Advanced Science Letters
Vol. 4, 400–407, 2011

Deep CNN Object Features for Improved Action Recognition in Low Quality Videos

Saimunur Rahman, John See*, Chiung Ching Ho
Centre for Visual Computing and Informatics, Faculty of Computing and Informatics
Multimedia University, Cyberjaya 63100, Malaysia

Human action recognition from low quality video remains a challenging task for the action recognition community. Recent state-of-the-art methods such as space-time interest point (STIP) uses shape and motion features for characterization of action. However, STIP features are over-reliant on video quality and lack robust object semantics. This paper harness the robustness of deeply learned object features from off-the-shelf convolutional neural network (CNN) models to improve action recognition under low quality conditions. A two-channel framework that aggregates shape and motion features extracted using STIP detector, and frame-level object features obtained from the final few layers (i.e. FC6, FC7, softmax layer) of a state-of-the-art image-trained CNN model is proposed. Experimental results on low quality versions of two publicly available datasets – UCF-11 and HMDB51, showed that the use of CNN object features together with conventional shape and motion can greatly improve the performance of action recognition in low quality videos.

Keywords: Action Recognition, Low Quality Video, Feature Representation, STIP, Deep Learning, CNN.

1. INTRODUCTION

Video based action recognition has been an active area of research in recent years. This is due to the many potential applications such as video surveillance, video content analysis, human computer interaction and video archiving. The ongoing trend of research deals with many complex action recognition problems such as appearance, pose and illumination variations but problem video quality is still considered unexplored. Under low quality conditions, the major challenge is to develop robust feature representation methods that possess discriminative capacity for modeling actions. Many feature representation methods that have been proposed in recent years can be classified into two types: *hand-crafted* and *deeply-learned* features.

Among various *hand-crafted* methods proposed in literature, STIP[1], Cuboid[2], Hessian[3], dense sampling[4], dense trajectory[5] (DT) and improved dense trajectories[6] (IDT) are popular choices. These methods generally rely on two distinct steps for transforming spatio-temporal visual information into feature representations: *feature detection* and *feature description*. In the *feature detection* step, important interest points or motion trajectories are detected or tracked using well-known techniques such as Harris3D[1] and Dense trajectories[5], and then the visual patterns across the detected or tracked point or trajectory patches are described by feature descriptors such as

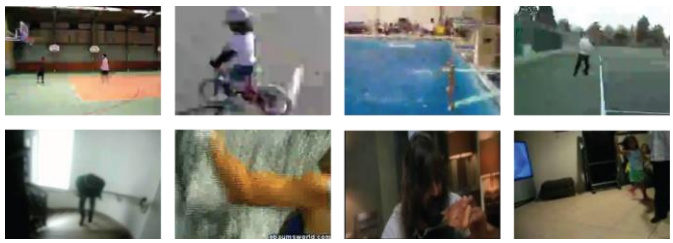

Fig.1. *Sample low quality videos form UCF-11 'compressed' version (top row) and HMDB51 'bad' and 'medium' quality subsets (bottom row).*

*Email Address: johnsee@mmu.edu.my

 1936-6612/2011/4/400/008 doi:10.1166/asl.2011.1261

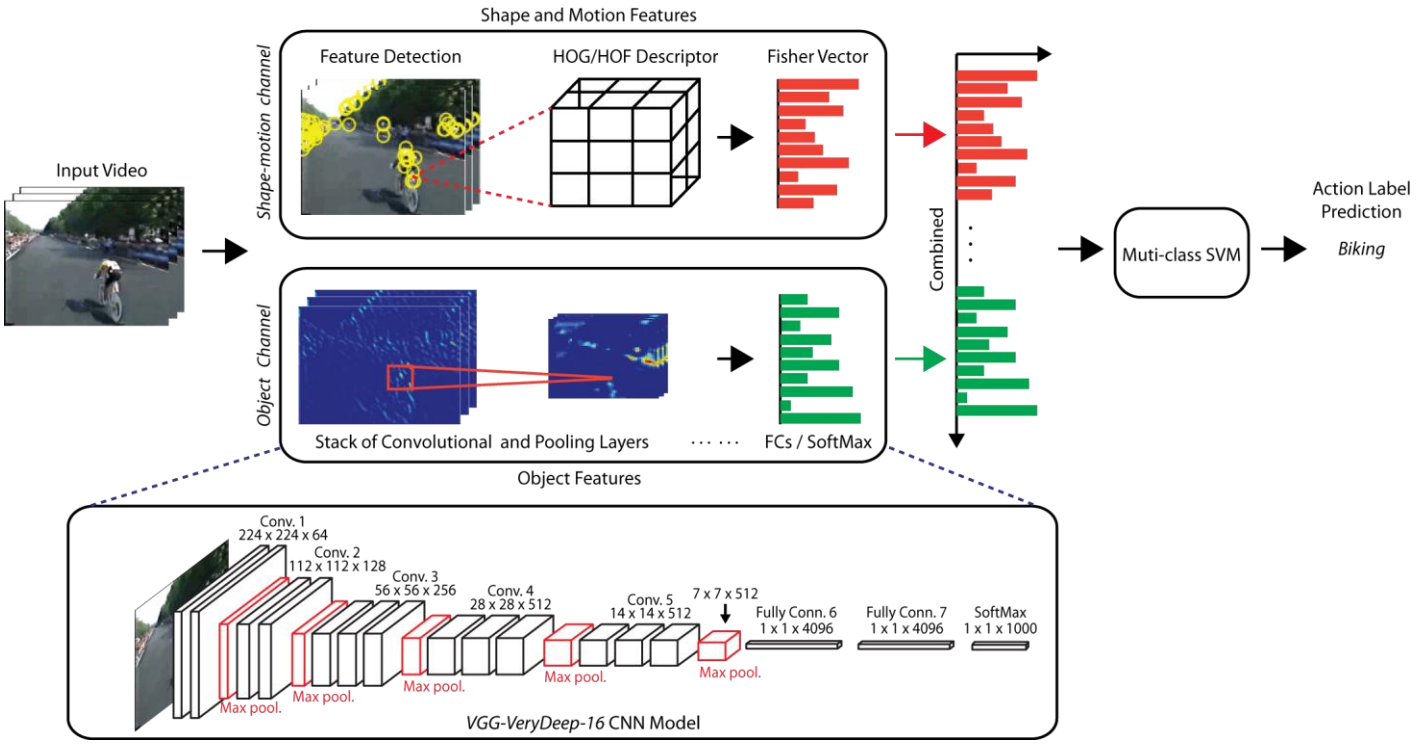

Fig. 2. *Overall framework for activity recognition in low quality videos*

Histogram of Oriented Gradients (HOG)[1] and Histogram of Optical Flow (HOF)[1] in the *feature description* step. However, despite excellent performances reported[4,5,6], *hand-crafted* methods have a number of limitations.

Firstly, they lack robust structural information. Most methods only utilize gradient based descriptor such as HOG[1] for representing shape information; this does not work well when spatial resolution becomes poor as the orientations of gradients are highly dependent on the image spatial quality[7]. Secondly, these methods mainly rely on local feature detection to represent essential information from moving body parts or complex person-object interactions. In the case of low video quality, the feature detector may fail to detect important interest points or motion trajectories. As such, scene and object information in a video may potentially provide additional clues as to the context of the action-of-interest. The aforementioned drawbacks of handcrafted methods motivates us to explore the role of object related features in improving existing methods and how it affects the performance of action recognition in low quality videos.

With the recent breakthrough in machine learning, *deeply-learned* features have grown in popularity for their excellent results across a diverse range of computer vision problems including action recognition. A number of methods based on deep convolutional neural network (CNN) have been proposed in recent literature with most of them reportedly outperform classic hand-crafted methods. Among popular methods, Karpathy et al.[8] used 3DCNN model to encode motion information in video. In another two-stream CNN model[9], two separate channels (considered as two separate CNN models) were used for encoding shape and motion information. However, obtaining an efficient CNN model is subject to tweaking millions of parameters[10] and a large amount of data samples are required for training in order to avoid over-fitting the classifier. Also, it requires a lot of computational resources and time for training. On the contrary, some researchers have proposed to use image-trained spatial CNN models as an alternative[10,11,12,13], which were reportedly able to match the performance of those trained directly with videos[8].

This paper investigates the role of deeply-trained CNN object features in improving the performance of action recognition under low quality conditions. The state-of-the-art 16-layer '*VGG-VeryDeep-16*' image-trained CNN model[14] is utilized to extract frame-wise object features. The concept of utilizing object features has been introduced in recent works[11,12] but never explored for the case of low quality videos. Moreover, for a richer feature set, a two-channel framework that combines the hand-crafted shape-motion features and deeply-learned CNN object features is proposed. Experiments were conducted on low quality versions of two datasets, followed by an analysis into the feature choice and computational cost.

2. FRAMEWORK FOR ACTION RECOGNITION

The overall framework for action recognition is shown in Figure 2. There are two channels: 1) shape-motion channel, and 2) object channel. In shape-motion channel, spatio-temporal interest points (STIPs) are detected across the video frames and then these points are described using both HOG and HOF local descriptors. The descriptor feature vectors are then encoded using Improved Fisher Vector (IFV) encoding[15]. The IFV is chosen over Bag of features (BoF) due to its superior performance[12]. In object channel, deep object features are extracted from video frames using an image-trained CNN model. Then, both features are combined to form the final feature vector. For classification, a non-linear multi-class support vector

machine (SVM) with χ^2 kernel[16], adopting a one-versus-all strategy is used. The class with the highest score is selected as the classified action label. A detailed elaboration of how the feature representations (STIP – HOG and HOF, and CNN feature layers) are extracted is further described in Section 3.

3. FEATURE REPRESENTATION

This section briefly discuss various feature detection and their representation strategies that are utilized in the proposed framework.

SHAPE AND MOTION FEATURES. For detection of shape and motion features Harris3D[1] interest point detector is used, which is known to be robust towards complex scenes[4] and is able to produce good detection results in low quality videos[17]. Briefly, the Harris3D detector computes a spatio-temporal second moment matrix at every video point μ defined as:

$$\mu(.;\sigma;\tau) = g(.;s\sigma;s\tau) * (\nabla L(.;\sigma;\tau) L(.;\sigma;\tau))^T \quad (1)$$

where, g is a separable Gaussian smoothing function and ∇L is the space-time gradient. The location of final STIPs are calculated by finding the local maxima H defined as:

$$H = \det(\mu) - k\,trace^3(\mu),\ H > 0 \quad (2)$$

The implementation and parameters from the author's original work[18] is used in this paper.

In order to characterize the local shape and motion information accumulated in space-time neighborhoods of the detected STIPs, the Histogram of Oriented Gradient (HOG) and Histogram of Optical Flow (HOF) features is extracted. They are chosen because of their good performance over other descriptors such as Cuboid[2] and ESURF[4]. The size of the HOG/HOF descriptor volumes is defined as: $\Delta_x(\sigma) = \Delta_y(\sigma) = 18\sigma$, $\Delta_t(\tau) = 8\tau$. Each volume is subdivided into a $n_x \times n_y \times n_t$ grid of cells; for each cell, a 4-bin HOG and 5-bin HOF are computed. As suggested by Wang et al.[4], $n_x = n_y = 3$ and $n_t = 2$ is used as descriptor grid parameters for all videos.

OBJECT FEATURES. For extraction of object features from action videos, an image-trained spatial CNN model named *'VGG-VeryDeep-16'*[14] *is used*. There are many image trained CNN models for object recognition available, but *'VGG-VeryDeep-16'* network is used due to its excellent performance[11] and low *top1*-error and *top5*-error in comparison with most of the recent pre-trained deep networks available publicly. The '*VGG-VeryDeep-16'* has 16 layers and is trained on the ImageNet dataset[18] which has 1000 image categories. The architecture of this model is shown in Figure 2 (bottom part of figure) while some sample features produced by different convolutional layers (CONV1 to CONV5) are shown in Figure 3.

MatConvNet – a freely available deep learning toolbox[19] is utilized for extraction of features from this network. The final output (softmax) of the model gives the decision probabilities of the object classes in an image frame, but it is not sufficiently discriminative for describing objects at the feature level. However, the last few convolutional layers (especially FC6 and FC7) keeps a more stable representation and offers relatively richer information of the frame which might be suitable for building object features. These layers provide features that can be comparable with mid-level encoded information. Under low quality conditions, these features may be able to complement local space-time feature representations that have less discriminative capacity when detected features greatly diminish. After extracting the features for all frames in the video, mean temporal pooling is applied to obtain the object features. Finally, features from both channels are concatenated into a final feature vector for classification thereafter.

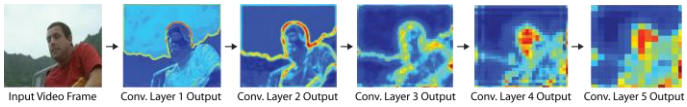

Fig. 3. *Sample features from the first five convolutional layers of 'VGG-VeryDeep-16 CNN' deep object model*[14]

4. DATASETS

This section describes the datasets that have been used for experiments. The proposed methods are evaluated on low quality versions which were methodologically created from two publicly available benchmark datasets: UCF-11 and HMDB51.

UCF-11[20] has 1,600 videos and 11 action classes. The videos are challenging and were collected from YouTube. The original resolution is 320x240 and further compression on each video sample in this dataset is applied by re-encoding the videos using the x264 encoder[21] with uniformly distributed constant rate factor (CRF)[**] values ranging from 23 to 50, across all action classes. (The higher the CRF value, the more compression). This compressed version of UCF-11 is denoted as 'UCF-LQ' hereafter. For evaluation, leave-one-group-out (LOGO) cross validation as specified by the original authors is used and the average accuracy across all action classes is reported.

HMDB51[22] has 6,766 videos and 51 action classes. The videos are annotated by quality-based meta-labels i.e. good, medium and bad. For evaluation purposes, the authors provided three training-test splits. For training, videos (from all quality types) as specified in all three training splits are used, while for testing, only the 'bad' and 'medium' quality videos from the testing splits[17] are used. These two poor quality test subsets are denoted as 'HMDB-BQ' and 'HMDB-MQ' hereafter. The average accuracy across all action classes from three splits is collected and the average of the splits is reported.

Some sample video frames from UCF-11 'compressed' and HMDB51 'bad' and 'medium' quality subsets is shown in Fig. 1.

[**]The complete distribution of all CRF values across all UCF-11 action classes is available for download at: http://saimunur.github.io/YouTube-LQ-CRFs.txt

1936-6612/2011/4/400/008 doi:10.1166/asl.2011.1261

4. EXPERIMENTAL RESULTS AND ANALYSIS

This section describes the experimental results on various feature combinations. A concise analysis of the results and demonstrated how the object features are able to aid the performance of conventional shape-motion features in low quality videos is also given.

RESULTS OF OBJECT FEATURES. Table 1 shows the results of using frame-level object features. Object features from three different layers of the *'VGG-VeryDeep-16'* deep model namely, the (final) softmax, FC6 and FC7 layers are obtained. From the results, the FC6 and FC7 layer features clearly performed better than the softmax score features. It is also noticeable that the concatenation of features from FC6 and FC7 layers improves the performance by a large margin. In the UCF-11-LQ, FC6 features performed better than the FC7 features. It is observed that the visual details of UCF-LQ compressed video frames can be quite distorted and hence, it loses a lot of valuable information when it passes through to a higher level of representation (FC7 and softmax). Interestingly, the softmax does help to increase the performance by a small amount if the quality of video is relatively better i.e. HMDB-MQ. Overall, FC6+FC7 performs well at a computationally feasible level.

Table 1. Experimental Results of Various Object Features on the Low Quality Datasets

METHOD	Dim.	UCF-LQ	HMDB	
			BQ	MQ
Softmax	1000	77.42	23.31	30.46
FC6	4096	**83.54**	23.31	30.50
FC7	4096	81.33	28.41	38.02
FC6+FC7	8192	83.13	**31.99**	**39.63**
FC6+FC7+softmax	9192	83.08	31.98	**39.70**

RESULTS OF SHAPE AND MOTION FEATURES. Table 2 shows the results of using shape and motion features. Compared to the results reported for object features, shape and motion features when taken individually, do not perform well. Shape features, in particular, performed poorly across all evaluated datasets; interestingly, it also performs badly when videos contained complex scenes in the case of HMDB-BQ videos. On the contrary, motion features performed better (than shape) but the scenario is completely opposite when videos are compressed (UCF-LQ).

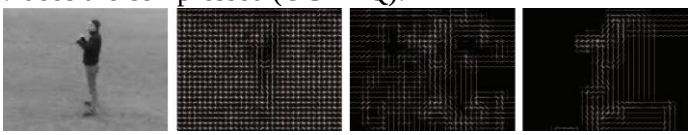

Fig. 4. *Performance of HOG features on compressed videos. The first image is the reference; second image is its associated HOG features; the third and fourth image are respectively, the HOG features when videos are compressed with constant rate factor (CRF) value of 40 and 50 using x264 video encoder[21].*

The shape and motion features used here are based on gradients and are highly dependent on the gradient magnitudes. If the visual details are good then naturally it offers good changes of magnitudes, but if the visual details are distorted then it will not offer magnitudes with significant changes (or sometimes even no variations even) hence, resulting in less discriminative features. Figure 4 shows the performance of shape features i.e. HOG when video frames are compressed. However, it is also interesting to see that the feature fusion between shape and motion features seems complementary, as it helps alleviate their individual limitations. Referring to the feature dimensionality, the dimensionality of shape and motion features is relatively higher than CNN object features. This is due to IFV encoding which produces a feature size of *2DK* (D is the dimension of feature vector i.e. for this paper HOG=72 and HOF=90, and K is the number of GMM clusters i.e. for this paper K=256).

Table 2. Experimental Results of Shape and Motion Features on Various Low Quality Datasets

METHOD	Dim.	UCF-LQ	HMDB	
			BQ	MQ
HOG	36864	63.57	8.15	10.40
HOF	46080	59.10	11.41	10.65
HOG+HOF	82944	**70.27**	**26.02**	**30.53**

Table 3. Experimental Results of Combination of Shape, Motion and Object Features on Low Quality Datasets

METHOD	Dim.	UCF-LQ	HMDB	
			BQ	MQ
HOG+FC6+FC7	45056	84.03	33.02	40.05
HOF+FC6+FC7	54272	85.16	32.80	40.41
HOG+HOF+FC6+FC7	91136	**86.34**	33.74	**40.55**
HOG+HOF+LBP-TOP[17]	85248	70.99	23.88	30.71
HOG+HOF+LPQ-TOP[23]	86016	71.65	25.02	30.75
STEM (w/o saliency)[24]	87040	75.04	33.78	38.76
STEM[24]	87040	77.50	**34.08**	38.94

RESULTS OF COMBINING SHAPE, MOTION AND OBJECT FEATURES. Table 3 shows the results of combining shape, motion and object features. For object features, we choose to use only the combination of FC6 and FC7 layers as it gives a marginally better performance compared to other combinations with a reasonable computational cost. The results show that the performance of shape or motion features greatly improved after combining with the object features. It improves even more when we combine all three (shape, motion and object) features i.e. HOG+HOF+FC6+FC7 together. However, this would increase the feature dimensionality and hence, computational more expensive. For all dataset except HMDB-BQ where STEM[24] alone

only slightly performs better, proposed methods achieves better or comparable results with the existing methods. This is due to the use of salient textures with shape and motion features that discriminately extracts textures from salient region of the video frame.

5. COMPUTATIONAL COMPLEXITY

This section compares the estimated computational cost of extracting shape-motion and CNN object features used in this paper. A sample video from 'bike_riding' action class of HMDB51 dataset is used, which has a resolution of 240x320 pixels and 246 video image frames at 30 frames per second *(fps)*. An Intel Core i7 PC with 24GB memory is used estimate the run-time. Table 4 shows the computational cost of the said methods. The extraction of shape-motion features is almost half of that of object features. The extraction of object features takes more time due to the feed-forward process through the CNN layers (16 layers deep). Note that the *'VGG-VeryDeep-16'* model used in this paper is based on the MatConvNet[19], which is comparably faster than those ported out to other frameworks.

Table 4. Computational Cost of Feature Extraction by shape-motion descriptors (feature detection+description) and *'VGG-VeryDeep-16'* object model

METHOD	Harris3D+ HOG+HOF (shape-motion)	VGG model (object)
Time per frame (sec.)	0.156	0.303

6. CONCLUSIONS

This paper proposed to use image-trained deep CNN model to obtain object features. These features have been proven to complement conventional shape and motion features very well in improving recognition of human actions in low quality videos. Experimental results on low quality versions of the UCF-11 and HMDB51 datasets have demonstrated the effectiveness of the proposed technique. As a future work, proposed method can be further improved by fine-tuning the image-trained CNN models using various action images to better tune the weights in the deep network.

ACKNOWLEDGMENTS

This work was supported, in part, by the Ministry of Education, Malaysia under Fundamental Research Grant Scheme (FRGS) project FRGS/2/2013/ICT07/MMU/03/4.

REFERENCES

[1] Laptev I. On space-time interest points. IJCV. 2-3, 75 **(2005)**.
[2] Dollár P, Rabaud V, Cottrell G, Belongie S. Behavior recognition via sparse spatio-temporal features. Proceedings of the 2nd Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance, **(2005)** October 15
[3] Willems G, Tuytelaars T, Van Gool L. An efficient dense and scale-invariant spatio-temporal interest point detector. In ECCV, **(2008)** Oct 12; Springer Berlin Heidelberg
[4] Wang H, Ullah MM, Klaser A, Laptev I, Schmid C. Evaluation of local spatio-temporal features for action recognition. In BMVC, **(2009)**; BMVA Press
[5] Wang H, Kläser A, Schmid C, Liu CL. Action recognition by dense trajectories. In CVPR, **(2011)** Jun 20; IEEE
[6] Wang H, Schmid C. Action recognition with improved trajectories. In CVPR, **(2013)**
[7] Rahman S, See J, Ho CC. Action Recognition in Low Quality Videos by Jointly Using Shape, Motion and Texture Features. In IEEE Int. Conf. on Signal and Image Processing Applications, **(2015)** October 19-21; Kuala Lampur, Malaysia
[8] Karpathy A, Toderici G, Shetty S, Leung T, Sukthankar R, Fei-Fei L. Large-scale video classification with convolutional neural networks. In CVPR, **(2014)**
[9] Simonyan K, Zisserman A. Two-stream convolutional networks for action recognition in videos. In NIPS, **(2014)**
[10] Ma S, Bargal SA, Zhang J, Sigal L, Sclaroff S. Do Less and Achieve More: Training CNNs for Action Recognition Utilizing Action Images from the Web. arXiv preprint arXiv:1512.07155, **(2015)**, Dec 22
[11] Murthy OV, Goecke R. Harnessing the Deep Net Object Models for Enhancing Human Action Recognition. arXiv preprint arXiv:1512.06498, **(2015)**
[12] Zha S, Luisier F, Andrews W, Srivastava N, Salakhutdinov R. Exploiting image-trained cnn architectures for unconstrained video classification. arXiv preprint arXiv:1503.04144, **(2015)**
[13] Ye H, Wu Z, Zhao RW, Wang X, Jiang YG, Xue X. Evaluating two-stream cnn for video classification. In ACM ICMR, **(2015)** Jun 22; ACM.
[14] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556, **(2014)**
[15] Wang X, Wang L, Qiao Y. A comparative study of encoding, pooling and normalization methods for action recognition. In ACCV, **(2012)** Nov 5; Springer Berlin Heidelberg.
[16] Vedaldi A, Zisserman A. Efficient additive kernels via explicit feature maps. IEEE PAMI. 3, 34 **(2012)**
[17] See J, Rahman S. On the Effects of Low Video Quality in Human Action Recognition. In DICTA, **(2015)** Nov 23; IEEE
[18] Deng J, Dong W, Socher R, Li LJ, Li K, Fei-Fei L. Imagenet: A large-scale hierarchical image database. In CVPR, **(2009)** Jun 20; IEEE.
[19] Vedaldi A, Lenc K. MatConvNet: Convolutional neural networks for matlab. In ACMMM, **(2015)** Oct 13; ACM
[20] Liu J, Luo J, Shah M. Recognizing realistic actions from videos "in the wild". In IEEE CVPR, **(2009)** Jun 20; IEEE
[21] Wiegand T, Sullivan GJ, Bjøntegaard G, Luthra A. Overview of the H. 264/AVC video coding standard. IEEE Transactions on Circuits and Systems for Video Technology, **(2003)** Jul; IEEE
[22] Kuehne H, Jhuang H, Garrote E, Poggio T, Serre T. HMDB: a large video database for human motion recognition. In IEEE ICCV, **(2011)** Nov 6; IEEE
[23] Rahman S, See J, Ho CC. Leveraging Textural Features for Recognizing Actions in Low Quality Videos. In Int. Conf. on Robotics, Vision, Signal Processing & Power Applications (ROVISP), **(2016)** February 2-3; Penang, Malaysia
[24] Rahman S, See J. Spatio-Temporal Mid-Level Feature Bank for Action Recognition in Low Quality Video. In IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP), **(2016)** March 20-25; Shanghai, China

Received: xx April 2016. Accepted: xx April 2016