Two person Interaction Recognition Based on Effective Hybrid Learning

Minhaz Uddin Ahmed[1], Yeong Hyeon Kim[1], Jin Woo Kim[1], Md Rezaul Bashar[2] and Phill Kyu Rhee[1]*

[1]Department of Computer Engineering, Inha University,
Incheon, South Korea
[Email: pkrhee@inha.ac.kr]
[2]Science,Technology and Management Crest, Sydney, Australia
*Corresponding author: Phill Kyu Rhee

Received April 5, 2018; revised August 23, 2018; accepted October 9, 2018; published February 28, 2019

Abstract

Action recognition is an essential task in computer vision due to the variety of prospective applications, such as security surveillance, machine learning, and human–computer interaction. The availability of more video data than ever before and the lofty performance of deep convolutional neural networks also make it essential for action recognition in video. Unfortunately, limited crafted video features and the scarcity of benchmark datasets make it challenging to address the multi-person action recognition task in video data. In this work, we propose a deep convolutional neural network–based Effective Hybrid Learning (EHL) framework for two-person interaction classification in video data. Our approach exploits a pre-trained network model (the VGG16 from the University of Oxford Visual Geometry Group) and extends the Faster R-CNN (region–based convolutional neural network a state-of-the-art detector for image classification). We broaden a semi-supervised learning method combined with an active learning method to improve overall performance. Numerous types of two-person interactions exist in the real world, which makes this a challenging task. In our experiment, we consider a limited number of actions, such as hugging, fighting, linking arms, talking, and kidnapping in two environment such simple and complex. We show that our trained model with an active semi-supervised learning architecture gradually improves the performance. In a simple environment using an Intelligent Technology Laboratory (ITLab) dataset from Inha University, performance increased to 95.6% accuracy, and in a complex environment, performance reached 81% accuracy. Our method reduces data-labeling time, compared to supervised learning methods, for the ITLab dataset. We also conduct extensive experiment on Human Action Recognition benchmarks such as UT-Interaction dataset, HMDB51 dataset and obtain better performance than state-of-the-art approaches.

Keywords: Action Recognition, Convolutional Neural Network, Deep Architecture, Transfer Learning

This work was supported by INHA UNIVERSITY Research Grant. (INHA-59176)

http://doi.org/10.3837/tiis.2019.02.015 **ISSN : 1976-7277**

1. Introduction

Action recognition in video data is a significant research subject in the computer vision research area due to its numerous applications, such as video surveillance, action recognition, human activity analysis, and human–computer interaction. To solve such a challenging task, researchers have tried hand-crafted video features for the last several years in order to recognize human actions [1][2], where performance is not satisfactory due to a static feature model [3]-[4]. In recent years, however, the convolutional neural network (CNN) [5]-[7] has shown a great deal of improvement in processing video data.

In this paper, we propose a deep learning–based framework for two-person action classification to solve this challenge. We build a Faster R-CNN (region-based convolutional neural network)-based [8] action recognition framework using the matconvnet [9] library and a trained CNN for each action. We used the VGG16 [10] pre-trained model from the University of Oxford Visual Geometry Group, which has higher batch normalization competence during training. Among the recent successful state-of-the-art methods for human action recognition is the two-stream architecture [5], which takes advantage of both spatial and temporal models. Lately, R*CNN [11] showed an improved map on the PASCAL (Pattern Analysis, Statistical Modelling and Computational Learning) VOC (Visual Object Classes) 2012 dataset over existing human action–recognition methods, despite limited training images per class.

Action recognition in a benchmark dataset is challenging because the training dataset is too small or there are too many variations in poses and objects. Therefore, large amounts of training examples are required to accurately identify the actions. A good number of training images is required in order for each action to be recognized. However, in some of the benchmark datasets, the number is still not sufficient. Moreover, a convent architecture is not capable of taking complete advantage of the limited number of image sets.

In this paper, we propose a more sophisticated way—combined (semi-supervised learning) SSL [12] and active learning (AL) [12][13]—to tackle this problem. Active semi-supervised learning (ASSL) works in two steps. First is SSL, where a number of batch image sets are trained with a deep CNN that evaluates each set of images. If the SSL evaluation is satisfactory, we consider it for the new model; if performance is not satisfactory, we apply AL. Secondly, AL checks which images are responsible for low performance, replaces them with a new image set, and then, trains again. This training procedure continues until the predictor reaches saturation. We consider that as the final model. Our proposed framework uses an incremental improvement over the existing model, with less human labeling, to produce excellent performance.

The main contributions of this work are summarized below.

i. Use deep learning to train the two-person action recognition model and improve the human action–detection accuracy.
ii. Propose a new approach that combines SSL and AL to tackle the issue of limited training examples.
iii. Propose multiple human interaction recognition that works both online and offline.

iv. Represent a unique dataset that is the first of its kind, with an experimental methodology to support the work also compare with benchmark datasets.

This paper is organized as follows. In Section 2, we present the relevant previous work on human action recognition. Section 3 describes the proposed system's overview and method in detail. Section 4 presents the experiments and results, and Section 5 offers a conclusion and future work.

2. Related Work

Human action recognition is an extensive research area where numerous algorithms exist. We review prior works on human action recognition that include manually engineered features, such as Histogram Oriented gradient (HOG) [1], On Space Time Interest Point (STIP) [2], dense trajectory [14], Fisher vector [15], and Deformable Part Model (DPM) [16], Bag of Words (BOW) [17], which are popular for human action recognition. But due to the popularity of deep models such as Long-term temporal convolutions (LTC-CNN) [18] [19] that learn multiple layers and produce high-level classification, convnet [20][8] received a great reputation. However, CNNs [7][21] have developed a lot in recent years. In this paper, we use AL [13] on top of SSL. The significant difference in our approach is iterative, and combines both SSL and AL. We call our framework Active Semi Supervised Learning (ASSL), which outperforms these traditional existing methods.

2.1 Semi-supervised learning

SSL is a self-training method where the model iteratively trains with a small amount of labeled data, and later, classifies unlabeled data to retrain. There are the most confident, unlabeled instances, together with predicted labels, included in the training dataset, and then, the process repeats [22].

SSL aspires to improve upon a large number of unlabeled data—a set of l images, $x_1 \ldots x_l \in X$, with corresponding labels (hugging, fighting, linking arms, talking, and kidnapping: $y_1, \ldots y_l \in Y$). Moreover, given u unlabeled images, $x_{l+1}, \ldots, x_{l+u} \in X$, SSL uses the combined x_1 and y_1 to surpass classifications [22]. On the other hand, semi-supervised learning attempts to teach the model itself by what it has learned so far from unlabeled instances. SSL use a latent structure in the data to improve the superiority of the model [23].

Ahsan et al. presented a SSL based action representation from video using Generative Adversarial Networks (GAN) which does not need weak supervision and expensive steps in the form of video encoding[24].Though, better sampling strategy and computationally expensive approach can outperform this SSL based GAN. Zhang et al. proposed an SSL based adaptation method which leverage both labeled and unlabeled data[25]. One advantage of their method is it works with few labeled training data and possible to use in other domain too. Jones et al. constructed a spectral method known as Feature Grouped Spectral Multigraph (FGSM) which is generated through clustering[26]. FGSM performs well on unsupervised and semisupervised tasks because of the use of latent structural information. However, when performs on fully supervised action recognition task, it can not find latent structure of fully labeled data. Zhang et al. proposed a boosting based multiclass SSL where

formulate the action recognition with insufficient labeled data. The co-EM algorithm is use for semisupervised model making. A weighted multiple discriminant analysis (WMDA) is adopt in order to make the co-EM algorithm efficiently learn parameters. One demerit of their work is large unlabeled samples are required for performance evaluation[27].

Our approach works in two steps. In the first phase, SSL evaluates each image set, where we use 10 images for each of five actions, with 50 images in total in one set. Each set of images trains with a pre-trained model and checks the performance. Based on the performance evaluation, we apply AL to each imageset.

2.2 Active Learning

In the active learning method, we take several observations into account, such as working with an equal number of image sets in SSL, but falsely classified images are replaced with true images. While preparing the image set, we only label those images on which a present model poorly performed and misclassified them. We gather an arbitrary number of unlabeled instances in sequential order (from good to worse). The main objective is to find falsely classified action recognition from the sequence of images, but with fewer possible testing images [13]. Thus, the approaches to set the threshold value that we consider will play a pivotal role in the active learning technique. Batch mode active learning is better than single mode when working with a large amount of data because it is less expensive [28]. It is a balanced approach where AL targets minimizing the labeling work by posing queries. Generally works in two steps, selecting a set of queries and requesting their labels[29]. Sourati et at. propose a practical querying algorithm based on Fisher Information Ratio (FIR) which offers a probabilistic way of pool-based active learning. They obtain the best possible query distribution in terms of approximation of FIR. One advantage of this work is to use a large batch size which improves the overall performance. Bernard et al. conduct three part labeling strategies those are identify different Visual Interactive Labeling (VIL) approaches, investigate strength and weakness of labeling strategies and effect of using different encoding to guide the visual interactive labeling process[30]. VIL support techniques can compete with the performance of AL. But first, fifty labeling iteration make cold start problem of AL which is intense.In future empirical performance of VIL can rely on user based data selection procedure. Huo et al. present a framework which is known as Second Order Active Learning (SOAL) for the binary classification problem[31]. SOAL use both First order information and Second order information for effective query strategy to predict the margin of incoming unlabeled data and confidence of the learner. SOAL address the limited label query budget and maximize the predictive accuracy with minimum cost.

We apply SSL and AL to each set of data by assuming that the training dataset is not prearranged and the evaluation outcome is less than satisfactory. In our framework, using only a little training data (e.g. less than five images per action) will not produce satisfactory performance, whereas 10 imges per action in a single batch of the dataset demonstrates a better outcome. Therefore, we increased the number of images for each human action to obtain a better outcome. Another fact is that fewer iterations demonstrated poor performance. As a result, we set our minimum number of iterations at 50,000, where the evaluation showed outstanding performance with the Intelligent Technology Laboratory (ITLab) dataset. Usually, a large number of iterations requires a long training time due to subsets of training images, called mini-batches, to update the weights. We also find that the proposed ASSL method can be applied to a number of computer vision applications.

2.3 Transfer Learning

Recently, transfer learning has emerged as a new learning framework that can use different domain learning models that can be used in another precise domain. For example, we have a classification task in some specific domains of interest, but on the other hand, we have a diverse dataset with different features in another domain. In this situation, transfer learning [26] works successfully, and improves performance significantly, without much effort for data labeling. Machine learning algorithms usually use training and test data from the same domain and feature space. When the distribution changes, most of the models require rebuilding from the beginning, using newly gather training data. This collecting of new training data and making new models is very expensive. In these situations, transfer learning [26] can play an essential role, and can reduce the effort and expense.

Transfer learning apply in many domains, such as Web document classification [27] where a web document is classify into several predefine categories. Wu and Dietterich [28] apply transfer learning to low-quality source domain data for an image classification problem. Transfer learning also propose for knowledge extraction [29] from Wi-Fi localization. Another diverse application of transfer learning is sentiment classification, where product reviews offer both positive and negative viewpoints. This kind of work requires a large amount of data, which is very expensive, but transfer learning aids a significant amount of the labeling effort [30].

3. System Overview

The flow diagram of our framework shows in **Fig. 1** where a video frame is converted into a sequence of images, and then noise is removed, and the Region of Interest (ROI) of the action area is selected to make training and test datasets. During the training process, we pass labeled input images through the convolutional layers (specifically, fully connected layers) to make the concluding prediction.

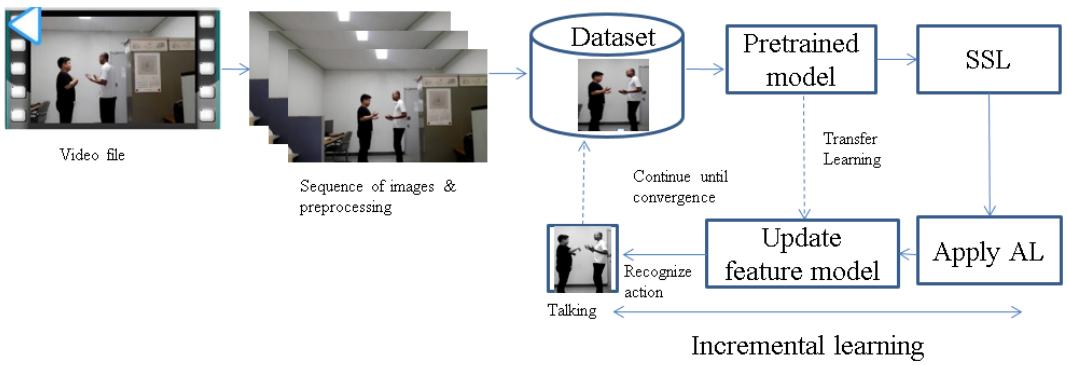


Fig. 1. Block diagram of the proposed method for two-person action recognition

3.1 Image pre-processing

We capture a video file using a web camera, and then converted it into a sequence of image files. We apply pre-processing that includes unwanted noise removal, such as intensity normalization. Intensity normalization [31] and cropping play a big role in noise reduction in our framework. Initially, we remove unnecessary parts of the image, such as the door and desk, which have no impact on human action, and only preserve the ROI, which captures the detailed appearance of human interaction, normalized to size 224×224. The eliminated parts of the images have no involvement in human action recognition, so in order to improve action recognition performance, it is necessary to remove noise.

(a) Original image (b) Preprocessed image

Fig. 2. Two-person interaction input image. (a) Original image (b) preprocessed image, cropped by a horizontal factor and a vertical factor to remove non-action features, for example, the background, the door, the desk, and the ceiling.

Image intensity and dissimilarity act as an important role for action recognition because the same human action has variations in the feature vector. Lighting conditions increase the difficulty in correct classification of actions. Therefore, to tackle this issue, we apply intensity normalization.

We minimize interference from illumination by changing the range of pixel intensity values, which helps reduce contrast. Normalization transforms [36] a color image to a grayscale image, I, with the intensity value range (Min , Max):

$$I: \{X \subseteq R^n\} \rightarrow \{Min, .., Max\} \tag{1}$$

into a new image where intensity values change to the range ($newMin$, $newMax$).

$$I_n: \{X \subseteq R^n\} \rightarrow \{newMin, .., newMax\} \tag{2}$$

In our work, we eliminate illumination interference via intensity normalization using the Gaussian weighted average [32] of light intensity. It works in two steps. Initially, each pixel

value is deducted from a Gaussian-weighted average of its neighbor. Secondly, each pixel is divided by the standard deviation of the neighborhood. Equation (1) represents the pixel value calculation of intensity normalization:

$$X' = \frac{X - \mu_{nhgx}}{\sigma_{nhgx}} \tag{3}$$

where X' represents a new pixel value, X is the unusual pixel value, μ_{nhgx} is the Gaussian weighted average of the neighbor pixel of X, and σ_{nhgx} is the standard deviation of the neighbors of X.

3.2 Deep Convolutional Neural Network

In this section we present the network architectures use in our experiments. The main advantage of using a deep convolutional network in our work is that we can fine-tune the existing pre-trained network model instead of training the network from scratch. In our work, we use the openly available VGG16 pre-trained model from the Oxford Visual Geometry Group [10][33] and Inception model[38].Our pre-processed image sets are trained through a convolutional neural network where the pre-trained model is VGG16.The pre-processing inputs are passed through the network. Our network architecture uses the similar principals inspire by Simonyan & Zisserman et al. [39]. First, each pre-processed image of size $(H \times W)$ is determined as the input of the network. The rescaled image is passed through the number of convolutional layers where filter size is small only 3×3. The padding used here is 1 pixel for 3×3 convolutional layers. Pooling is completed by five max-pooling layers as shown in fig 3. Let convolution transpose y to get from x here H' is the length of the filter[9].

$$Y_{i"j"d} = \max_{1 \le i' \le H', 1 \le j' \le W'} x_{i"+i'-1, j"+j'-1, d.} \tag{4}$$

For max pooling stride is 2 and window is 2×2 pixel. The first two fully connected layer consists of 4096 channels each. The last fully connected layer covers 1000 channels for classification[40]. The final layer is the softmax layer. Softmax works as a combination of a normalize operator and an activation function. Softmax operator [9] can be calculate as follows:

$$y_{ijk} = \frac{e^{x_{ijk}}}{\sum_{t=1}^{D} e^{x_{ijt}}} \tag{5}$$

All hidden layers are equipped with Rectified Linear Unit (RelU) activation function. ReLU operator can be expressed as matrix notation

$$y_{ijd} = \max\{0, x_{ijd}\}. \tag{6}$$

Fig. 3 portrays the VGG16 model for ImageNet [40]. It has 13 convolutional layers and three fully connected layers. Adjacent convolution layers are connected through max-pooling layers. Each group contains a sequence of 3×3 convolution layers, enhanced from 64 at the beginning to 512 in the last group. This network was prepared for the ImageNet 1000 category classification challenge. For detection, ASSL uses Faster R-CNN [8].

Fig. 3. VGG16 model for ImageNet

After each dataset training, performance evaluation is ensure by the SSL prediction. If the prediction result is less than the threshold value of 0.9 and with false classifications, we applied AL. The entire process continued for each dataset until performance reached saturation.

We adopted the VGG16 [10] pre-trained network model, which performs well for person classification. We fine tune our network, and find that, during training, about 50,000 iterations produce optimal performance. We set the learning rate at 0.0001, and the batch size at 8. For AL human labeling, we consider a threshold value around 0.9 for optimal performance.

3.3 Proposed Algorithm

The proposed ASSL method is represent in Algorithm 1. This algorithm classifies human action with the smallest error. Let the input image datasets be $Unlabelled\ Data\ (UD_i)$. We consider unlabeled datasets for training sequentially, e.g. one to four. It is worth mentioning that labels for training datasets are known $Pretrained\ Model\ (M_{i-1})$. The VGG16 model is train with a two-person action dataset for the initial model, M_0. N refers to the maximum number in the dataset. Each unlabeled dataset is train with model M_0, then initialize to LD_{temp}. It should be noted that in the next step, C$urrent\ Dataset\ (CD_0)$ is combine with LD_{temp} and assigned to CD_{temp}. It is worth mentioning that in the second step, AL is applied where the threshold value is over 0.9, and is denote LD_i. Subsequently, CD_{i-1} is join with LD_{temp} and trained with model M_{temp}. Finally, model M_{temp} is compare with the previous model, and if the current model's performance is better than the previous model, training continues until convergence; otherwise, dataset is rolled back. Roll back procedure works as a catalyst to obtain better performance. If the current batch of data performs poorly we can come back to the previous better model and replace that current

batch data with noise free data which improves the performance.

Algorithm 1. Effective Hybrid Learning (EHL) Algorithm for Action Classification

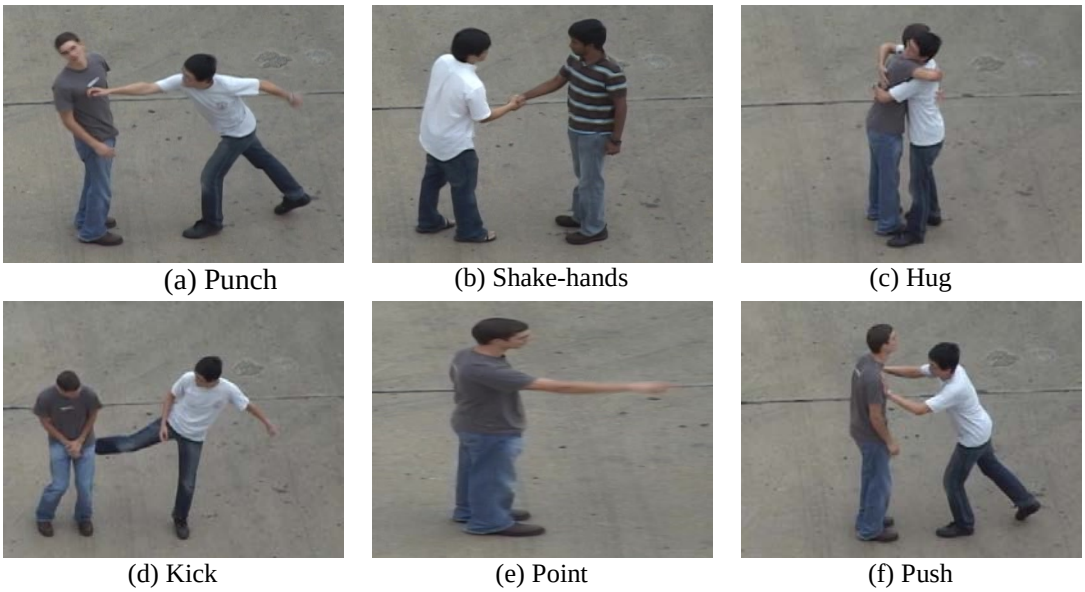
Input: C$urrent\ Dataset(CD_0) = \{\},\ Unlabelled\ Dataset\ (UD_i),$
$Pretrained\ Model\ (M_{i-1})$
Output: $Optimized\ Model(M_i), Labeled\ Dataset\ (LD_i)$
Begin
Step 1. Initialize model M_0 for training
Foreach i=1 to N **do** [N =DataSize]
LD_{temp}=$M_{i-1}(UD_i)$
$CD_{temp} = CD_{i-1}\ U\ LD_{temp}$
M_{temp} =$M_{i-1}\{CD_{temp}\}$
Step 2. Active Learning
LD_i=Active Learning (LD_{temp})
$CD_i = CD_{i-1}\ U\ LD_i$
$M_i = M_{temp}\ \{CD_i\ \}$
Step 3. Compare $M_i < M_{i-1}$
If M_i<M_{i-1}
Roll back to M_{i-1}
End if
End for
End

4. Experiments and Results

The main objectives of our experiments are to prove the efficiency of the framework with different conditional data of various numbers. We observe whether the performance improve with added data in the training set. Depending on the pre-trained model with different environments, our framework produce different accuracies. We carry out a rigorous experiment on a number of benchmark datasets such as UT-Interaction dataset[41], HMDB: a large human motion database [42] and the ITLab action datasets to check the overall performance of our framework. We consider these two benchmark dataset due to similar human action as ITLab action dataset such as hugging and fighting.

4.1 Dataset Overview

A. UT-Interaction dataset
Total six classes of human action interactions: shake-hands,point,hug, push,kick and punch are exist in UT-Interaction dataset [41].There are total 20 video clips whose lengths are around 1 minutes. Each video clips contains one interaction which has a different background, scale and illumination. Among them, hug and punch has similarity with our work. In order to train the network models, we use 835 images.



(a) Punch (b) Shake-hands (c) Hug

(d) Kick (e) Point (f) Push

Fig. 4. Examples of UT-Interaction dataset.

B. HMDB: a large human motion database

There are 51 action categories and 6766 video clips exist in this dataset where 7000 manually annotated clips extracted from different sources such as movies and youtube videos[42]. This dataset can be grouped into five categories such as 1) General facial actions: smile, laugh, chew, talk; 2) Facial actions with object manipulation: smoke, eat, drink; 3) General body movements: cartwheel, clap hands, climb, climb stairs, dive, fall on the floor, backhand flip, handstand, jump, pull up, push up, run, sit down, sit up, somersault, stand up, turn, walk, wave; 4) Body movements with object interaction: brush hair , catch, draw sword, dribble, golf, hit something, kick ball, pick, pour , push something, ride bike, ride horse, shoot ball, shoot bow , shootgun, swing baseball bat, sword exercise, throw; 5) Body movements for human interaction: fencing, hug, kick someone, kiss, punch, shake hands, sword fight. Among them, we consider hug and punch for experimental evaluation. We use 1133 images for training and testing this dataset.

(a)Punch (b) Punch (c) Punch

(d)Hug (e)Hug (f) Hug

Fig. 5. Example of HMDB dataset where action category is punching and hugging.

C. ITLab Action dataset

The ITLab action dataset consist of five different human actions: Hugging, Fighting, Linking arms, Talking, and Kidnapping. We consider two environments for data gathering: simple with a fair background and cluttered with a noisy background. We make 20 video clips where each action is include. From the video capture, we extracted an image sequence from the video frames. Every video clip contains 1000 JPEG files (altogether, 20,000 images). Each batch has 50 images where each action has 10 images.

4.2 Experiment Setup

Our experiment use a publicly available VGG network [10] that has five convolutional layers and three fully connected layers and Inception network [38]. These networks are pre-trained ImageNets for a 1000-category classification task. We aslo use adaptive gradient algorithm such as RMSprop[43], Adam[44] for optimization. For person detection, we use the Faster R-CNN [8] convolutional neural network based on an object detector and an ASSL framework that is implemented on the popular Caffe deep learning library [45]. All implementations were on a single server with cuDNN (Deep Neural Netowrk Library) [46] and a single NVIDIA GeForce GTX 970. We also used Matlab 2017a with the matconvnet [9] library for both Windows 7 and Ubuntu 14.4 operation systems in our experiments. We trained our initial model for 50,000 iterations with a batch size of eight and a learning rate of 0.001.

4.3 Evaluation

For the evaluation of our framework, we consider five types of actions: Hugging, Fighting, Linking arms, Talking, and Kidnapping in a simple environment, where the background is noise-free. From those 20,000 video frames, we selecte noise-free images from the unlabeled data, and then, incrementally updated the model. For active learning, we consider a threshold value of 0.9. In order to get accuracy, we divide the number of true positive images by the total number of images and multiplied by 100.

(a)Hugging (b)Fighting (c)Linked arm (d)Talking (e)Kidnapping

Fig. 6. Two-person action in simple environments of the ITLab action dataset.

For a simple background, we consider four training sets and one test set. The training dataset and the test dataset are labeled before training. Performance is shown in Fig. 7, where the initial training set's performance is not good (around 44% accuracy) due to noisy dataset. But after active learning apply with an updated model, performance increases gradually. For a second dataset, AL performance is 52% accuracy, and in the same way, after a third dataset, accuracy reached 78%. After the final dataset training is complete, the performance attain up to 96%. We find that the best case, with training and validation set using a simple background, is 98%; however, on average, performance is 96% accurate, where our model reached saturation.

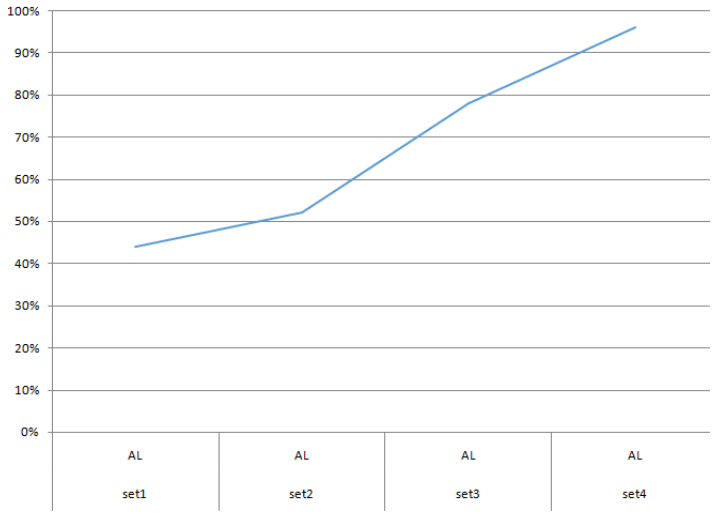


Fig. 7. The performance graph based on a simple-environment dataset.

(a) Hugging (b) Fighting (c)Linking arms (d) Talking (e)Kidnapping

Fig. 8. Two-person action in a complex environment of the ITLab action dataset

For the complex-background dataset where background is noisy, we consider four training sets and one test set. The training dataset and test dataset are labeled before training, as with the simple environment. Performance results are shown in Fig. 9, where the initial training set's performance is around 67% accuracy. But after active learning apply to the second set with an updated model, accuracy is boost to 75%, and gradually reach 77% for the third set. After the final dataset training is complete, performance reach up to 81% accuracy.

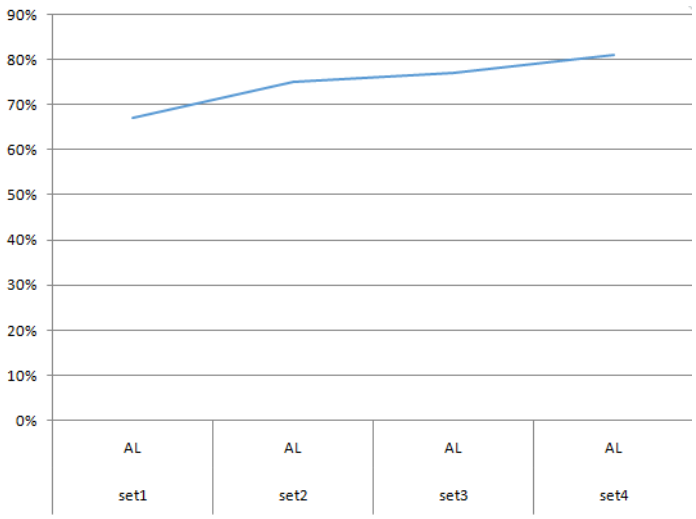


Fig. 9. Performance graph based on a complex-environment dataset.

We divided the entire simple-environment dataset into k equal-sized subsets. A single subset retain the validation data for testing the model, and the remaining k-1 subsets are use as training data. The cross validation process continued k times, with each of the k subsets use as a validation set, averaged to produce a single estimation. After k-fold cross validation, we obtained the results shown in Table 1, where average accuracy is around 95.7%.

Table 1. k-fold cross validation based on a simple environment dataset.

Training set	Test set	Average accuracy
6,7,8,9	10	92%
7,8,9,10	6	96.5%
6,8,9,10	7	94%
6,7,9,10	8	98%
6,7,8,10	9	98%

Among the five batches of ITLab action dataset we consider four as training sets and one test set. The respective average accuracy for all four training datasets of ITLab datasets are showed in **Table 1**.

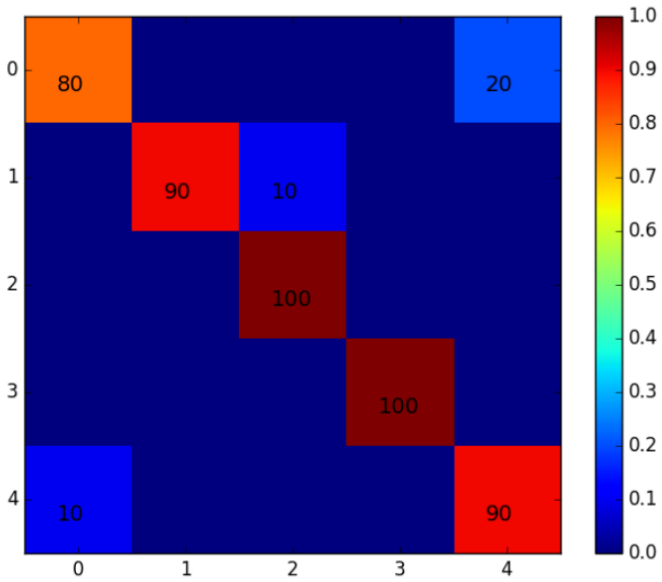


Fig. 10. Confusion matrix for two-person action in a complex environment

Fig. 10 represents a confusion matrix for two-person action in a complex environment, where the first row shows hugging, with true positive classification at 80% and false positive classification at around 20% for kidnapping. The second row represents the action of fighting, where true positive classification is 90% and false positive is 10% for linking arms. The third and fourth rows depict linking arms and talking, where true positives are 100%. Finally, false positives for kidnapping are classified as hugging about 10% of the time.

Table 2. Performance of the two environments in the ASSL framework.

Training set number	Test set	Environment	Performance
4	1	Simple environment	96%
4	1	Complex environment	81%

Table 2 Shows the results of our proposed method, using the ASSL framework for both simple and complex environments.

4.4 Benchmark dataset comparison

Table 3. Comparison method of action classification on UT-interaction dataset using VGG model with Adam optimizer.

Method	UT-Interaction dataset	precision	Recall	F-measure
VGG [ADAM]	Fight	1	0.95	0.97
	Handshake	0.97	1.00	0.99
	Hug	1.00	1.00	1.00
	Kick	1.00	0.96	0.98
	Point	1.00	0.97	0.99
	Push	0.94	1.00	0.97

Table 3 shows the performance using VGG model where adaptive gradient algorithm is Adam [44] used to network optimization. We show that our proposed method trained with VGG model and Adam optimizer performs well. In UT-interaction dataset has six classes of action in two different sets such as set 1 and set 2. Here we consider set1 and got better performance for hug, push, kick and punch. The precision, recall and F-measure shows that the proposed method outperforms previous work [17] where they manually extract the location of each person but we did not consider extract the location.

Table 4. Comparison method of action classification on UT-interaction dataset using VGG model with RMSProp optimizer.

Method	UT-Interaction dataset	Precision	Recall	F-measure
VGG [RMS]	Fight	0.94	0.85	0.89
	Handshake	0.94	1.00	0.97
	Hug	1.00	1.00	1.00
	Kick	0.96	0.92	0.94
	Point	1.00	0.94	0.97
	Push	0.92	1.00	0.96

Table 4 demonstrates the performance using VGG model where adaptive gradient algorithm RMSProp [43] use for network optimization. We show that our proposed method trained with VGG model and RMSProp optimizer performs well. Among the six human action Hug and Point performs well.The precision, recall and F-measure shows that the proposed method outperforms previous work[17].

Table 5. Few comparison methods of action classification on UT-interaction dataset.

Method	Accuracy on (UT-Interaction dataset)
Ryoo et al. 2009 [47]	70.8%
Branden et al. 2011[48]	78.9%
RPT+HV Yu et al.2015 [48]	85.4%
LMDI Sahoo et al. 2018[49]	87.5%
Ours	92.1%

The result of Table 5 shows the comparison method and their performance on UT-interactiond dataset. Our proposed Effective Hybrid Learning framework trained with UT-interaction dataset outperform other methods.

Table 6. Comparison of action classification on HMDB51 dataset using proposed method.

Approach	Average accuracy					
	Test set (200,0)	Test set (200,50)	Test set (200,300)	Test set (200,550)	Test set (200,800)	Test set (0,1050)
VGG [RMS]	98	76	63	60	61	60
VGG [ADAM]	94	77	71	72	70	69
Inception[RMS]	84	76	65	61	62	61
Ours	96	76	66	61	64	63

We evaluate the performance on HMDB51 dataset on a limited scale such as hug and fight due to many of the actions are the single person and poorly visible. In Table 6 “Test set (X,Y)” means the test data which is combined with Y from HMDB51 dataset and X test data from the ITLab action dataset. Common Effective Hybrid Learning (EHL) parameters are the number of batch size,total number of epoch and learning rate. Notably, these results outperform previously published result on HMDB51 dataset. Our method gains the lowest accuracy for Test set (0,1050) and gain better performance with ADAM optimizer 69% which is better than Varol et al. [18].We obtain the best performance on Test set(200,0) with VGG RMSProp optimizer.

5. Conclusion

In this paper, we propose a deep learning–based framework that produces outstanding performance for two-person action recognition. This framework is known as Effective Hybrid Learning framework, which decreases the human labeling that consumes a great deal of time. Our experiment shows state of the art performance on two benchmark Action recognition datasets UT-Interaction and HMDB51. The limited training data with various background and scale of image influences the result of the performance. Our proposed framework is significantly applicable to other domains in the computer vision area, such as object classification and video surveillance. In future, we intend to improve the accuracy of the proposed method in more complex environments with diverse human action recognition dataset.

Acknowledgments

This work was supported by INHA UNIVERSITY Research Grant. (INHA-59176)

References

[1] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," in *Proc. of 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, 2005. Article (CrossRef Link)

[2] I. Laptev, "On space-time interest points," *International Journal of Computer Vision*, 2005. Article (CrossRef Link)

[3] M. Hasan and A. K. Roy-Chowdhury, "A Continuous Learning Framework for Activity Recognition Using Deep Hybrid Feature Models," *Ieee Tmm*, vol. 17, no. 11, pp. 1909–1922, 2015. Article (CrossRef Link)

[4] B. Yao and L. Fei-Fei, "Modeling mutual context of object and human pose in human-object interaction activities," in *Proc. of IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 1, pp. 17–24, 2010. Article (CrossRef Link)

[5] C. Feichtenhofer, A. Pinz, and A. Zisserman, "Convolutional Two-Stream Network Fusion for Video Action Recognition," *Cvpr*, no. i, pp. 1933–1941, 2016. Article (CrossRef Link)

[6] A. Richard, "A BoW-equivalent Recurrent Neural Network for Action Recognition Bag-of-Words Model as Neural Network," *Bmvc2015*, 2015. Article (CrossRef Link)

[7] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proc. of 2016 IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 770–778, 2016. Article (CrossRef Link)

[8] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, 2017. Article (CrossRef Link)

[9] A. Vedaldi and K. Lenc, "MatConvNet Convolutional Neural Networks for MATLAB," 2016. Article (CrossRef Link)

[10] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps," *Iclr*, p. 1-, 2014.

[11] G. Gkioxari, U. C. Berkeley, R. Girshick, and U. C. Berkeley, "Contextual Action Recognition with R*CNN," *Cvpr*, 2015. Article (CrossRef Link)

[12] M. Stikic, K. Van Laerhoven, and B. Schiele, "Exploring semi-supervised and active learning for activity recognition," *Wearable Comput. 2008. ISWC 2008. 12th IEEE Int. Symp.*, pp. 81–88, 2008. Article (CrossRef Link)

[13] B. Settles, *Active Learning*, vol. 6, no. 1. 2012. Article (CrossRef Link)

[14] H. Wang, A. Kläser, C. Schmid, and C. L. Liu, "Action recognition by dense trajectories," in *Proc. of IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 3169–3176, 2011. Article (CrossRef Link)

[15] X. Peng, C. Zou, Y. Qiao, and Q. Peng, "Action Recognition with Stacked Fisher Vectors," *Eccv*, pp. 581–595, 2014. Article (CrossRef Link)

[16] P. F. Felzenszwalb, R. B. Girshick, D. Mcallester, and D. Ramanan, "Object Detection with Discriminatively Trained Part-Based Models." Article (CrossRef Link)

[17] K. N. E. H. Slimani, Y. Benezeth, and F. Souami, "Human interaction recognition based on the co-occurrence of visual words," in *Proc. of IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work.*, pp. 461–466, 2014. Article (CrossRef Link)

[18] I. Laptev and C. Schmid, "Long-term Temporal Convolutions for Action Recognition To cite this version : Long-term Temporal Convolutions for Action Recognition," vol. 40, no. 6, pp. 1510–1517, 2015. Article (CrossRef Link)

[19]Y. LeCun, K. Kavukcuoglu, and C. Farabet, "Convolutional networks and applications in vision," *ISCAS 2010 - 2010 IEEE Int. Symp. Circuits Syst. Nano-Bio Circuit Fabr. Syst.*, pp. 253–256, 2010. Article (CrossRef Link)

[20] G. Gkioxari, B. Hariharan, R. Girshick, and J. Malik, "R-CNNs for Pose Estimation and Action Detection," *arXiv Prepr. arXiv1406.5212*, pp. 1–8, 2014.

[21] C. Szegedy *et al.*, "Going Deeper with Convolutions," pp. 1–9, 2014. Article (CrossRef Link)

[22] D. Yarowsky, "Unsupervised word sense disambiguation rivaling supervised methods," in *Proc. of 33rd Annu. Meet. Assoc. Comput. Linguist.* -, pp. 189–196, 1995. Article (CrossRef Link)

[23] A. B. Goldberg, "Multi-Manifold Semi-Supervised Learning," pp. 169–176, 2009. Article (CrossRef Link)

[24] U. Ahsan, C. Sun, and I. Essa, "DiscrimNet: Semi-Supervised Action Recognition from Videos using Generative Adversarial Networks," *Computer Vision and Pattern Recognition*, 2018. Article (CrossRef Link)

[25] J. Zhang, Y. Han, J. Tang, Q. Hu, and J. Jiang, "Semi-Supervised Image-to-Video Adaptation for Video Action Recognition," *IEEE Trans. Cybern.*, vol. 47, no. 4, pp. 960–973, 2017. Article (CrossRef Link)

[26] S. Jones and L. Shao, "A Multigraph Representation for Improved Unsupervised / Semi-supervised Learning of Human Actions," *Cvpr*, 2014. Article (CrossRef Link)

[27] T. Zhang, S. Liu, C. Xu, and H. Lu, "Boosted multi-class semi-supervised learning for human action recognition," *Pattern Recognit.*, vol. 44, no. 10–11, pp. 2334–2342, 2011. Article (CrossRef Link)

[28] M. Li and I. K. Sethi, "Confidence-Based Active Learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 8, pp. 1251–1261, 2006. Article (CrossRef Link)

[29] J. Sourati, M. Akcakaya, D. Erdogmus, T. K. Leen, and J. G. Dy, "A Probabilistic Active Learning Algorithm Based on Fisher Information Ratio," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 8, pp. 2023–2029, 2018. Article (CrossRef Link)

[30] J. Bernard, M. Hutter, M. Zeppelzauer, D. Fellner, and M. Sedlmair, "Comparing Visual-Interactive Labeling with Active Learning: An Experimental Study," *IEEE Trans. Vis. Comput. Graph.*, vol. 24, no. 1, pp. 298–308, 2018. Article (CrossRef Link)

[31] S. Hao, J. Lu, P. Zhao, C. Zhang, S. C. H. Hoi, and C. Miao, "Second-Order Online Active Learning and Its Applications," *IEEE Trans. Knowl. Data Eng.*, vol. 30, no. 7, pp. 1338–1351, 2018. Article (CrossRef Link)

[32] Sinno Jialin Pan, Qiang Yang, "a Survey on Transfer Learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010. Article (CrossRef Link)

[33] G. P. C. Fung, J. X. Yu, H. Lu, and P. S. Yu, "Text classification without negative examples revisit," *IEEE Trans. Knowl. Data Eng.*, vol. 18, no. 1, pp. 6–20, 2006. Article (CrossRef Link)

[34] P. Wu and T. G. Dietterich, "Improving SVM accuracy by training on auxiliary data sources," in *Proc. of Int. Conf. Mach. Learn.*, pp. 110–118, 2004. Article (CrossRef Link)

[35] Y. Jie, Y. Qiang, and N. Lionel, "Adaptive Temporal Radio Maps for Indoor Location Estimation," *Pervasive Comput. Commun. 2005. PerCom 2005. Third IEEE Int. Conf.*, vol. 7, no. 7, pp. 85–94, 2005. Article (CrossRef Link)

[36] R. Gonzalez and R. Woods, Digital image processing. 2002. Article (CrossRef Link)

[37] A. T. Lopes, E. de Aguiar, A. F. De Souza, and T. Oliveira-Santos, "Facial expression recognition with Convolutional Neural Networks: Coping with few data and the training sample order," *Pattern Recognit.*, vol. 61, pp. 610–628, 2017. Article (CrossRef Link)

[38] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the Inception Architecture for Computer Vision," in *Proc. of 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. Article (CrossRef Link)

[39] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *Computer Vision and Pattern Recognition*, pp. 1–14, 2014. Article (CrossRef Link)

[40] O. Russakovsky *et al.*, "ImageNet Large Scale Visual Recognition Challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, 2015. Article (CrossRef Link)

[41] J. K. Ryoo, M. S. and Aggarwal, "Interaction Dataset, ICPR contest on Semantic Description of Human Activities (SDHA)," 2010. Article (CrossRef Link)

[42] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre, "HMDB: A large video database for human motion recognition," in *Proc. of IEEE Int. Conf. Comput. Vis.*, no. November 2011, pp. 2556–2563, 2011. Article (CrossRef Link)

[43] T. and G. H. Tieleman, "Lecture 6.5-rmsprop: Divide the gradient by a running average ofits recent magnitude.," *COURSERA Neural Networks Form. Learn.*, 2012. Article (CrossRef Link)

[44] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in *Proc. of conference paper at the 3rd International Conference for Learning Representations*, pp. 1–15, 2014. Article (CrossRef Link)

[45] Y. Jia *et al.*, "Caffe: Convolutional Architecture for Fast Feature Embedding," in *Proc. of the 22nd ACM international conference on Multimedia*, pp. 675-678, 2014. Article (CrossRef Link)

[46] S. Chetlur *et al.*, "cuDNN: Efficient Primitives for Deep Learning." Article (CrossRef Link)

[47] M. S. Ryoo and J. K. Aggarwal, "Spatio-temporal relationship match: Video structure comparison for recognition of complex human activities," in *Proc.of IEEE Int. Conf. Comput. Vis.*, no. Iccv, pp. 1593–1600, 2009. Article (CrossRef Link)

[48] W. Brendel and S. Todorovic, "Learning spatiotemporal graphs of human activities," in *Proc. of IEEE Int. Conf. Comput. Vis.*, no. Iccv, pp. 778–785, 2011. Article (CrossRef Link)

[49] S. Prakash Sahoo and S. Ari, "On an algorithm for Human Action Recognition," *Expert Syst. Appl.*, 2018. Article (CrossRef Link)

Minhaz Uddin Ahmed received his B.S. and M.S. degrees in Computer Science from the National University, Bangladesh, in 2006 and 2010. He is currently a Ph.D. student in the Intelligent Technology Laboratory, Inha University, Korea. His research interests include Human Action Recognition, Facial Expression Recognition, Machine Learning and Deep Learning.

Yeong Hyeon Kim received his B.S. degree in Computer Engineering from Inha University, Incheon, Korea in 2018. He is currently pursuing a Masters' degree at Inha University, where he is majoring in computer engineering. His research interests include Object Detection, Tracking and Localization, Deep Learning, Machine Learning, computer vision.

Jin Woo Kim received his B.S. degree in Computer Science and Information Engineering from Inha University, Incheon, Korea in 2013. He completed Masters' degree in 2017 from Inha University. His research interests include Image Classification, computer vision, deep learning, machine learning.

Md. Rezaul Bashar received the B.Sc. and M.Sc. degrees in Computer Science and Technology from the University of Rajshahi, Rajshahi, Bangladesh, in1996 and 1997, respectively. He was an assistant professor in Information and Communication Technology, Islamic University, Bangladesh. He completed his PhD from University of Southern Queensland in 2010. He joined as a Postdoctoral Research Fellow in The University of Sydney in 2010, currently, he is working as a data scientist in Science, Technology and Management Crest, Sydney, Australia. His research interest includes Face Recognition, Human Computer Interaction, Computer Vision, and Object Recognition

Phill Kyu Rhee received his B.S. degree in Electrical Engineering from the Seoul National University, Seoul, Korea, in 1982, an M.S. degree in Computer Science from the East Texas State University, Commerce, Texas, in 1986, and a Ph.D. degree in Computer Science from the University of Louisiana, Lafayette, Louisiana, in 1990. From 1982 to 1985, he worked as a research scientist in the Systems Engineering Research Institute, Seoul, South Korea. In 1991, he joined the Electronic and Telecommunication Research Institute, Seoul, South Korea, as a senior research staff member. From 1992 to 2001, he was an associate professor in the Department of Computer Science and Information Engineering of Inha University, Incheon, South Korea, and since 2001, he has been a professor in the same university and department. His current research interests are pattern recognition, machine intelligence, and autonomic cloud computing. Dr. Rhee is a member of the IEEE Computer Society and the Korea Information Science Society (KISS).