Human Activity Recognition using PCA and BiLSTM Recurrent Neural Networks

Amir A. Aljarrah
MATHCOMP, University of Kufa
Najaf, Iraq
amir.aljarrah@student.uokufa.edu.iq

Ali H. Ali
Department of Electronic and Communications Engineering ,University of Kufa
Najaf, Iraq
alih.alathari@uokufa.edu.iq

Abstract— **This paper presents an original approach to Human Activity Recognition (HAR) tasks based on wearable sensors data. We have trained a Bidirectional Long-Short Term Memory (BLSTM) Recurrent Neural Network (RNN) model to predict the physical activity of subjects taken from the mHealth dataset. PCA has been used to reduce dimensionality of the dataset while keeping the total variation at no less than 85%. The resulting dataset consisted of 15 features instead of features at a reduction rate of 34.7%. The mhealth has 12 different activities recorded out of 10 subjects. We present a comparison of the PCA-BLSTM approach to several classical machine learning models. Results indicated that the PCA-BLSTM model has registered the highest accuracy of 97.64%. The SVM algorithm has registered the least accuracy of 64.67. We recommend the application of this procedure to other HAR datasets which have higher dimensionality and complexity of human activities.**

***Keywords**—- Principal component analysis, Health dataset, human activity recognition, recurrent neural network, deep learning*

I. Introduction

In the recent years, Human Activity Recognition (HAR) using multiple wearable sensors has been one of the highest demanded application in monitoring many real-world industries like healthcare, personal fitness, gaming, …, etc. HAR is categorized into two major classes: vision-based and sensor-based activity recognition. In vision-based class, images of human skeleton are constructed by using techniques such as neural network. One example of such class is in[1] where activities are inferred from angles and distances between joints out of a dataset consisting of 146 images. Their accuracy was up to 95%. A live video feed can also be used to acquire video sequences and to apply various algorithms to classify features out of it such as the MLP-NNs algorithm [2].

In the sensor-based class, various approaches have been proposed as to the sensors attachment to body, types, and quantities. For instance, Hsu, Wei-Chun and others have placed seven sensors at both sides of the thigh, lower back, foot, and distal tibia (shank) and achieved an accuracy of 89.13% by using decision trees [3]. Another approach used one accelerometer attached to waist for gait speed and energy [4]. An alternative approach to wearable sensors is to use environment for estimating parameters such as the number of persons in a lab or room by machine learning using features like heat and properties of pressure [5]. Other methods in machine learning have also been investigated, e.g. k-Nearest Neighbors (k-NN) classifiers and multiclass Support Vector Machine (SVM) using data from accelerometer and gyroscope sensors [6]. In addition, random forests, SVM and decision trees, have been used to model data acquired from sensors attached to the right-wrist and left-wrist which resulted in 60 features and an accuracy of 81.1% when the decision tree method is used [7]. Hidden Markov Models for classifying human physical activity from five bi-axial accelerometers. The accuracy of this approach was found to be 69.18% [8] and 76.79% when SVM is used [9].

Deep learning is one important family of machine learning methods based on learning from data. One of widely used member of that family is the Convolutional Neural Networks (CNNs). CNN for HAR has been proposed in many recent papers. For instant, using CNN for shallow feature extraction equipped with simple statistical features. When applied to the WISDM and UCI dataset, accuracy was reported as up to 94.35% [10]. In addition, smartphones which have triaxial accelerometer has been used for recognizing human activity with 1 dimensional CNN [11], the use of accelerometers and gyroscopes or combination of 1D and 2D CNNs in two different architecture is discussed in[12][13][14][15]. In order to facilitate the use of CNN for streaming data, such as the case of HAR, inputs are buffered and converted into blocks which are then fed to the CNN [12][13]. The other member of deep learning family is the so-called Recurrent Neural Networks (RNN). RNN is specifically crafted to handle streaming data. For instance, [16] have constructed Deep ConvLSTM model for human activity recognition based on the Long Short-Term Memory (LSTM) model and Multimodal Wearable sensors.

In this paper, we propose BiLSTM RNN model equipped with PCA to classify human activities out of the mHealth dataset [8]. Both of which have received little attention. The feasibility of this method can be useful in applying the same procedure to other HAR datasets and in order to reduce complexity of training and processing by reducing the number of features in these datasets.

The rest of this paper is divided as follows: explanation of our approach is in Section II. Section III contains description of datasets along with discussion and experimental results. Finally, conclusion and future research directions are given in Section IV.

978-1-7281-4105-3/19/$31.00 ©2019 IEEE

II. METHODOLOGY

This section presents the details of the PCA and RNN methods that have been used for the purpose of human activity recognition. Our approach has been applied specifically on the mhealth dataset [8]. The mhealth dataset consists of three 3-axis accelerometer sensors, two 3-axis gyroscope sensors, two 3-axis magnetometer sensors, and one 2-leads electro-cardiogram sensor. It is one the most feature rich dataset and represent quite a challenge for HAR because of the high computational power requirements for training and predicting activates out of it. Fig. 1 shows a flowchart of the steps performed in this research in order to predict the activities associated with the mhealth. For simplicity, we will discuss it in terms of three major steps as follows:

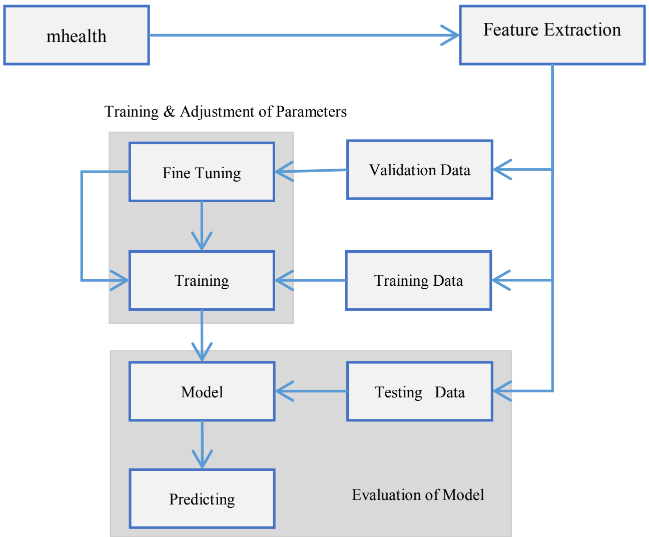

Fig. 1: flowchart of the three steps performed in this research.

Step one: Feature Extraction

In this step, we have used PCA transformation in order to reduce the number of dimensions that the RNN has to deal with, since deep learning is very computationally costly. However, some of the mhealth samples are classless. These samples were originally labelled as “unknown”. After several training attempts, we have opted to omit these samples from training, training and testing. The fraction of unlabeled data is very small that we think would not tip the balance of accuracy to our side. We have used the following simple formula to clean out the dataset:

$$D = \sum_{i=1}^{n} Da_i | (y_i \, != 0) \qquad (1)$$

Where D is the sample to be processed and y_i is its label. A label of (0) value indicate the unknown class and is, hence, removed from the dataset. Then, we used simple random sampling to divide the dataset into three holdouts: training, which accounts for 60% of the dataset, validation, and testing each with a fraction of 20%. Finally, we have performed standardization of data using the z-score method as in equation 2:

$$Z_i^{\mathcal{F}} = \frac{e_i^{\mathcal{F}} - \mathcal{M}_{\mathcal{F}}}{\delta_{\mathcal{F}}} \qquad (2)$$

Where $e_i^{\mathcal{F}}$ is element i in feature $\mathcal{F}$, $\mathcal{M}_{\mathcal{F}}$ is the mean of feature $\mathcal{F}$ and δ is the standard deviation [17].

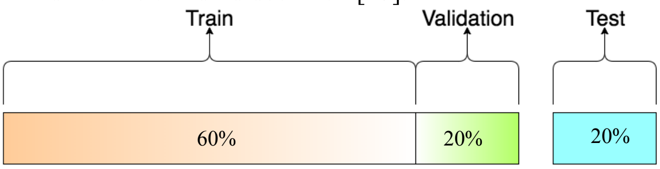

Fig. 2 Proportion of training, validating and testing sets

Step two: Training

We have used a BiLSTM RNN architecture which we thought is more suitable for HAR. Our design of the BiLSTM RNN had six layers (sequence input, Bidirectional long short-term memory (BiLSTM) layer, long short-term memory (LSTM), fully-connected, Softmax, and classification) layer. Fig. 3 shows an exploded diagram of the layers and their specifications.

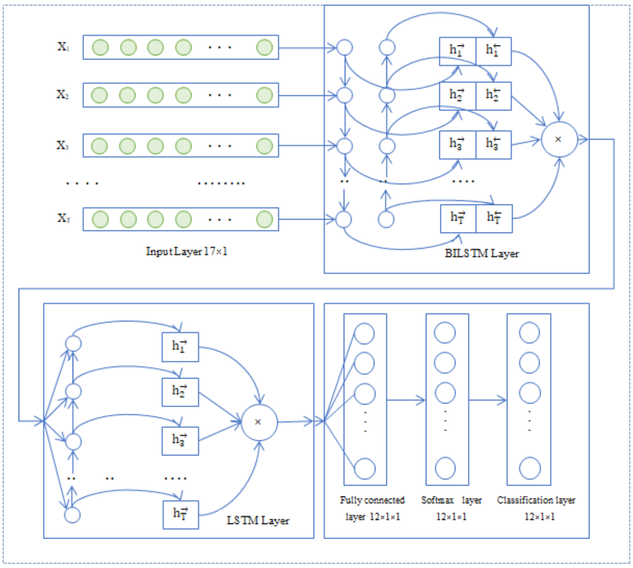

Fig. 3. The BiLSTM RNN model designed in our esearch.

There are 17 input sequence layers. The output of layer BiLSTM equal 300 with 150 hidden units and number of parameters 201,600. In the LSTM layer, the output equals 150 with 150 hidden units and number of parameters 270,600. The fully connected network has input size of 150 and output size of 12 since there is 12 classes in the mhealth dataset.

We have used MATLAB to train the model using following set of parameters: max epochs was 60, mini batch size = 1000, gradient decay factor = 0.9000, squared gradient decay factor =

0.999, epsilon= 0.00000001, initial learn rate= 0.001, l2regularization = 0.0001, gradient threshold method: 'l2norm', gradient threshold = 1, shuffle = 'never', execution environment: 'GPU', sequence length=longest and an adaptive learning rate (ADAM).

Step Three: Predicting and Tuning

In order to measure the performance of our design against others while avoiding bias, we have used the validation subset to validate the accuracy of our design and terminate the training accordingly and the testing subset for comparison with others. The number of layers and their details shown in fig. 3 were the result of several trail and error procedure so as to optimize the performance of the model and reach to the best accuracy possible.

III. EXPERIMENT AND RESULT

This section will present the results of training our BiLSTM RNN model to classify mhealth dataset. The mhealth dataset contains data collected out of wearable sensors attached to ten subjects while performing twelve physical activities (standing still (1 min), sitting and relaxing (1 min), walking (1 min), lying down (1 min), climbing stairs (1 min), waist bends forward (20x), knees bending (crouching) (20x), frontal elevation of arms (20x), running (1 min), cycling (1 min), jogging (1 min), and jumping front & back (20x)) [8]. Activities are recorded with a sampling frequency of 50Hz from 4 different types of sensors: three 3-axis accelerometer sensors (placed on chest, right wrist, left ankle), two 3-axis gyroscope sensors (placed on right wrist, left ankle), two 3-axis magnetometer sensors (placed on chest, right wrist, left ankle), and one 2-lead Electrocardiogram sensor (placed on chest) [8]. The total number of features is 23. as discussed before, we have divided the dataset using simple random sampling into training data 60%, validation data 20%, and test data 20%. The training proportion was used to train the model, validation portion was used to decide on when to stop the training. Finally the test set is used to calculate the accuracy.

In the PCA steps, we have assumed a variation of no less than 95% which resulted in dimensionality reduction to 17 instead of 21. We used MATLAB v2018a to train and test the model on an Nvidia GPU. The max epochs were set as 60 using trial and error, "Mini batch size", was put at 1000. Fig.4 shows the interactive training window result during training. The validation set accuracy was found to be 97.64% when PCA variance is set at 95%, mini batch size of 1000, max epochs of 60 and a number of iterations of 205. Fig. 5 shows the MATLAB's loss & iteration window.

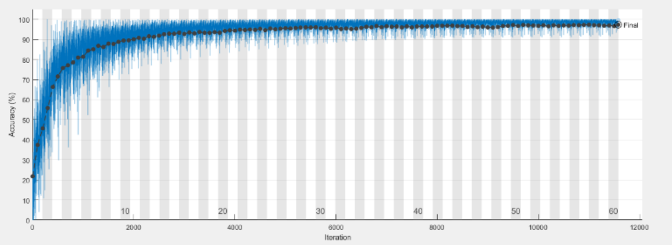

Fig. 4. Accuracy improvement during the training phase BiLSTM RNN

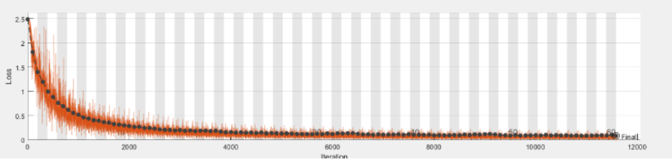

Fig. 5 Loss improvement during the training phase BiLSTM RNN

Accuracy of predicting class of test set is found to be 97.64. The confusion matrix of our model showing false and true positives of the testing set is show in table I.

TABLE I

CONFUSION MATRIX OF THE DESIGNED MEDAL. PLEASE NOTE THAT: STANDING STILL=1, SITTING AND RELAXING=2, LYING DOWN=3, WALKING=4, CLIMBING STAIRS=5, WAIST BENDS FORWARD=6, FRONTAL ELEVATION OF ARMS=7, KNEES BENDING =8, CYCLING=9, JUMPING=10, RUNNING=11, AND JUMP FRONT & BACK=12

	1	2	3	4	5	6	7	8	9	10	11	12
1	5389	0	0	0	0	11	0	0	0	0	0	0
2	0	7350	0	0	0	0	0	0	0	0	0	0
3	0	0	7050	0	0	0	0	0	0	0	0	0
4	2	0	0	4752	32	6	0	4	0	3	1	0
5	4	0	0	95	5572	34	10	120	0	3	0	12
6	1	0	0	2	25	4512	45	64	0	0	0	1
7	3	0	0	0	6	54	5159	28	0	0	0	0
8	7	3	0	4	238	171	150	5872	5	0	0	0
9	0	1	0	0	6	1	4	19	4618	0	0	1
10	0	0	0	2	0	0	0	0	0	5766	73	9
11	0	0	0	2	1	0	0	0	2	317	5369	9
12	0	1	1	12	15	3	2	0	0	57	28	1681

In order to compare the performance of the designed model, We have trained several standard machine learning models to predict the activities in the mhealth dataset, namely: kNN, Decision trees, Naive Bayes and SVM. Their results are shown in tables II through VI.

TABLE II
CONFUSION MATRIX OF KNN CLASSIFICATION METHOD. SEE TABLE I FOR LEGEND

	1	2	3	4	5	6	7	8	9	10	11	12
1	6536	0	0	0	0	15	8	25	0	0	0	0
2	0	6088	0	0	0	0	0	0	0	0	0	0
3	0	0	6090	0	0	0	0	0	0	0	0	0
4	1	0	0	5289	369	8	5	108	17	12	3	16
5	2	0	0	571	4626	236	73	457	94	61	34	66
6	63	1	0	21	210	5050	314	371	10	0	2	1
7	24	2	0	2	18	167	6253	68	4	0	0	0
8	56	0	0	113	498	512	207	4168	241	7	3	6
9	0	1	0	35	113	42	13	189	5664	1	0	2
10	0	0	0	72	37	2	13	11	5	5733	608	92
11	0	0	0	40	17	0	7	3	1	914	4173	79
12	0	0	0	109	145	15	24	42	28	229	120	819

TABLE III
CONFUSION MATRIX OF DECISION TREES METHOD. SEE TABLE I FOR LEGEND

	1	2	3	4	5	6	7	8	9	10	11	12
1	5173	0	0	0	0	16	30	19	0	0	0	0
2	0	6267	0	0	9	0	3	0	0	0	3	2
3	0	0	7018	0	0	0	0	0	0	0	0	0
4	5	0	0	5109	151	6	1	30	0	9	1	17
5	11	5	0	392	6536	143	32	293	57	59	26	93
6	13	11	0	7	83	5522	58	66	0	5	0	3
7	18	0	0	2	18	58	5227	31	0	11	5	10
8	7	3	0	49	182	86	39	5233	24	11	4	17
9	0	0	0	1	14	1	4	15	5518	1	2	4
10	0	2	1	13	22	8	3	30	1	6115	360	106
11	0	2	0	20	30	3	6	1	0	274	5595	113
12	0	2	0	14	42	14	4	22	2	145	63	1708

TABLE IV
CONFUSION MATRIX OF NAIVE BAYES. SEE TABLE I FOR LEGEND

	1	2	3	4	5	6	7	8	9	10	11	12
1	6465	2	0	0	0	87	30	0	0	0	0	0
2	0	6023	0	0	0	0	39	26	0	0	0	0
3	0	0	6059	0	0	0	0	0	0	0	29	2
4	0	0	0	4904	525	88	0	193	0	18	71	29
5	0	0	0	1156	3876	160	13	746	17	116	38	98
6	67	0	0	11	41	5341	489	74	1	0	0	19
7	83	1	0	0	1	62	6374	0	0	0	4	13
8	110	0	0	69	425	1430	208	3494	0	62	0	13
9	0	0	0	0	14	1	0	484	5530	20	3	8
10	0	0	0	8	0	0	0	0	0	4644	1664	257
11	0	0	0	0	0	0	0	0	0	743	4372	119
12	0	0	0	2	0	0	0	0	0	583	325	621

TABLE V
CONFUSION MATRIX OF SUPPORT VECTOR MACHINES. SEE TABLE I FOR LEGEND

	1	2	3	4	5	6	7	8	9	10	11	12
1	6525	0	0	9	0	41	0	9	0	0	0	0
2	0	1458	0	336	0	0	670	0	0	0	1039	2585
3	0	0	6090	0	0	0	0	0	0	0	0	0
4	188	0	0	3674	349	268	300	823	1	75	92	58
5	101	45	0	2411	772	251	61	1278	62	588	379	272
6	148	0	0	245	1	5296	62	259	1	0	0	31
7	652	72	0	114	6	29	4896	153	5	171	227	213
8	302	0	0	552	8	342	47	4126	123	56	48	207
9	0	0	0	3	2	1	0	36	6014	1	3	0
10	0	84	0	35	601	22	174	72	2	3164	1578	841
11	8	56	2	261	560	3	76	21	15	2039	1846	347
12	10	19	3	53	94	50	21	26	5	428	322	500

Table VI summarizes the overall accuracy of each of the methods shown in tables I through VI using the testing set. It is clear that our designed model, which was built upon the BiLSTM RNN architecture, has performed the best at 97.64% out of all other methods in this study. Whereas, SVM scored the worst at 64.67%. The result is a clear indication of the feasibility of this model in classifying human activities based on wearable sensors.

TABLE VI
THE OVERALL ACCURACY OF EACH METHOD IN TABLES I TO VI

	Name method	Accuracy of validation	Accuracy of test
1.	kNN	88.78%	88.18%
2.	Decision Trees	93.54%	94.78%
3.	Naïve Bayes	83.34%	84.12%
4.	SVM	64.12%	64.67%
5.	**PCA-BiLSTM RNN**	**97.32%**	**97.64%**

Finally, table VII shows the individual accuracy of each class in the mhealth dataset when our model is used for prediction.

TABLE VII
ACCURACY OF PREDICTING EVERY CLASS IN MHEALTH DATASET

Physical activities	**Accuracy**
standing still	99.79%
sitting and relaxing	100%
lying down	100%
walking	99.04%
climbing stairs	97.10%
waist bends forward	95.57%
frontal elevation of arms	98.26%
knees bending	92.99%
cycling	99.71%
jumping	98.58%
running	94.19%
jump front & back	93.91%

The lower accuracy can be simply explained by remembering that some of the classes are very similar to each other. For example, the physical postures of climbing stairs and knees bending are to some extent similar because climbing stairs requires one to bend their knees.

IV. Conclusions

This study has presented a deep learning recurrent neural network algorithm for human activity recognition tasks. It consists of two stages. The first is a standard PCA to reduce the number of features and the other is Bilstm RNN. In addition, we have used the mhealth dataset which consists of three 3-axis accelerometer sensors placed on chest, right wrist, and left ankle, two 3-axis gyroscope sensors placed on right wrist and left ankle, two 3-axis magnetometer sensors placed on chest, right wrist, and left ankle. It has twelve physical activities recorded from 10 subjects which are standing still, sitting and relaxing, walking, lying down, climbing stairs, waist bends forward, knees bending, frontal elevation of arms, running, cycling, jogging, and jumping front & back. We have registered an accuracy of 79.64% using our design which is higher than many standard machine learning models in the literature such as KNN, decision trees, and SVM. The performance results of our model prove the feasibility of this study. Further work should include experimentation on large-scale and complex human activities.

References

[1] S. Ghazal and U. S. Khan, “Human posture classification using skeleton information,” 2018 Int. Conf. Comput. Math. Eng. Technol. Inven. Innov. Integr. Socioecon. Dev. iCoMET 2018 - Proc., vol. 2018-Janua, pp. 1–4, 2018.

[2] K. K. Htike, O. O. Khalifa, H. A. M. Ramli, and M. A. M. Abushariah, “Human activity recognition for video surveillance using sequences of postures,” 2014 3rd Int. Conf. e-Technologies Networks Dev. ICeND 2014, pp. 79–82, 2014.

[3] W. C. Hsu et al., “Multiple-wearable-sensor-based gait classification and analysis in patients with neurological disorders,” Sensors (Switzerland), vol. 18, no. 10, 2018.

[4] A. Panagiota, S. Layal, and H. Stefan, “Assessment of human gait speed and energy expenditure using a single triaxial accelerometer,” Proc. - BSN 2012 9th Int. Work. Wearable Implant. Body Sens. Networks, pp. 184–188, 2012.

[5] A. Mannini and A. M. Sabatini, “Machine learning methods for classifying human physical activity from on-body accelerometers,” Sensors, vol. 10, no. 2, pp. 1154–1175, 2010.

[6] A. Jain and V. Kanhangad, “Human Activity Classification in Smartphones Using Accelerometer and Gyroscope Sensors,” IEEE Sens. J., vol. 18, no. 3, pp. 1169–1177, 2018.

[7] Y. C. Huang, C. W. Yi, W. C. Peng, H. C. Lin, and C. Y. Huang, “A study on multiple wearable sensors for activity recognition,” 2017 IEEE Conf. Dependable Secur. Comput., pp. 449–452, 2017.

[8] T. Brezmes, J. L. Gorricho, and J. Cotrina, “Activity recognition from accelerometer data on a mobile phone,” Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 5518 LNCS, no. PART 2, pp. 796–799, 2009.

[9] D. Das and A. Chakrabarty, “Human gait based gender identification system using Hidden Markov Model and Support Vector Machines,” Int. Conf. Comput. Commun. Autom. ICCCA 2015, pp. 268–272, 2015.

[10] A. Ignatov, “Real-time human activity recognition from accelerometer data using Convolutional Neural Networks,” Appl. Soft Comput. J., vol. 62, pp. 915–922, 2018.

[11] Song-Mi Lee, Sang Min Yoon, and Heeryon Cho, “Human activity recognition from accelerometer data using Convolutional Neural Network,” 2017 IEEE Int. Conf. Big Data Smart Comput., pp. 131–134, 2017.

[12] G. Ogbuabor and R. La, “Human activity recognition for healthcare using smartphones,” ACM Int. Conf. Proceeding Ser., pp. 41–46, 2018.

[13] A. Jordao, L. A. B. Torres, and W. R. Schwartz, “Novel approaches to human activity recognition based on accelerometer data,” Signal, Image Video Process., vol. 12, no. 7, pp. 1387–1394, 2018.

[14] M. Panwar et al., “CNN based approach for activity recognition using a wrist-worn accelerometer,” Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. EMBS, no. July, pp. 2438–2441, 2017.

[15] C. A. Ronao and S. B. Cho, “Human activity recognition with smartphone sensors using deep learning neural networks,” Expert Syst. Appl., vol. 59, pp. 235–244, 2016.

[16] F. J. Ordóñez and D. Roggen, “Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition,” Sensors (Switzerland), vol. 16, no. 1, 2016.

[17] C. Vargas, “Face Recognition using Principal Component Analysis, OpenCV and EmguCV,” Int. J. Adv. Res. Comput. Eng. Technol., vol. 1, no. 9, pp. 135–139, 2012.