International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 10, No.01 (Nov-2021)

http://dx.doi.org/10.12785/ijcds/1001107

Development of A Human Posture Recognition System for Surveillance Application

Olayiwola F Arowolo[1], Ezekiel O Arogunjo[1], Daniel G Owolabi[1] and Elisha D Markus[2]

[1]Department of Electrical and Electronic Engineering, University of Ibadan, Ibadan, Oyo State, Nigeria.
[2]Department of Electrical and Electronic and Computer Engineering, Central University of Technology, Free State, South Africa

Received 31 July. 2020, Revised 4 April. 2021, Accepted 10 April. 2021, Published 25 Nov. 2021

Abstract: A method of recognizing human posture in traditional camera images for surveillance application is presented in this paper. Recognition of human posture using a camera has been considered as a cue for modelling human activity in automated surveillance systems. The aim of this study is to analyze the use of joint angles between key body points and machine learning algorithms to classify human posture into three categories; Standing, Sitting, and Lying. Positions of key body points obtained from a deep convolutional neural network were used. The novelty of this approach is in the use of existing traditional cameras without depth sensors. This overcomes the limitations of joint tracking using depth sensors such as Kinect. Distances measured between two key body points, hip and knee, of persons in 2D images were also used for posture recognition. The result shows that 2D information of angles between certain joints can be used to recognize human posture. This approach achieved higher accuracy than simple distance measurement between joints and is computationally efficient. Our approach can be adopted using security cameras and computer hardware already in place.

Keywords: Posture Recognition, Surveillance, Joint Angle, Machine Learning, Traditional Cameras

1. Introduction

Computers are used to process large amounts of data in everyday life with greater ease than manual human effort. However, problems that appear simple to humans, like recognizing an object or obtaining high-level information from a scene can be difficult for a computer to solve. This makes it difficult for computers to interact with the environment autonomously. Computer vision research has developed techniques and technologies for obtaining, processing, and understanding images with application in industrial quality control, autonomous driving, computer-human interaction and security, safety surveillance, and design of intelligent assistive mobility devices [1].

Using video cameras for security surveillance is already a common practice, but existing video camera systems have the limitation of needing a human operator to constantly monitor them. This can lead to mistakes as humans are not naturally well suited to monitoring suspicious activities [2]. Also, in common surveillance cameras, the feeds are hardly monitored; they are only referred to after a crime has been carried out. Intelligent surveillance cameras will go a step further because instead of merely recording footage, they can be used to identify suspicious events requiring attention in real-time and notify people to act quickly [3].

Achieving a truly intelligent surveillance system involves recognizing human behaviour and understanding intents and motives from observation alone. This can be a difficult problem, even for humans, and mistakes are common [4]. Many approaches have been considered for solving this problem over the years. They all involve object detection, classification, tracking of objects in an image sequence, as well as comprehending the target's behaviour or activity in different surveillance scenarios [3].

Motion analysis is how an object's behaviour in a surveillance scenario is determined. Human motion analysis is broad and involves the motion of all body parts. Pose estimation, however, is an aspect of human motion analysis limited to certain body parts e.g., head,

E-mail address: arogunjo.eo@gmail.com, layirowolo@gmail.com, danielgbenga814@gmail.com, emarkus@cut.ac.za

trunk, and limb. It involves the detection of certain key body points. Human body configuration can be determined by pose estimation. Traditionally, recognizing human postures utilizes systems that require markers or sensors to be attached to the body of targets, but this would be of no benefit in a surveillance situation. [5]. The use of a vision-based pose recognition system can be used to detect situations that are of security concern, such as people lying down on the floor of a banking hall.

This study is aimed at posture recognition of a single person in 2D images with controlled backgrounds, using angles between joints and machine learning algorithms for surveillance application. A lot of research focus has been dedicated to the use of depth sensors and 3D images to carry out pose recognition. To the best of our knowledge, no work has been carried out combining angles between key points obtained from a 2D neural network-based pose estimation model and machine learning algorithms for posture recognition.

2. LITERATURE SURVEY

A review of previous research efforts shows that different techniques and tools have been used for human posture recognition.

In [4], the authors proposed a technique of inferring human posture by classification of three-dimensional human shape. This technique determines three-dimensional human body shape using a set of silhouettes, and the shape description is determined by the corresponding visual hull. The 3D shape illustration accounts for variability in body size and proportion while it is also invariant to rotation, translation, and scale. In their experiment, a support vector machine learning algorithm was used to classify twelve postures with a good level of accuracy.[6] approached the problem by using a shape descriptor based on MPEG-7 contours and projection histogram. Different postures were recognized by passing the extracted descriptors through a hierarchy of classifiers. Four major postures (sitting, standing, bending, and laying) were recognized by matching descriptors to their labels.

For the purpose of human-robot interaction, [7] developed a system for posture recognition and for tracking people. Their technique for posture recognition was based on the use of Eigen spaces. Since different human postures could possess characteristics that look similar, an image containing a posture was described with lower dimensionality using principal component analysis (PCA). PCA was used to find the vectors which best represent the human posture image. By using only twelve principal eigenvectors, the system was able to recognize seven postures with an average accuracy of 95%.

The introduction of Microsoft Kinect into the consumer electronics market for Xbox games stimulated research attention into its usage for posture recognition. Kinect had a joint tracking tool and simplified many of the problems researchers face in detecting joint positions. An open-sourced software development kit (SDK) known as open Kinect was developed, allowing researchers to focus on non-gaming applications of Kinect. [8] used the skeleton information provided by Kinect SDK to recognize four different postures (sitting, standing, bending, and lying). Using the skeletal tracking tool, information about 20 joints in the body were captured to form the feature vector that was passed to an SVM classifier.

Using Microsoft Kinect, [9] developed a two-stage fall detection system. The vertical state of a three-dimensional human subject image is first determined through segmentation. In the following stage, a random forest model is applied to features obtained from the first stage to determine if a fall has happened. The system provided practical application of recognizing the ‘standing’ human posture.

In the work of [10], skeleton information from a 2D image was translated to a 3D space by using depth information from a Kinect camera. An interpolating algorithm was also used to prevent self-occlusion. A multi-class support vector machine was used to recognize five variations of standing posture with 98% accuracy.

In [11], a method of estimating three human postures - sitting, standing, and bending - was put forward. The method uses features extracted by fast Fourier transforms (FFT) of 2D body scans from Kinect v2 sensor and a two-layered neural network.

The use of Kinect for posture recognition suffers from limitations such as self-occlusion, bone length variation, and artificial vibration [12]. Kinect’s skeleton tracking tool also has limited range and performs poorly in outdoor conditions.

More recently, the use of neural networks has gained popularity in adoption for the problem of pose recognition in humans due to the results obtained in literature, and this method is available on many publicly available pose recognition datasets. The authors in [13] developed a method that uses deep learning regression architecture for the structured prediction of a 3D human pose from monocular images. By using auto-encoders in combination with traditional convolutional neural network (CNN) to account for joint dependencies, the method achieved good performance and fast computation time.

The work of [14] introduced convolutional pose machines (CPM) that use sequential layers of convolutional neural networks (CNN) to produce refined estimates for the location of body parts.

Our proposed approach overcame some of the limitations of pose recognition using Kinect by combining the use of neural networks for pose estimation and machine learning algorithms for pose recognition with good accuracy. The novelty of this approach is in the use of existing traditional cameras without depth sensors. This overcomes the limitations of joint tracking using depth sensors such as Kinect.

3. METHOD

Our system architecture is as shown in Fig. 1. It consists of image acquisition, image processing, pose estimation, feature extraction, and posture recognition.

A. Image Acquisition

For the data acquisition process, we collected 280 pictures of single persons in different postures while they perform ordinary day-to-day activities such as walking, sitting, dancing, lying on the floor. These pictures were taken with a Samsung phone camera. The pictures were collected in controlled sceneries with diverse backgrounds. Objects in the background sometimes occluded parts of the person in the image. Some of the images collected are shown in Fig. 2

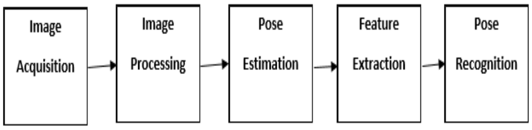

Figure 1. The system architecture

(a) (b)

Figure 2. Sample of images in the dataset (a) standing subject (b) sitting subject

B. Image Preprocessing

The collected pictures were resized to a size of 300 pixels by 400 pixels while maintaining the aspect ratio.

C. Pose Estimation

For this study, we made use of Openpose[15] for 2D pose estimation. Openpose can be considered as the state-of-the-art approach for the estimation of a pose. The implementation of Openpose is open-source. It makes use of Convolutional Neural Networks (CNN) to predict the locations of key points for each person in an image. Its implementation has three stages.

Openpose uses Part Affinity Field (PAF) network architecture. A PAF is a 2D vector describing the orientation of one key point with respect to another.
The first stage of the network uses a ten layered neural network built on a pre-trained VGG-19 network to create a feature map **F** for the input image. The second stage uses a two-branched CNN to produce a set of detection maps **S** and vector fields of part affinity **L**. For the first level, this is given by:

$$S^1 = \rho^1(F) \qquad (1)$$

$$L^1 = \emptyset^1(F) \qquad (2)$$

For the next levels, **S** and **L** are defined as:

$$S^t = \rho^t(F, S^{t-1}, L^{t-1}), \forall t \geq 2 \qquad (3)$$

$$L^t = \emptyset^t(F, S^{t-1}, L^{t-1}), \forall t \geq 2 \qquad (4)$$

The last stage uses greedy parsing to produce 2D key joint positions of persons in the image. When an image is passed through the network, the network returns an output image with the positions of key joints indicated on the human in the image. An example of an output image from the network is shown in Fig. 3.

The Openpose implementation used produces key points on each detected person in the following format: Nose -1, Neck - 2, Shoulder(R) - 3, Elbow (R) - 4, Wrist (R) - 5, Shoulder(L) - 6, Elbow(L) - 7, Wrist(L) - 8, Hip(R) - 9, Knee (R) - 10, Ankle (R)- 11, Hip (L) - 12, Knee (L) - 13, Ankle(L) - 14, Eye (R) - 15, Eye (L) - 16, Ear (R) - 17, Ear (L) - 18, Background - 19.

For each of the output key point, the coordinates of the key point (x_i, y_i) is returned by the network.

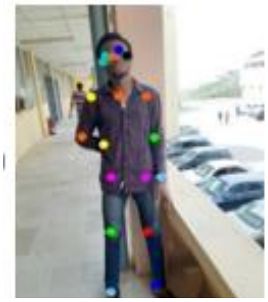

Figure 3. Key joint positions indicated on standing subject

D. Feature Extraction

We draw the skeleton formed by the human in each image by connecting the collected key points. We form 18 joint pairs from these key points. The joint pairs are: Left Elbow-Left Wrist, Nose-Neck, Neck-Left Shoulder, Right Shoulder, Right Elbow, Right Elbow-Right Wrist, Left Shoulder-Left Elbow, Neck-Right Hip, Right Knee-Right Ankle, Right Hip-Right Knee, Neck-Left Hip, Left Knee-Left Ankle, Neck-Nose, Left Hip-Left Knee, Right Eye-Right Ear, Nose-Right Eye, Left Eye-Left Ear, Nose-Left Eye, Left Shoulder-Right Ear, Right Shoulder-Left Ear. An example of the skeleton's form is shown in Fig. 4.

For two key points (x_i, y_i) and (x_j, y_j), the angle between the line formed by joining the two key points and the x-axis is given by:

$$\tan^{-1}\left(\frac{y_{i} - y_{j}}{x_{i} - x_{j}}\right) \qquad (5)$$

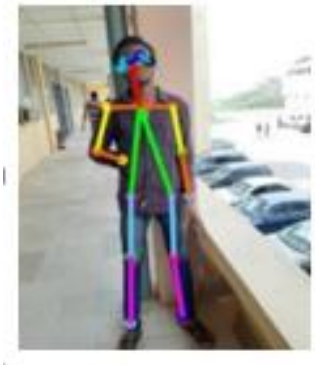

Figure 4. Skeleton drawn on standing subject

The joint pair angles are measured in a clockwise manner from the x-axis are taken as negative, while measurements taken in an anti-clockwise from the x-axis are recorded as positive. Key points without connection to the next corresponding point in the key point pair due to occlusion or absence of such point will return no angles and are left blank in the recording. Analysis was carried out on the angle information collected from the images in the dataset.

E. Pose Estimation

A dataset of extracted features from 280 images was used to train four classifiers used to recognize the three postures of interest. The distribution of the images in the training dataset is shown in Table I.

There is a high proportion of images with standing posture (59.2%) in the dataset compared to sitting (32.8%) and lying posture (8.0%). It is assumed that in a security or safety scenario which our work models, lying posture is an anomaly that rarely occurs and is therefore, of particular interest. Hence, it is important that the machine learning algorithm of choice performs well in detecting the lying class when trained with small samples of the lying class.

From observation, some of the key point pairs play no part in the classification of postures into standing, sitting, or lying. These key point pairs are removed from the dataset. Angle information from 11 key point pairs are selected for use. The 11 key point pairs used are shown in Table II.

We selected four machine learning algorithms for building a model that can predict the target class and made a comparison between their levels of performance:

- Adaboost classifier
- Catboost classifier
- Multi-layer perceptron
- Random forest classifier

TABLE I. DISTRIBUTION OF IMAGES IN DATASET

Posture	Count	% of Total
Standing	166	59.2
Sitting	92	32.8
Lying	22	8.0

TABLE II. SKELETON LINES OF KEY POINT PAIRS USED

S/N	Key Point Pairs
i	Neck-Shoulder(L)
ii	Shoulder(R)- Elbow(R)
iii	Elbow(R) – Wrist(R)
iv	Shoulder(L)-Elbow(L)
v	Elbow(L)- Wrist(L)
vi	Neck-Hip(R)
vii	Hip(R)-Knee(R)
viii	Knee(R)-Ankle(R)
ix	Neck- Hip(L)
x	Hip(L)- Knee(L)
xi	Knee(L)-Ankle(L)

F. Joint Distance Measurement for Pose Recognition

We hypothesized that distance between certain joint pairs could be used to classify the posture of a human in a two-dimensional image. From simple observation of our dataset, the distance between the hip key joint and the knee key joint along the Y-plane appeared most suitable for separating the postures into the three classes. The distance between the right hip key joint (x_{rh}, y_{rh}) and the right knee key joint (x_{rk}, y_{rk}) as well as the distance between left hip key joint (x_{lh}, y_{lh}) and left knee key joint (x_{lk}, y_{lk}) was measured for images in the dataset. This was used to determine the threshold or distance limit for each class of posture. We therefore developed the following conditional statements:

If $(y_{rk} - y_{rh})$ and $(y_{lh} - y_{lk})$ > 30 pixels, then the person is standing.

Else if 10 pixels < $(y_{rk} - y_{rh})$ and $(y_{lh} - y_{lk})$ < 30 pixels, then the person is sitting.
Else person is lying.

We made a comparison between the results obtained when simple distance measurement was used for posture recognition and the results obtained from machine learning algorithms.

4. RESULT AND DISCUSSION

A. Pose Recognition Using Joint Distance

Using a test dataset of 54 images, we obtained results from the experiment using the distance between the hip key point and the knee key point on the Y-plane to classify postures. Some of the correctly classified images are shown in Fig. 5. The result from the test was put in a confusion matrix as shown in the Tab. III

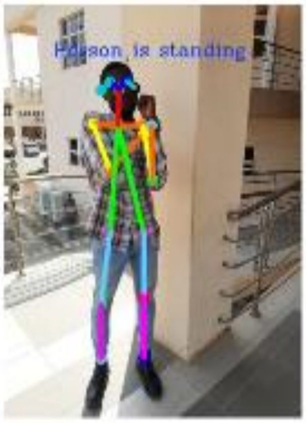

Figure 5.1. Subject correctly recognized as standing.

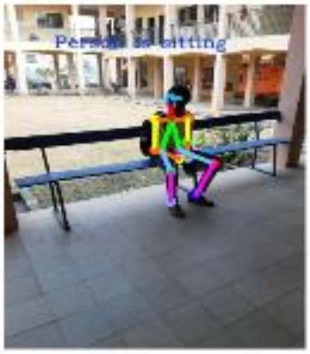

Figure 5.2. Subject correctly recognized as sitting.

TABLE III. CONFUSION MATRIX FOR POSE RECOGNITION USING DISTANCE MEASUREMENT

	Score Standing	Score Sitting	Score Lying
Actual Standing	29	0	0
Actual Sitting	13	8	4
Actual Lying	0	0	2

We evaluated the precision, recall metrics of all the target classes as shown in Tab. IV:

TABLE IV. EVALUATION METRICS FOR POSE RECOGNITION USING JOINT DISTANCE

	Standing	Sitting	Lying
Number of cases	29	25	2
Precision	0.69	1.000	0.33
Recall	1.00	0.32	1.00

From the table above, it can be seen that the use of simple distance measurement only has a fair precision on the standing class, poor recall of the sitting class, and poor precision of the lying class. This can be attributed to varying heights of the persons, camera position, etc. This method achieved an overall accuracy of 69.64%.

B. Pose Recognition Using Machine Learning

We evaluated the performance of the four classifiers used for posture recognition. The results obtained and the performance metrics are shown in Tab. V-XII:

TABLE V. CONFUSION MATRIX FOR ADABOOST CLASSIFIER

	Score Standing	Score Sitting	Score Lying
Actual Standing	17	14	0
Actual Sitting	2	19	0
Actual Lying	0	1	3

TABLE VI. EVALUATION METRICS FOR ADABOOST CLASSIFIER

	Standing	Sitting	Lying
Number of cases	31	21	4
Precision	0.89	0.56	1.00
Recall	0.55	0.90	0.75

TABLE VII. CONFUSION MATRIX FOR CATBOOST CLASSIFIER

	Score Standing	Score Sitting	Score Lying
Actual Standing	29	2	0
Actual Sitting	8	13	0
Actual Lying	0	1	3

TABLE VIII. EVALUATION METRICS FOR CATBOOST CLASSIFIER

	Standing	Sitting	Lying
Number of cases	31	21	4
Precision	0.78	0.81	1.00
Recall	0.61	0.81	0.50

TABLE IX. CONFUSION MATRIX FOR MULTI-LAYER PERCEPTRON CLASSIFIER

	Score Standing	Score Sitting	Score Lying
Actual Standing	19	11	1
Actual Sitting	4	17	0
Actual Lying	1	1	2

TABLE X. EVALUATION METRICS FOR MULTI-LAYER PERCEPTRON CLASSIFIER

	Standing	Sitting	Lying
Number of cases	31	21	4
Precision	0.79	0.59	0.67
Recall	0.61	0.81	0.50

TABLE XI. CONFUSION MATRIX FOR RANDOM FOREST CLASSIFIER

	Score Standing	Score Sitting	Score Lying
Actual Standing	29	2	0
Actual Sitting	5	16	0
Actual Lying	0	1	3

TABLE XII. EVALUATION METRICS FOR RANDOM FOREST CLASSIFIER

	Standing	Sitting	Lying
Number of cases	31	21	4
Precision	0.85	0.84	1.00
Recall	0.84	0.76	0.85

The Random Forest Classifier had the highest accuracy of the four algorithms, with an overall accuracy of 85.71%. The others were: Adaboost – 69.64%, Catboost – 80.35%, Multi-layer Perceptron – 67.86%. When we also considered the recall and precision, especially of the lying class, Random Forest Classifier achieved the best overall performance Comparison of the four classifiers is made using accuracy, recall, and precision metric. The result is shown in the Fig. 6.

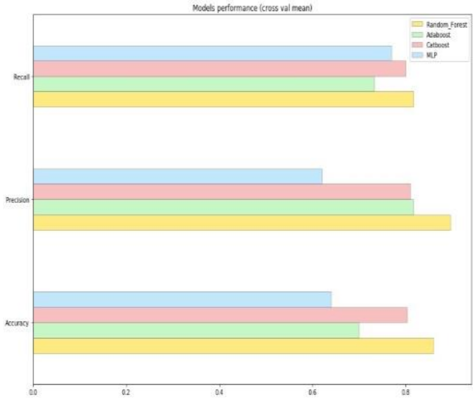

Figure 6. Performance comparison of four classifiers

In terms of computational speed, our approach achieved a processing speed of 0.43 frames per second on a Hewlett-Packard computer with a Core i7 processor maximum clock speed of 2.8Ghz.

C. MODEL ASSESSMENT WITH REFERENCE TO EXISTING WORKS

The evaluation metrics obtained from the two experiments carried out have been presented and compared. The dataset used has an imbalanced number of cases in each posture class, hence accuracy may not be the best measure of model performance. The use of angles between skeletons and machine learning algorithms performed considerably better the use of distance between hip and knee key points in classifying the postures with far fewer misclassified cases. The highest accuracy we recorded in our work was 85.7% using 11 joint skeletal angles to recognize postures. When compared with [8], which achieved an average accuracy of 72.71% using nine skeletal angles without scaling, our approach gave a better result. [10] achieved an average accuracy of 93.2% in recognizing five variations of standing posture using joint angles alone, but like [8], experiments were restricted to an indoor setting where lightning variations would not affect the Kinect camera, a key limitation our approach overcame. We utilized images in both indoor and outdoor settings with varying lighting conditions, underlying the promise of our approach.

5. CONCLUSION

In this study, we have proposed a method of recognizing 3 human postures (standing, sitting, and lying) in 2D images taken with a traditional camera using angle between skeleton lines and the horizontal axis. Skeleton lines were obtained from the key points returned by Openpose pose estimation model. The results obtained show that there is promise in using the angles between these skeletal lines to classify postures. The performance of machine learning algorithms in classifying postures correctly is however dependent on accurate estimation of key points on the body by the pose estimation model. Experimental result showed that the use of joint angles from body skeleton gives a better accuracy than using distance between key joints in the body when recognising posture in 2D images. In the future, we hope to extend this work by using a larger and more variant dataset as well as use posture recognition to model criminal activity.

REFERENCES

[1] E. O. Arogunjo, E. D. Markus and H. Yskandar, "Development of a Holonomic Robotic Wheeled Walker for Persons with Gait Disorder," *2019 Open Innovations (OI)*, Cape Town, South Africa, 2019, pp. 159-164, doi: 10.1109/OI.2019.8908169Phil. Trans. Roy. Soc. London, vol. A247, pp. 529–551, April 1955. (references)

[2] O. Javed and K. Shafique, "Visual Surveillance in Realistic Scenarios," pp. 30–39, 2007.

[3] R. Poppe, "Vision-based human motion analysis : An overview," vol. 108, pp. 4–18, 2007, doi: 10.1016/j.cviu.2006.10.016.

[4] Q. Tian *et al.*, "Human Detection using HOG Features of Head and Shoulder Based on Depth Map," pp. 2223–2230, 2013, doi: 10.4304/jsw.8.9.2223-2230

[5] I. Cohen and H. Li, "Inference of Human Postures by Classification of 3D Human Body Shape Institute for Robotics and Intelligent Systems Integrated Media Systems Center University of Southern California," 2003.

[6] L. Goldmann, M. Karaman, and T. Sikora, "Human Body Posture Recognition Using MPEG-7 Descriptors," vol. 5308, pp. 177–188, 2004

[7] A. Chella, H. Dindo, and I. Infantino, "People Tracking and Posture Recognition for Human-Robot Interaction."

[8] T. Le, "Human posture recognition using human skeleton provided by Kinect," no. January 2013, 2014, doi: 10.1109/ComManTel.2013.6482417.

[9] E. E. Stone and M. Skubic, "Fall Detection in Homes of Older Adults Using the Microsoft Kinect," no. c, 2014, doi: 10.1109/JBHI.2014.2312180.

[10] B. Cao, S. Bi, J. Zheng and D. Yang, "Human Posture Recognition Using Skeleton and Depth Information," WRC Symposium on Advanced Robotics and Automation (WRC SARA), 275-280, 2018.

[11] S. Maryam *et al., "*A novel human posture estimation using single depth image from Kinect v2 sensor" Annual IEEE International Systems Conference (SysCon), 2018.

[12] Y. Kwok-Yan *et al*., "Improved Skeleton Tacking by Duplex Kinects: A Practical Approach for Realtime Applications." Journal of Computing and Information Science in Engineering. October, 2016.

[13] B. Tekin *et al*., " Structured Prediction of 3D Human Pose with Deep Neural Networks" arXiv: 1605.05180 [cs.CV] , 2016

[14] S. Wei *et al*., "Convolutional Pose Machines." Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, 4724-4732, 2016.

[15] Z. Cao, T. Simon, S-E. Wei, and Y. Sheikh, "Real time Multiperson Pose Estimation Using Part Affinity Fields," arXiv: 1161.08050v2, 2014

Ezekiel O. Arogunjo is currently a Lecturer in the Department of Electrical and Electronic Engineering at the University of Ibadan, Oyo State, Nigeria. His
research interests are Adptive Control, Robotics, Artificial Intelligence, Biomedical Engineering, Network Design Optimization. Email: arogunjo.eo@gmail.com

Olayiwola F. Arowolo is a graduate of Electrical and Electronic Engineering from the University of Ibadan, Nigeria. He is currently a Research Assistant at Robotics and Artificial Intelligence Nigeria (RAIN). His research interests are Computer Vision, Artificial intelligence and Sustainable Energy Systems Design.

Daniel G. Owolabi is a graduate of Electrical and Electronic Engineering from the University of Ibadan, Nigeria. His research interests are primarily in the areas of Computer Vision, Artificial intelligence and Embedded Systems Design.

Elisha D. Markus (Dr.) is currently a Senior Lecturer in the Electrical, Electronic and Computer Engineering Department at the Central University of Technology, Free State, South Africa. His research interests are Non-linear control, Robotics, Power Systems, Differential Flatness, Telecommunication and Artificial Intelligence.