Action Recognition in the Dark

Tan Jian An
EE6222 Machine Vision
Matric No. G2201738D

***Abstract*—In recent years, the application of human action recognition has been increasingly popular in many different fields such as surveillance and self-driving. Despite its rapid development, most of the research are done with some constraints on the dataset. While many models can achieve good performance for HAR in videos that are taken under adequate illumination, some of them might lose their effectiveness in dark videos. In this paper, image enhancement techniques are used to improve the contrast and illuminance of the image frames extracted from the ARID dataset through a fixed interval. Then, the enhanced images are inputted into the OpenPose model for the human heatmaps to be extracted. Finally, the generated heatmaps are fed through the ResNet-18 (2+1)D neural network to classify them into 10 different classes. The top-1 and top-5 accuracy of the proposed architecture is computed.**

***Index Terms*—Human Action Recognition, ARID, CLAHE, gamma correction, OpenPose, R(2+1)D**

I. Introduction

Nowadays, video data has become more and more common in our daily lives. As such, there is an increasing need for automatic video analysis tasks due to the ever increasing number of video files. Among those, human action recognition (HAR) is one of the most popular tasks due to its wide applicability and usefulness. HAR aims to intepret human actions through the use of computer and machine vision technology and classfify them automatically. While there is significant progress in current HAR research, in reality a lot of the video data is taken under poor conditions, i.e. insufficient illumination and contrast. Hence it can be challenging for HAR tasks if the video is taken under challenging environments [1]. Examples of applications of HAR in the dark include autonomous driving in the dark, night surveillance etc.

Dark images can be processed through pre-processing and segmentation. These two processes help to treat the images such that more valuable information can be extracted from it. First, pre-processing is implemented to enhance the image by removing unwanted noise. Next, image segmentation clarify and simplify the image representation for easier analysis [6].

This paper aims to introduce an efficient method to perform human action recognition in video clips in the dark. Dark video dataset ARID (Action Recoognition in Dark) is utilized for model training and validations [1]. The video in the dataset is split into image frames and image processing techniques are developed to enhance the image frames. Finally, the images frames are inputted to a 3D-CNN based network for training and inferencing.

In this paper, Section 2 briefly reviews the previous works related to HAR and HAR in the dark. Section 3 introduces the methodology on the proposed architecture, Section 4 gives detailed demonstrations on the experiment results and analysis, and Section 5 concludes the findings.

II. Related Work

A. Human Action Recognition (HAR)

According to [2], Human Action Recognition can be categorized into two different approaches, namely representation based solutions and deep networks based solutions. Representation based solutions are techniques that are constructed using handcrafted features. Depending on the level of locality captured, they can be further categorized into holistic representations as well as local representations and aggregation methods. Furthermore, Deep network based solutions can be sub-divided into three categories, namely 3D-CNN, Two-Stream networks, and self-attention mechanism To summarize, the techniques that are used for HAR is shown in fig.1.

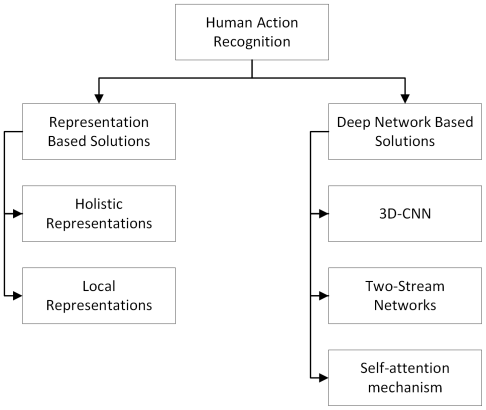

Fig. 1. Summary of HAR architectures

1) Representation based solutions

Human activity in a video can be translated into a three-dimensional space-time shape, which contains the human pose spatial information as well as its dynamic information [3]. However, As building an accurate three-dimensional model from videos can be complex and expensive, many solutions prefer representing actions at holistic or local level instead of 3D modeling [2].

a) Holistic representations

Holistic representation methods utilize this information to capture a global human body structure and movements representation to obtain a holistic motion information for HAR. The weakness of the holistic representation is that it can

be sensitive to noise. Moreover, as it captures the entire predefined rectangular region, unrelated information and noise may be introduced.

Bobick and Davis [4] introduced the Motion Energy Image (MEI) and Motion History Image (MHI) to allow the encoding of dynamic information of human activity into a single image. In MEI, the spatial information of the motion across the images frames is accumulated and combined into a single frame. Presented as a binary image, the MEI provides insights into the shape of the regions where movement is occuring [4].

While MEI is effective in representing the motion over time, it does not show how it is moving. On the contrary, MHI defines the pixel intensity with distinctive values according to the temporal information of the motion at that point. As such, a scalar image can be formed with more recently moving pixels having higher intensity. The major drawbacks of this method is that the background must be static; it is not view invariant; the motion in the video has to be short and simple; fine-grained actions cannot be easily detected; and it is sensitive to parameter selection.

b) Local representations

In local representations, local features are extracted from the motion. Laptev [24] proposed his work on the Space-Time Interest Points (STIPs). First, the interest point is detected. These points are points where the data variation in its space-time neighbourhood is high. This indicates the presence of a motion. Then, a cuboid known as the local descriptor is constructed from pixels detected around these interest points. These set of keypoints and descriptors are used to construct dictionaries so that the video can be represented through the keypoint frequencies and their corresponding descriptors. This method is less popular due to its complexities.

2) Deep Architectures

a) 3D-CNN

Unlike 2D CNNs, the filters in 3D convolution are designed to be 3-dimensional, such that the spatial and temporal information can be represented in separate dimensions. C3D [13] was first developed for action recognition, followed by larger and deeper networks such as 3D-ResNet [14] and 3D-ResNext [15]. Moreover, R(2+1)D [16] is proposed in attempt to improve video feature extractions by decomposing 3D spatio-temporal convolutions into spatial and temporal convolutions.

b) Two-stream methods

As videos possess various useful information with different modalities such as depth [17], RGB [18], optical flow [19] and skeleton [20] information. Two-stream methods make use of these different information and pass them into the feature extraction pipeline parallel through two or more pathways. Through this, video features rich in diversified modality information can be obtained. Among the modality information, Optical flow is one of the most popular option to be used in complement with RGB information. The major flaw of this method, however, is that apart from the RGB information, it is more difficult to obtain the information from other modalities - additional sensors are needed to obtain depth information whereas skeleton and optical flow are computationally expensive.

c) Self-attention mechanism

Gedas Bertasius et al. [25] first introduced self-attention mechanism to extract video features. It utilized standard Transformer architecture and applied temporal and spatial attention separately within each block. The drawbacks of the self-attention mechanism is that it requires huge amount of training data. As such, more researches prefer integrating self-attention into feature extractions instead.

B. HAR in the dark

To improve the low-light image quality, K. Dabov et al. [21] and Kin Gwn Lore et al. [22] introduced denoising techniques and autoencoders respectively.These pre-processing techniques are implemented before model training to improve results. In [23], a gaussian denouncer is used to create a probabilistic based image denoising model.

Komal et al. [6] implemented various denoising techniques on SONY SID dataset before applying ResUnet as well as Unet for model training. Although they managed to achieve 90% accuracy, it cannot be verified due to the lack of ground truth images.

Rui Chen et al. [12] proposed the DarkLight Networks consisting of a dual pathway structure to utilize both the raw dark video input and its enhanced version to represent the video effectively. They also introduced a self-attention mechanism to obtain the features from both pathways.

III. Methodology

There are a total of three different phases in the proposed video classification architecture, namely the frames extraction, data pre-processing and model prediction.

A. Frames extraction

In frames extraction, the raw input video is converted into an image sequence. Then, every 5th image frame is extracted from the sequence, until a total of 16 frames are collected. In other words, an image sequence should have at least 80 frames for complete frames extraction. For image sequence that has less than 80 frames, blank images are used to fill the remaining empty sampling frames; for those with more than 80 frames, the front and back end of the input clip are trimmed to shorten it to the correct number of frames. Extracting the frames at a constant interval can give good representation of the action flow and captures vital temporal information for deep learning model training. The sampling interval was obtained through trial-and-error method, and it had shown that inputs with greater interval was too sparse for model to yield a good result. Example of the image frames is shown in fig.2

Fig. 2. Image frames for the videoset Drink_3_16.mp4. As the total frames extracted from the video is less than 80, the last two frames are filled with blank images.

B. Data pre-processing

As most of the videos in the ARID dataset were captured in very dark environment with poor visibility, image enhancement is implemented to improve the visual quality of the image frames.

1) Contrast Limited Adaptive Histogram Equalization (CLAHE)

Historgram equalization (HE) is an effective way obtain a uniform histogram for the output image. The output image $g(x, y)$ is obtained by performing transformation on the gray-level of the input image $f(x, y)$:

$$g = T(f)$$
$$c(f) = \sum_{t=0}^{f} p_f(t) = \sum_{t=0}^{f} \frac{n_f}{n}, f = 0, 1, ..., L \quad (1)$$
$$g = T(f) = round[\frac{c(f) - c_{min}}{1 - c_{min}} L], c(f) \geq c_{min}$$

where c_{min} is the smallest positive value of all $c(f)$ computed.

Although HE is easy to implement and effective in enhancing the contrast of an image, it also posseses some drawbacks. Through HE, the intensity of each image pixel is distributed based on the information obtained from the entire image. This could cause over-enhancement where low-occuring pixel intensities are transformed and merged wiith its neighbouring pixels which occur more frequently [7]. Also, images could also face brightness preservation issue under HE transformation if mean shift problems occur [8].

Contrast Limited Adaptive Histogram Equalization (CLAHE) is adapted from the Adaptive Histogram Equalization (AHE) [9]. While there are some similarities in the operation between CLAHE and the conventional HE, several tweaks were made to improve the weakness found in HE. The input image is divided into several non-overlapping regions with similar dimensions. The regions can then be categorized into three groups - inner regions (IR), the border regions (BR)and the corner regions (CR) [9]. A histogram is computed from each region, then a clip limit for histogram clipping is obtained considering the desired limit for contrast expansion. The histogram from each region is recomputed such that it does not exceed the clip limit. Lastly, the CDFs of each histogram is obtained for greyscale mapping. CLAHE can limit the contrast amplification effectively, reducing the risk of noise amplification [10]. The transformation can be seen in fig.3

Fig. 3. CLAHE transformation with clip size of 10 and tile grid size of 8

2) Gamma Correction

After CLAHE, gamme correction is implemented. Gamma correction is often used to improve the luminance of an image. It is done by encoding and decoding the pixels of the image through non-linear statistical technique. The most common method is to perform the power transform given by the equation:

$$g = cf^{\gamma} \quad (2)$$

where g is the image output and f is the image input. The gamma correction is known as gamma compression when encoding gamma ($\gamma < 1$) is implemented as compressive power law non-linearity is introduced. Likewise, the process is known as gamma expansion when decoding gamma ($\gamma > 1$) is used [11]. The effect of gamma correction can be seen in fig.4

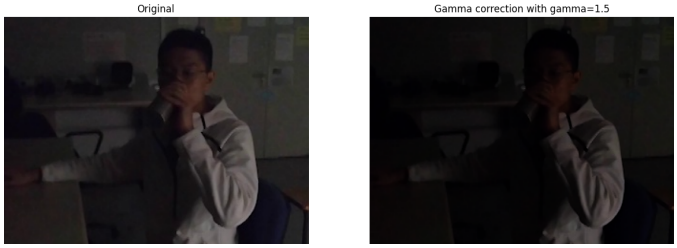

Fig. 4. Gamma correction with gamma of 1.5

3) CLAHE + Gamma Correction

In general, while CLAHE makes the dark background brighter, it is unable to differentiate the foreground from

the background if the images are dark. On the other hand, gamma correction enhances the overall luminance of the image through pixel encoding/decoding using non-linear statistical formula. To enhance both the contrast and luminance of the image, both CLAHE and gamma correction methods are combined to give a better image, as shown in fig.5

Fig. 5. Combination of CLAHE and gamma correction

The histogram of the original image and the enhanced image are depicted in fig.6. As can be seen, the distribution is more even across all pixel values in the enhanced image as compared to the original image, showing that the visibility of the image has been improved.

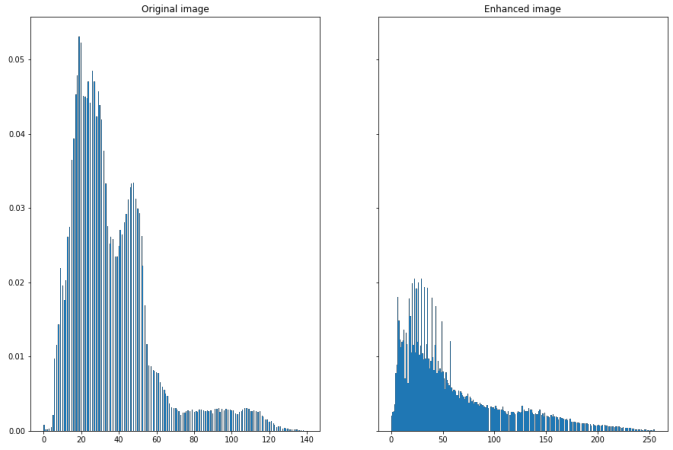

Fig. 6. Histogram of the original image and the enhanced image

4) Randomness in Dataset

After the enhancement of the image, random rotation, horizontal and vertical flip are applied to the image at a probability of 50% during training. By introducing randomness to the dataset, the risk of overfitting can potentially be reduced. The rotation, horizontal and vertical flips increase the variation of the training data and prevent the model from memorizing the data instead of learning it.

5) OpenPose Extractions

After the transformation of the image frames, the OpenPose algorithm (https://github.com/Hzzone/pytorch-openpose) with Python API is used to generate human skeleton and heatmaps from the images. OpenPose is a Real-time multiple-person detection library [26]. It is capable of detecting human body, face and foot keypoints by generating part affinity fields of the body parts. The OpenPose model selected for the human feature extraction was Coco default detection with a threshold of 0.05. In order for a clean OpenPose information to be generated, the background is disabled. Examples of keypoints and heatmaps generated from the image frames are shown in fig.7 and fig.8. Instead of the original image frames, the action classification problem is modelled as an image analysis, and heatmap images are used as the training data. The reason that keypoints are not used is because the keypoint image is too sparse and will not yield a good training result.

Fig. 7. OpenPose Keypoints

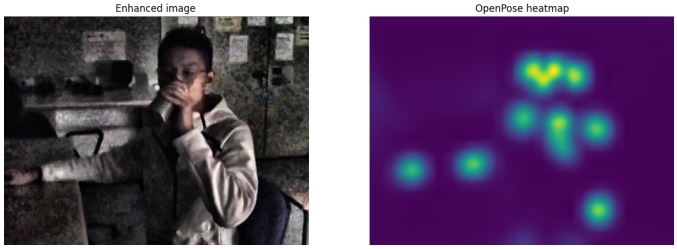

Fig. 8. OpenPose Heatmap

C. Modelling

1) Model architecture

Through novel deep residual learning paradigm, the ResNet models have become one of the best Convolutional Neural Networks (CNN) which have won the ImageNet Large Scale Visual Recognition Challenge 2015 in image classification, detection, and localization.

ResNet-18 is then adopted with (2+1)D blocks, which factorizes the 3D convolution explicitly into a 2D spatial convolution, and succesively a 1D temporal convolution [16]. Through this, the number of non-linearities can be doubled due to the additional non-linear rectification between the two operations. Hence, it is more capable of representing more complex functions as compared to networks using full 3D convolutions for the same number of parameterss. Besides, decomposing the the 3D convolution facilitates the optimizaition, thus reduce both the training and testing loss.

Due to the fact that the dataset available is small, pre-trained weights from Kinetics 400 without the top classification layer are used. The outputs are transformed into an array of 10 nodes per image through a linear layer, corresponding to the confidence level of the 10 classes available in the dataset. The node with the maximum value will be taken as the class prediction of the image.

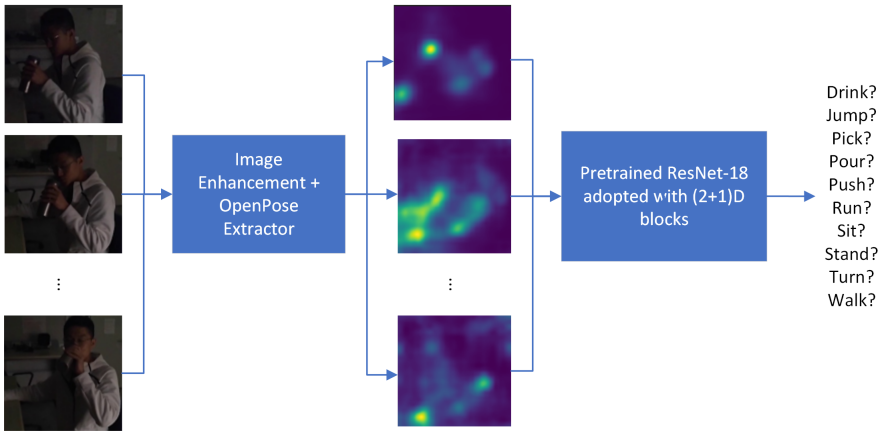

Fig. 9. Overview of the model architecture

D. Experiment

1) Experimental Details

The experiment was conducted on the ARID dataset which consists of 750 low brightness and contrast video clips distributed into 10 action classes. The Top-1 and Top-5 accuracies are reported. The experiment was done using PyTorch. The input consists of a sequence of frames with size of 8 x 3 x 16 x 224 x 224, i.e. 16 frames of RGB square images with size of 224 pixels in batches of 8. The heatmaps generated from the OpenPose are stacked once on each channel to form the RGB image. Cross Entropy Loss between the input logits and target was used as the loss function. For training, AdamW optimizer with adaptive learning rate was deployed. The experiment was carried out with 40 epochs.

2) Adaptive Learning rate

Learning rate is one of the most crucial parameters to train an effective model. Learning rate that is too high will cause the weights of the model to oscillate due to big changes and become unable to converge; learning rate that is too low will cause the model to train too slowly and risk getting stuck at the local minimum. To solve this problem, adaptive learning rates are adopted. Cosine learning rate decay is used in the training process. At first, the learning rate decay undergoes a warm-up stage where the rate slowly increases from 0 to 1e-4 for the first 10 epochs. This is because the first few epochs usually are the ones with the highest error rate. Starting the training with small learning rates would help to reduce the error. Once the learning rate peaks at the 5th epochs, it starts to decay following the cosine function towards zero at the last epoch, as shown in fig.10

3) Results

The results obatined by the proposed method are summarized in 3 tables: table.I shows the Top-1 and Top-5 accuracy of the proposed method on the validation as well as the test set; table.II and table.III depict the confusion matrix of the proposed method and the accuracy score of each class on the validation and test data respectively.

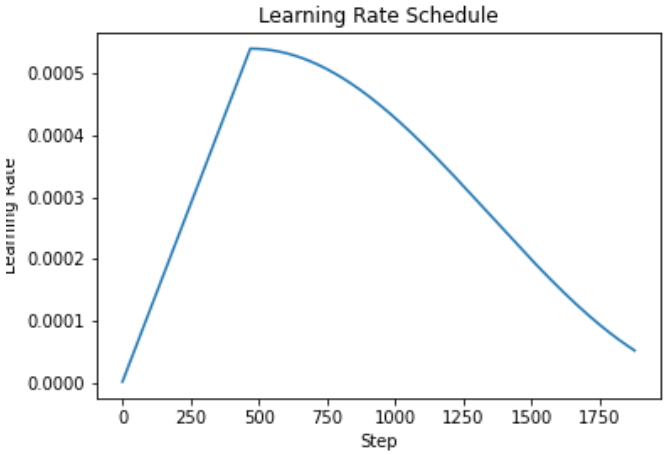

Fig. 10. Learning rate scheduler for the experiment

TABLE I
THE TOP-1 AND TOP-5 ACCURACY OF THE PROPOSED METHOD

Method	**Accuracy**	
OpenPose-R(2+1)D	***Top-1***	***Top-5***
Validation Data	51.250%	89.375%
Test Data	50.677%	93.333%

4) Discussion

The Top-1 accuracy scores are 51.25% and 50.67% on validation and test set respectively. In terms of Top-5 accuracy, the proposed method managed to produce circa 90% accuracy on both dataset. The fact that similar accuracies was achieved on different dataset as depicted in table.I show that the proposed method was able to produce consistent performance. To examine the performance of the proposed method on each class thoroughly, the confusion matrix is computed for both the validation and test set. The results obtained can be discussed in different aspects as follow:

TABLE II
CONFUSION MATRIX OF THE VALIDATION SET

Actual Class	Predicted Class										Class Accuracy
	Drink[0]	Jump[1]	Pick[2]	Pour[3]	Push[4]	Run[5]	Sit[6]	Stand[7]	Turn[8]	Walk[9]	
Drink[0]	13	0	0	4	4	0	0	1	2	0	54.167%
Jump[1]	0	24	1	0	3	0	0	0	2	1	77.419%
Pick[2]	12	1	5	6	1	0	1	2	2	5	17.857%
Pour[3]	11	0	0	33	1	0	1	0	0	0	71.739%
Push[4]	1	3	0	1	16	0	1	0	1	24	34.043%
Run[5]	0	5	0	0	0	6	0	0	0	15	23.077%
Sit[6]	0	4	0	2	4	0	17	3	0	3	51.515%
Stand[7]	2	3	0	0	2	0	1	23	0	1	71.875%
Turn[8]	3	6	0	0	7	0	0	1	10	1	35.714%
Walk[9]	0	0	0	6	1	0	0	1	0	17	68.000%

TABLE III
CONFUSION MATRIX OF THE TEST SET

Actual Class	Predicted Class										Class Accuracy
	Drink[0]	Jump[1]	Pick[2]	Pour[3]	Push[4]	Run[5]	Sit[6]	Stand[7]	Turn[8]	Walk[9]	
Drink[0]	9	0	0	15	0	0	0	0	16	0	22.500%
Jump[1]	1	49	0	0	3	1	0	0	1	3	84.483%
Pick[2]	9	2	4	9	6	0	0	0	3	0	12.121%
Pour[3]	4	0	5	30	1	0	0	0	0	2	71.429%
Push[4]	4	4	2	1	14	1	0	0	12	13	27.451%
Run[5]	0	8	0	0	3	12	0	0	0	7	40.000%
Sit[6]	0	4	2	1	14	0	13	9	0	2	28.889%
Stand[7]	1	3	0	2	6	0	5	20	1	0	52.632%
Turn [8]	2	0	1	0	4	0	0	0	48	0	87.273%
Walk[9]	3	2	0	0	19	1	0	1	3	29	50.000%

a) Effectiveness of the image enhancement

While many dark images are enhanced effectively through the methods discussed in Section 3, some images are originally too dark and their quality remain poor even after enhancement. Fig.11 shows one of the image frames in "Drink_6_16" that was poorly illuminated at first place. As can be seen in the figure, even though the position of the subject was visible for human eyes after enhancement, the contrast between the subject and the background was too low that the OpenPose model failed to recognize any body shape in the image.

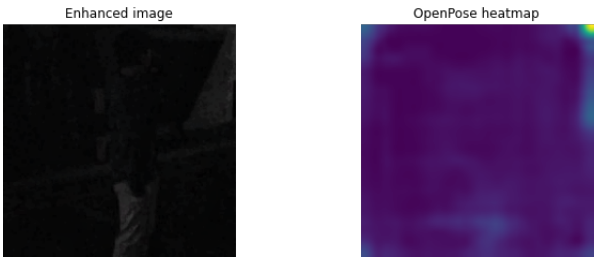

Fig. 11. Ineffective image enhancement on "Drink_6_16" data

b) Dataset shift

Fig.12 depicts the histogram against class label for the train, validation and test set respectively. It can be seen that the distribution of class label are different for all three dataset. This could cause dissimilarity between the train set and the validation/test set. As a result, the relationship between the input and output of the model could be data-sparse and misrepresented, which would affect the accuracy of the model predictions.

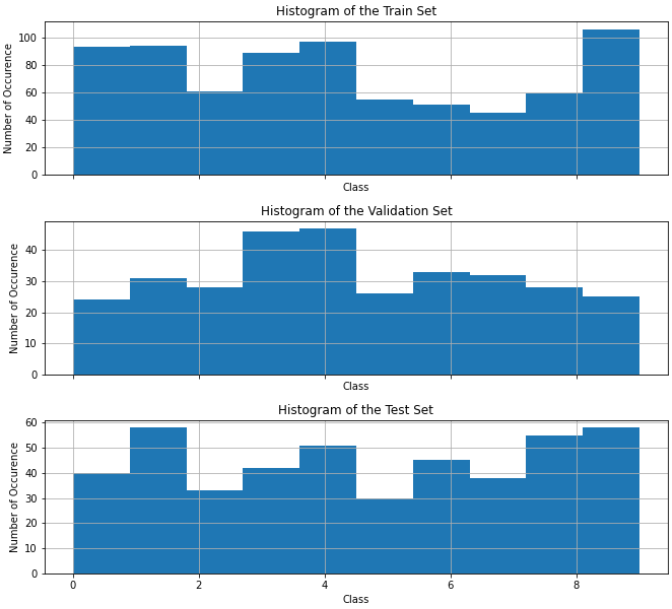

Fig. 12. Histogram of the Dataset

c) Number of image frames

One of the classes that had low accuracy score is the "Run" class. On the validation set, more than half of the class data were mistaken as the "Walk" class; on the test set, 23% of them were predicted as "Walk". This is intuitive as both the running and walking process are very similar, i.e. they have similar spatial information. The main difference comes from the temporal information, where "Run" motion is faster compared to "Walk".

This could be caused by the insufficient number of image frames. As discussed in Section 3, the proposed method extracts 1 image for every 5 frames from each video clip for training and inferencing. With a total of 16 images, the input might not be robust enough to represent the temporal information of the data. This can be improved by increasing the frequency of image extraction as well as the total number of frames. For example, instead of 1 image every 5 frames, images can be extracted once every three frames for a total of 32 frames. However, this can be computationally expensive and increases the runtime.

d) Similarity between different classes

From table.II and table.III, it can be seen that the class accuracy can be affected by other classes that are similar in nature. For example, at 17.86% and 12.12% on the validation and test set respectively, the proposed method did not perform well on the "Pick" class. For both dataset, most of the "Pick" motion were mistaken as "Drink" and "Pour". This could be due to the fact that all "Pick", "Drink" and "Pour" motions have similar nature in the sense that all of them focus on the upper part of the body, particularly the hand movement.

This could also partly explain the low accuracy score for the "Drink" class on test set.

On the contrary, the action classes that have good accuracy are "Jump", "Pour", "Stand" and "Walk". Particularly, "Jump" class scored 77.42% and 84.48% on the validation and test set respectively. This is because the "Jump" motion is unique among the classes and none of the other classes have similar movements.

This problem could potentially be reduced by increasing the image frame's extraction frequency as well as the input's total number of image frames. By increasing the input size, more spatial and temporal information can be fed through the model for training.

e) Amount of data available for training

Although transfer learning approach was adopted, where pre-trained weights from Kinetics 400 were used, the accuracy scores obtained from the validation and test set show that the provided train dataset consisting of 750 videos across 10 classes was inadequate for the proposed method to have an effective training. As a consequence of insufficient data, the model could potentially be overfitted. This caused the model to memorize the pattern of the training set instead of learning them, resulting in low training errors but high test errors. Therefore, increasing the data size would certainly help to improve the training of the model.

CONCLUSION

In this paper, an architecture is introduced for human action recognition in the dark. The image frames are extracted from the ARID video dataset using a fixed interval. Conventional image processing techniques such as CLAHE and gamma correction are used to improve the contrast and illuminance of the dark images. OpenPose model is deployed to extract human heatmaps from the improved images, which are then fed through the ResNet-18 R(2+1)D deep learning neural network. Experiment shows that while good results can be obtained for many action classes, the architecture has poor performance on some of the classes. Nevertheless, the architecture could be improved if more dataset are used for training.

REFERENCES

[1] Yuecong Xu, Jianfei Yang, Haozhi Cao, Kezhi Mao, Jianxiong Yin, Simon See, "ARID: A New Dataset for Recognizing Actions in the Dark in 2020".

[2] Samitha Herath, Mehrtash Harandi and Fatih Porikli, "Going deeper into action recognition: A survey", Image and vision computing, 2017, 60, 4-21.

[3] J. Yu Kong and Yun Fu, "Human Action Recognition and Prediction: A Survey", Int J Comput Vis 130, 1366–1401 (2022).

[4] A.F. Bobick and J.W. Davis, "The recognition of human movement using temporal templates", IEEE Trans. Pattern Anal. Mach. Intell., 23 (3) (2001), pp. 257-267

[5] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, "Electron spectroscopy studies on magneto-optical media and plastic substrate interface," IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].

[6] Komal Mourya, Sharda Patil, Tabasum Nadaf, Divya Voccaligara, Harsha Chari, Shailendra Aswale, "Techniques for Learning to See in the Dark: A Survey in 2020".

[7] Gonzalez RC, Woods RE, "Digital image processing", Prentice Hall, Upper Saddle River, 2002.

[8] Hussain, K., Rahman, S., Rahman, M.M. et al., "A histogram specification technique for dark image enhancement using a local transformation method", IPSJ T Comput Vis Appl 10, 3 (2018).

[9] Ali M Reza, "Realization of the Contrast Limited Adaptive Histogram Equalization (CLAHE) for Real-Time Image Enhancement", Journal of VLSI Signal Processing 38, 35–44, 2004.

[10] S. M. Pizer, E. P. Amburn, J. D. Austin, et al, "Adaptive Histogram Equalization and Its Variations", Computer Vision, Graphics, and Image Processing 39 (1987) 355-368.

[11] Harsh Rakesh Patel, Jash Tejaskumar Doshi, "Human Action Recognition in Dark Videos", 2021 International Conference on Artificail Intelligence and Machine Vision (AIMV).

[12] Rui Chen, Jiajun Chen, Zixi Liang, Huaien Gao, Shan Lin, "DarkLight Networks for Action Recognition in the Dark" 2021.

[13] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, Manohar Paluri, "Learning spatiotemporal features with 3d convolutional networks" In Proceedings of the IEEE international conference on computer vision, pages 4489–4497, 2015.

[14] Kensho Hara, Hirokatsu Kataoka, Yutaka Satoh, "Learning spatio-temporal features with 3d residual networks for action recognition" In Proceedings of the IEEE International Conference on Computer Vision Workshops, pages 3154–3160, 2017.

[15] Kensho Hara, Hirokatsu Kataoka, Yutaka Satoh, "Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet?" In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, pages 6546–6555, 2018.

[16] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, Manohar Paluri, "A closer look at spatiotemporal convolutions for action recognition" In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, pages 6450–6459, 2018.

[17] Hossein Rahmani, Arif Mahmood, Du Huynh and Ajmal Mian, "Histogram of oriented principal components for crossview action recognition", IEEE transactions on pattern analysis and machine intelligence, vol. 38, no. 12, pp. 2430-2443, 2016.

[18] Junnan Li, Yongkang Wong, Qi Zhao and Mohan S Kankanhalli, "Unsupervised learning of view-invariant action representations", arXiv preprint arXiv:1809.01844, 2018.

[19] Zhaofan Qiu, Ting Yao, Chong-Wah Ngo, Xinmei Tian and Tao Mei, "Learning spatio-temporal representation with local and global diffusion", Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12056-12065, 2019.

[20] Bin Li, Xi Li, Zhongfei Zhang and Fei Wu, "Spatio-temporal graph routing for skeleton-based action recognition", Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, pp. 8561-8568, 2019.

[21] K. Dabov, A. Foi, V. Katkovnik and K. Egiazarian, "Image Denoising by Sparse 3-D Transform-Domain Collaborative Filtering", in IEEE Transactions on Image Processing, vol. 16, no. 8, pp. 2080-2095, Aug. 2007.

[22] Kin Gwn Lore, Adedotun Akintayo, Soumik Sarkar, "LLNet: A deep autoencoder approach to natural low-light image enhancement," Pattern Recognition, Volume 61, 2017, Pages 650-662.

[23] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, Lei Zhang, 'Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising," IEEE Transactions on Image Processing 26(7), 2017.

[24] Ivan Laptev, " On space-time interest points," Int. Journal of Computer Vision, 64(2):107–123, 2005.

[25] Gedas Bertasius, Heng Wang and Lorenzo Torresani, "Is space-time attention all you need for video understanding?", arXiv preprint arXiv:2102.05095, 2021

[26] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, Yaser Sheikh, "OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields", arXiv preprint arXiv:1812.08008