as we are using the underlying 3D motion field, it is possible to modify our feature encoding to be invariant to such changes.

We firstly encode invariance to camera roll (i.e. rotation around the z axis) by cycling the order of the subregion histograms $H_{1\ldots n}$ such that the the subregion containing the largest amount of motion occurs first. This re-arranged, roll-invariant, descriptor is referred to as $\bar{H}^r$ (see Figure 6).

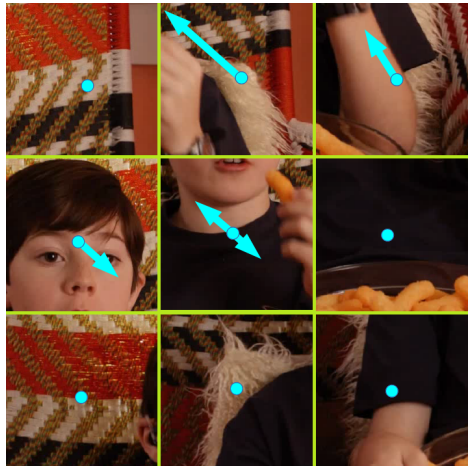


Fig. 6: $\bar{H}^r$ The subregions of the encoded motion field are re-arranged such that the region of maximum motion occurs first. This provides some degree of invariance to camera roll.

We can follow a similar approach for the flow vectors within the subregion histograms, to make the direction of the motions as well as their positions, rotationally-invariant. If we find the strongest motion vector in H and label its 3D orientation as $\hat{\phi}$,$\hat{\psi}$ then we can redefine our local orientations in relation to this flow vector,

$$\phi^p = \arctan\left(\frac{v}{u} - \hat{\phi}\right) \qquad \text{and} \qquad \psi^p = \arctan\left(\frac{w}{v} - \hat{\psi}\right). \tag{9}$$

The resulting descriptors $\bar{H}^p$ obtained when encoding ϕ^p,ψ^p makes the flow vectors robust to camera pitch (rotation around the x axis) in addition to roll, as shown in Figure 7.

However, due to the separation of ϕ and ψ our descriptors are still not resistant to camera pans (rotation around the y axis, which at 90 degrees causes ϕ orientation to

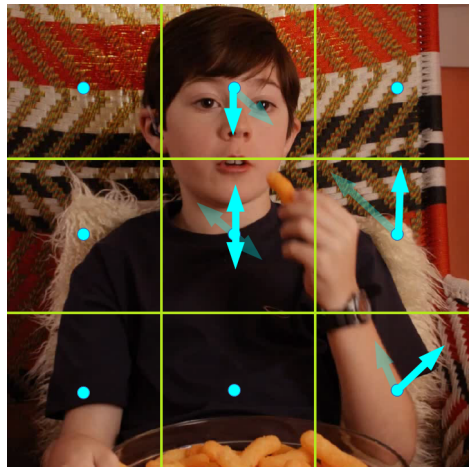


Fig. 7: $\bar{H}^p$ The orientation ($\hat{\phi}$,$\hat{\psi}$) of the strongest motion vector in the scene, is used to normalise the orientation histograms, providing invariance to camera pitch and roll.

become ψ orientation). In addition, normalising based on the maximum flow vector is sensitive to outliers in the flow field. As such, our final approach is to perform PCA on the local region of the motion field, extracting 3 new basis vectors u', v', w'. Computing orientation using these basis vectors,

$$\phi' = \arctan\left(\frac{v'}{u'}\right) \quad \text{and} \quad \psi' = \arctan\left(\frac{w'}{v'}\right), \tag{10}$$

leads to a descriptor $\bar{H}'$ which is invariant to all 3 types of camera viewpoint change, and also robust to outlier motions. See Figure 8 for an illustration.

4.2 Holistic Sequence Encoding

Whichever local descriptors are used, the final representation of the sequence is formed by a holistic Bag-of-words encoding. The sequence is described in terms of the frequency of occurrence for various exemplar descriptors (called the codebook). As in the case of the local descriptors, this space-time accumulation serves to provide invariance to spatio-temporal translations, scaling *etc*. but also implies a loss of relational information. To somewhat mitigate this, the sequence is divided into space-time blocks, each of which is encoded independently to provide the final description of the sequence.

5 Results

We evaluate our technique on the recently released Hollywood 3D dataset, which contains over an hour of "in the wild" action footage, taken from 3D broadcasts, covering 14 action categories. We compare our 3D motion features estimated using a single

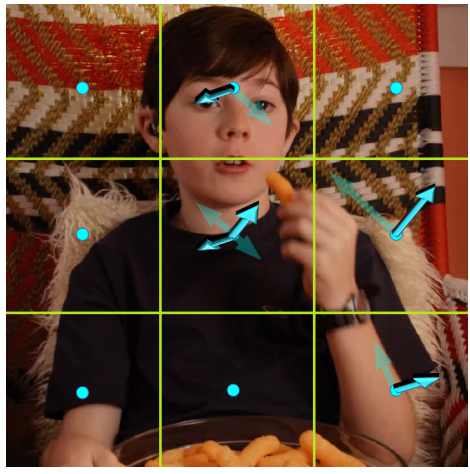


Fig. 8: $\bar{H}'$ A new set of 3D axes is chosen using PCA, relating to the dominant 3D motion orientations in the scene. This provides complete invariance to camera viewpoint change.

generic calibration, and estimated with sequence specific auto-calibration[1], against the current state of the art results on the dataset [12] which uses auto-encoders to implicitly model uncalibrated structural information. We also include results for a baseline method using 2D motion information from optical flow.

Performance is evaluated in terms of Average Precision [5] for each class. Classification is performed using an SVM with an RBF kernel, and encoding uses a Mahalonobis distance function and a codebook of 40000 elements, facilitating comparison with [9]. For the feature descriptors each subregion histogram uses 4×4 bins in the ϕ and ψ orientations, leading to a motion feature vector of 144 elements.

In Table 1 we can see that the raw 3D motion features ($\bar{H}$ -uncal), directly attainable from the dataset with a generic calibration, perform rather poorly, offering only a minor improvement over 2D motion based features (HOF[9]). The use of our proposed stereo sequence auto-calibration ($\bar{H}$) dramatically improves performance, more than doubling the average precision, by removing the projective distortion effects on the motion field. This helps to explain why 3D motion estimation techniques have not previously been exploited for "in the wild" action recognition, despite the fact that actions are generally defined by their 3D motions. The results also show that the unnormalised features (H), which are not scale invariant, perform uniformly worse than their normalised counterparts. It's worth noting, however, that Hollywood 3D doesn't contain the *Run*/*Jog*/*Walk* ambiguities of some datasets. Instead the wide range of viewpoints and zooms present in the data favour the more consistent $\bar{H}$ features.

The viewpoint invariant encoding schemes of Section 4.1 (upgrading the motion fields to fully consistent, rather than "up to a rotation") provide more modest improvements. Including roll invariance ($\bar{H}^r$) gives only a small performance increase, probably

[1] estimated calibrations are available at http://cvssp.org/Hollywood3D/

because broadcast footage such as that contained in the Hollywood-3D dataset contains few camera rolls. It may be expected that this scheme would prove more valuable in other scenarios such as on mobile devices. Attempting to include pitch invariance ($\bar{H}^p$) by normalising motion orientations actually reduces performance on many of the action classes. This is likely because normalising by the maximum motion makes the technique susceptible to outliers in the motion field. It is interesting to note however, that there is a marked improvement for a small number of actions such as *Run* and *Swim*. This may be because these actions experience greater variation in camera pitch (for example running shots being seen from above, and swimming shots from underwater). The final scheme ($\bar{H}'$), including full viewpoint invariance by estimating new motion orientation axes, provides the greatest performance, with more than 40 % improvement over the previously state of the art SAE-MD(av) [12] technique. It is interesting to note that all of these encoding schemes actually *throw away* some of the information present within the original features. However, for the task of "in the wild" action recognition, camera viewpoint invariance outweighs this, by making it easier to generalise between sequences.

Table 1: Per class Average Precision scores using various types of features encodings, including 2D motions [9], implicit depth and motion encoding [12], uncalibrated 3D motions, Unnormalised 3D motions, and calibrated motions encoding varying levels of invariance to camera viewpoint change.

Action	HOF [9]	SAE-MD(av) [12]	$\bar{H}$ -uncal	$\bar{H}$	H	$\bar{H}^r$	$\bar{H}^p$	$\bar{H}'$
NoAction	12.5	12.8	13.0	18.0	16.2	17.2	15.3	**21.2**
Run	18.0	50.4	21.5	44.3	41.1	40.8	55.9	**63.1**
Punch	2.9	38.0	10.9	48.7	45.6	51.6	52.1	**54.2**
Kick	3.6	7.9	8.1	18.2	18.2	**19.9**	18.1	**19.9**
Shoot	16.3	35.5	24.4	27.1	26.5	30.2	27.9	**31.0**
Eat	3.6	7.0	5.5	24.2	24.1	24.0	23.1	**24.2**
Drive	35.1	59.6	45.4	**62.3**	58.4	62.0	50.2	60.8
UsePhone	8.1	23.9	7.8	18.8	18.2	19.3	18.2	**22.3**
Kiss	6.7	16.4	7.0	24.2	24.1	24.0	26.3	**31.3**
Hug	2.6	7.0	3.5	21.8	21.0	22.2	23.8	**32.4**
StandUp	8.8	34.2	7.1	49.1	47.0	**51.8**	49.0	50.0
SitDown	4.3	7.0	4.8	16.3	14.1	17.9	16.9	**18.1**
Swim	6.4	29.5	14.0	28.8	27.1	30.0	**43.2**	43.0
Dance	2.8	36.3	3.7	45.3	41.8	44.2	**48.1**	44.9
Overall	9.4	26.1	12.6	31.9	30.2	32.5	33.4	**36.9**

6 Conclusions

In this paper we have demonstrated that 3D motion is a powerful tool for recognising the actions being performed in a scene. However, in order for it to be truly exploited within the field of "in the wild" action recognition, appropriate sequence calibration techniques must be employed. To this end we introduce an approach for stereo sequence calibration which is robust to near parallel cameras setups, and we make available the estimated calibrations for the entirety of the Hollywood-3D dataset.

We have also shown that one of the biggest issues for "in the wild" recognition, is the intra-class variability. By using viewpoint invariant encoding schemes, we can significantly improve the value of our 3D motion features, particularly for actions which are commonly viewed from different angles.

In the future it would be useful to explore more advanced holistic encoding schemes for sequences, preserving the invariances encoded in our 3D motion features without discarding so much relational information. It would also be interesting to investigate online approaches to auto-calibration, allowing the calibration to change within sequences. This would prove valuable for sequences which include zooming cameras, and also in domains where the cameras are not rigidly attached together and may move independently (for example surveillance cameras and co-operating drones).

Acknowledgements

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

References

1. Basha, T., Avidan, S., Hornung, A., Matusik, W.: Structure and motion from scene registration. In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on. pp. 1426 –1433 (june 2012)
2. Belongie, S., Malik, J., Puzicha, J.: Shape matching and object recognition using shape contexts. PAMI 24(4), 509–522 (2002)
3. Cheng, Z., Qin, L., Ye, Y., Huang, Q., Tian, Q.: Human daily action analysis with multi-view and color-depth data. In: Computer Vision–ECCV 2012. Workshops and Demonstrations. pp. 52–61. Springer (2012)
4. Dalal, N., Triggs, B., Schmid, C.: Human detection using oriented histograms of flow and appearance. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) Proc. ECCV. Lecture Notes in Computer Science, vol. 3952, pp. 428–441. Springer Berlin / Heidelberg, Graz, Austria (May 7 – 13 2006)
5. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL visual object classes (VOC) challenge. International Journal of Computer Vision 88(2), 303–338 (Jun 2010)
6. Gilbert, A., Illingworth, J., Bowden, R.: Action recognition using mined hierarchical compound features. PAMI 33(5), 883 –897 (may 2011)
7. Gorelick, L., Blank, M., Shechtman, E., Irani, M., Basri, R.: Actions as space-time shapes. PAMI 29(12), 2247–2253 (2007)

8. Hadfield, S., Bowden, R.: Kinecting the dots: Particle based scene flow from depth sensors. In: In Proceedings, International Conference on Computer Vision. Barcelona, Spain (6-13 Nov 2011)
9. Hadfield, S., Bowden, R.: Hollywood 3d: Recognizing actions in 3d natural scenes. In: In Proceedings, Conference on Computer Vision and Pattern Recognition (CVPR). Oregon, USA (Jun 22 – 28 2013)
10. Hadfield, S., Bowden, R.: Scene particles: Unregularized particle based scene flow estimation. IEEE Trans. on Pattern Analysis and Machine Intelligence 36(3), 564–576 (March 2014)
11. Hartley, R., Zisserman, A.: Multiple View Geometry in computer vision. Cambridge University press (2000)
12. Konda, K., Memisevic, R.: Learning to combine depth and motion. arXiv preprint arXiv:1312.3429 (2013)
13. Kukelova, Z., Bujnak, M., Pajdla, T.: Polynomial eigenvalue solutions to the 5-pt and 6-pt relative pose problems. In: BMVC. pp. 1–10 (2008)
14. Laptev, I., Lindeberg, T.: Space-time interest points. In: Proc. Ninth IEEE Int Computer Vision Conf. pp. 432–439 (2003)
15. Laptev, I., Marszalek, M., Schmid, C., Rozenfeld, B.: Learning realistic human actions from movies. In: Proc. IEEE Conf. Computer Vision and Pattern Recognition CVPR 2008. pp. 1–8 (2008)
16. Laptev, I., Perez, P.: Retrieving actions in movies. In: Proc. IEEE 11th Int. Conf. Computer Vision ICCV 2007. pp. 1–8 (2007)
17. Lebeda, K., Matas, J., Chum, O.: Fixing the locally optimized ransac. In: Bowden, R., Collomosse, J., Mikolajczyk, K. (eds.) Proc. BMVC. pp. 1013–1023. BMVA, London, UK (September 2012)
18. Li, W., Zhang, Z., Liu, Z.: Action recognition based on a bag of 3D points. In: Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on. pp. 9–14. IEEE (2010)
19. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. IJCV 60(2), 91 – 110 (November 2004)
20. Lukins, T., Fisher, R.: Colour constrained 4D flow. In: Proc. BMVC. pp. 340–348. Oxford, UK (Sep 6 – 8 2005)
21. Messing, R., Pal, C., Kautz, H.: Activity recognition using the velocity histories of tracked keypoints. In: Proc. IEEE 12th Int Computer Vision Conf. pp. 104–111 (2009)
22. Oreifej, O., Liu, Z.: Hon4d: Histogram of oriented 4d normals for activity recognition from depth sequences. In: Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on. pp. 716–723. IEEE (2013)
23. Oshin, O., Gilbert, A., Bowden, R.: Capturing the relative distribution of features for action recognition. In: Proc. IEEE Int Automatic Face & Gesture Recognition and Workshops (FG 2011) Conf. pp. 111–116 (2011)
24. Saff, E.B., Kuijlaars, A.B.: Distributing many points on a sphere. The Mathematical Intelligencer 19(1), 5–11 (1997)
25. Sapienza, M., Cuzzolin, F., Torr, P.: Learning discriminative space-time actions from weakly labelled videos. In: Proc. BMVC. Surrey, UK (Sep 3 – 7 2012)
26. Schuchert, T., Aach, T., Scharr, H.: Range flow in varying illumination: Algorithms and comparisons. PAMI pp. 1646–1658 (2009)
27. Schuldt, C., Laptev, I., Caputo, B.: Recognizing human actions: a local SVM approach. In: Proc. 17th Int. Conf. Pattern Recognition ICPR 2004. vol. 3, pp. 32–36 (2004)
28. Scovanner, P., Ali, S., Shah, M.: A 3-dimensional SIFT descriptor and its application to action recognition. In: Proceedings of the 15th international conference on Multimedia. pp. 357–360. MULTIMEDIA '07, ACM, New York, NY, USA (2007)

29. Torr, P., Zisserman, A.: Robust computation and parametrization of multiple view relations. In: Computer Vision, 1998. Sixth International Conference on. pp. 727–732. IEEE (1998)
30. Vieira, A.W., Nascimento, E.R., Oliveira, G.L., Liu, Z., Campos, M.F.: Stop: Space-time occupancy patterns for 3d action recognition from depth map sequences. Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications pp. 252–259 (2012)
31. Wang, J., Liu, Z., Chorowski, J., Chen, Z., Wu, Y.: Robust 3d action recognition with random occupancy patterns. In: Computer Vision–ECCV 2012. pp. 872–885. Springer (2012)
32. Willems, G., Tuytelaars, T., Van Gool, L.: An efficient dense and scale-invariant spatio-temporal interest point detector. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) Proc. ECCV, Lecture Notes in Computer Science, vol. 5303, pp. 650–663. Springer Berlin / Heidelberg, Marseille, France (Oct 12 – 18 2008)
33. Yang, X., Zhang, C., Tian, Y.: Recognizing actions using depth motion maps-based histograms of oriented gradients. In: Proceedings of the 20th ACM international conference on Multimedia. pp. 1057–1060. ACM (2012)