Comparitive Study On Various Technique Used For Gait Analysis

Afroz Mohammed, Ayush Garg, Dvijay Shanbagh
College of Engineering - Northeastern University
EECE 5552: Assistive Technology

Abstract

In this paper we intend to detect gait abnormalities for human motion. In the past, research has been conducted on various algorithms leveraging data from different techniques such as Motion Capture systems, IMUs and alike to detect gait abnormalities. Our main objective here is to do a comparative study of gait analysis using Inertial sensors, Optical sensor(camera) and Convolution Neural Networks. Primarily, we are using computer vision tools to obtain the Gait parameters from a video and analyzing it to obtain a Gait Score that can detect abnormality. Further, we used inertial sensors to determine the accuracy of the data acquisition process. Lastly, we have trained a Neural Network model to classify between a normal and abnormal gait.

1. Introduction

As the population is aging and moving towards a lazy generation, posture problems and Musculoskeletal disorders on the rise. Around 3 to 4 percent of people in United States have scoliosis, which results in bending of the backbone, where 64% of these cases are seen in adolescents and can be treated if detected at an early stage. There are various ways to detect Musculoskeletal disorders, most of which includes visiting a physician, which is expensive and tedious and also this way our goal to detect disorders at an early stage might not be accomplished.

Hence, we wanted to make a system which can be used by a layman, sitting at their home to check if he/she/they might have a disorder or not. Keeping this in mind we have worked on various approaches to make this possible.

To solve this problem, firstly, we worked on vision based model, where we extract joint angles of a walking human from a video file using a deep learning framework and use these fetched angles to predict if the person in scope2 has normal or abnormal gait. This approach follows the structure defined by Gillette Gait Score benchmark which checks the efficiency of gait of a human. Gillette has defined 16 parameters which define the gait score which is used by professional to study human gait. The difference is that our approach is completely vision based which makes the technology accessible to a layman.The only fault in choosing a vision based approach is that the angles fetched depends on factors like distance of camera from human, angle of camera placement and also height at which camera is kept. We tried to remove these constraints by introducing constraint in the way humans can move, which being perpendicular to the normal of camera lens and also the distance between the camera and human must be around 4m. Also to ensure the accuracy of angles fetched using the deep learning framework, we used a sensor based approach. We used two IMUs to extract the angles between shank and thigh, which were time based. These angles fetched were compared with the angles fetched using vision method and the results were consistent. The process of extracting the angles using IMUs was made possible by converging the angles to fit.

Secondly, we approached a vision based deep learning framework, where we read the whole of a video and attempt to predict if the Gait of that particular human is normal or abnormal. We used a dataset comprising of 30 videos out of which 15 were of normal humans and 15 of abnormal human gait. To ensure the privacy of the humans the dataset provided was silhouette and hence we couldn't use the first approach. We carried out a hefty amount of data augmentation to ensure that every time the input to model is different from the previous seen and this also removed the dependence of camera position and angle. The model gave an 78% of accuracy of test dataset.

Further we tried implementing our approach on a 5 link biped. We aim to fetch the angles from a human video and mimic those angles on a five link biped. This approach is still under process, but once done this would ease the development of assistive technologies for lower-body. Figure 1 represents a five link biped model.

2. Methods

2.1. Vision based Joint Angle measurement for Gait Analysis

The scoring criteria we selected for Gait Efficiency was Gillette Gait Score [1]. Our objective was to analyze how

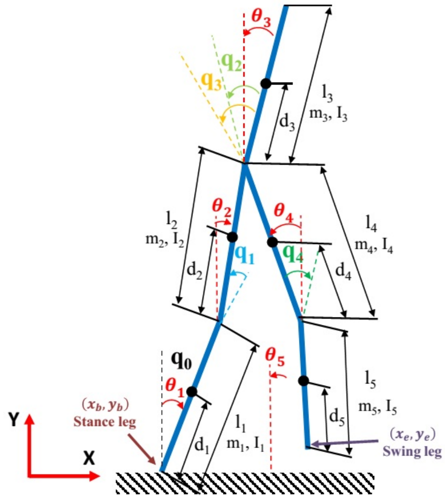

Figure 1. A five-link biped model.

efficient a gait is, for which computing a score seems the best option. Once we have this score, it can have a large number of use cases such as scoring robot gaits. Gillette Score is one of the benchmark score used by medical professionals to study human gaits. It is effectively scoring a gait based on its deviation from a set of ground truth values. The major challenge in computing a score, is to first identify the parameters which manipulate a gait the most. The Gillette Score techniques has found 16 parameters which are paramount to computing the gait score, recommended by medical professionals. Out of these 16 parameters, we were able to find 9 due to the limitations of using a single camera. The parameters used are as follows

- Peak Abduction Angle
- Time of Toe-off in a single Gait as a percentage
- Cadence in Steps per Second
- Minimum Hip Flexion Angle
- Range of Hip Flexion
- Knee Flexion at Initial Contact
- Range of Knee Flexion
- Peak Dorsiflexion during Stance Phase
- Peak Dorsiflexion during Swing Phase

To calculate these parameters, we first used OpenPose[2] to get the exact Joint data in the video. OpenPose is a state-of-the-art pose estimation Deep Neural Network which outputs the Joint location of a human in every image. After we have the joint positions, we use simple geometrical formulas to

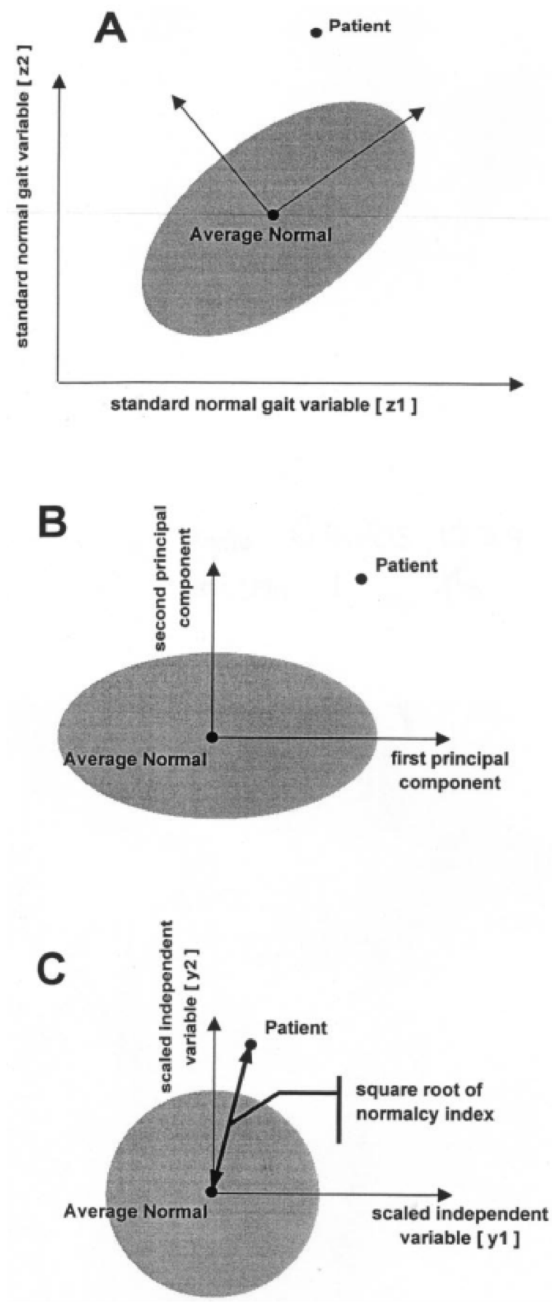

Figure 2. A hypothetical 2-variable demonstration of the method used for Gait Score calculation. (A) 2 Gait variables shown with mean and standard deviation as the ellipse. A specific patient is represented by a combination of the 2 variables. (B) The Principal Components (eigenvectors) are now the co-ordinate axes aligned with the ellipse. The specific patient is projected to the new basis. (C) The scaled uncorrelated variables are shown. Gait score is the Euclidean distance from the origin of the basis.

find these parameters. The output of OpenPose can be seen in Figure 1. After getting the Gait Variables, the following algorithm is used

1. Let μ_j and σ_j be the mean and standard deviation of x_j measured on M normal subjects.

2. Let z_j be the set of standardized gait variables (zero mean, unit S.D.) defined by,

3. Calculate the Covariance Matrix, C_ij, for the N standardized gait variables.

4. Calculate the N eigenvectors from C_ij which are effectively the Principal Components of the data.

5. Let a subject's gait be represented by the same N discrete variables $\tilde{x_j}$, j = 1, N.

6. Scale $\tilde{x_j}$ by the average normal gait variables as follows:
$$\tilde{z_j} = (\tilde{x_j} - \mu_j)/\sigma_j, j = 1, N \quad (1)$$
where ${}_j$ and σ_j are from the normal population.

7. Change basis for $\tilde{z_j}$ to the normal population's principal components.

8. Find the square root of Euclidean Distance of transformed $\tilde{z_j}$
$$d = \sqrt{\sum_{j=1}^{N} y_j^2} \quad (2)$$
This will give us the Gait Score.

2.2. IMU-Based joint angle measurement for Gait Analysis

This section is concerned with joint angle calculation based on inertial measurement data in the context of human motion analysis. For walking motion on a flat surface, flexion/extension angle (joint angle between the thigh and shank) is calculated using the algorithm described below. In a similar way, other joint angles of human motion could be obtained and used as metrics for gait analysis. The method used for joint angle calculation is dependent upon the identification of joint axis and its position within the coordinate systems of individual IMUs.
As presented by authors Seel et al [3], the joint angle measurement using the method described here has a good accuracy, i.e. root mean square error of the knee flexion/extension angles are found to be less than 3 in comparison to results obtained using an optical 3D motion capture system. Therefore, results from our gait trials is provided and used as a landmark for validating the joint angle obtained from videos using Deep Pose, open-source tool.

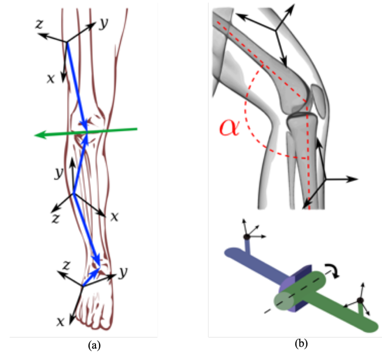

Figure 3. IMU mounting position on joint segments.

2.2.1 Inertial Measurement Units

Inertial sensors, also known as inertial measurement units (IMUs), measure acceleration, angular velocity and the magnetic vector in their own three- dimensional coordinate system. It has been demonstrated in many publications that inertial measurement data can be used to calculate hinge joint angles when at least one IMU is attached to each side of the joint. The hinge joint angle is calculated by integrating the difference of both angular rates along the joint axis. Since, the measurements from IMU contains noise which introduces drift into the calculated angles, it can be removed using additional information from acceleration and/or the magnetometers.

2.2.2 Mounting position of IMUs

The position of the IMUs on the adjacent segments is such that its z-axis is pointing outward from the segment. Figure 3 shows the placement of inertial sensors on the human body, the definition of joint angle and a model of a hinge joint. (a) the coordinates of the joint axes direction (green arrows) and the joint position (blue arrows) in the local coordinate systems of the sensors characterize the sensor-to-segment mounting. (b) definition of the joint angle, α.

2.2.3 IMU-based Knee angle Estimation

The flexion/extension angle of the knee joint is the angle between the upper and lower leg along the main axis of relative motion. Integrating the difference of the upper and lower sensor's angular rates around that axis will yield a drifting flexion/extension angle. The knee axis is approximated to be a pure hinge joint. Further, its kinematic constraint is exploited to obtain the complete orientation of each IMU with respect to the global reference frame. From the IMUs, we get, accelerations $(a_1(t),\ a_2(t)) \in R^3$ and angular rates $(g_1(t),\ g_2(t)) \in R^3$, at some sample period Δt. Additionally, we calculated $\dot{g_1}(t),\ \dot{g_2}(t) \in R^3$ of angular rates using third order approximation:

$$\dot{g_{1/2}}(t) \approx (g_{1/2}(t-2\Delta t) - 8g_{1/2}(t-\Delta t)+ \\ 8g_{1/2}(t + \Delta t)\ g_{1/2}(t + 2\Delta t))/12\Delta t \tag{3}$$

2.2.4 Identification of the Joint Axis Coordinate

Identification data is collected for a walking motion, with the IMUs is mounted on the segments as described before. About every one-hundredth of a second, a dataset, $S(i)$, of the from:

$$\{a_1(t_i),\ a_2(t_i),\ g_1(t_i),\ g_2(t_i),\ g_1\dot{(t_i)},\ g_2\dot{(t_i)}\} \tag{4}$$

is recorded. The dataset, $S(i), i \in [1, N]$, are used to identify the unit-length direction vectors, j_1, j_2, of the knee flexion/extension axis in the local coordinates of both sensors. In spherical coordinate,

$$\begin{aligned} j_1 &= (cos(\phi_1)\,cos(\theta_1),\ cos(\phi_1)\,sin(\theta_1),\ sin(\phi_1))^T \\ j_2 &= (cos(\phi_2)\,cos(\theta_2),\ cos(\phi_2)\,sin(\theta_2),\ sin(\phi_2))^T \end{aligned} \tag{5}$$

The projection of angular rate $(g_1(t),\ g_2(t))$ into the joint plane have the same lengths for each instant in time, which is equivalent to:

$$||g_1(t)\ \times\ j_1||_2 - ||g_2(t)\ \times\ j_2||_2\ =\ 0\ \forall t \tag{6}$$

Therefore, every dataset S(i), i [1, N] must fulfill the above equation. Hence, j_1 and j_2 are identified by minimizing the left-hand side of the equation using Gauss-Newton algorithm,

$$\beta^{(s+1)} = \beta^s\ -\ \left(J_f^T\ J_f\right)^{-1}\ J_f^T\ r\left(\beta^s\right) \tag{7}$$

2.2.5 Identification of the Joint Position Coordinates

As the acceleration of the joint center must be the same in both local frames, considering vectors $O_1, O_2 \in R^3$ as the origin of the first and second frame from the joint axis, the following equation is minimized using Gauss-Newton method to obtain $O_1, O_2 \in R^3$.

$$\begin{aligned} &||a_1(t) - \Gamma_{g_1(t)}(o_1)||_2 - ||a_2(t) - \Gamma_{g_2(t)}(o_2)||_2\ =\ 0\ \forall t \\ &\Gamma_{g_i(t)}(o_i) := g_i(t) \times (g_i(t) \times o_i) + \dot{g_i}(t) \times o_i,\ i = 1, 2 \end{aligned} \tag{8}$$

Since the result of that optimization, denoted by $\hat{o}_1$, $\hat{o}_2$, refers to an arbitrary point along the joint axis, we shifted it as close to the sensors by applying,

$$o_1\ =\ \widehat{o_1} - j_1\frac{\hat{o}_1.j_1\ +\ \hat{o}_2.j_2}{2},\ o_2\ =\ \widehat{o_2} - j_2\frac{\hat{o}_1.j_1\ +\ \hat{o}_2.j_2}{2} \tag{9}$$

2.2.6 Calculation of the Flexion/Extension Angle

The joint axis coordinates j_I, j_2 and the local joint position coordinates o_1, o_2 using the method described above. Further, gyroscope-based flexion/extension angle is calculated by integrating the difference of the angular rates around the joint axis , i.e.,

$$\alpha_{gyr}(t)\ =\ \int_0^t (g_1(t)\cdot\ j_1\ \ g_2(t)\cdot\ j_2)\ dt \tag{10}$$

Accelerometer-based flexion/extension angle is calculated by shifting the measured accelerations onto the joint axis by applying:

$$\tilde{a}_1(t) = a_1(t)\Gamma_{g_1(t)}(o_1),\ \tilde{a}_2(t) = a_2(t)\Gamma_{g_2(t)}(o_2) \tag{11}$$

The flexion/extension angle can be approximated by the angle between the projections of $\widetilde{a_1}(t)$ and $\widetilde{a_2}(t)$ into the joint plane. Defining a pair of joint plane axes $x_{1/2},\ y_{1/2}$ for each local frame:

$$x_1\ = j_1\ \times c,\ y_1\ = j_1\ \times x_1,\ x_2\ = j_2\ \times c,\ y_2\ = j_2\ \times x_2 \tag{12}$$

The accelerometer-based joint angle is obtained by calculating the signed angle between the two vectors in the quation below.

$$\alpha_{acc}(t)\ =\ \left(\begin{bmatrix} \tilde{a}_1(t).x_1 \\ \tilde{a}_1(t).y_1 \end{bmatrix}, \begin{bmatrix} \tilde{a}_2(t).x_1 \\ \tilde{a}_2(t).y_1 \end{bmatrix}\right) \tag{13}$$

The gyroscope-based angle exhibits drift, while accelerometer-based angle does not but it is affected by the accelerometer noise. Therefore, both the angles are combined using a complementary filter as given below,

$$\alpha_{acc+gyr}(t)\ =\ \lambda\alpha_{acc}(t) + (1-\lambda)(\alpha_{acc+gyr}(t-\Delta t)+ \\ \alpha_{gyr}(t) + \alpha_{gyr}(t-\Delta t)),\ \lambda \in [0, 1] \tag{14}$$

2.3. Gait abnormality detection using Convolution Neural Network

In this section we aim to define a deep learning model to detect if a particular human has normal or abnormal gait. For this the dataset used is ”Gait-A dataset”, which contains side view silhouette videos of 30 people walking along the hall. Our main aim is to create a system that can be used by a layman, and this required heavy amount of data augmentation. Also the data augmentation helped in not over-fitting the model. An example of single silhouette frame of a video is shown below in Figure 4.

2.3.1 Network Architecture

Table 1 below shows the network architecture. We employed 3 dimensional CNN to capture the temporal features

Figure 4. Original Image

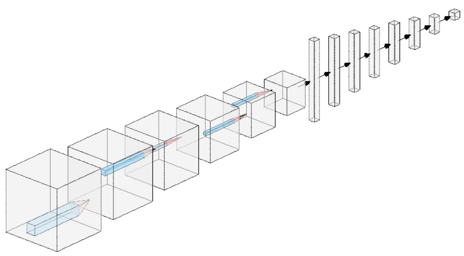

Figure 5. Model Structure.

of the videos. One block of convolution includes 2 3D CNN followed by concatenation and pooling. The concatenation is done to capture the learning of previous layers and learn new features. Also concatenation helps in easy flow of gradient.

The input to the system is a video with frame size 512x512 and 384 number of frames. The main aim is to reduce the dimension of the input as soon as possible and to capture all temporal and spatial features to ensure accurate model. After each convolution block the image dimensions are pooled down. After the last pool layer the features are completely flattened and given as input to fully connected layers. The number of neurons in the fully connected layers keep on reducing and finally produce a two bit output. Here after each convolution and fully connected layer we have applied RELU activation function. Only the output nodes are left without any activation/nonlinear functions. Figure 5 shows the visual structure of model implementation.

2.3.2 Training Parameters

The network weights were initialized with He et al initialization technique. We haven't used batch normalization or dropout because our data augmentation was pretty heavy and dropout or batch normalization would have been redundant. We used ADAM optimizer with a constant learning rate of 0.0001.

We used cross entropy as loss function for our model. Though using Mean Square Error would have worked fine,

MODEL STRUCTURE	
NAME	NODE Description
Input	
3D conv1	Convolution 3x3x3
3D conv2	Convolution 3x3x3
Concat 1	Concatenation of 3D conv2 and Input
Pool1	Maxpool 2x2x2
3D conv3	Convolution 3x3x3
3D conv4	Convolution 3x3x3
Concat 2	Concatenation of 3D conv4 and Pool1
Pool2	Maxpool 2x2x2
3D conv5	Convolution 3x3x3
3D conv6	Convolution 3x3x3
Concat 3	Concatenation of 3D conv6 and Pool2
Pool3	Maxpool 2x2x2
3D conv7	Convolution 3x3x3
3D conv8	Convolution 3x3x3
Concat 4	Concatenation of 3D conv8 and Pool3
Pool4	Maxpool 2x2x2
3D conv9	Convolution 3x3x3
3D conv10	Convolution 3x3x3
Concat 5	Concatenation of 3D conv10 and Pool4
Pool5	Maxpool 2x2x2
FLATTEN1	Flatten Pool5 layer
Fully con1	Fully Connected layer 1x8000
Fully con2	Fully Connected layer 1x3000
Fully con3	Fully Connected layer 1x956
Fully con4	Fully Connected layer 1x500
Fully con5	Fully Connected layer 1x260
Fully con6	Fully Connected layer 1x120
Fully con7	Fully Connected layer 1x50
Fully con8	Fully Connected layer 1x15
Output	Output of size 1x2

Table 1. Model Network.

but we preferred a probabilistic approach.

2.3.3 Data augmentation

Our main aim is to make a model which can be used by a layman and therefore we had to ensure all type of possible data into the system while training. The images in Figure 6 show all the techniques used.

To remove the dependence on the height angle and movement of the camera we padded each frame with a random number of rows and column and also rotated the images.

2.3.4 Training

The training was done on GPU's provided by google colab. The learning rate was initially kept constant and then gradually reduced with every epoch. This reduction in learning

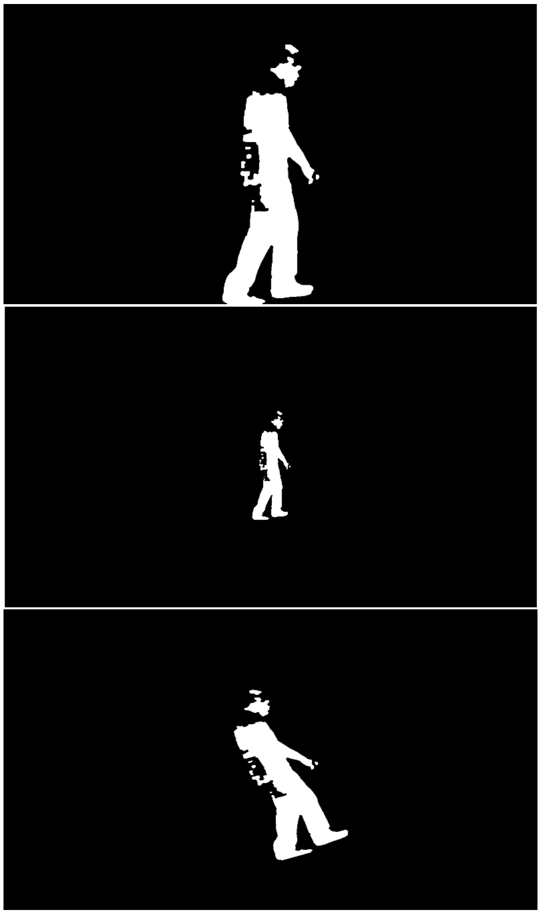

Figure 6. Data Augmentations (randomly paded and randomly tilted.

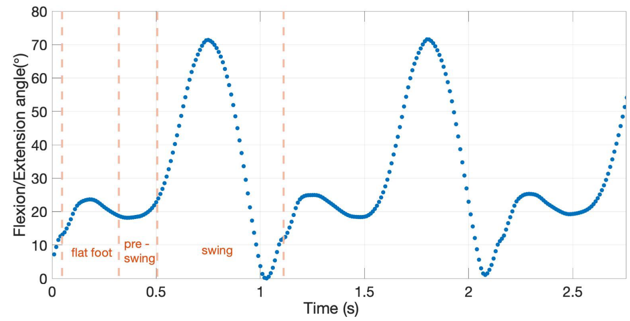

Figure 7. Flexion/Extension angle.

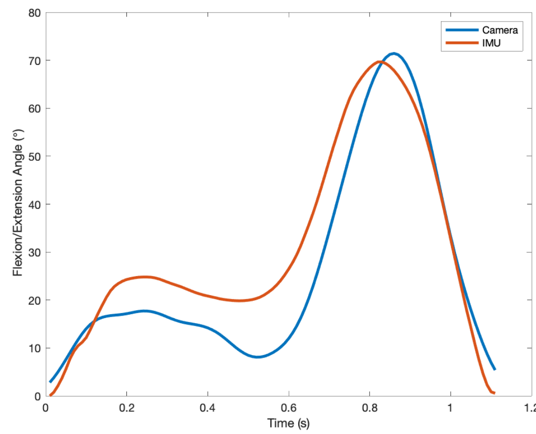

Figure 8. Overlapping plots of joint angles obtained using IMUs and Camera.

rate was done to ensure that the model fits.

The data augmentation along with huge input size and float initialization makes the model take a lot of time in training which also sometimes results in RAM overload or OOM error. To stop that from happening the whole dataset was divided into different parts using h5py library. This resulted in loading less number of variables at a time in the RAM. This also made scope for input pipeline, which helped in speeding up the training process. So, while one batch was used for training the other batch was randomly fetched and the stored when the system is ready to process.

3. Results

3.1. Vision based Joint Angle measurement

We used the CASIA Gait Dataset B as the ground truth data, but we didn't use the complete dataset. We first trained our model on 20 different subjects and then tested on 2 subjects - one normal outside of the dataset and another with a walking disability. The data extracted from videos using OpenPose can be visualized in Figure 9. The Probability distribution graphs of all the variables can be visualized in Figure 10. For a normal person tested, our Gait Score was **7.9** and for the person with a walking abnormality, we got a score of **17.7**. The farther the score is from zero mean value, the greater is the abnormality.

3.2. IMU-Based joint angle measurement

The flexion/extension angle calculated from the experiment is shown in Figure 7. Absolute joint angles for hip, knee and ankle joints were derived from videos using Deep Pose, open-source tool. Figure 8, shows the angle trajectory for these joints for both legs. Further, for one case, knee joint angle obtained from videos is compared with joint angles obtained using IMU as shown in Figure 8. The joint angles followed similar trajectory in both the methods used and the root mean square difference between these were below 8° for five trials.

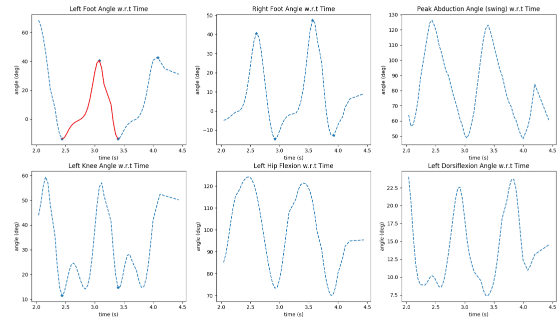

Figure 9. Joint angles extracted from videos.

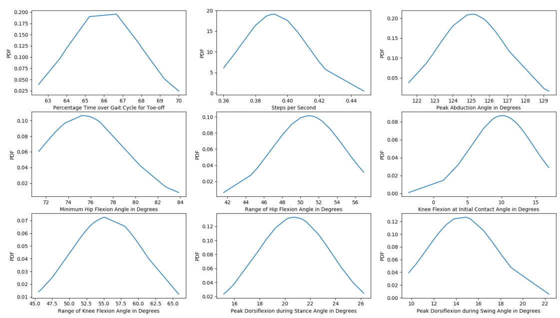

Figure 10. Probability distribution curves.

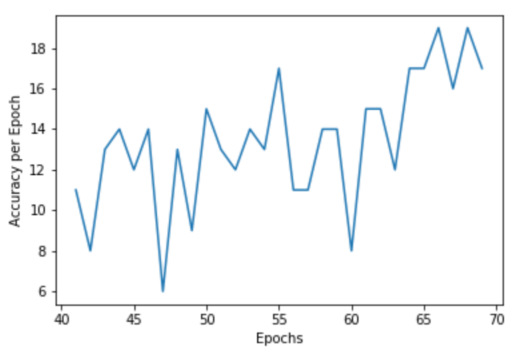

Figure 11. Average accuracy per epoch.

3.3. Gait abnormality detection using Convolution Neural Network

The model was successfully trained to provide an average accuracy of 78% in 70 epochs. As the training of the model demanded more time and resources, in each epoch we couldn't increase the accuracy any further. Figure 4 shows the plot for average accuracy and loss of the model during epochs after the model started to converge.

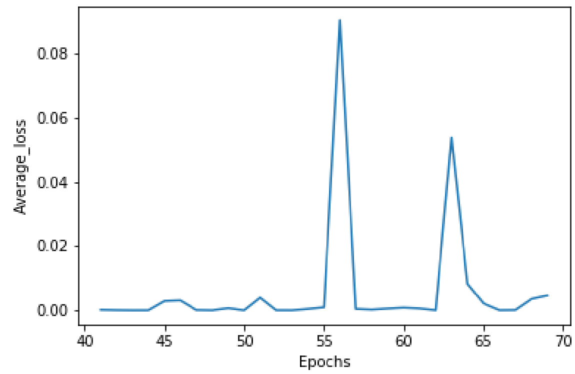

Figure 12. Average loss per epoch.

4. Conclusion

Three different approaches to Gait Analysis is discussed here, namely, IMU-based, vision-based and CNN-based. In Vision-based approach a Gait score criteria is used to define the Gait of a human motion. Further, the parameters used in this approach is validated using Inertial sensors, where similar trajectory of joint angles were obtained in both methods with the root mean square difference of less than 8°. Finally, in the CNN-based approach, an accuracy of 78 % is achieved to detect gait abnormalities.

References

[1] Schutte LM, Narayanan U, Stout JL, Selber P, Gage JR, Schwartz MH. An index for quantifying deviations from normal gait. Gait Posture. 2000;11: 25-31.

[2] Zhe Cao, Tomas Simon, Shih-En Wei, Yaser Sheikh; Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields;The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 7291-7299

[3] Seel T, Raisch J, Schauer T. IMU-based joint angle measurement for gait analysis. Sensors (Basel). 2014;14: 6891-6909.

[4] Seel T, Schauer T, Raisch J. Joint axis and position estimation from inertial measurement data by exploiting kinematic constraints. 2012 IEEE International Conference on Control Applications, 2012:45-49.