# Classification of Body Position during Prayer using the Convolutional Neural Network

Dr. M. Nalini, Professor, Department of Electronics and Instrumentation Engineering, Sri Sairam Engineering College, Chennai, India,Email: nalini.ei@sairam.edu.in

T.J.Nagalakshmi , Associate Professor , ECE , Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, Chennai. Email: t.j.nagalakshmi@gmail.com

**Abstract:** A Muslim must perform Salat (prayer) five times a day as the most fundamental and important form of religious devotion, as it is the second pillar of Islam. EEG recordings of brain activity during a Namaz can be used to study the effects of rapid changes in body position and a 14-channel EEG recorder monitors the brain activity of 40 Muslim participants during a four-cycle Namaz. Different Namaz positions were used to measure brain connectivity in several frequency bands. An artificial intelligence-assisted framework to assist worshippers in assessing the accuracy of their prayer postures is one solution to these problems. Using Convolutional Neural Networks to recognise basic Islamic prayer movements is the first step in achieving this goal. A YOLOv3 neural network was trained on a dataset of Salat positions to recognise the gestures in this paper. According to the experimental results, for a training dataset of 764 photos.

**Keywords:**Body position; Namaz; convolutional neural network; cross correlation images

**1. INTRODUCTION:** To better understand human activity recognition, sensors [1–3], computers [4–5], and deep learning [5–6] have all been used. Data collected by sensors is used to classify a person's activities. It has been possible to use human activity recognition advancements in a wide range of fields, such as healthcare, sporting activities (such as detecting aggression), elderly monitoring, and posture identification among others. The recognition of Salat is a major issue for Muslims around the world, and this study focuses on this application of human activity. All Muslims perform Salat, Islam's second pillar and most essential act of worship, five times a day. On the other hand, it is an orderly sequence of postures that must be performed in accordance with the instructions of the Prophet Muhammad, peace be upon him. Due to the fact that beginners and children may not be able to perform the poses correctly, this effort is put forth. Many factors contribute to this, including a lack of knowledge about prayer movements or a lack of attention. While the Quran is being read and invocations are being made, each position in Salat must be held for a sufficient amount of time. In the literature, Salat's activity recognition has only been addressed in a few studies. When it came to detecting Salat activities, the researchers in [1,2] used smartphone technology. Salat's activity was examined by the authors of [3,7] using electromyographic (EMG) data. Wearable sensors-based applications such as sports, healthcare, and well-being can benefit from the use of deep learning algorithms to recognise human activity [5,6,8]. Deep learning methods have never been used before to monitor Islamic prayer activity, to our knowledge. Stances of Qiam (Ruku), bowing, prostration and sitting (Sujud) are identified using the most advanced Convolutional neural networks (CNN) algorithms (Julus). Self-driving cars and other autonomous vehicles have also been aided by the use of CNN in a variety of applications [9,10]. The main objective of this project is to develop an AI-based tool for assessing Islamic prayer and an assistance system to assist beginners and children in correcting their Salat postures. Step one in that direction is taken here. It should be highlighted that in our neural network model, we only evaluate the recognition of right postures and neglect incorrect postures, which will be addressed in a future extension of this study that would address anomalies during prayers. Human behaviour in the context of Islamic prayer is the focus of this study, which has created a dataset of four classes for each of the Salat postures. We put the trained network to the test on films of people praying, and it accurately identified all of the postures in the vast majority of cases.

The rest of the paper is arranged in this manner: It is discussed in Section II how to recognise human activity and how to monitor Islamic prayer through sensing activities are related. The YOLOv3 algorithm is briefly mentioned in Section III. Section IV includes Salat's various positions and the accompanying datasets. Scattered throughout Section V are discussions of the findings of the experiments. Section VI brings the paper to a close and discusses future developments in the final paragraphs.

## II. RELATED WORKS

Research published in 2009 by El-Hoseiny et al. is the first to address the task of automatically recognising the gestures of prayer.[13]. The side view of the prayer was captured with the help of a cameraman. A morphological procedure extracted the polygon representing the prayer's main body outline from the original image. Backbone angle and four important human body points are determined by using polygon data. At an angle of y-y with the axis of rotation of the identified body, this angle is known as backbone axis. From the polygon, four primary points are determined:

the polygon's centre point, ankle point, head and back points. The backbone axis angle and the coordinates of the key points can be used to generate a series of inequations that can be used to calculate prayer postures and movements. There are no machine learning classifiers used in this approach; everything is done by hand. Only in this project was the task completed using a standard camera sensor. Others used accelerometers and Kinect sensors as well. The algorithms that were put to the test were correct 95% of the time or more. Accelerometer data was used by Eskaf et al.[14] to develop a daily activity framework (sitting, standing, walking, etc.). It was then possible to combine these behaviours using supervised machine learning classifiers to recognise prayer, as demonstrated here. According to Ali et al.[15], a smartphone's triaxial accelerometer data can be used to detect and track a person's prayer posture automatically. Analysis of group prayer actions was also done using dynamic temporal warping techniques.

In [16], the authors proposed a deep learning model for low-power devices as a method for recognising human activities. The acknowledgement serves as an important foundation for a healthy lifestyle. Mobile accelerometer data was studied by Alobaid et al.[1] for the purpose of identifying prayer activities. Three feature extraction approaches and eight machine learning classifiers were compared for their overall performance. For this assignment, they discovered that Random Forest had a 90% accuracy rate and was the best method. In order to remove the ambiguity between two similar stages of prayer, they developed a two-level classifier that improved accuracy to 93%. Their research also included an investigation into human variables such as height and age. Kinect RGB-Depth cameras were used by Jaafer et al.[17] to capture images. The Kinect Software Development Kit is used to collect skeleton data after two Kinect sensors are placed in a fixed location on the body. In order to learn the skeleton's prayer movements, they used a machine learning classifier called the Hidden Markov Model.

For all practicality and ease of use, only one of the works listed above utilised a standard camera sensor. For this study, machine learning was not considered because it was only concerned with geometric properties. Recognizing prayer positions with a video camera will be much easier thanks to recent developments in machine learning, particularly deep learning algorithms. Because of this, we focused on this flaw in current state-of-the-art techniques. YOLOv3[18], a one-stage deep learning algorithm for object detection, was selected as one of the most effective and used in the recognition of prayer postures here. During our research, we considered that the camera could be placed in a variety of positions rather than being fixed in one position.

## III. ALGORITHMS BACKGROUND

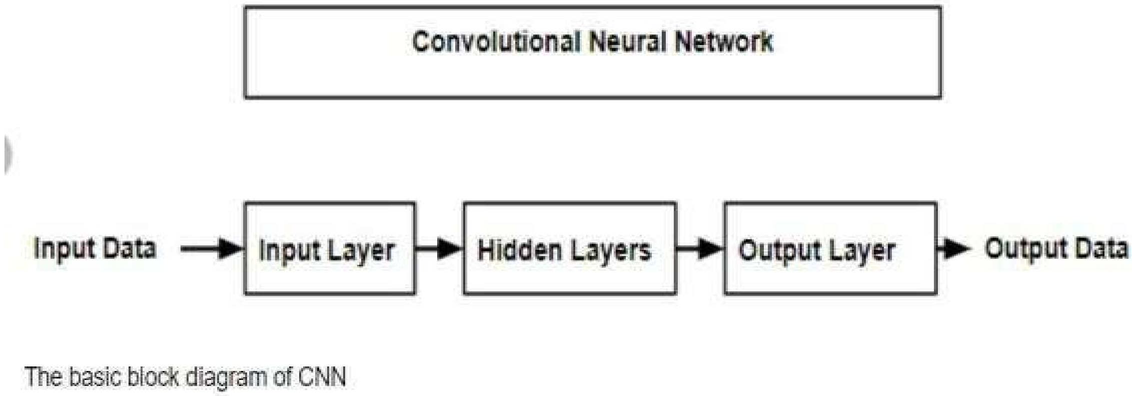


**Figure 1 Block Diagram**

YOLOv3[18] is the most enticing of the deep learning algorithms used in computer vision. This is one of the two supporting facts for my conclusion. YOLOv3[18] has already been proven to outperform other object detection algorithms[9] in comparison. Second, it is quick to reach conclusions (up to 45 frames per second). As a result, the various prayer positions can be identified in real time. These sections detail the YOLOv1[19] architecture, as well as the many improvements that were made in YOLOv2[20] and YOLOv3[18], respectively.

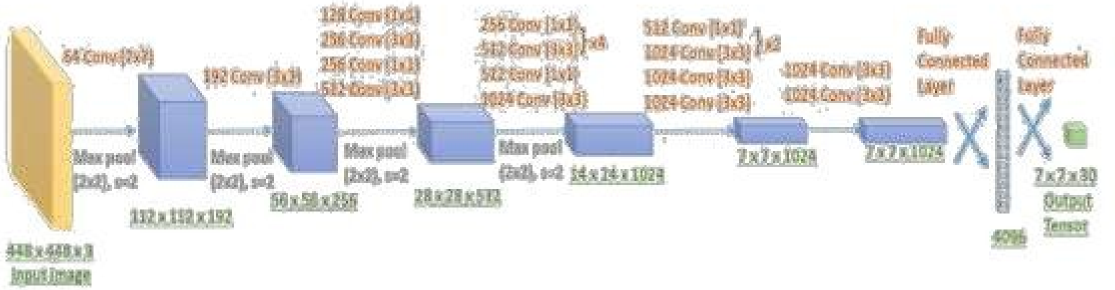


Figure 1. *YOLO v1 Architecture*

**A. YOLOv1**

A new approach to solving object detection problems was first introduced in 2016. For both localization and classification, a single CNN must be trained simultaneously. Two fully linked layers and 24 convolutional layers make up YOLOv1's feature extraction architecture. As depicted in Figure 1, the overall architecture is shown An S * S grid is created from the input image. Grid cells can only be linked to one object at a time. The grid cell for this item also has a set number B of border boxes. Each bounding box is assigned a confidence level. A vector of class probabilities is generated for each grid cell based on the C classes we're interested in. YOLO additionally calculates a vector containing 5 parameters for each bounding box of the cell:

(x, y, w, h, box_confidence_score). For each image, the YOLO network generates an output tensor with the following structure:

where:

- S × S: The number of grid cells corresponds to this value.
- B: The number of bounding boxes corresponds to this value.
- C: The number of targeted classes corresponds to this value.

The YOLO network was constructed by combining three different loss functions. Second, there is a loss of classification . There is also a lack of localization in the app

$$\begin{aligned}
Loss = {} & \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{obj} [(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2] \\
& + \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{obj} [(\sqrt{\omega_i} - \sqrt{\hat{\omega}_i})^2 + (\sqrt{h_i} - \sqrt{\hat{h}_i})^2 \\
& + \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{obj} (C_i - \hat{C}_i)^2 \\
& + \lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{noobj} (C_i - \hat{C}_i)^2 \\
& + \sum_{i=0}^{S^2} \mathbb{1}_{i}^{obj} \sum_{c \in classes} (p_i(c) - \hat{p}_i(c))^2
\end{aligned}$$

where:

.A decrease in self-assurance is another factor. YOLO uses the sum-squared error metric to measure the discrepancy between the expected and actual values. Equation 2 shows the loss expression in detail.

• $\lambda_{coord}$ is the reduction in the predicted weight of the bounding box coordinates. During practise, keep it at 5.

- $\lambda_{noobj}$ is the weight and set to 0.5 for the duration of the training.
- $1l^{obj}$ indicates the present of the material in cell i
- $1^{lobj}$ Indicating that the prediction is based on the index j bounding box
- $x_i$ x is the actual value of x, and xi is the predicted value .
- C indicates how confident you are in the results of your analysis
- $p_i(c)$ Probability that cell I will be classified as belonging to the class C

In the beginning, YOLO was faster than any other object detection algorithm. The mAP (mean average precision) of this algorithm was on par with or better than that of other leading-edge methods.

YOLOv2[20] has undergone a number of changes to improve its accuracy and speed of processing. We can find the following among them:

- The application of batch normalisation (BN). Training loss convergence was improved by using this method in 2015. To improve the mAP in YOLO, BN was introduced to all convolutional layers.
- Replacing the input image size of 224 224 with 448 448 results in a 4% increase in mAP.
- The convolution of anchor boxes. It is now possible to predict the class at the boundary box level rather than at the grid cell level. While recall increased from 81 percent to 88 percent, this resulted in a 0.3% increase in mAP. Improved object recognition in addition to reducing false negatives.
- Analyzing the training set to build an anchor box using K-means clustering. The IoU distance replaces the Euclidian distance in clustering (Intersection Over Union).
- The projections are created using the anchor offsets. In YOLOv2, (x, y, C) is predicted rather than (x, y, C) (tx, ty, tw, th, tC). Convergence is improved as a result of this.
- Fine-grained characteristics can be applied. Low-resolution and high-resolution features are combined in YOLOv2 like ResNet's identity mapping to improve the detection of small objects. This causes mAP to rise by one percent.
- Scales of instruction are employed in training. The image size isn't set in stone with YOLOv2, which instead uses a random algorithm every ten batches. This improves the capacity to anticipate accurately across a wide range of input image sizes.

### C. YOLOv3

The YOLOv3[18] was released in April 2018 as an incremental upgrade to prior versions. Among the enhancements performed, we can mention the following:

- The application of a multi-label classification system. When deciding whether or not an object belongs to one of the previously defined labels, YOLOv3 uses a logistic classifier instead of the mutually exclusive labelling that was used in previous versions.
- A new method of determining the size of the bounding box. It is linked to the best-fitting bounding box anchor during training with YOLOv3's objectness score 1. To make matters worse, if the IoU (Intersection Over Union) is less than a predetermined threshold, it is ignored (0.7 in the implementation). Each ground truth object has its own anchor.
- The implementation of darknet-53. 53 layers and skip connections are used in the same way as ResNet [21]. Both 3* 3 and 1* 1 convolutions are used in this process. Accuracy was top-notch, but it was both faster and less computationally intensive than previous methods.

## IV. DATASET

Videos of people in prayer found online were combined with images and videos taken by laboratory members using their mobile phones to create the Salat Postures Dataset, which includes various Salat positions. People who use smartphones to record videos of themselves should have that footage immediately examined for flaws and suggestions for improvement. This is our ideal scenario. Because of this, most of the photos taken were taken using cell phone data, with a few more wide-angle shots thrown in for good measure. For this reason, we've included photos that show the entire body of a person praying, as it is necessary for us to be able to identify and classify each individual.

Photographs were taken and sorted into four categories by using rectangular bounding boxes, which were manually labelled by researchers. The four classes are based on the four most common Muslim prayer stances:

- Qiyam: Standing up from a seated or bowed position, as in prayer or reading the Quran, is a common practise (Ruku). One to three minutes is typical. A guy in Qiyam stance is depicted in Figure 2's subfigure 5.
- Ruku: It refers to a stance taken after making invocations using Qiam. It usually lasts only a few seconds. Figure 2's sub-figures 1 and 3 depict an example of a guy in Ruku stance.
- Sujud: It is a form of prostration that occurs after rising from a bow in order to make an invocation. It lasts for a few seconds. A man in Sujud stance is depicted in Figure 2's sub-figure 4.

• Julus: It is the position in which one sits after finishing Sujud in order to make an invocation and close Salat. Figure 2's subfigures 2 and 6 depict a man in the Julus pose.
Table I and Figure 2 show the number of images and instances in the training and testing datasets. 90% of the data were used for training and 10% were used for testing.
Performance of object detection networks was evaluated using these metrics:

• IoU: This technique is used to determine how much of a gap exists between the expected bounding box and what is actually there.

• mAP: The mean average precision is represented by the area under the precision vs. recall curve. mAP was calculated based on data from multiple IoU measurements (0.5, 0.6, 0.7, 0.8 and 0.9).

• On test images, the inference time (in milliseconds per image) is measured.

• TP (True Positives): This is the total number of objects that were correctly identified and categorised.

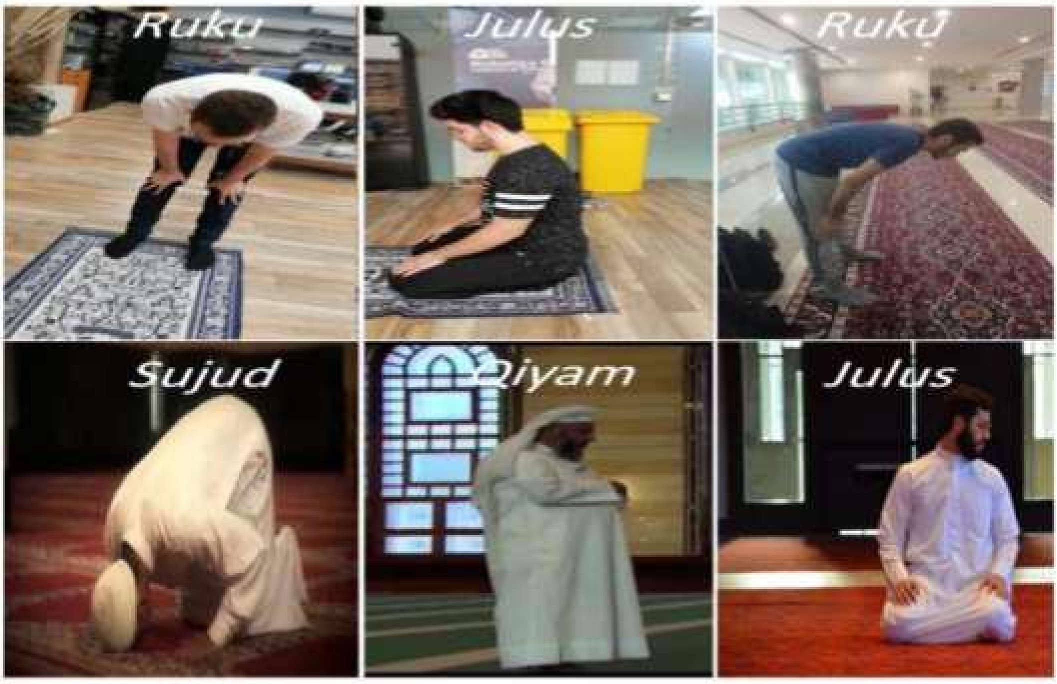
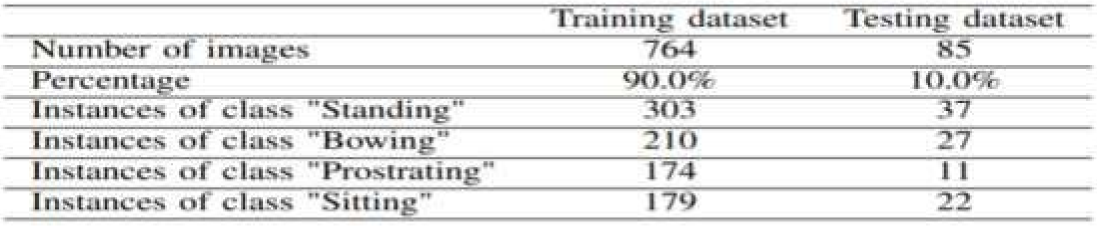


Figure 2. Sample images of our dataset, showing the different prayer positions. The images on the top were captured specifically for this study, while the images on the bottom were collected from the Internet.

Table I
NUMBER OF IMAGES AND INSTANCES IN THE TRAINING AND TESTING DATASETS.



| | Training dataset | Testing dataset |
|---|---|---|
| Number of images | 764 | 85 |
| Percentage | 90.0% | 10.0% |
| Instances of class "Standing" | 303 | 37 |
| Instances of class "Bowing" | 210 | 27 |
| Instances of class "Prostrating" | 174 | 11 |
| Instances of class "Sitting" | 179 | 22 |

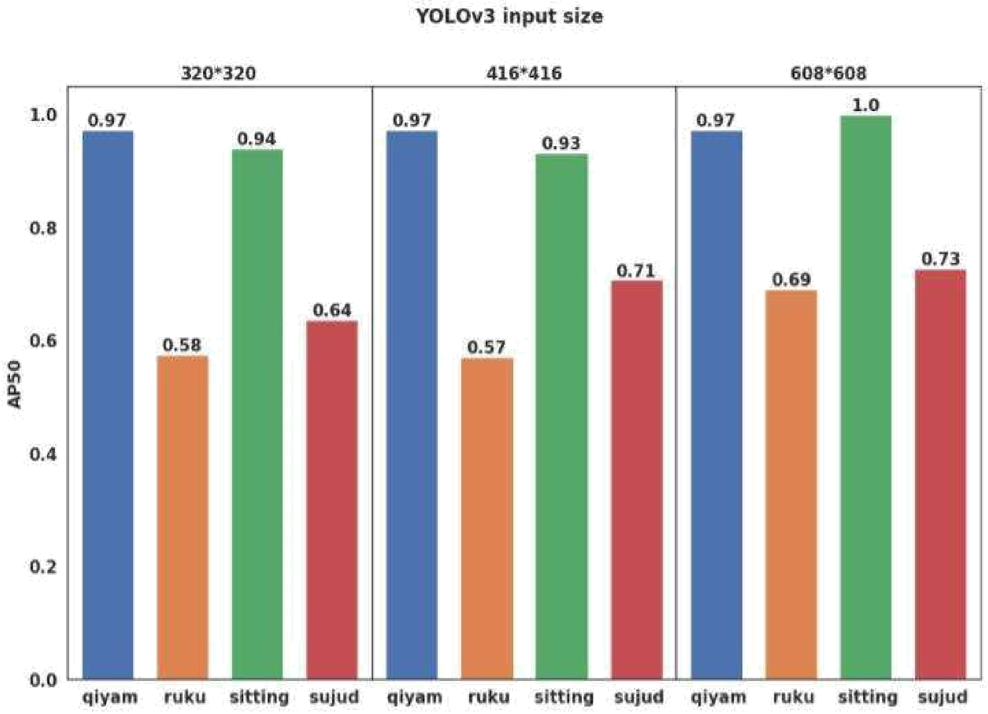


Figure 3. Average Precision (AP) at an IoU threshold of 0.5, for each input size and each class.

- FP (False Positives): a large number of items have been found, but their classifications have been incorrect.
- FN (False Negatives): undetected occurrences.

For each size and class of input, the AP50 value can be seen in Figure 3. All three networks perform significantly worse than qiyam and sitting in detecting Ruku and Sujud classes. Their confusion may be due to a visual resemblance between the two classes.

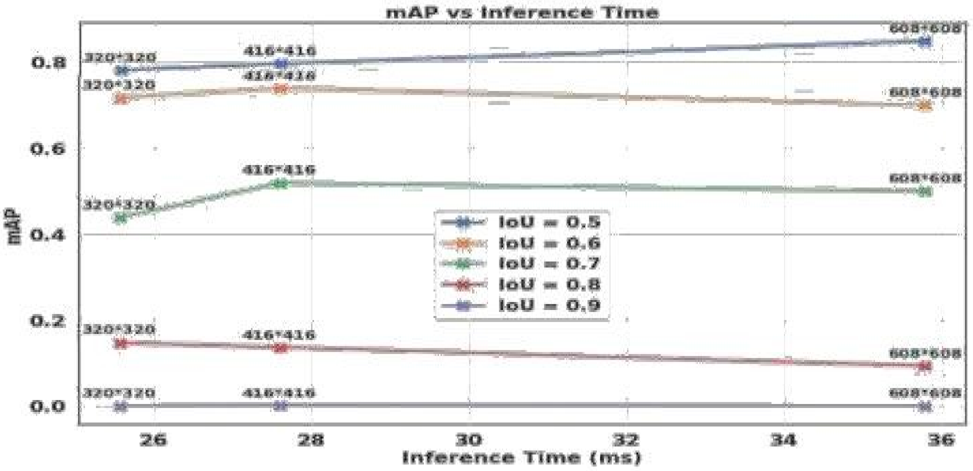


Figure 4. Comparison of the trade-off between mAP and inference time for the three different input sizes of YOLO v3, and for different values of IoU threshold ranging from 0.5 to 0.9.

For the three input sizes and different IoU threshold values ranging from 0.5 to 0.9, the mAP and inference time trade-offs are shown in Figure 4. At 416*416, the IoU threshold increases by 8 percent, which results in an increase in the mAP50 by 2 percent. The mAP has little or no effect on IoU thresholds above a certain point. Using the average IoU per input size presented in Figure 5, bigger input sizes have no effect on the accuracy of the bounding boxes. MAP90 (at IoU=0.9) indicates that YOLOv3 has difficulty aligning the bounding boxes precisely with an object, as observed in YOLO's original paper[18]. An increasing operations are required to achieve an output when the input size is increased in a network. As a result, inference takes longer.

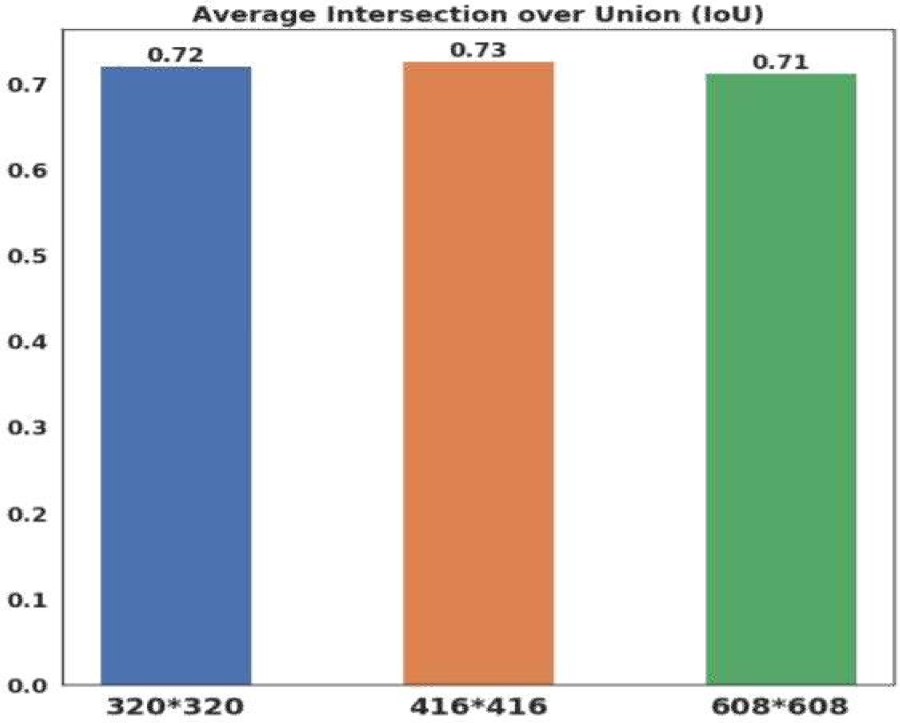


Figure 5. Average intersection over Union (IoU) for the three input sizes.

Using an IoU threshold of 0.5 and a 320x320 input size, Figure 6 shows the number of true positives (TP), false positives (FP), and false negatives (FN) (FN). In comparison to the other two classes, the ruku and sujud have the highest percentage of false negatives. As a result, we should broaden the training dataset to include more images.

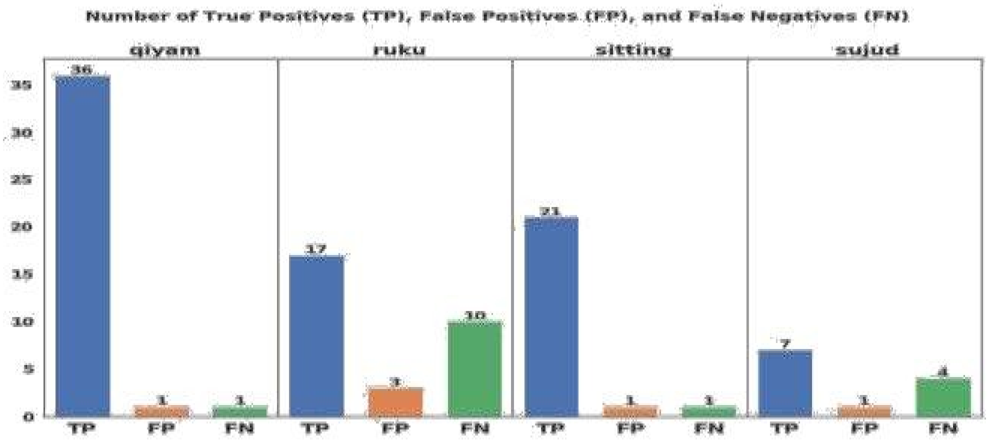


Figure 6. Number of true positives (TP), false positives (FP), and false negatives (FN) for an IoU threshold of 0.5, and an input size of 320x320.

## VI. CONCLUSION

In this paper, we present the results of a study that evaluated how well a computer could detect Islamic prayer postures. The YOLOv3 network was trained using a dataset that was specifically gathered for this purpose (in various settings). The various network configurations were assessed based on a number of metrics. On a training set of 764 images, the mAP ranges from 78 percent to 85 percent depending on the network input size (e.g., 320x320 or 608x608). Initial steps include expanding the training dataset, exploring other network architectures, optimising hyper parameters, and assisting Muslim worshippers in analysing their postures during prayer by using an artificial intelligence assistive framework that includes the object detection model.

## REFERENCES

O. Alobaid and K. Rasheed, "Prayer activity recognition using an accelerometer sensor," in Proceedings on the International Conference on Artificial Intelligence (ICAI), pp. 271–277, The Steering Committee of The World Congress in Computer Science, Computer . . . , 2018.

R. Al-Ghannam and H. Al-Dossari, "Prayer activity monitoring and recognition using acceleration features with mobile phone," Arabian Journal for Science and Engineering, vol. 41, no. 12, pp. 4967–4979, 2016.

M. F. Rabbi, N. Wahidah Arshad, K. H. Ghazali, R. Abdul Karim, M. Z. Ibrahim, and T. Sikandar, "Emg activity of leg muscles with knee pain during islamic prayer (salat)," in 2019 IEEE 15th International Colloquium on Signal Processing Its Applications (CSPA), pp. 213–216, March 2019.

M. Ramzan, A. Abid, H. U. Khan, S. M. Awan, A. Ismail, M. Ahmed, M. Ilyas, and A. Mahmood, "A review on state-of-the-art violence detection techniques," IEEE Access, vol. 7, pp. 107560–107575, 2019.

D. Ravì, C. Wong, B. Lo, and G. Yang, "A deep learning approach to on-node sensor data analytics for mobile or wearable devices," IEEE Journal of Biomedical and Health Informatics, vol. 21, pp. 56–64, Jan 2017.

W. Xu, Z. Miao, J. Yu, and Q. Ji, "Deep reinforcement learning for weak human activity localization," IEEE Transactions on Image Processing, pp. 1–1, 2019.

F. Ibrahim and S. A. Ahmad, "Assessment of upper body muscle activity during salat and stretching exercise: A pilot study," in Proceedings of 2012 IEEE-EMBS International Conference on Biomedical and Health Informatics, pp. 412–415, Jan 2012.

A. Gumaei, M. M. Hassan, A. Alelaiwi, and H. Alsalman, "A hybrid deep learning model for human activity recognition using multimodal body sensing data," IEEE Access, vol. 7, pp. 99152–99160, 2019.

B. Benjdira, T. Khursheed, A. Koubaa, A. Ammar, and K. Ouni, "Car Detection using Unmanned Aerial Vehicles: Comparison between Faster R-CNN and YOLOv3," in 2019 1st International Conference on Unmanned Vehicle Systems-Oman (UVS), pp. 1–6, IEEE, 2019.

A. Ammar, A. Koubaa, M. Ahmed, and A. Saad, "Aerial Images Processing for Car Detection using Convolutional Neural Networks: Comparison between Faster R-CNN and YoloV3," arXiv pre-print 1910.07234, October 2019.

B. Benjdira, Y. Bazi, A. Koubaa, and K. Ouni, "Unsupervised Domain Adaptation Using Generative Adversarial Networks for Semantic Segmentation of Aerial Images," Remote Sensing, vol. 11, no. 11, 2019.

B. Schoettle and M. Sivak, "A survey of public opinion about autonomous and self-driving vehicles in the us, the uk, and australia," tech. rep., University of Michigan, Ann Arbor, Transportation Research Institute, 2014.

M. H. El-Hoseiny and E. Shaban, "Muslim prayer actions recognition," 2009 Second International Conference on Computer and Electrical Engineering, vol. 1, pp. 460–465, 2009.

K. Eskaf, W. M. Aly, and A. Aly, "Aggregated activity recognition using smart devices," in 2016 3rd International Conference on Soft Computing & Machine Intelligence (ISCMI), pp. 214–218, IEEE, 2016.

M. Ali, M. Shafi, U. Farooq, et al., "Salat activity recognition using smartphone triaxial accelerometer," in 2018 5th International MultiTopic ICT Conference (IMTIC), pp. 1–7, IEEE, 2018.

D. Ravi, C. Wong, B. Lo, and G. Yang, "Deep learning for human activity recognition: A resource efficient implementation on low-power devices," in 2016 IEEE 13th International Conference on Wearable and Implantable Body Sensor Networks (BSN), pp. 71–76, June 2016

N. A. Jaafar, N. A. Ismail, and Y. A. Yusoff, "An investigation of motion tracking for solat movement with dual sensor approach," 2015.

J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," CoRR, vol. abs/1804.02767, 2018.

J. Redmon, S. K. Divvala, R. B. Girshick, and A. Farhadi, “You Only Look Once: Unified, Real-Time Object Detection,” in 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016, pp. 779–788, 2016.

J. Redmon and A. Farhadi, “YOLO9000: better, faster, stronger,” in 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017, pp. 6517–6525, 2017.

K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” Arxiv.Org, 2015.

Faisal Aqlan, Abdulaziz Ahmed, Wen Cao, and Mohammad T. Khasawneh, "An ergonomic study of body motions during Muslim prayer using digital human modelling " , International Journal of Industrial and Systems Engineering 2017 25:3, 279-296