International Journal of Computer Vision manuscript No.
(will be inserted by the editor)

Hollywood 3D: What are the best 3D features for Action Recognition?

Simon Hadfield · Karel Lebeda · Richard Bowden

Received: 29/07/2015 / Accepted: 23/05/2016

Abstract Action recognition "in the wild" is extremely challenging, particularly when complex 3D actions are projected down to the image plane, losing a great deal of information. The recent growth of 3D data in broadcast content and commercial depth sensors, makes it possible to overcome this. However, there is little work examining the best way to exploit this new modality.

In this paper we introduce the Hollywood 3D benchmark, which is the first dataset containing "in the wild" action footage including 3D data. This dataset consists of 650 stereo video clips across 14 action classes, taken from Hollywood movies. We provide stereo calibrations and depth reconstructions for each clip. We also provide an action recognition pipeline, and propose a number of specialised depth-aware techniques including five interest point detectors and three feature descriptors. Extensive tests allow evaluation of different appearance and depth encoding schemes. Our novel techniques exploiting this depth allow us to reach performance levels more than triple those of the best baseline algorithm using only appearance information. The benchmark data, code and calibrations are all made available to the community.

Keywords Action recognition · in the wild · 3D · structure · depth · 3D motion · Hollywood 3D · benchmark

This work was supported by the EPSRC project "Learning to Recognise Dynamic Visual Content from Broadcast Footage" (EP/I011811/1) and the Swiss National Science Foundation project "SMILE - Scalable Multimodal sign language technology for sign language learning and assessment".

CVSSP, University of Surrey, GU2 7XH Guildford, UK
Tel.: +44 1483 682260
E-mail: {s.hadfield, k.lebeda, r.bowden}@surrey.ac.uk

1 Introduction

Recognising actions "in the wild" is useful for many applications including surveillance, automatic video indexing/search and assisted living. Huge intra-class variation is inherent to recognition in the wild, caused by the wide variety of environments, actors, viewpoints and action styles. We address this issue by exploiting the invariances inherent in 3D data, and proposing new approaches to using and encoding this information, to provide better generalisation capability.

There has been much previous work on "in the wild" action recognition in 2D data. Likewise, there has been significant work in recent years, on action recognition for 3D data "in the lab", due to the introduction of cheap consumer depth sensors. However, the crossover between the two areas, 3D recognition in the wild, has rarely been considered. It is important to address this problem as new sources of 3D data such as mobile devices are emerging, in addition to increasing levels of 3D broadcast data from television networks & film studios.

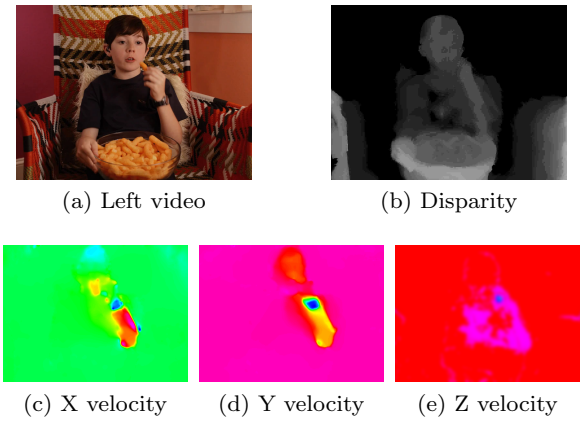
In the past, benchmark datasets such as KTH [45], Weizmann [3] or Kinect based datasets [33, 5] have been invaluable in providing comparative benchmarks to examine how competing approaches perform in action recognition and detection. However, these staged datasets now routinely have reported performance rates over 90 %, suggesting that they are reaching the end of their service to the community. "In the wild" datasets such as Hollywood [30], Hollywood2 [36] and our Hollywood 3D [15] provide a more challenging problem due to huge variability in appearance. These more natural datasets consist of actions extracted from a variety of Hollywood feature films. They provide a new level of complexity to the recognition community, arising from the natural within-class variation of unconstrained data, in-

cluding unknown camera motion, viewpoint, lighting, background and actors, and variations in action scale, duration, style and number of participants. While this natural variability is one of the strengths of the data, the lack of structure or constraints make classification an extremely challenging task.

The use of depth information can help to mitigate some of these factors. Lighting variations are generally not expressed in depth data, and actor appearance differences are eliminated (although differences in body shape remain). Additionally, depth provides useful cues for background segmentation, and occlusion detection. However, this also introduces new problems such as the inconsistency of 3D data obtained from disparate sources with unknown calibrations.

This paper discusses the Hollywood 3D benchmark dataset for 3D action recognition in the wild. In addition a broad experimental baseline is produced; many techniques for 2D "in the wild" recognition are extended to operate on depth data, including 2 feature descriptors and 5 interest point detectors. The effect of incorporating this depth data is comprehensively examined and the full source code of all these baseline techniques is provided to stimulate further research. Another novel feature descriptor is also proposed based on recent advances in the estimation of 3D motion fields (shown in Figures 1 and 2) [17]. This is coupled with a robust stereo auto-calibration framework to remove calibration inconsistencies from the resulting features without human intervention. The resulting calibrations are also provided to accompany the dataset. Finally a novel viewpoint-invariant feature encoding scheme is proposed to make it easier to recognise the similarities between different shots of the same action.

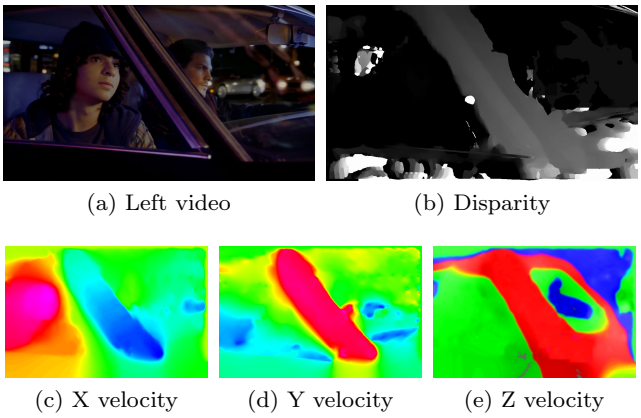
Compared to the initial presentation of the Hollywood 3D benchmark [15] and subsequent extensions [18], this paper includes makes two primary contributions. Firstly the evaluation has been significantly expanded including the popular dense trajectories [51] encoding technique, and a comparison of techniques other authors have proposed for this benchmark, including deep-learning based approaches [24,23], implicit calibration [27] and new feature descriptors [35,27]. This helps to provide additional insight into the field, beyond what was possible with the initial baseline experiments. Secondly, deeper discussion is also included of the scene-flow encoding, compared to the initial work in [18]. This includes a discussion of the 3D motion estimation pipeline.



(a) Left video (b) Disparity

(c) X velocity (d) Y velocity (e) Z velocity

Fig. 1: The appearance and disparity (top row) for a *Eat* action from the Hollywood 3D dataset. The 3D motion field is also shown on the bottom row. The primary motion is concentrated on the arm and head, which move towards each other.



(a) Left video (b) Disparity

(c) X velocity (d) Y velocity (e) Z velocity

Fig. 2: The appearance and disparity (top row) for a *Drive* action from the Hollywood 3D dataset. The 3D motion field is also shown on the bottom row. Note that the primary motion occurs on the foreground regions of the car, with secondary x and y motion on the passengers.

2 Related Work

A common practice in action recognition is to focus attention on parts of the scene which are identified as salient by interest point detectors [29]. One advantage is tractability, reducing the quantity of data for subsequent processing [11,30,33]. In the past, this approach has also helped to suppress irrelevant background information. Note that in this paper we make a distinction between saliency and interest points. The estimated "interest" of any image point is a continuous number referred to as the point's *saliency*. The parts of an image

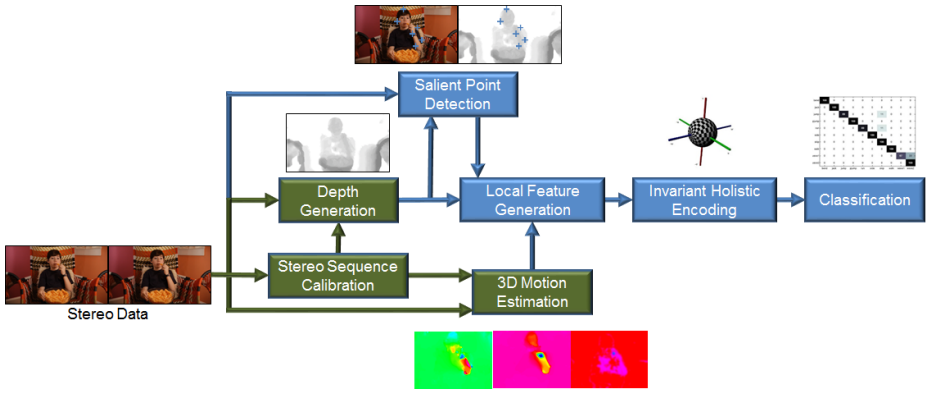


Fig. 3: "In the wild" action recognition pipeline, making use of depth information at various stages. The green elements refer to dataset pre-processing, while blue elements relate to the recognition pipeline.

which have a saliency score greater than a particular threshold are then collectively referred to as the set of *interest points*. Generally speaking interest points are also filtered via non-maxima suppression, to prevent duplicate entries.

Some approaches segment the actor, for example using the Kinect's user mask [12,33,5]. This enables complex "volumetric" descriptions of the actor's body over time [54,52,50,39]. However, for "in the wild" action recognition it remains challenging to segment the actor reliably, due to noisy 3D data, cluttered environments, and scenes containing multiple people. This is still an active area of research, with current techniques [7,55, 21] only able to provide rough bounding boxes, rather than the per pixel segmentations generally used in volumetric description. As such, it is still common to use interest point detectors for recognition in the wild. Recently an intermediate encoding approach, called Dense Trajectories, has been successful [51,10]. As the name implies, this is based on densely sampled local features, with temporal accumulation along the trajectories. In this paper we explore both interest point and dense trajectory encoding schemes.

When the salient parts of the sequence have been detected, it is common to compute local descriptors of these spatio-temporal region. Some of the most popular descriptors have been based on the gradient of the appearance information [45,31], spatio-temporal extensions to SIFT and SURF descriptors [46,53] and 2D motion information [6,37]. However, there has been little previous work on feature descriptors including depth information (which is generally encoded directly at the holistic level, with the aid of user masks).

The local descriptors from salient regions can then be accumulated into a single holistic description of the sequence. One way to achieve this is to take inspiration from highly successful Bag-of-Words techniques in the field of object recognition, and include an additional temporal dimension. This entails creating a codebook of exemplar features, and then accumulating occurrences of these exemplars spatially and temporally across the sequence. This is invaluable for generalisation, as it provides invariance to a range of important deformations, such as spatial and temporal translation, stretching and reflection. However, the accumulation 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 [30]. 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) [43]. 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 specific to our 3D motion features, which incorporate additional invariances such as scale and viewpoint invariance.

Some approaches avoid the bag-of-words holistic description route. These include data-mining and voting schemes [11] and chains of single-frame recognitions (for example using HMMs [4]). The lack of the accumulation step makes the learning task more complex due to the lack of invariances. However, it has the advantage that the exact location and time of the action is estimated. There has also been a recent rise in techniques based on deep learning, where bag-of-words is obviated. These range from simply pre-processing the input images [47, 26] to convolving over time [25] and estimating motion patterns [47].

3 Paper overview

The structure of the paper roughly mirrors the flow of the approach, as shown in Figure 3 (green elements indicate dataset pre-processing steps, and blue elements relate to the recognition pipeline). First, Section 4 covers the stereo data extraction approach, with details of the Hollywood 3D dataset. The proposed auto-calibration technique is then described in Section 5 and the results are used to extract 3D structure and motion information from the dataset as described in Section 6 using scene flow estimation.

The recognition pipeline is performed in 3 stages. Firstly salient points are detected using a range of detection schemes including a number of new schemes which incorporate the depth information, as discussed in Section 7. Next, feature descriptors are extracted from these salient points, encoding both appearance and depth information. These are discussed in detail in Section 8, and include extensions of two well known techniques and a novel motion descriptor based on 3D motion fields. These local descriptors are then accumulated over the sequence, with 3D motion features using viewpoint invariant encoding. In Section 9 we present results classifying these holistic descriptors using a Support Vector Machine (SVM), while exploiting depth data at various stages of the pipeline. Our conclusions about the benefits of depth data in natural action tasks, and the relative merits of the presented approaches, are then summarised in Section 10 where the use of depth consistently outperforms appearance-only recognition.

4 Extracting 3D Action Clips from Movies

There has been a sharp rise in commercially available 3D content, due to the emergence of high definition home media such as BluRay™and the introduction of 3D displays into the consumer market. Unfortunately much of this data is not suitable for action recognition. Most of the earlier 3D films were constructed via post-processing techniques (i.e. rotoscoping) from the original 2D data, and is fundamentally artificial, created for effect only. When depth data is extracted from these films, any fine-detailed depth variations within objects are missing, with scene depth simplified into a number of discrete depth planes. Additionally, films generated entirely through CGI (where 3D versions are much easier to produce), are unlikely to provide transferable information on human actions. For this dataset, we have focused on content captured using commercial camera rigs such as James Cameron's Fusion Camera System™or products from 3ality Technica. These use real stereo cameras rigs, making it possible to reconstruct accurate 3D depth maps.

Table 1: The number of unique training and test sequences for each action in the dataset. The top row is the training set and the bottom row is the test set.

NoAction	*Run*	*Punch*	*Kick*	*Shoot*	*Eat*	*Drive*	*UsePhone*	*Kiss*	*Hug*	*StandUp*	*SitDown*	*Swim*	*Dance*	Total
44	38	10	11	47	11	51	21	20	9	22	14	16	45	359
34	39	9	11	50	11	47	20	20	8	21	13	17	7	307

Most 3D films are too recent to have publicly available transcriptions, and subtitles alone rarely offer action cues, so automatic extraction techniques such as those employed by Marszalek *et al.* [36] are currently not feasible and manual labelling was used. This also ensures that all examples are well segmented from their carrier movies. This approach led to 650 clips (after class balancing) spread across 13 action classes, and a collection of 78 sequences containing no actions. These *NoAction* clips were automatically extracted as negative data, while ensuring no overlap with positive classes. The dataset was extracted from 14 films[1] and is publicly available [13]. In total the dataset contains over an hour of footage.

Within the dataset, actions are temporally localized to the frame level, ensuring non-discriminative frames at the start and end of sequences do not confuse training, and also improving separation of the *NoAction* class. The data is high definition (1920 by 1080 resolution) and is provided for both the left and right viewpoints at 24 frames per second.

To emphasize generalization, the 14 films comprising the dataset were split between the train and test sets on a per action basis. As such, each action is tested on actors and settings not seen in the training data. Certain actions are more common than others, and as in the Hollywood and Hollywood2 datasets, this prior distribution is reflected in the dataset. The number of training and test clips is shown in Table 1.

5 Stereo Sequence Auto-calibration

Every sequence in the dataset comprises an appearance from both the left and right viewpoint (extracted from

[1] Drive Angry, Tron: Legacy, My Bloody Valentine, Spy Kids: All The Time In The World, A Very Harold and Kumar Christmas, Final Destination 5, Underworld: Awakening, Step Up 3D, Sanctum, Avatar, Resident Evil: Afterlife, Pirates Of the Caribbean: On Stranger Tides, The Three Musketeers and Fright Night

the original film). For the stereo calibration of these viewpoints (i.e. the rotation and translation between the views and the intrinsics parameters of both cameras), the initial release of Hollywood 3D included only a single generic calibration for the entire dataset. This was based on the types of cameras and lenses commonly used in broadcast footage. However, this is overly simple as in reality the calibration varies greatly between films (where different cameras may be used), and even between different shots of the same film (the stereo-rig may be modified to change the strength of the perceived depth). This makes extracting consistent 3D information difficult from sequence to sequence, introducing a huge amount of artificial variation to the action classes and making recognition even more challenging. To mitigate this issue, we employ a stereo auto-calibration stage operating on the video pairs, which ensures extracted 3D features are more comparable across sequences.

The first step towards calibrating the 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 [34] points are extracted, where

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

based on the threshold λ_s. Non-maxima suppression is also applied to avoid prevent multiple interest points being generated by the same feature. Each resulting SIFT point comprises a space-time location $\boldsymbol{s}_i = (x, y, \tau)$ (where x, y is the image position and τ is the frame number). Additionally each point 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 [32]. The fundamental matrix is estimated by

$$\boldsymbol{F} = \arg\min_{\boldsymbol{F}'} \sum \epsilon_s\left(\boldsymbol{s}^l_i, \boldsymbol{s}^r_i | \boldsymbol{F}'\right), \tag{3}$$

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 [48].

Given the estimated $\boldsymbol{F}$ we can also extract the set of inlier correspondences,

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

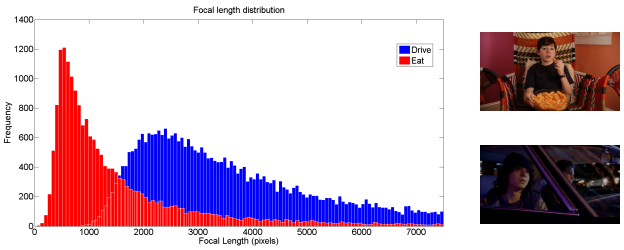


Fig. 4: Distribution of estimated focal lengths over 20000 repetitions, on the 2 different sequences pairs shown (wide-angle, close-up *Eat* shot, and extreme zoom *Drive* shot).

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.

5.1 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, and it would be jarring for the audience if one eye was zoomed differently to the other). 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 [28]. As with the estimation of $\boldsymbol{F}$, this is solved in a RANSAC framework, using the inliers to the epipolar geometry $\hat{C}$.

Unfortunately, for 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 $\boldsymbol{E}$. Due to the offline nature of the auto-calibration system, coupled with efficient PEP solvers, 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 4 shows the distribution of focal lengths estimated over a large number of 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. This is reasonable as it only needs to be performed once per sequence and is part of the pre-processing (i.e. it does not need to be repeated for each new experiment).

Finally, given our robust estimate of $\boldsymbol{E}$, it is possible to estimate the projection matrices $\boldsymbol{P}^l$ and $\boldsymbol{P}^r$ for the cameras [20]. This leads to 4 possible solutions. 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 within a single shot). 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. In practice the "no zooming" assumption is reasonable as most modern broadcast footage prefers to zoom between shots rather than within shots. Additionally, zooming is particularly unlikely while important actions are being performed. As a result only around 1% of the sequences in the Hollywood 3D dataset contain a zoom. Hence, the robustness gained by utilising this assumption is far more significant than any limitations it imposes.

The final limitation is that the reconstructions obtained by our calibration technique (available online), are only consistent with each other up to a similarity transform (for comparison, reconstructions using the original generic calibration was consistent up to a homography). The removal of projective distortions does reduce the variability in the data, but the remaining scale ambiguity still must be addressed during encoding.

6 Structure and 3D Motion Estimation

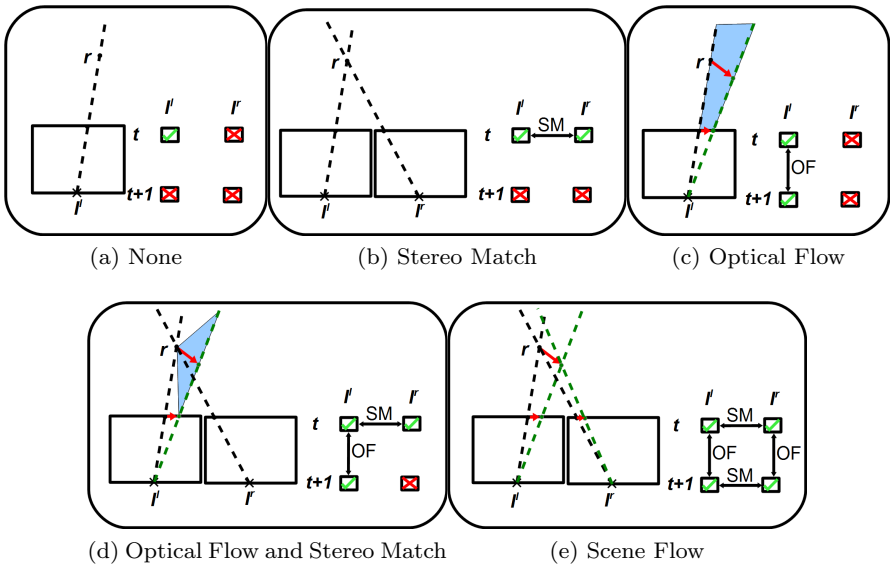
Once we have an estimated calibration, we can extract 3D structure and motion information from the dataset. For the structural information (i.e. stereo matching) we use a GPU-accelerated bilateral grid filtering approach, as described by Richardt *et al.* [41]. This technique attempts to estimate smooth but edge-preserving scene structures based on filtering theory, and unlike many other modern stereo techniques, scales well to the large amount of high-resolution content in the dataset.

To replace the 2D optical flow descriptor with a 3D "scene flow" descriptor, the 3D motion field was estimated. At time t this motion field is related to 4 different input frames. As shown in Figure 5 these are $\boldsymbol{I}_t^l$, $\boldsymbol{I}_t^r$, $\boldsymbol{I}_{t+1}^l$ and $\boldsymbol{I}_{t+1}^r$, and scene-flow estimation can be seen as trying to find the scene structure at 2 different times, while also estimating exactly how one structure warps into the other. As such, the task is then to find a set of 6D vectors $\boldsymbol{w}$ (comprising 3D world position x, y, z and 3D world velocity $\dot{x}, \dot{y}, \dot{z}$), which are consistent with all 4 images observations. There are many approaches to achieve this, but for estimating 3D motion fields for action recognition, we use the sampling based Scene Particles [16] approach. This is well suited to action recognition as it provides excellent performance at object boundaries [14] (where most salient point detections occur during human action sequences). In addition the approach is orders of magnitude faster than competing variational techniques, which is vital when dealing with the large quantities of data present in action recognition datasets.

For a given $\boldsymbol{w}$ we can use the estimated calibrations to find its projection in all 4 relevant images. We can then measure the quality of this $\boldsymbol{w}$ based on the Brightness Constancy Assumption which is common in motion estimation algorithms (i.e. for a real point on the scene structure, its appearance should not change over time, or with viewpoint). We measure conformance to this assumption using the variance of the appearance of the 4 projected points,

$$\boldsymbol{p}(\boldsymbol{I}^l, \boldsymbol{I}^r \mid \boldsymbol{w}) = \Gamma\left(\sum_{m \in \{l,r\}} \sum_{\tau=t\text{-}1}^{t} \frac{\left(\boldsymbol{I}_\tau^m(\boldsymbol{P}^m \boldsymbol{w}_\tau) - \bar{I}\right)^2}{4} \right), \tag{7}$$

where $\bar{I}$ is the mean observed appearance across all 4 frames, and $\boldsymbol{w}_\tau$ is the 3D end point of the flow cor-



(a) None (b) Stereo Match (c) Optical Flow

(d) Optical Flow and Stereo Match (e) Scene Flow

Fig. 5: Detection of correspondences between the two cameras, and between two points in time (shown in black and green). This illustrates the scene flow estimation task, and it's relation to optical flow (OF) and stereo matching (SM).

responding to frame τ. The Γ is an Intelligent Cost Function (ICF) [17] which is a type of robust scoring function using Gaussian Processes to reflect the distribution of motion errors in real scenarios.

This ICF is trained using the Middlebury stereo dataset [44]. Real correspondences are extracted using the ground truth, alongside deliberate "erroneous matches". These are then used to train a matching function which ideally separates correct and incorrect correspondences, while accounting for realistic lightning effects such as specularities and under/over-exposure. Conditioning the error on the observations in this way greatly improves robustness when compared to simply penalising the squared error.

We can embed this likelihood function in an efficient estimation scheme, such as a particle filter, in order to estimate consistent sets of motions $\boldsymbol{w}$ over the sequence, which maximize $\boldsymbol{p}(\boldsymbol{I}^l, \boldsymbol{I}^r \mid \boldsymbol{w})$. This efficient scheme is important, as the size of most action recognition datasets preclude the use of variational optimization based techniques [22], which our experiments indicated would take thousands of years to complete.

When using particle filtering to estimate the scene flow probability distribution, special attention must be paid to the coverage of the estimate. If all the "Scene Particles" are collected within a standard approximation framework, the particles will accumulate over time. In this case they will approximate the probability distribution in confident regions of the scene (such as strong edges) with great accuracy, while areas of low accuracy such as untextured regions will be represented sparsely by a small number of samples. To resolve this issue we adopt the "ray resampling" strategy, where the population of each viewing ray is treated independently during resampling to ensure even coverage of the scene. Another way to view this procedure is that every ray is assigned its own particle filter, but with the particles being capable of moving between rays at every frame increment. For further details see [17].

7 Interest Point Detection

Once the dataset has been preprocessed by extracting the structure, 3D motion and calibration information, the additional information present in this data may be exploited. This can be done at various stages of the pipeline, but we first look at interest point extraction, in order to detect more salient features, and discount irrelevant detections. The extended algorithms discussed in this section are based on the Harris Corners work by Laptev and Lindeberg [29], the Hessian points algorithm by Willems *et al.* [53] and the Separable Filters technique by Dollar *et al.* [8]. For a comparison of the original 2D interest point detection schemes (without the proposed depth-aware extensions), see the survey paper by Tuytelaars and Mikolajczyk [49].

7.1 4D Interest Points

The Harris Corner [19] is a frequently used interest point detector, which was extended into the spatio-temporal domain by Laptev *et al.* [29]. The detector is based on the second-moment-matrix (ψ) of the Gaussian smoothed spatio-temporal volume (I). Interest points are detected in the spatio-temporal volume as locations where ψ contains 3 large eigenvalues, i.e. there is strong intensity variation along 3 distinct spatio-temporal axes. To avoid eigenvalue calculation at every point, the following approximate formulation is used where (u, v, w) is a spatio-temporal position and k is typically 0.001:

$$H(u,v,w) = \det\left(\psi\left(u,v,w\right)\right) - k\ \mathrm{trace}\left(\psi\left(u,v,w\right)\right)^3 . \tag{8}$$

To extend the operator into 4D, the power of the trace is increased, and ψ must be expanded to a 4 by 4 matrix, incorporating the differential (I_z) along z. However, the combination of appearance and depth streams does not constitute volumetric data (i.e. measurements are not dense along the new dimension as in an MRI scan). This is referred to as 3.5D rather than 4D data, and gradients cannot be directly calculated along the z axis. Instead, the relationship between the spatio-temporal gradients of the depth stream and those of the appearance stream are exploited. If I_x,I_y,I_t are intensity gradients along the spatial and temporal dimensions and D_x,D_y,D_t are the gradients of the depth stream (and omitting the spatio-temporal location (u, v, w)) a simple application of the chain rule allows us to estimate I_z.

$$I_z = \frac{\frac{dI}{dx}}{\frac{dz}{dx}} + \frac{\frac{dI}{dy}}{\frac{dz}{dy}} + \frac{\frac{dI}{dt}}{\frac{dz}{dt}} = \frac{I_x}{D_x} + \frac{I_y}{D_y} + \frac{I_t}{D_t} \tag{9}$$

This allows us to define ψ as

$$\psi = g(\sigma^2,\tau^2) * \begin{pmatrix} I_xI_x & I_xI_y & I_xI_t & I_xI_z \\ I_xI_y & I_yI_y & I_yI_t & I_yI_z \\ I_xI_t & I_yI_t & I_tI_t & I_tI_z \\ I_xI_z & I_yI_z & I_tI_z & I_zI_z \end{pmatrix}, \tag{10}$$

where $g(\sigma^2, \tau^2)$ is a Gaussian smoothing function, with spatial and temporal scales defined by σ and τ respectively.

The set of 4D Harris interest points $F_{4D\text{-}Ha}$ is defined as the set of spatio-temporal locations within the sequence, for which H is greater than the threshold $\lambda_{4D\text{-}Ha}$

$$F_{4D\text{-}Ha} = \{u,v,w | H\left(u,v,w\right) > \lambda_{4D\text{-}Ha}\}. \tag{11}$$

In [53], Willems *et al.* extended the Beaudet Saliency Measure [1] into the spatio-temporal domain. Rather than the second-moment-matrix of Laptev *et al.* they calculated the Hessian (μ) of the Gaussian smoothed spatio-temporal volume (I). The detected interest points relate to areas with strong second order intensity derivatives, including both blobs and saddles.

As in the 4D Harris scheme, gradients along z are estimated using the relationships between the depth and intensity stream gradients. This allows the 4D Hessian μ to be calculated as

$$\mu = g(\sigma^2,\tau^2) * \begin{pmatrix} I_{xx} & I_{xy} & I_{xt} & I_{xz} \\ I_{xy} & I_{yy} & I_{yt} & I_{yz} \\ I_{xt} & I_{yt} & I_{tt} & I_{tz} \\ I_{xz} & I_{yz} & I_{tz} & I_{zz} \end{pmatrix}. \tag{12}$$

The set of interest points $F_{4D\text{-}He}$ is calculated as the set of spatio-temporal locations, for which the determinant of μ is greater than the threshold $\lambda_{4D\text{-}He}$

$$F_{4D\text{-}He} = \{u,v,w | \det\left(\mu\left(u,v,w\right)\right) > \lambda_{4D\text{-}He}\}. \tag{13}$$

7.2 Interest Points in 3.5D

In part, the Harris and Hessian interest point operators are motivated by the idea that object boundary points are highly salient, and that intensity gradients relate to boundaries. However, depth data directly provides boundary information, rendering the estimation of the intensity gradient along z somewhat redundant. An alternative approach would be to employ a "3.5D" representation, using a pair of complimentary 3D spatio-temporal volumes, from the appearance and depth sequences. This can be applied to the Harris measure,

$$F_{3.5D\text{-}Ha} = \{u,v,w | \theta(u,v,w) + \alpha\phi(u,v,w) > \lambda_{3.5D\text{-}Ha}\}, \tag{14}$$

and the Hessian measure

$$F_{3.5D\text{-}He} = \{u,v,w | \det(\xi) + \alpha\det(\zeta) > \lambda_{3.5D\text{-}He}\}. \tag{15}$$

Where θ and ϕ are Equation 8 applied to the appearance and depth streams respectively, while ξ and ζ are the 3 by 3 Hessians. The relative weighting of the appearance and depth information is controlled by α. This approach exploits complimentary information between the streams, to detect interest points where there are large intensity changes and/or large depth changes.

7.3 3.5D Separable Filters

A third highly successful approach to interest point detection, is the Separable Linear Filters technique of Dollar *et al.* [8]. Peaks are detected within a spatio-temporal volume, after filtering with a 2D Gaussian in

the spatial dimensions, and a quadrature pair of Gabor filters h_{ev} and h_{od} along the temporal dimension,

$$S(I) = (I * g(\sigma^2) * h_{ev})^2 + (I * g(\sigma^2) * h_{od})^2. \quad (16)$$

Employing the same 3.5D approach used for the Harris and Hessian detectors, leads to

$$F_{3.5D\text{-}S}=\{u, v, w | S(I(u, v, w))+\alpha S(D(u, v, w))>\lambda_{3.5D\text{-}S}\}, \quad (17)$$

where I and D are the appearance and depth streams respectively.

7.4 Dense Trajectories

An alternative scheme to detecting interest points which we explore is accumulation via densely sampled trajectories. In this approach, feature points are densely sampled in the first frame, and are tracked over time by median filtering of the motion field [51]. Any trajectories which are static are assumed to be part of the background and are ignored. To prevent drift during tracking, an upper limit is placed on the length of the trajectories, after which a new grid of dense points is sampled.

Local features can then be accumulated over the trajectory to form a single descriptor, encompassing the spatio-temporal behaviour of a particular part of the scene.

8 Feature Descriptors

Once feature points have been detected, the next stage is to extract descriptors to encode the characteristics of these salient regions for classification. The descriptors can be based on various types of information, including appearance, motion and saliency, but we wish to also include our additional depth information.

8.1 RMD

The Relative Motion Descriptor (RMD) introduced by Oshin *et al.* [40] has been shown to perform well in a large range of action recognition datasets, while making use of only the saliency information obtained during interest point detection. A spatio-temporal volume η is created, containing the interest point detections and their strengths. The saliency content of a sub-cuboid, with origin at (u, v, w) is defined for a sub-cuboid of dimensions $(\hat{u}, \hat{v}, \hat{w})$ as

$$c(u, v, w) = \sum_{\boldsymbol{\gamma}=0}^{(\hat{u},\hat{v},\hat{w})} \eta([u, v, w] + \boldsymbol{\gamma}). \quad (18)$$

For efficiency this is implemented as an integral volume. The descriptor δ of the saliency distribution at a position (u, v, w) can then be formed, by performing N comparisons of the content of two randomly offset spatio-temporal sub-cuboids, with origins at $(u, v, w) + \boldsymbol{\beta}$ and $(u, v, w) + \boldsymbol{\beta}'$:

$$\delta(u,v,w)=\sum_{n=0}^{N}\begin{cases} 2^n \text{ if } c([u, v, w]+\boldsymbol{\beta}_n) > c([u, v, w]+\boldsymbol{\beta}'_n) \\ 0 \text{ otherwise} \end{cases} \quad (19)$$

Note that the collections of offsets $\boldsymbol{\beta}_{0..N}$ and $\boldsymbol{\beta'}_{0..N}$ are randomly selected prior to training, and then maintained, rather than selecting new offsets for each clip.

By extracting δ at every location in the sequence, a histogram may be constructed, which encodes the occurrences of relative saliency distributions within the sequence, without requiring appearance data or motion estimation. Increasing the number of comparisons N leads to improved descriptiveness, however the resulting histograms also become more sparse. A common alternative is to compute several δ histograms, each using different collections of random offsets $\boldsymbol{\beta}_{0..N}$ and $\boldsymbol{\beta'}_{0..N}$. The resulting histograms are then concatenated, with the result encoding more information without sparsifying the histogram. However, this comes at the cost of the independence between bins, i.e. introducing some possible redundancies.

We propose extending the standard RMD described above, by storing the saliency measurements within a 4D integral hyper-volume, so as to encode the behaviour of the interest point distribution across the 3D scene, rather than within the image plane. The 4D integral volume can be populated by extracting the depth measurements at each detected interest point. RMD-4D descriptors can then be extracted, using comparisons between pairs of sub-hypercuboids. The resulting histogram encodes relative distributions of saliency, both temporally, and in terms of 3D spatial location. As with the original RMD, the descriptor can be applied in conjunction with any interest point detector and is not restricted to the extended interest point detectors described in Section 7 (provided that a depth video is available during descriptor extraction).

8.2 Bag of Visual Words

One of the most successful approaches in action recognition is to concatenate a range of local descriptors and to calculate a bag of words representation. Laptev *et al.* [30] used this approach to great effect to combine HOG and HOF descriptors (defined as G and F). Both histograms are computed over a small window, storing coarsely quantized image gradient and optical flow vectors, respectively. This provides a descriptor ρ of the visual appearance and local motion around the salient point at $I(u,v,w)$.

$$\rho(u,v,w) = \left(G\left(I\left(u,v,w\right)\right), F\left(I\left(u,v,w\right)\right)\right) \tag{20}$$

When accumulating ρ over space and time, a Bag of Words (BOW) approach is employed. Clustering is performed on all ρ obtained during training, creating a codebook of distinctive descriptors. During recognition, all newly extracted descriptors are assigned to the nearest cluster center from the codebook, and the frequency of each clusters occurrences are accumulated. In this work K-Means clustering is used, with a Euclidean distance function as in [30]. To extend ρ to 4D, we include a Histogram of Oriented Depth Gradients (HODG):

$$\rho(u,v,w) = \Big(G\big(I(u,v,w)\big), F\big(I(u,v,w)\big), G\big(D(u,v,w)\big)\Big). \tag{21}$$

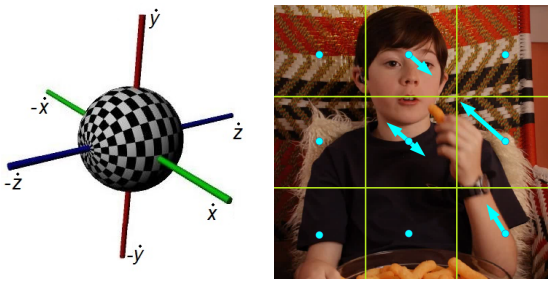
Thus the descriptor encapsulates local structural information, in addition to local appearance and motion. The bag of words approach is applied to this extended descriptor, as in the original scheme. Importantly, this descriptor is not dependent on the interest point detector, provided the HODG can be calculated from the depth stream D. By normalising these local descriptors, we are able to resolve the scale ambiguity which remained in our auto-calibration of Section 5.

8.3 3D motion descriptors

The inclusion of structural (depth) features into the bag of words descriptor does not fully exploit the additional information in the Hollywood 3D dataset. During pre-processing we also extracted the 3D motion fields for the dataset, which can be used as a replacement for the optical flow features F. We refer to these 3D motion descriptors as "Histograms of Oriented Sceneflows" (HOS).

Given the dense 3D flow field $(\dot{x},\dot{y},\dot{z})$, we can extract a local 3D motion descriptor using the spherical coordinate system

$$\gamma = \operatorname{atan}\left(\frac{\dot{y}}{\dot{x}}\right) \quad \text{and} \quad \delta = \operatorname{atan}\left(\frac{\dot{z}}{\dot{y}}\right), \tag{22}$$



(a) Orientation histogram (b) Encoded motion field

Fig. 6: (a) Orientation bins visualised with alternating white and black squares. γ is rotation around the w axis. δ is rotation around the u axis. (b) a scene divided into a 3 by 3 grid of subregions, with the motion of each subregion aggregated.

to describe the 3D orientation of flow vectors. Note that γ refers to the "in plane" orientation (from the viewpoint of the left camera) *i.e.* when γ is 0°, the motion is toward to the top of the image, when γ is 90° the motion is toward the right of the image, etc. In contrast δ 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 $\mathbf{H}$ (see Figure 6) which can be included into the bag of words descriptor ρ. 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. As with HoG, HoF and HoDG this histogramming discards much of the spatial information. However, some general attributes are maintained by separating the region into several neighbouring blocks, and encoding each of them independently as $\mathbf{H}_{1\ldots n}$. These sub-region spherical histograms are then combined to form the overall descriptor $\mathbf{H}$. 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 6a, 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 mitigate this [42].

As above we normalise the descriptors to resolve the scale ambiguity between sequences. Thus, even though motion fields are consistent only up to a similarity trans-

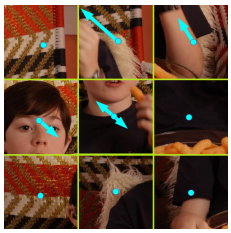


Fig. 7: $\bar{\mathbf{H}}^{\mathbf{r}}$ The subregions of the encoded motion field are re-arranged such that the region of maximum motion occurs first. This provides some degree of invariance to camera roll.

form, the normalised spherical histograms,

$$\bar{\mathbf{H}} = \frac{\mathbf{H}}{|\mathbf{H}|}, \tag{23}$$

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 important for "in the wild" recognition, with many different actors, each of whom have their own action style.

8.4 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 $\mathbf{H}_{1\ldots n}$ such that the subregion containing the largest amount of motion occurs first (similar to the orientation normalisation used in shape context [2], SIFT [34], Uniform LBPs [38] etc.). This re-arranged, roll-invariant, descriptor is referred to as $\bar{\mathbf{H}}^{\mathbf{r}}$ (see Figure 7).

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 $\mathbf{H}$ and label its 3D orientation as $\hat{\phi}$,$\hat{\psi}$ then we can redefine

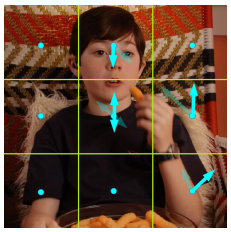


Fig. 8: $\bar{\mathbf{H}}^{\mathbf{p}}$ The orientation of the strongest motion vector in the scene is used to normalise the histograms, providing robustness to camera pitch and roll.

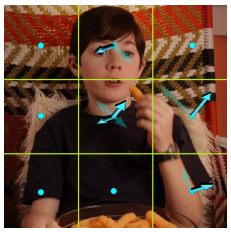


Fig. 9: $\bar{\mathbf{H}}'$ A new set of 3D axes is chosen using PCA, relating to the dominant 3D motion orientations in the scene. This provides complete invariance to camera viewpoint change.

our local orientations in relation to this flow vector,

$$\gamma^p = \text{atan}\left(\frac{\dot{y}}{\dot{x}} - \hat{\phi}\right) \quad \text{and} \quad \delta^p = \text{atan}\left(\frac{\dot{z}}{\dot{y}} - \hat{\psi}\right). \tag{24}$$

The resulting descriptors $\bar{\mathbf{H}}^{\mathbf{p}}$ obtained when encoding γ^p,δ^p makes the flow vectors robust to camera pitch (rotation around the x axis) in addition to roll, as shown in Figure 8.

However, due to the separation of γ and δ our descriptors are still not resistant to camera pans (rotation around the y axis, which at 90 degrees causes γ orientation to become δ 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 $\dot{x}', \dot{y}', \dot{z}'$ (the eigenvectors of the motion field covariance). Computing orientation using these basis vectors,

$$\gamma' = \text{atan}\left(\frac{\dot{y}'}{\dot{x}'}\right) \quad \text{and} \quad \delta' = \text{atan}\left(\frac{\dot{z}'}{\dot{y}'}\right), \tag{25}$$

leads to a descriptor $\bar{\mathbf{H}}'$ which is invariant to all 3 types of camera viewpoint change, and also robust to outlier motions. See Figure 9 for an illustration.

9 Results

Classification was performed for all tests, with a multi-class SVM. Note that tests were performed using other

Table 2: Average precision per class, on the 3D action dataset, for a range of sparse interest point detectors, including simple spatio-temporal interest points, depth aware extensions and Dense Trajectory encoding. The Bag of Visual Words (HoG/HoF/HoDG) feature encoding was used. Classes are shown in bold, for schemes outperforming both of the simple spatio-temporal interest point schemes.

Action	3D-S	3D-Ha	**4D-He**	**4D-Ha**	**3.5D-S**	**3.5D-He**	**3.5D-Ha**	Dense-Traj
NoAction	11.4	12.1	**12.2**	**12.9**	11.4	12.0	**13.7**	**14.2**
Run	12.6	19.0	15.9	**22.4**	12.7	**21.8**	**27.0**	**27.3**
Punch	2.9	10.4	2.9	4.8	2.9	5.7	5.7	9.4
Kick	3.6	9.3	4.2	4.3	3.8	3.7	4.8	**9.9**
Shoot	16.2	27.9	18.9	17.2	16.2	16.2	16.6	**33.8**
Eat	3.6	5.0	3.6	**5.3**	3.6	**7.7**	**5.6**	**11.9**
Drive	15.3	24.8	**25.6**	**69.3**	15.5	**76.5**	**69.6**	**56.8**
UsePhone	6.5	6.8	**14.7**	**8.0**	6.5	**17.7**	**7.6**	**12.4**
Kiss	6.5	8.4	**8.5**	**10.0**	6.5	**9.4**	**10.2**	**11.4**
Hug	2.6	4.3	3.5	**4.4**	2.6	3.4	**12.1**	5.3
StandUp	6.8	10.1	7.0	7.6	6.9	9.1	9.0	**17.6**
SitDown	4.2	5.3	4.5	4.2	4.2	4.3	**5.6**	**6.7**
Swim	5.5	11.3	7.8	5.5	5.5	5.9	7.5	8.4
Dance	2.3	10.1	4.2	**10.5**	2.2	3.8	7.5	**26.5**
Average	7.1	12.6	9.8	**13.3**	7.1	**13.4**	**14.1**	**17.9**

SVM kernels including linear, χ^2 and multi-χ^2. However, linear kernels were found to perform poorly while χ^2 kernels greatly increased computation time and had little effect on performance. Thus for clarity we only present the results using RBF kernels. To facilitate comparisons with the Hollywood 1 and 2 datasets, the Average Precision (AP) measure was used, as explained in the PASCAL VOC [9]. Relevant source code is available online along with the data [13]. RMD tests were performed with 4 binary comparisons per histogram ($N = 4$), concatenating 10 descriptor histograms. BOW tests were performed with 4,000 cluster centres (as suggested in [30]), with the local histogram descriptors calculated using a block size of 3 by 3, with 8 orientation bins. For the 3D motion (HOS) features, each subregion histogram uses 4×4 bins in the γ and δ orientations.

To aid clarity, we will first examine 3 components of the framework independently; the interest point detection, the local descriptors, and the motion feature encoding. We will then summarise the most effective technique in each area and compare against other techniques designed for "in the wild" 3D action recognition, including [27] which uses auto-encoders to implicitly model uncalibrated structural information, and [24, 23] which use Extreme Learning Machines on Dense Trajectory encoded HoG/HoF/MBH descriptors.

9.1 Interest Point Analysis

First we examine the benefits of including depth information during interest point detection. Clearly this is the least significant stage of the pipeline to include depth; no information from the depth stream is encoded in the descriptors, and depth information cannot be used by the classifier to distinguish actions. Instead, the depth stream is used only to make more informed decisions about which regions of the appearance stream to encode and which to discard. This is particularly true after the encoded regions are accumulated into a single holistic descriptor.

For these comparisons the standard Bag of Words descriptor (see Section 8.2, HoG/HoF/HoDG) is used. The traditional spatio-temporal interest points (Separable Filters 3D-S and Harris Corners 3D-Ha) are compared to the proposed depth aware interest point detectors and the currently state-of-the-art Dense Trajectories approach (also using the 3D Bag of Words descriptor). The AP for each class is shown in Table 2, with bold entries indicating performance greater than both of the standard spatio-temporal schemes.

The type of saliency measure used has a surprisingly large effect on the performance, with the average performance for the best scheme being more than double that of the worst, even using the same feature encoding. For the standard spatio-temporal schemes, Harris points (3D-Ha) outperform separable filter points (3D-S) for all actions. This is also reflected in the depth aware schemes, and is unsurprising, as separable filters were designed primarily for computational speed. Hessian based interest points prove less informative than the extended Harris operators in both the 4D and 3.5D case. For all detectors, the 4D schemes outperform their standard spatio-temporal counterpart, while the 3.5D approach proves more informative than the direct 4D

extension. This confirms the belief that the calculation of intensity gradients along z is redundant, and that the combination of intensity and structure gradients is a stronger measure of saliency. The Dense-Trajectory approach proves to be the more descriptive than the sparse interest points, when used to encode the same feature descriptors. However this comes at the price of significantly increased computational cost due to it's dense nature.

Interestingly, certain actions consistently perform better, when described by depth aware interest points. These are actions such as *Kiss*, *Hug*, *Drive* and *Run* where there is an informative foreground object, which depth aware interest points are better able to pick out. In contrast, actions such as *Swim*, *Dance* and *Shoot* are often performed against a similar depth background, or within a group of people, and the inclusion of depth in the saliency measure is less valuable. This suggests that a combination of standard spatio-temporal, and depth aware schemes may prove valuable.

The complexity of the depth aware interest point detectors remains of the same order as their spatio-temporal counterparts (linear with respect to u, v and w). Naturally the multiplicative factor is increased however, with 3.5D techniques being roughly twice as costly, and 4D techniques taking 4 times as long.

9.2 Structural Descriptor Analysis

Next, the use of structural information at the feature level was explored, using the best performing Interest Point detection schemes from the previous analysis. These results are shown in Table 3. The best performing descriptor for each saliency measure is shown in bold. The RMD descriptor has it's own holistic accumulation scheme and so does not fit well with Dense Trajectory encoding. However, the Bag of words descriptor can be evaluated in both sparse and dense scenarios.

It should be noted that previous work using Dense Trajectories has employed them as both an accumulation scheme, and as a feature descriptor. It is possible that this could provide additional information, further improving performance. However, the purpose of this analysis is to quantify the value of depth based features. Further, it should be noted that features such as Eigenjoints [54] or Actionlets [52] (which are currently state-of-the-art for non-"in the wild" 3D action recognition) cannot be evaluated as the user masks and body skeletons normally provided by the Kinect, cannot be produced in this more complicated scenario.

Both types of descriptor show a consistent improvement when incorporating structural information, with increases of around 20 % in both average precision and correct classification. This demonstrates the value of such features for recognizing actions in the wild. Overall, the Bag of Words descriptors perform somewhat better than the RMD descriptors. This is unsurprising as the RMD relies only on interest point detections, without the inclusion of any visual and motion information.

Table 3: Correct Classification rate and Average Precision for different local features using the 2 top performing saliency measures. The best feature for each saliency measure is shown in bold.

Descriptor	Saliency	CC	AP
RMD	3.5D-Ha	12.3	11.9
RMD-4D	3.5D-Ha	17.2	14.4
HoG/Hof	3.5D-Ha	17.9	13.0
HoG/Hof/HoDG	3.5D-Ha	**21.8**	14.1
HoG/Hof/HoDG	Dense-Traj	20.8	**17.9**

The complexity of the RMD-4D is greater than the standard RMD (being linear in the range of depth values, as well as in u, v and w). This is somewhat mitigated by the use of integral volumes however, meaning that runtimes are still on the order of seconds using a single CPU. In contrast the extraction of HoDG features relates to only a 50 % increase in runtime of the standard bag of words descriptor (although the complexity remains linear). However the increased feature vector length does lead to an increased cost during codebook generation, as K-Means is generally linear in the number of dimensions.

9.3 Motion Descriptor Analysis

Taking the best descriptor so far (HoG/HoF/HoDG), we next investigate improvements to the motion based portion of the descriptor, in light of the available depth information. This set of experiments was performed using the 3.5D-Ha interest point detector.

In Table 4 we can see that the raw 3D motion features ($\bar{\mathbf{H}}$ -uncal), directly attainable from the dataset with a generic calibration, perform rather poorly, offering only a minor improvement over 2D motion based features (HOF[15]). The use of our proposed stereo sequence auto-calibration ($\bar{\mathbf{H}}$) dramatically improves performance, almost tripling 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.

Table 4: Per class Average Precision scores using various types of motion features, including 2D motions, uncalibrated 3D motions, unnormalised 3D motions, and calibrated motions encoding varying levels of invariance to camera viewpoint change.

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

The results also show that the unnormalised features ($\mathbf{H}$), which are not scale invariant, perform uniformly worse than their normalised counterparts. It is worth noting, however, that Hollywood 3D does not 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 $\bar{\mathbf{H}}$ features.

The viewpoint invariant encoding schemes of Section 8.4 (upgrading the motion fields to fully consistent, rather than "up to a rotation") provide more modest improvements. Including roll invariance ($\bar{\mathbf{H}}^{\mathbf{r}}$) gives only a small performance increase, probably 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 ($\bar{\mathbf{H}}^{\mathbf{p}}$) 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). In addition, the motions are generally stronger for these actions which may make the dominant direction more reliable. The final scheme ($\bar{\mathbf{H}}'$), including full viewpoint invariance by estimating new motion orientation axes, provides the best performance. It is interesting to note that all of these encoding schemes actually discard 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.

It should be noted that there are more advanced features such as Motion Boundary Histograms (MBH) which have proven very powerful for 2D action recognition in recent years. However, depth-aware extensions of these complex features are beyond the scope of this work.

9.4 Current State of the Art

From this extensive analysis, our best approach to "in the wild" 3D action recognition is to use the Dense Trajectory encoding scheme, combined with the bag-of-words descriptors including 3D structure and motion. In addition, we found that calibrated 3D motion features are far more powerful than their 2D counterparts, especially when encoded with full viewpoint invariance ($\bar{\mathbf{H}}'$). In Table 5 we show these 3 techniques independently (without using depth aware components in the rest of the pipeline) against the combination of all 3 techniques in a single framework. We also show results for the other techniques currently submitted to our online leaderboard [13], including several deep-learning based techniques. Clearly the calibrated 3D motion features (HOS) offer the largest improvements, but all of the proposed techniques offer significant improvement over their spatiotemporal counterparts (quantified in parentheses). In addition these gains are complementary, and in combination provide a more than three-fold improvement in performance.

10 Conclusion

In this paper, we introduced a large publicly available corpus of 3D data (and code) for the action recognition community to compare techniques in natural environments. Further, we have demonstrated the intrinsic value of this 3D information throughout the Natural Action Recognition pipeline. Specifically, a variety of new interest point detection algorithms incorporating depth data have been shown to improve action recognition rates, doubling performance in some cases, even using standard features. Additionally, popular feature descriptors have been modified to encode structural information, demonstrating an average of 20 % additional improvement in performance. We have also discussed the use of 3D information for estimating a new,

Table 5: The current state of the art for in the wild 3D action recognition. For each of our depth-aware extensions, the improvement over spatio-temporal techniques is shown in parentheses.

Algorithm	SAE-MD (Av)[27]	MVRELM [24]	Disp-Pyr {1,3}[23]	Enriched IPs [35]	3.5D IPs	Structure Features	3D Motion (HOS')	Den-Traj HOS HoG/HoDG
mAP	26.1	29.9	30.5	30.1	14.1 (+12%)	17.9 (+22%)	36.9 (+293%)	**37.4**

more advanced class of motion features based on scene-flow. These provide recognition rates significantly better than previously state-of-the-art techniques, particularly when utilising the proposed viewpoint-invariant encoding.

In fact, our results demonstrate that invariances are vital for features used in recognition "in the wild". The proposed robust stereo sequence calibration step is needed to fully exploit the power of 3D information in the presence of large intra-class variation. As a result, the estimated calibrations for the dataset have been made publicly available, in addition to the stereo data, reconstructed depth and code for the baseline techniques.

Future work should focus on more complex feature descriptors, particularly focused on mitigating the sparsity which may arise in higher dimensional feature spaces. It would also be useful to develop further the invariant holistic encoding schemes for local features, preserving the invariances encoded in our 3D motion features without discarding so much relational information.

References

1. Beaudet, P.: Rotationally invariant image operators. In: Joint Conference on Pattern Recognition (1978)
2. Belongie, S., Malik, J., Puzicha, J.: Shape matching and object recognition using shape contexts. PAMI (2002)
3. Blank, M., Gorelick, L., Shechtman, E., Irani, M., Basri, R.: Actions as space-time shapes. In: ICCV (2005)
4. Brand, M., Oliver, N., Pentland, A.: Coupled hidden Markov models for complex action recognition. In: CVPR (1997)
5. Cheng, Z., Qin, L., Ye, Y., Huang, Q., Tian, Q.: Human daily action analysis with multi-view and color-depth data. In: ECCV Workshop (2012)
6. Dalal, N., Triggs, B., Schmid, C.: Human detection using oriented histograms of flow and appearance. In: Proc. ECCV. Graz, Austria (2006)
7. Desai, C., Ramanan, D.: Detecting actions, poses, and objects with relational phraselets. In: ECCV (2012)
8. Dollar, P., Rabaud, V., Cottrell, G., Belongie, S.: Behavior recognition via sparse spatio-temporal features. In: Visual Surveillance and Performance Evaluation of Tracking and Surveillance Workshop (2005)
9. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL visual object classes (VOC) challenge. IJCV (2010)
10. Gilbert, A., Bowden, R.: Data mining for action recognition. In: ACCV (2014)
11. Gilbert, A., Illingworth, J., Bowden, R.: Action recognition using mined hierarchical compound features. PAMI (2011)
12. Gorelick, L., Blank, M., Shechtman, E., Irani, M., Basri, R.: Actions as space-time shapes. PAMI (2007)
13. Hadfield, S., Bowden, R.: Hollywood 3d dataset, code and calibration. DOI 10.15126/surreydata.00808228. URL `http://cvssp.org/Hollywood3D`
14. Hadfield, S., Bowden, R.: Kinecting the dots: Particle based scene flow from depth sensors. In: ICCV (2011)
15. Hadfield, S., Bowden, R.: Hollywood 3D: Recognizing actions in 3D natural scenes. In: CVPR (2013)
16. Hadfield, S., Bowden, R.: Scene flow estimation using intelligent cost functions. In: BMVC (2014)
17. Hadfield, S., Bowden, R.: Scene particles: Unregularized particle based scene flow estimation. PAMI (2014)
18. Hadfield, S., Lebeda, K., Bowden, R.: Natural action recognition using invariant 3D motion encoding. In: ECCV (2014)
19. Harris, C., Stephens, M.: A combined corner and edge detector. In: Alvey Vision conference, pp. 147–152 (1988)
20. Hartley, R., Zisserman, A.: Multiple View Geometry in Computer Vision. Cambridge University press (2000)
21. Hoai, M., Ladicky, L., Zisserman, A.: Action recognition from weak alignment of body parts. In: BMVC (2014)
22. Huguet, F., Devernay, F.: A variational method for scene flow estimation from stereo sequences. In: ICCV (2007)
23. Iosifidis, A., Tefas, A., Nikolaidis, N., Pitas, I.: Human action recognition in stereoscopic videos based on bag of features and disparity pyramids. In: European Signal Processing Conference (2014)
24. Iosifidis, A., Tefas, A., Pitas, I.: Multi-view regularized extreme learning machine for human action recognition. In: Artificial Intelligence: Methods and Applications. Springer International Publishing (2014)
25. Ji, S., Xu, W., Yang, M., Yu, K.: 3d convolutional neural networks for human action recognition. Pattern Analysis and Machine Intelligence, IEEE Transactions on **35**(1), 221–231 (2013)
26. Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., Fei-Fei, L.: Large-scale video classification with convolutional neural networks. In: Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on, pp. 1725–1732. IEEE (2014)
27. Konda, K., Memisevic, R.: Learning to combine depth and motion. arXiv preprint arXiv:1312.3429 (2013)
28. Kukelova, Z., Bujnak, M., Pajdla, T.: Polynomial eigenvalue solutions to the 5-pt and 6-pt relative pose problems. In: BMVC (2008)
29. Laptev, I., Lindeberg, T.: Space-time interest points. In: ICCV (2003)
30. Laptev, I., Marszalek, M., Schmid, C., Rozenfeld, B.: Learning realistic human actions from movies. In: CVPR (2008)
31. Laptev, I., Perez, P.: Retrieving actions in movies. In: ICCV (2007)

32. Lebeda, K., Matas, J., Chum, O.: Fixing the locally optimized RANSAC. In: BMVC (2012)
33. Li, W., Zhang, Z., Liu, Z.: Action recognition based on a bag of 3D points. In: CVPR Workshops (2010)
34. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. IJCV (2004)
35. Mademlis, I., Iosifidis, A., Tefas, A., Nikolaidis, N., Pitas, I.: Stereoscopic video description for human action recognition. In: Symposium on Computational Intelligence (2014)
36. Marszalek, M., Laptev, I., Schmid, C.: Actions in context. In: CVPR (2009)
37. Messing, R., Pal, C., Kautz, H.: Activity recognition using the velocity histories of tracked keypoints. In: ICCV (2009)
38. Ojala, T., Pietikäinen, M., Mäenpää, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. PAMI (2002)
39. Oreifej, O., Liu, Z.: HON4D: Histogram of oriented 4D normals for activity recognition from depth sequences. In: CVPR (2013)
40. Oshin, O., Gilbert, A., Bowden, R.: Capturing the relative distribution of features for action recognition. In: Automatic Face & Gesture Recognition (2011)
41. Richardt, C., Orr, D., Davies, I., Criminisi, A., Dodgson, N.: Real-time spatiotemporal stereo matching using the dual-cross-bilateral grid. In: ECCV (2010)
42. Saff, E.B., Kuijlaars, A.B.: Distributing many points on a sphere. The Mathematical Intelligencer (1997)
43. Sapienza, M., Cuzzolin, F., Torr, P.: Learning discriminative space-time actions from weakly labelled videos. In: BMVC (2012)
44. Scharstein, D., Szeliski, R.: High-accuracy stereo depth maps using structured light. In: IEEE Computer Society Conf. CVPR, vol. 1 (2003)
45. Schuldt, C., Laptev, I., Caputo, B.: Recognizing human actions: a local SVM approach. In: ICPR (2004)
46. Scovanner, P., Ali, S., Shah, M.: A 3-dimensional SIFT descriptor and its application to action recognition. In: International conference on Multimedia (2007)
47. Simonyan, K., Zisserman, A.: Two-stream convolutional networks for action recognition in videos. In: Advances in Neural Information Processing Systems, pp. 568–576 (2014)
48. Torr, P., Zisserman, A.: Robust computation and parametrization of multiple view relations. In: ICCV (1998)
49. Tuytelaars, T., Mikolajczyk, K.: Local invariant feature detectors – a survey. Foundations and Trends in Computer Graphics and Vision (2008)
50. 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 (2012)
51. Wang, H., Kläser, A., Schmid, C., Liu, C.L.: Action Recognition by Dense Trajectories. In: CVPR (2011)
52. Wang, J., Liu, Z., Wu, Y., Yuan, J.: Mining actionlet ensemble for action recognition with depth cameras. In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, pp. 1290–1297. IEEE (2012)
53. Willems, G., Tuytelaars, T., Van Gool, L.: An efficient dense and scale-invariant spatio-temporal interest point detector. In: ECCV. Springer (2008)
54. Yang, X., Zhang, C., Tian, Y.: Recognizing actions using depth motion maps-based histograms of oriented gradients. In: ACM international conference on Multimedia (2012)
55. Yao, B., Fei-Fei, L.: Action recognition with exemplar based 2.5 d graph matching. In: ECCV (2012)