Combining Pose-Invariant Kinematic Features and Object Context Features for RGB-D Action Recognition

Manoj Ramanathan, Jaroslaw Kochanowicz, and Nadia Magnenat Thalmann

***Abstract*—Action recognition using RGB-D cameras is a popular research topic. Recognising actions in a pose-invariant manner is very challenging due to view changes, posture changes and huge intra-class variations. This study aims to propose a novel pose-invariant action recognition framework based on kinematic features and object context features. Using RGB, depth and skeletal joints, the proposed framework extracts a novel set of pose-invariant motion kinematic features based on 3D scene flow and captures the motion of body parts with respect to the body. The obtained features are converted to a human body centric space that allows partial viewinvariant recognition of actions. The proposed pose-invariant kinematic features are extracted for both foreground (RGB and depth) and skeleton joints and separate classifiers are trained. Borda-count based classifier decision fusion is employed to obtain an action recognition result. For capturing object context features, a convolutional neural network (CNN) classifier is proposed to identify the involved objects. The proposed context features also include temporal information on object interaction and help in obtaining a final action recognition. The proposed framework works even with non-upright human postures and allows simultaneous action recognition for multiple people, which are topics that remain comparatively unresearched. The performance and robustness of the proposed pose-invariant action recognition framework are tested on several benchmark datasets. We also show that the proposed method works in real-time.**

***Index Terms*—Real-time action/activity recognition, poseinvariant kinematic features, object context, non-upright postures.**

I. Introduction

With the advent of Microsoft Kinect, usage of depth cameras for vision-based research has attracted a significant amount of attention. The availability of depth and skeleton tracking has allowed for several applications in industries, such as gaming, automation and entertainment among others. The effectiveness of human computer interactions is governed by the computer's ability to understand human gestures. However, action recognition is affected by several factors, including view angle changes, pose variations and occlusion among others [1]. Furthermore, recognition methods are limited because they assume there is only one person in the frame and are not real-time in nature.

In this paper, we introduce a novel framework for pose-invariant action recognition using RGB-D cameras that includes RGB, depth modalities and ability to track skeleton joints. We introduce a new set of kinematic features computed with scene flow [2] as a basis for representing the motion of each body part. A novel algorithm is proposed to convert the computed features to human body centric space using the depth and skeletal joints information. In this body-centric space, the motion of body parts is captured with respect to the body frame, allowing viewinvariant action recognition. By converting to the proposed human body centric space, we can also handle non-upright human body postures during an action, which is a topic in need of further research. These kinematic features can be extracted for both the foreground region and skeletal joints directly, each for which a separate extreme learning machine (ELM) [3] classifier is trained for them. Both classifier's output are combined using Borda-count index to obtain an initial action recognition result.

Action recognition using only motion features can be difficult since multiple actions may share similar motion patterns. For example, *throw and pull* involve similar movements of hands. In the proposed method, we intend to initially classify both actions as same and then differentiate between them using the objects that are involved in the action. First, using the confusion matrix of the initial action recognition, we create clusters of actions with similar motions. Then, we introduce temporal object context features to encode the object that is involved and its temporal involvement. For this purpose, inception model [4] and transfer learning [5] have been used. Based on the foreground and skeletal data, we re-train the inception model to determine object presence in key positions. On the contrary, we only identify whether the subject is holding the object or not, and exclude all other interactions For each frame, we can identify the objects that are being held. To capture temporal object context, video is divided into smaller segments and the object context in each segment is combined with action representation to produce a final action recognition result. Most available methods assume that only a single person is present in the frame, therefore, it cannot be applied in real-time. We extend our framework to work for multiple people using Kinect's tracking and its applicability is tested in real-time.

The rest of the paper is organised as follows: Related work in the area of action recognition is discussed in Section II. Our proposed action recognition method is explained in Section III. Experimental results and discussion are provided in Section IV. We conclude the paper in Section V.

Manuscript received on July 12, 2018; revised October 24, 2018. This research is supported by the BeingTogether Centre, a collaboration between Nanyang Technological University (NTU) Singapore and University of North Carolina (UNC) at Chapel Hill. The BeingTogether Centre is supported by the National Re- search Foundation, Prime Minister's Office, Singapore under its International Research Centres in Singapore Funding Initiative.

The authors are with Institute for Media Innovation, Nanyang Technological University, Singapore (e-mail: mra-manathan@ntu.edu.sg, jarek108@gmail.com, nadiathalmann@ntu.edu.sg).

doi: 10.18178/ijmlc.2019.9.1.763

II. Related Work

Action recognition using RGB-D cameras is researched growing topic of research due to current and possible applications in several industries. A fundamental aspect of any action is the articulated motion of one or more body parts. To capture motion, [6] extracted optical flow from RGB and depth data and represented actions using spatial pyramid histogram of optical flow. Meanwhile, [7] introduced a depth motion map, that accumulates foreground motion re- gions to capture global activities. To achieve view-invariance, motion regions in front, side and top were accumulated. Skeleton joints based action recognition is a widely re- searched area. Many researchers attempt to represent the action using the relative position of joints [8] (EigenJoints), joint trajectories [9], joint positions and velocities [10]. [11]

III. Proposed Framework

For action recognition, the proposed framework captures motion as pose-invariant kinematic features (foreground and skeleton) encoded into a human body centric space and object interactions. In the following subsections, the kinematic motion features and its conversion to a human body cen- tric space based on estimated body orientation and the person's viewing direction (front, back, left,or right) are explained.

A. Kinematic Features

Once the foreground region has been obtained, the next step is computing the kinematic features, namely, divergence, curl, projection and rotation [14]. These features were previously used with optical flow to characterise 2D motion and assumed only frontal face. Herein, we extend these features to 3D and apply them to extracted scene flow [2] of the identified fore- ground region. The following equations give the 3D kinematic features:

Proposed compatibility kernels to compare spatio-temporal and action dynamics compatibility between two sequences. Several methods based on RNN and LSTM [12], [13] have also used skeleton joint sequences to learn deep networks for action recognition. These can be easily trained with large-scale datasets as well. In this study, we propose using scene flow to obtain velocity profile of the foreground and introduce a new set of kinematic features, which is borrowed from emotion recognition [14]. Other than the relative positions of body parts, we focus on capturing the relative motion of body parts using these features. Also, we propose to include contextual information using temporal object context features.

$$Div(p) = \frac{\partial SF_X(p)}{\partial x} + \frac{\partial SF_Y(p)}{\partial y} + \frac{\partial SF_Z(p)}{\partial z} \quad (1)$$

$$Curl(p) = \left(\frac{\partial SF_Z(p)}{\partial y} - \frac{\partial SF_Y(p)}{\partial z}\right)\hat{i} + \left(\frac{\partial SF_Z(p)}{\partial x} - \frac{\partial SF_X(p)}{\partial z}\right)\hat{j} + \left(\frac{\partial SF_Y(p)}{\partial x} - \frac{\partial SF_X(p)}{\partial y}\right)\hat{k} \quad (2)$$

View angle changes and pose variations are some of the most impor- tant bottlenecks in RGB-D action recognition. [15] proposed HOJ3D for posture representation and a spherical coordinate system to achieve a view invariant action recognition. How- ever, methods relying on only skeletal joints suffer under severe occlusion. As a result many researchers have combined depth and skeleton information such as [16] and [17] for action recognition. [18] extended the self-similarity matrices to depth streams to build a viewinvariant action recognition.

Capturing contextual information can be essential in recognizing some actions. [16] introduced a novel local occupancy pattern feature based on the 3D point cloud around particular joints. [19] proposed human interactive object features based on hand, edge and motion detection. To represent the motion of each body part and capture contextual information, [20] estimates the parts and employs a BoW-Pyramid feature representation. While these methods rely on local features for context, we use CNN model to focus on the global foreground region and determine the objects involved and include temporal information about the object. In this paper, for body part estimation, we use the skeletal joints only as guidance and assign part labels to foreground based on simple distance measures. Additionally, using depth and skeleton information, we encode the kinematic motion features into a human body centric space that forms the basis of our pose- invariant action representation. [21] provide more detailed surveys on action recognition based on depth data.

$$Proj(p) = \overrightarrow{SF_p}.\hat{P}neck \quad (3)$$

$$Rot(p) = \hat{P}neck \times \overrightarrow{SF_p} \quad (4)$$

where $SF_x(p)$, $SF_y(p)$ and $SF_z(p)$ are x, y and z-components of scene flow obtained for foreground pixel p respectively. Div and $Curl$ mainly focus on expansion and circular motion in local neighbourhood. To capture the relative motion between parts, we use $Proj(p)$ and $Rot(p)$ features. These features compute the motion of the foreground pixel p with respect to a stable reference point, neck in this case. We have chosen neck because it is stable and easily seen from all directions. The neck point can be obtained from the skeletal joints from Microsoft Kinect. The unit vector at neck $\hat{P}_{neck}$ is computed and is used for obtaining $Proj$ and Rot. Due to the articulated nature of the human body, the variation of velocities in depth can be assumed to be negligible or zero, which means that the equations can be further reduced.

As the next step, we assign a body part label to each foreground pixel. For this purpose, we measure the distance of each pixel from each of the skeletal joints and assign it a label (Head, Torso, L-arm, R-arm, L-leg, R-leg) based on minimum distance measure. Based on the assigned part labels, we compute a centre point for each body part label. Similar to $Proj$ and Rot, we introduce two more features body part referenced projection ($BodyProj(p_i)$) and rotation ($BodyRot(p_i)$) that aim to capture motion of pixel p classified as body part i with respect to its centre point. These are given by the following equations

$$BodyProj(p_i) = \overrightarrow{SF_{p_i}}.\widehat{P_c} \qquad BodyRot(p_i) = \widehat{P_c} \times \overrightarrow{SF_{p_i}} \quad (5)$$

where for each body part i, (P_c) is computed from the centre pixel computed using the label assigned previously.

B. Human Body Centric Space

Human body centric space is a three dimensional cylinder characterised by up-down, left-right and forward-backward directions. An RGB image is a 2D cross-section of this cylinder and only 2D motion can be recovered. With the available depth and skeleton joints, we aim to convert the observed motion features into body-centric space so that pose-invariant action representation can be obtained. The body orientation and viewing direction of the person determines which cross-section of the cylinder is observed.

To estimate the body orientation, we use the Neck and SpineBase skeletal joints obtained from Kinect. Based on the orientation between these joints, we can estimate the posture as either upright or non-upright. For view direction estimation, we use the depth values of the body centre (neck and spinebase), right (R-shoulder, R-arm) and left (L-shoulder, L-arm) side of the body. Based on these values we can determine if the person is front, right or left. We use a face detector to differentiate between front and back views. Based on the body orientation and view direction, we can estimate the three dimensions of the human body centric space observed from the RGB-D cameras. The kinematic features computed above lie in one of these 3 dimensions. We assume Div, $Proj$ and $BodyProj$ lie in the direction of scene flow and $Curl$, Rot and $BodyRot$ have separate components for each direction.

For action representation, first, the foreground region is divided into three grids, head, torso and legs, using the skeletal joints. For accurate representation, we further divide them into cells (3 x 2 is used in our implementation). To account for variations due to view point and body posture, these cells are labeled with a specific number. This allows us to compare corresponding cells while recognising the actions. The kinematic features can be classified into three based on the reference point used for measuring, namely, local neighbourhood (Div, $Curl$), neck ($Proj$, Rot) and body part referenced ($BodyP roj$, $BodyRot$). For action representation, we compute the weighted (sign and magnitude) and unweighted (only sign) histograms of the three classes of kinematic features in each of the three directions of human body centric space. L2- normalisation is applied to the features after accumulating over the total video length.

C. Skeleton Pose Kinematic Features

In this subsection, we explain the extraction of the proposed pose-invariant kinematic features for each skeleton joint to obtain an action representation. The frame- work mentioned above has two limitations. First, the foreground is not always detected correctly. Second, since the action representation involves histograms, the temporal evolution of the pose might not be captured. Also, some actions differ in the way only one or two joints move which might not be captured when complete foreground is taken.

For extracting skeleton pose kinematic features, we require the motion vector of the neck joint. For this purpose, we use 3D neck joint location for two consecutive frames and obtain the motion vector in each direction. In each frame, the other skeletal joints are converted to coordinate space with neck as an origin. Motion vector for each skeletal joint is computed using 3D world coordinates obtained for two consecutive frames. Projection and rotation features are computed for all available skeletal joints using the above mentioned equations and converted to the human body centric space as discussed. The weighted and unweighted histograms for each skeletal joint in each of the three directions of human body centric space are computed and L2-normalised.

For combining proposed features, we have employed Borda-count based classifier fusion approach. Separate ELM classifiers are trained for pose-invariant kinematic motion features of foreground pixels and skeletal joints, respectively. For a particular class c_k, Borda Count $B(c_k)$ is defined as a sum of the number of classes ranked below class c_k by each classifier. The magnitude of the BC reflects the level of agreement that the input pattern belongs to the considered class [22]. For every test video, both features are extracted and passed through respective classifiers. From the output of both classifiers, BC of each class c_k is identified and the class with highest BC is given as the action recognised.

D. Temporal Object Context Features

The Borda-count based method is completely dependent on only motion features. It is difficult to differentiate all actions based on only motion as similar patterns may be observed for different action. To overcome this issue, we introduce temporal object context features to include contextual information on objects and temporal interactions involved. Before applying these features, we first need to create action clusters, i.e., actions with similar motion patterns are grouped as a single class. Initially, we run the proposed Borda-count method with all the action classes. During training based on the confusion matrix, we can determine the actions poorly recognised and actions confused with. A threshold can be set to determine if two classes are confused. Based on the initial run, we first create action clusters and train new classifiers to classify test videos into one of the clusters. During testing after the video is classified into one of the clusters, temporal object context features are used to determine a final action. In this subsection, we elaborate on the training object detection CNN and extract temporal object context features using it.

Object Recognition CNN - Retraining Procedure: Detection of objects held by subject can help in differentiating actions with similar motion. Due to low resolution, occlusion and variety of objects being used, object detection is difficult in action datasets. Hence, we train a CNN with a whole person holding 16 objects (including no object) seen in the action training videos. Using the skeletal joints, we crop the person out and resize to 640 x 480. The extracted set of images are used for retraining the Inception v3 [4] classifier, pre-trained on ImageNet [23], to recognise various objects held by the subject.

During retraining, the original network's last layer is replaced by two layers, namely, hidden layer and new classification layer (for objects of interest in action dataset). Only parameters of these two new layers need to be trained. From the training videos, we collected images for each object category and chose around 500 sample images per object category which were then divided into training (70%) and

validation (30%) sets. The network was retrained and parameters with the best performance were selected for testing on an unlabeled dataset. Fig. 1 shows some of the images from training videos used for retraining.

Fig. 1. Sample images with person holding different objects used for retraining Inception v3 classifier.

For extraction of temporal object context features, we divide the video into *N* temporal segments and consider the objects detected in each. Given a video frame, we crop the person out using skeletal joints, resize to 640 x 480 and pass through the trained object classifier. Thus, each frame is asso- ciated with an object. In each temporal segment, we compute the ratio of the number of occurrences of each identified object and the total number of frames. The final features are combination of these features for all temporal segments.

With help of Kinect, we extend our action recognition framework to handle multiple people (6 skeletons) using the unique tracking ID provided. For real-time action recognition, the most time consuming part would be computation of scene flow. Using the CUDA implementation of [2] using NVIDIA GTX 650, we were able to achieve real-time action recognition.

IV. Experimental Results and Discussions

The proposed framework for action recognition is based on two important components, namely, pose-invariant kinematic and object context features. For testing our frame- work's performance, we selected three benchmark datasets, namely, MSR DailyActivity 3D [16], UT Kinect dataset [15] and NTU RGB+D dataset [12]. These datasets provide RGB, depth and skeleton streams from Kinect for different actions that may or may not include object interactions. For real-time implementation, we collected our own dataset of nine actions. For object context, we train a global classifier with 16 objects (including no object) seen in the datasets from the action training videos. As mentioned earlier, we use 3 x 2 cell configuration for dividing the body grids. For classification, we use ELM [3] due to its faster training time. The only parameter to tune in ELM is the number of hidden layer neurons. We train ELM with different number of neurons and report the best performance here.

A. MSR Daily Acivity 3D

MSR Daily Activity 3D [16] is a popular dataset with ten subjects performing 16 activities in standing and sitting postures in an office environment. Activities can include interaction with objects, such as laptop, vacuum cleaner, guitar, paper and soda can among others. For validation, we have used the leave-one-person- out setting. First, we run without the object context module to identify the action clusters. In this setting, we achieved a recognition performance of 64%, which is very low in comparison with other available methods. We analysed the results obtained by constructing the confusion matrix, as shown in Fig. 2 (a). The confusion matrix showed that seven out of 16 actions (*read, call cellphone, write, use laptop, play games, play guitar, throw*) did not perform well. *Read, write, use laptop and play games* were mainly confused with sit still action. In each of these actions, we observed that there is no discernible motion. In all these actions, the motion observed is negligible making it difficult to detect without object detection results. These five actions *read, write, use laptop, play games and sit still* can be one action cluster. A similar cluster was obtained with actions *drink, eat and call cellphone*. In the case of *call cellphone, play guitar and throw actions*, there is no unique motion pattern to represent them. For example, in the *call cellphone* action, after the subject lifts the phone to the ears, he/she is free to move in standing position thus introducing erroneous motion. Similarly, *play guitar* can be done in different ways and cannot be fixed to single discriminative motion pattern. All of these actions can be recognised in a better manner by including an object detection module. The remaining actions are recognised well by our framework and can be considered as separate action clusters.

Second, we ran the proposed framework with object context module. The actions were classified into either one of the identified action clusters based on the motion(s) involved. The final action is then recognized based on the tempo- ral object context features. Also some actions can be recognised by simply identifying the objects. For example, the action *play guitar* can be recognised by identifying the object *guitar*. In this setting, we give more importance to the objects observed. The proposed framework shows a performance of 90.84%. The comparison of our method and state of art methods are shown in table I. We can observe that the proposed method performs better than most of the available methods. Heterogeneous feature learning [17] and multiple kernel learning [6] allow them to learn significantly important motion and appearance in an action, thus allowing for a better action representation. The proposed framework is pose-invariant whereas the other methods available are quite limited. Overall, our proposed method can be improved with better detection of significant motion patterns. Another limitation is that the object context module fails in cases where objects are too small or occluded.

B. UT Kinect Action Dataset

UT Kinect dataset [15] is a popular dataset with ten actions done by ten actors twice in an office environment. Only a few actions like *pick up, carry and throw* involve any objects. For validation, we have used the leave-one-person-out setting. Since the skeleton label is not provided for depth and RGB frames, we include a foreground detection based on the x,y

and z coordinates of the skeleton.

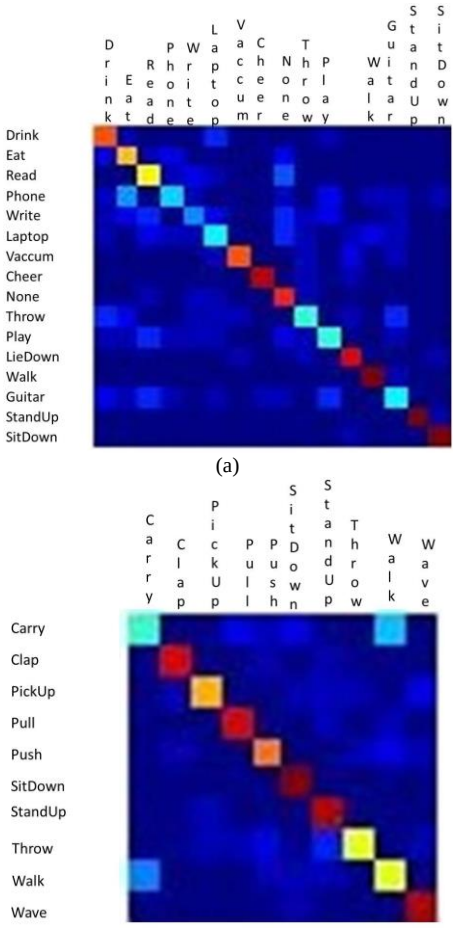

Fig. 2. Confusion matrix obtained for the proposed framework without object context module in (a) MSR Daily Activity 3D (b) UT Kinect dataset. *blue* indicates low value and *red* indicates a higher value.

TABLE I: PERFORMANCE COMPARISON IN MSR DAILY ACTIVITY 3D DATASET

Method	Performance (%)
Proposed Framework (with object context)	90.84
Range-Sample Depth [24]	95.63
Heterogeneous Learning [17]	95
Multiple Kernel Learning [6]	92.5
Actionlets [16]	85.75
Tensor Subspace[25]	80.63

Similar to MSR DailyActivity 3D, we first run without the object context module. We achieve a performance of 80.95%, which is very low in comparison to the state of the art methods. Using the confusion matrix, as shown in Fig. 2 (b), we identified the action clusters. We observed that only three actions had a very poor recognition performance, namely, *carry, walk and throw*. *Throw* is mainly confused with the *push* action since they have similar actions with the only difference being the object held in the hand. *Carry and walk* actions are also confused with each other. From the videos of both actions, we can see that action *carry* involves walking with a box in the hand. The rest of the actions have a better recognition performance.

Second, we ran the framework with object context module included. Also for actions *carry and throw*, it is not important which object is being held. We only need information as to whether the object is present or not, since any object can be carried or thrown. Our object context features include this information. The results obtained with object context and comparison to other available methods is tabulated in table II. We show a performance of 96.5%, which is comparable with the state of the art methods. [10] and [11] show better performance. [11] captures higher order relation- ships between actions and joints using tensors. Since, from [10], action representation involves identifying important skeletal joints and temporal stages of that action. Our framework can also be improved for identifying such attributes and we plan to do this as part of our future work. Also, our framework includes object context in contrast with the other methods.

TABLE II: PERFORMANCE COMPARISON IN UT KINECT DATASET

Method	Performance (%)
Proposed Framework (with object context)	96.5
GCA LSTM [13]	98.5
Tensor Representations[11]	98.2
ST NBNN [10]	98
Eigen Joints[9]	91.5

C. NTU RGB+D Dataset

NTU RGB+D [12] is one of the largest RGB-D action recognition datasets available with 56880 samples and 60 action classes. The actions are performed by 40 subjects and captured from three camera views. The actions can be either one-person actions or two-person interactions. The dataset are extremely challenging due to the view-point changes, pose variations and intra-class variations among others. In our implementation, we used RGB, masked depth image and skeleton information provided with the dataset. We followed two standard protocols, namely, cross-subject and cross-view, provided with the dataset for testing. For two person interactions, we compute average skeleton joint positions and use it for skeleton pose kinematic features. The RGB-D based kinematic features are computed based on the skeleton for each person in the frame. The results of our method and its comparison with other state of art methods are shown in table III. We can observe that the proposed method does not perform well in comparison with state of the art deep learning methods. With only kinematic motion features, the performance is 45.01% and 41.37% in both evaluation protocols, respectively. Similar to other datasets, our performance can increase with the inclusion of temporal object context features but it might not be comparable to state of the art methods. Due to this, we have not included the object context module. Our kinematic motion features represent the observed motion during the entire action. As a result, it might not be able to learn important, subtle motion involved in many of the action classes. Deep learning based methods using RNN, LSTM [12] and [13] can explore spatial and temporal structure. Also, they can learn long term temporal dependency of different body parts involved in an action. ELM classifier used by the proposed method is very simple and might not be able learn all variations and nuances of an action. This suggests that the proposed method might not be suitable for big datasets. In our future work, we will focus on developing a deep learning based method using the proposed features.

D. Own Collected Dataset

To test the proposed framework in real-time and handle multiple persons, we collected our own dataset with nine actions. They include *eat, drink, answerPhone, checkPhone, give object, extend to shake hands, takePhoto, walk and do*

nothing. For actions *eat, drink, answerPhone and walk*, we used the training videos from MSR DailyActivity 3D and UT Kinect dataset, respectively, and collected our own dataset for other actions using a Kinect v2. Some of the collected sample frames are shown in Fig. 3. The subjects were allowed to choose any object for performing the actions. We used global object context classifier for extracting the contextual information. Leave-one-out validation was used for measuring performance.

TABLE III: PERFORMANCE COMPARISON IN NTU RGB+D DATASET

Method	Cross-Subject (%)	Cross-View (%)
Proposed Framework (without object context)	45.01	41.37
Original Samples Skeleton Visualization [26]	75.97	82.56
Part-aware LSTM [12]	62.9	70.3
GCA LSTM [13]	74.4	82.8

Fig. 3. Sample frames from own dataset collected for *checkPhone, give object, shake hands and takePhoto* actions. Other actions were taken from other datasets.

Similar to other datasets, we ran the proposed framework without the object context module. The framework shows a performance of 64.72%. From the results obtained, we identified two action clusters, eat, drink and answerPhone and give, *shake hands and takePhoto*. Using these clusters, we ran the proposed framework with object context module. We observed an increase of 12% after inclusion of object context.

Fig. 4. Sample frames showing occlusion of objects and small sized objects in the dataset

An important observation is that the same object can be used for different actions. For example, a *phone* can be used for both *give* and *takePhoto*. When both actions are in the same action cluster, it would be difficult to identify the action. Similar to MSR DailyActivity 3D, another source of error is when the object is small or occluded. Also as shown in Fig. 3, the subject might be occluded making it difficult for the motion based action recognition to be accurate. Overall, the proposed method was able to function in real-time and handle multiple persons.

E. Discussion

During analysis of our results, we noticed that simple foreground detection based on skeleton and depth information can result in erroneous detection of cupboards and table as foreground. This would introduce erroneous motion features and thus reduce our recognition performance. Our framework also depends on the availability of neck point to compute pose- invariant motion features. The skeleton joints provided in some cases are such that the neck and other parts remain hidden or invalid. In these cases, it is not possible to extract motion from the videos. The availability of neck point in the frame is a significant limitation of our framework.

The proposed framework currently cannot handle occlusion of body parts. We observed that the performance drops in the presence of occlusion. In the own collected dataset, *give, shake hands and takePhoto* all included occlusion of legs. When we ran the proposed framework in real-time, we observed that the actions were getting confused to these 3 actions when the legs were occluded.

Object context is an important component in the proposed framework. In all three datasets, we have seen a significant increase after including the contextual information. But as shown in Fig. 4, it is difficult to detect occluded or small-sized objects. In the current setup, we have considered only one type of interaction, i.e. the subject holding the object. But this needs to be extended to other interactions such as sitting down in a chair. In the future, we would like to extend the framework to include object identity, type of interaction and temporal involvement of object. For action recognition, we would like to investigate deep learning for the proposed set of kinematic features also to encode object interactions into it.

V. CONCLUSION

Pose-invariance in RGB-D action recognition is difficult due to a variety of factors, such as view changes, occlusion, large amount of possible poses and object interactions among others. In this study, we propose a novel framework to combine motion features and temporal object context features. For motion, we introduce a new set of pose-invariant kinematic features that capture motion of body parts with respect to the body frame itself. Encoding in human body centric space allows us to recognise actions in non-upright postures and in a partially view invariant manner. The proposed features can be applied to both foreground and skeletal joints and are combined using Borda-count classifier fusion. Initial action clusters are formed using this output. For final action recognition, we include temporal object context features that distinguish what object is being held and for how long. For this purpose, we also train CNN that detects the object in each frame. Experiments have shown that the proposed framework performs well in benchmark datasets and can be applied in real-time to handle multiple subjects.

ACKNOWLEDGEMENTS

This research is supported by the BeingTogether Centre, a collaboration between Nanyang Technological University (NTU) Singapore and University of North Carolina (UNC) at Chapel Hill. The BeingTogether Centre is supported by the National Research Foundation, Prime Minister's Office, Singapore under its International Research Centres in Singapore Funding Initiative.

REFERENCES

[1] M. Ramanathan, W.-Y. Yau, and E. K. Teoh, "Human action recognition with video data: Research and evaluation challenges," *IEEE Trans. on Human Machine Systems*, vol. 44, pp. 650-663, October 2014.

[2] M. Jaimez, M. Souiai, J. Gonzalez-Jimenez, and D. Cremers, "A primal- dual framework for real-time dense rgb-d scene flow," in *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, pp. 98-104, 2015.

[3] G.-B. Huang, L. Chen, and C.-K. Siew, "Universal approximation using incremental constructive feedforward networks with random hidden nodes," *IEEE Trans. on Neural Networks*, vol. 17, pp. 879-892, July 2006.

[4] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 2818-2826, June 2016.

[5] H.-C. Shin, H. R. Roth, M. Gao, L. Lu, Z. Xu, I. Nogues, J. Yao, D. Mollura, and R. M. Summers, "Deep convolutional neural networks for computer-aided detection: Cnn architectures, dataset characteristics and transfer learning," *IEEE Trans. on Medical Imaging*, vol. 35, pp. 1285-1298, May 2016.

[6] V. H. Viet, L. Q. Ngoc, T. T. Son, and P. M. Hoang, "Multiple kernel learning and optical flow for action recognition in rgb-d video," in *Proc. Intl. Conf. on Knowledge and Systems Engineering*, pp. 222-227, October 2015.

[7] X. Yang, C. Zhang, and Y. Tian, "Recognizing actions using depth motion maps-based histograms of oriented gradients," in *Proc. ACM Intl. Conf. on Multimedia*, pp. 1057-1060, October 2012.

[8] X. Yang and Y. Tian, "Effective 3d action recognition using eigenjoints," *Journal of Visual Communication and Image Representation*, vol. 25, pp. 2-11, January 2014.

[9] M. Devanne, H. Wannous, S. Berretti, P. Pala, M. Daoudi, and A. D. Bimbo, "3d human action recognition by shape analysis of motion trajectories on riemannian manifold," *IEEE Trans. on Cybernetics*, vol. 45, pp. 1340-1352, August 2015.

[10] J. Weng, C. Weng, and J. Yuan, "Spatio-temporal naive-bayes nearest-neighbor (st-nbnn)for skeleton-based action recognition," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 1-10, July 2017.

[11] P. Koniusz, A. Cherian, and F. Porikli, "Tensor representations via kernel linearization for action recognition from 3d skeletons," in *Proc. European Conf. on Computer Vision*, pp. 37-53, October 2016.

[12] A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang, "NTU RGB+D: A large scale dataset for 3d human activity analysis," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pp. 1010-1019, June 2016.

[13] J. Liu, G. Wang, P. Hu, L.-Y. Duan, and A. C. Kot, "Global context-aware attention LSTM networks for 3d action recognition," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pp. 3671-3680, July 2017.

[14] S. Shojaeilangari, W.-Y. Yau, K. Nandakumar, J. Li, and E. K. Teoh, "Robust representation and recognition of facial emotions using extreme sparse learning," *IEEE Trans. on Image Processing*, vol. 24, pp. 2140-2152, July 2015.

[15] L. Xia, C.-C. Chen, and J. K. Aggarwal, "View invariant human action recognition using histograms of 3d joints," in *Proc. IEEE Computer Soceity Conf. on Computer Vision and Pattern Recognition WorkShops*, pp. 20–27, June 2012.

[16] J. Wang, Z. Liu, Y. Wu, and J. Yuan, "Mining actionlet ensemble for action recognition with depth cameras," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 1290-1297, June 2012.

[17] J.-F. Hu, W.-S. Zheng, J. Lai, and J. Zhang, "Jointly learning heterogeneous features for rgb-d activity recognition," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 5344-5352, June 2015.

[18] A.-R. Lee, H.-I. Suk, and S.-W. Lee, "View-invariant 3d action recognition using spatiotemporal self-similarities from depth camera," in *Proc. Intl. Conf. on Pattern Recognition*, pp. 501-505, August 2014.

[19] C. Wu, J. Zhang, S. Savarese, and A. Saxena, "Watch-n-patch: Unsupervised understanding of actions and relations," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 4362-4370, June 2015.

[20] J.-S. Tsai, Y.-P. Hsu, C. Liu, and L.-C. Fu, "An efficient part-based approach to action recognition from rgb-d video with bow-pyramid representation," in *Proc. IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems*, pp. 2234-2239, November 2013.

[21] M. Ye, Q. Zhang, L. Wang, J. Zhu, R. Yang, and J. Gall, *A Survey on Human Motion Analysis from Depth Data*, pp. 149–187. Berlin, Heidelberg: Springer Berlin Heidelberg, September 2013.

[22] D. Ruta and B. Gabrys, "An overview of classifier fusion methods," *Computing and Information Systems*, vol. 7, pp. 1-10, February 2000.

[23] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet large scale visual recognition challenge," *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.

[24] C. Lu, J. Jia, and C.-K. Tang, "Range-sample depth feature for action recognition," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 772-779, June 2014.

[25] C. Jia and Y. Fu, "Low-rank tensor subspace learning for rgb-d action recognition," *IEEE Trans. on Image Processing*, vol. 25, pp. 4641-4652, October 2016.

[26] M. Liu, H. Liu, and C. Chen, "Enhanced skeleton visualization for view invariant human action recognition," *Pattern Recognition*, vol. 68, pp. 346-362, August 2017.

Manoj Ramanathan received hi B.Tech degree in instrumentation and control engineering from the National Institute of Technology, Tiruchirapalli, India, in 2009. He worked as a software engineer at Toshiba Software India Pvt. Ltd till 2012. He received his PhD degree in 2017 from the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. His research interests include computer vision, action recognition, biometrics and Robotics. Currently, He is working as a research fellow in Institute for Media Innonvation, Nanyang Technological University, Singapore.

Jaroslaw Kochanowicz worked as research fellow in Institute for Media Innovation, Nanyang Technological University, Singapore. He received his PhD degree in 2017 from the School of Computer Science Engineering and the Institute for Media Innovation, Nanyang Technological University, Singapore, under the supervision of Dr Ah-Hwee Tan and Dr Daniel Thalmann. He studied computer science and math at the Jagiellonian University, Krakow, Poland, and Artificial Intelligence at the Vrije University, Amsterdam, Holland, and Katholieke Universiteit, Leuven, Belgium. He received a Master Degree in Computer Science in 2009 from the Institute of Computer Science, Jagiellonian University, Krakow. His fields of interest include social-cognitive modelling and simulation and formalisation of human-like irrationality of individuals and groups including investigation of its types, sources, and implications.

Nadia Magnenat Thalmann is chairing MIRALab at University of Geneva, and is presently Director of the inter-disciplinary Institute for Media Innovation in NTU in Singapore.She has authored dozens of books, published with her students more than 600 papers on virtual humans, virtual worlds and social robots, organised major conferences as CGI, CASA, and delivered more than 300 keynote addresses, some of them at global events such as the World Economic Forum in Davos. In NTU, Singapore, she revolutionised social robotics by un- veiling the first social robot Nadine that can have mood and emotions and remember people and actions. She has received numerous research Awards as the Humboldt Research Award in Germany. http://en.wikipedia.org/wiki/Nadia Magnenat Thalmann.