Fast Algorithm to Measure the Types of Foot Postures with Anthropometric Tests Using Image Processing

Husneni Mukhtar[#], Dien Rahmawati, Desri Kristina Silalahi, Ledya Novamizanti, Muhammad Rayhan Ghifari, Ahmad Alfi Adz Dzikri, Faris Fadhlur Rachman, Ahmad Akbar Khatami
Faculty of Electrical Engineering of Telkom University
Jl. Telekomunikasi, Terusan Buah Batu, Bandung, 40257, Indonesia
[#]husnenimukhtar@telkomuniversity.ac.id, dienrahmawati@ telkomuniversity.ac.id, desrikristina@ telkomuniversity.ac.id.

Article Info

Article History:
Received Feb 2, 2020
Revised Feb 10, 2020
Accepted Feb 20, 2020

Keywords:
Foot posture
Pronation
RFA
MLA
AHI
Image processing

Abstract

There are two types of tools for measuring the foot posture, uniplanar (anthropometric and radiographic types) and multiplanar tools (such as Foot Posture Index-6 and -8). The process of the foot posture measurement with both tools performed by a doctor was commonly carried out by using manual equipment such as ruler, arc, goniometer, marker and applying the observation skill by eyes. It needs time to measure for each foot. For research needs, a large number of samples have to be provided by a doctor to analyze data statistically which consumes much more time and exhaustion from the workload in the measurement process. Hence, the aim of this study is to significantly decrease the measurement time and minimizing human error by developing software of anthropometric measurements of foot posture based on digital image processing (DIP). The anthropometric tests used in this study consist of Rear Foot Angle (RFA), Medial Length Arc Angle (MLAA) and Arch Height Index (AHI). Instead of using equipment with a series of measurements to determine the foot posture, the DIP system only needs two pictures of the foot as the input of the system. The methods involved in the image processing are performed by a series of digital image processing, started from pre-image processing, noise filter, Sobel edge detection, feature extraction, calculation, and classification. The result of the image processing is able to determine the foot posture types for all tests based on the values of angle and length of the foot variables. The error measurements of length and angle are 6.22 % and (0.26-1.74) %, respectively. This study has demonstrated the development algorithm in MATLAB to measure the foot posture, which is named *Anthro-Posture v1.0* software. This software offers an efficient alternative way in measuring and classifying the foot posture in a shorter time and minimizing the human error in the measurement process. In the future, this study can be improved to be used by doctors in obtaining large amounts of data for research needed

Corresponding Author:
Husneni Mukhtar
Faculty of Electrical Engineering of Telkom University
Jl. Telekomunikasi, Terusan Buah Batu,
Bandung, 40257, Indonesia
Email: husnenimukhtar@telkomuniversity.ac.id

I. INTRODUCTION

The assessment of foot posture should play a notable role to observe its correlation with other fields, such as finding out the effect of foot posture in the gait analysis [1] where a distinct of foot pressure might be resulted by different foot posture, in data diagnosis on leg muscle activity and leg physical therapy [2], [3], in foot kinematic as walking [4] and in musculoskeletal examination in clinical practice and research [5], [6] for example the foot alignment examination as a clinical assessment of patients with pain and lower-limb injury [7]. However, the assessment of the foot alignment often meets difficulties for doctors and researchers because the results have concern about the validity, reliability and usefulness of all measures due to a variety of confounding factors [8].

Nevertheless, a static foot assessment in foot posture is generally carried out to categorize the foot type based on its anatomical characteristics. In these three decades, there were many foot posture classification tools developed which were divided into uniplanar and multiplanar tools. Examples of multiplanar tools are Foot Posture Index (FPI)-6 and FPI-8 that combine sagittal, frontal and transversal assessments of the feet [9], [10]. Meanwhile, there were many uniplanar tools that have also been developed and employed. The most frequently used tools were the anthropometric and radiographic types. Rearfoot angle (RFA), Medial Longitudinal Arch Angle (MLAA), navicular drop (ND), footprint (arch index) and malleolar valgus index are included in anthropometric measurements [8], [11].

Research on foot pronation has been carried out by previous researchers such as Langley et al. [9], Bailey et al. [12], and James et al. [13]. In general, their research was conducted by marking the tibia and calcaneus lines of the subject's feet using markers, then measuring the angle between the midline of the tibia and calcaneus using a goniometer. Another pronation foot study was done by Lin et al. [14][15] using image processing to determine the angle between the tibia and calcaneus midline, but this study did not directly classify the results of the angular measurements to determine the type of foot pronation.

For doctors or clinicians, the anthropometric measurements were widely performed by direct measurement [7], [13], [16], [17] such as evaluating the curvature and harmony of the foot [8] using a goniometer to measure RFA [9] or using a scanner device and other technologies [5], [15], [18] such as RFID sensors and data visualization with computer modeling [12]. It is rare to find studies that use image processing in measuring either angle or length of the measured foot variables. Furthermore, for the necessity of large medical data, utilization of those devices, such as goniometer and arc will spend more time or need more people to perform the measurement. Therefore, the purpose of this study is to provide and develop a fast algorithm for measuring and classifying the foot posture types based on anthropometric tests, namely RFA, MLAA and AHI using a set of DIP technique.

II. Materials and Methods

A. Experimental Setup

This study used ten participants with information on age, body weight, height, and body mass index (19.3 ± 1.76 years, 57.3 ± 10.39 kg, 166.8 ± 7.64 m, 20.50 ± 2.78 kg/m^2). All photographs of the feet of participants were taken in a standard studio setting.

1) Materials and Tool

A camera, Canon 200D with a 5.76 focus and a 25mm lens, was used to take the images of the participant’s standing foot of both sides of the rear and inner foot for each leg. During the process of taking photographs, the participant stood on a platform block. An adequate lighting system was provided to acquire a good contrast image between the object and the background (described in Fig. 1).

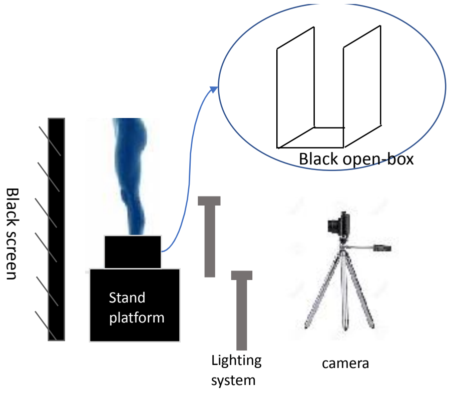


Fig. 1. Illustration of the system arrangement of taking pictures/ images.

2) Experiment

In this study, after the photo session was completed, the digital images were processed using a developed graphical user interface (GUI) based on *Matlab* software. In this tool, each image was processed and measured using techniques of RFA, MLAA, and AHI. Then, the classification of the foot type was resulted for each technique.

B. The Diagram Block

The technique of digital image processing was performed by several steps in this study as shown in Fig. 2. The acquired image was previously processed by converting the image format and reducing the noise in order to obtain a better extraction result of the feature.

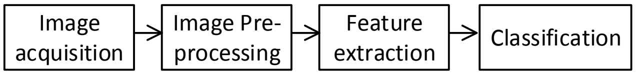


Fig. 2. Block diagram of Digital Image Processing of anthropometric tests.

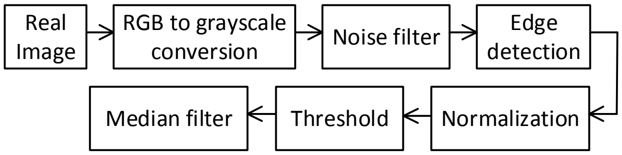


Fig. 3. Block diagram of Image Pre-processing.

C. Image Pre-processing

Image pre-processing is aimed to improve the image data by suppressing unwanted distortions such as color transformation, filtering, segmentation and scaling in order to prepare the image to be used in the next process. This process (Fig. 3) was carried out in six sequences.

i. Converting the RGB to grayscale image by eliminating the hue and saturation information while retaining the luminance.

ii. Decreasing the noise effect using the smooth filter by replacing each pixel with the average of its 3×3 neighborhood.

iii. Undertaking edge detection to identify the edges in an image using a Sobel edge detector. Thus, the resulting image becomes clearer.

iv. Doing the normalization process on the image by ranging the image on a scale of *0* and *1*.

v. Converting a grayscale image to binary image using the threshold method by replacing each pixel in the image with a value of *0* (for a typical black intensity) or *1* (for a typical white intensity). The profiles of the foot image and the background would be shown in white intensity and black intensity, respectively.

vi. Applying a non-linear filter, namely median filter, to remove the noise of the resulting error without reducing the image sharpness by replacing the gray level of each pixel by the median of the gray levels in the neighborhood of the pixels [20].

Sobel operator, written in Equation 1, used the derivative approximation to find or detect the edge. This operator applied a pair of horizontal (S_x) and vertical (S_y) gradient matrices (commonly in a 3×3 dimension) at the weight of the central pixels [21]. It works by calculating the gradient of the image intensity at each pixel within the image by calculating the magnitude (Equation 2) and finding the direction of the largest increase from light to dark and the rate of change in that direction (Equation 3).

$$S_x = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} * A\,,$$

$$S_y = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} * A \qquad (1)$$

$$S = \sqrt{S_x^2 + S_y^2} \qquad (2)$$

$$\Theta = a \qquad \frac{S_y}{S_x} \qquad (3)$$

The variable of A is the original source image, S is the gradient magnitude, and Θ is the gradient direction. Two kernels (3×3) of Sobel filter used for changes in the horizontal and vertical direction were convolved with the original image to calculate the approximations of the derivatives. To compute S_x and S_y, the appropriate kernel (window) was moved over the input image, then computing the value for one pixel, and shifting one pixel to the right. Once the end of the row was reached, moved down to the beginning of the next row. The results represented an edge by showing how abruptly or smoothly the image changes at each pixel and how the edge was likely to be oriented. A gradient value of *0* indicated the vertical edge that was much darker on the left side.

D. Feature Extraction

A process or method to reduce the dimensionality of initial raw data, but still accurately and completely depicting the initial data set, is called feature extraction. In this study, the practical use of feature extraction was performed by image processing using the algorithms to detect features such as points in a digital image. Feature detection was used to calculate the RFA, MLAA, and AHI, which will be explained separately as follows.

1) Rearfoot Angle (RFA)

RFA was measured as the acute angle between the projection of two lines, as shown in Fig. 4(a). The four marked points [21] were the base of the calcaneus, the Achilles tendon attachment, the Achilles tendon center at the height of medial malleoli and the center of shank posterior, 15 cm above the Achilles tendon center. The calculation process of RFA was described in the following steps.

i. It requires a binary image, then the user placed 4 points on each edge of calcaneus bone using marker tools of the developed algorithm,

ii. The algorithm calculates and marks the coordinate center of each distance owned by each pair of marker points,

iii. The algorithm calculates the gradient value using a numerical gradient (Equation 4) to estimate the values of the partial derivatives in each dimension using the known function values from the step ii.

$$\nabla \mathrm{F} = \frac{\partial}{\partial}\hat{\imath} + \frac{\partial}{\partial}\hat{\jmath} \qquad (4)$$

iv. The angle (in *radian* unit) is calculated and converted in degree.

2) Medial Longitudinal Arch Angle (MLAA)

MLAA is the midpoint of the medial malleolus (MM), the most prominent aspect of the navicular tuberosity (NT) and the most medial prominence of the first metatarsal head (MH) [5] as shown in Fig. 5. The calculation process of MLAA is described in the following steps.

i. It requires a grayscale image, then the user placed 3 points of MH, NT and MM using marker tools of Graphical User Interface in MATLAB,

ii. The algorithm calculates the distances of MM-NT, NT-MH, and MH-MM using Euclidean distance, the distance between two points in Euclidean space (relationship between the angle and the distance). The angle at NT (in radian) was calculated using Equation 5, then converted in degree. Descriptive classification of foot posture for RFA and MLAA calculation are presented in Table I.

$$\cos N \quad = \frac{a^2+b^2-c^2}{2a} \qquad (5)$$

3) Arch Height Index (AHI)

AHI, a clinical measure to assess the static foot posture and arc height [22] is a ratio of the dorsum height at 50%-foot length to the truncated foot length [23][24]. The calculation process of AHI is described in the following steps.

i. It requires a binary image, then the user places 4 points using marker tools to calculate the *AHI_total* and *AHI_instep* (on dorsum, base, the most posterior point of the calcaneus and the first metatarsophalangeal joint or the front end of the foot)
ii. The algorithm determines the coordinate center of the total foot length and calculates the *AHI_total* (Equation 6) and *AHI_instep* (Equation 7).

$$\mathrm{AHI}_{t} = \frac{I \quad he \quad ht}{t \quad f \quad le \quad h} \qquad (6)$$

$$\mathrm{AHI}_{i} = \frac{I \quad he \quad ht}{i \quad f \quad le \quad h} \qquad (7)$$

E. *Classification*

The types of foot postures (supination, neutral, or supination) were determined by the DIP results of RFA or MLAA rules, as presented in Table I.

TABLE I. Classification of foot posture (supination, neutral, pronation).

Type of tests	Type of foot posture		
	Supination	Neutral	Pronation
RFA [24]	RFA ≥ 5° varus	4° valgus to 4° varus	RFA ≥ 5° valgus
MLAA [27]	> 150°	130° to 150°	MLAA < 130°

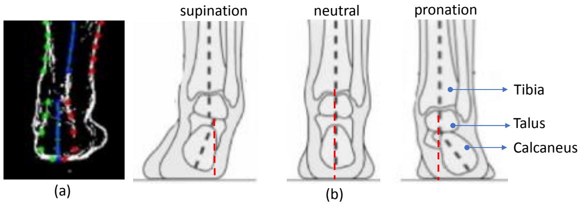


Fig. 4. Illustration of (a) Calculating RFA by the developed algorithm in MATLAB and (b) RFA from the anatomy of right rear foot [25].

III. Results

As shown in Fig. 6, each participant stood on a black open-box which was placed on the platform. The black open-box has three sides, which are the base, left and right sides. The black background was placed as a screen in front of the front legs (for taking pictures of the rear foot) and next to the legs (for taking pictures of the inner-side foot) to obtain a foot image that contrasts with the background color. The lighting system was positioned to provide sufficient luminance for the camera.

The results of taking pictures of the rear and inner-side foot are shown in Fig. 7(a). Each image was then processed by pre-processing technique (in the steps in Fig. 3.) in order to get a grayscale and binary images to conduct some measurements of angle and length based on RFA, MLAA and AHI tests. Fig. 7(b) shows an image pre-processing using the developed algorithm.

1) *Anthro-Posture v1.0* software

An algorithm was created in MATLAB to determine the foot posture from RFA and/ or MLAA tests. AHI was used as additional information to support the results of determining the classification of foot posture. The developed algorithm, which is named *Anthro-Posture v1.0* software, was built using a Graphical User Interface (GUI) and designed with the principle of user-friendly. Fig. 8 shows the main menu of the software where the user can choose one of the tests by clicking the selected button.

2) *Validity Test*

The angle and the length, the measurement of the variables of RFA, MLAA, and AHI, were validated by the specialists and a certain comparison method. The specialists, doctors of the medical rehabilitation in the Department of Medical Rehabilitation in Hasan Sadikin Hospital (RSHS) Bandung validated the result of length and angle measurement in this study using a ruler and arc as shown in Table II. The error means of length and angle from those comparisons were 6.22% and 0.26%, respectively.

Another validation technique was comparing the angle result of DIP with Kinovea software. This software, a valid and reliable tool [25, 26], is one of the test methods to accurately measure the angle range. Table III shows the comparison results of Konivea and *Anthro-Posture v1.0* software. The differences in angle measurements were caused by the resolution of Kinovea (the resolution is only 1 degree). Hence, it could be considered that its error means is 1.74%.

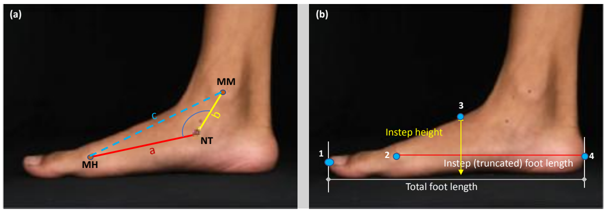


Fig. 5. Illustration of determining (a) MLAA and (b) AHI.

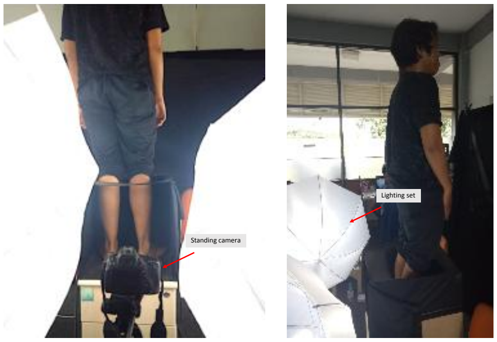


Fig. 6. The photograph of a participant's feet that stood on the standing platform was taken using a camera and lighting set.

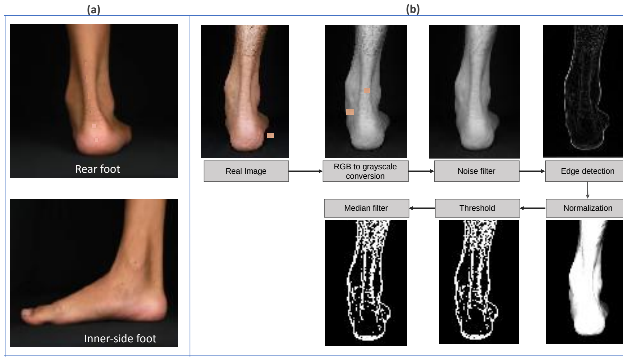


Fig. 7. (a)Photographs of the rear and inner side of the standing foot, (b) Results of each step of image pre-processing.

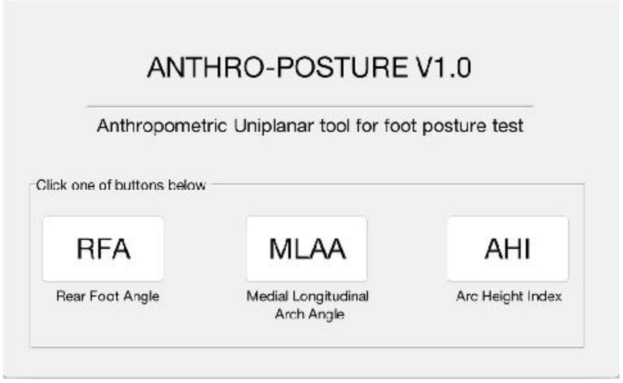
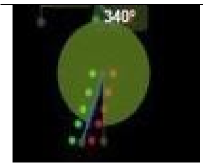
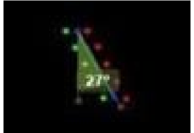
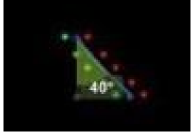
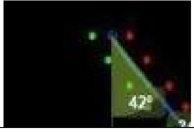

ANTHRO-POSTURE V1.0
Anthropometric Uniplanar tool for foot posture test
Click one of buttons below
RFA
Rear Foot Angle
MLAA
Medial Longitudinal Arch Angle
AHI
Arc Height Index

Fig. 8. The main menu of *Anthro-Posture v1.0* software.

TABLE II. ERRORS OF LENGTH AND ANGLE VALIDATIONS BETWEEN SPECIALISTS AND ANTHRO-POSTURE V1.0 SOFTWARE.

N	Length (cm)			Angle (°)		
	Ruler	Software	Error	Arc	Software	Error
1	6.1	6.1797	0.0797	140	140.7728	0.5520
2	5.1	5.2035	0.1035	125	125.5943	0.4754
3	5.9	5.9628	0.0628	152	152.1459	0.0960
4	6.0	6.0266	0.0266	128	128.3519	0.2749
5	4.8	4.9103	0.1103	138	138.9661	0.7001
6	4.9	5.0157	0.1157	146	146.1195	0.0818
7	5.2	5.2356	0.0356	142	142.1341	0.0944
8	4.9	4.9511	0.0511	139	139.2135	0.1536
9	4.2	4.2298	0.0298	145	145.2784	0.1920
10	4.6	4.6077	0.0077	157	157.0745	0.0475

TABLE III. COMPARISON RESULT OF ANGLE MEASUREMENT OF KONIVEA SOFTWARE AND SOFTWARE OF ANTHRO-POSTURE V1.0.

Tested angle (°) in *Anthro-Posture v.1* software	Angle measurement in *Konivea* software		Error (°)
	Description	Angle result (°)	
19.76°		20 °	0.24°
27.82°		27°	0.82°
40.79°		40°	0.79°
42.30°		42°	0.30°

3) The Listing Program of Anthro-Posture v1.0 for Anthropometrics Tests.

The listing program of the image pre-processing was shown in Listing Program 1 while the listing program for each chosen test in the main menu, namely RFA, MLAA or AHI, were shown in the Listing Program 2 to 4.

Listing program 1. Program of image pre-processing.

```
%import and resize image file
<code>
[FileName,Pathname]=uigetfile('*.jpg',sprintf('Pilih sampel
untuk scan'));
if FileName==0
   return
end

fullName=fullfile(Pathname,FileName);

disp(fullName);
imdat=imread(fullName);
imdat=imresize(imdat,[842 1500]);
imdat=rgb2gray(imdat);
preProcessing(hObject, eventdata, handles,imdat);

<\code>

%Pre-processing
<code>
```

```
%function filter binary

function gambar=filterBinary(gambar)
gambar=bwareaopen(gambar,10);

%function HPF

function gambar=doHPF(gambar)
kernelFilter=[0 -1/4  0;-1/4 2 -1/4;0 -1/4 0];
gambar=imfilter(gambar,kernelFilter,'conv');

%function LPF
function gambar=doLPF(gambar)
kernelFilter=[ 1/9 1/9 1/9;1/9 1/9 1/9;1/9 1/9 1/9; ];
gambar=imfilter(gambar,kernelFilter,'conv');

function preProcessing(hObject, eventdata, handles,gambar
)
% hObject    handle to selectFile (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUI
DATA)
% gambar     dalam bentuk matrix image

% Noise reduction
G = fspecial('gaussian',[5 5],5);
gambar=imfilter(gambar,G,'same');

% do High pass filter
gambar=doHPF(gambar);
C=double(gambar);

%Sobel Masking ( Edge Detection )
for i=1:size(C,1)-2
    for j=1:size(C,2)-2
        %Sobel mask for x-direction:
        Gx=((2*C(i+2,j+1)+C(i+2,j)+C(i+2,j+2))-
(2*C(i,j+1)+C(i,j)+C(i,j+2)));
        %Sobel mask for y-direction:
        Gy=((2*C(i+1,j+2)+C(i,j+2)+C(i+2,j+2))-
(2*C(i+1,j)+C(i,j)+C(i+2,j)));

        %The gradient of the image
        %B(i,j)=abs(Gx)+abs(Gy);
        gambar(i,j)=sqrt(Gx.^2+Gy.^2);

    end
end

% filtering implement low pass filter
gambar=doLPF(gambar);

%Convert image to binary and LPF
gambar=im2bw(gambar,0.0600);
gambar=doLPF(gambar);

% noise reduction
 gambar=filterBinary(gambar);
axes(handles.edgeDisplay);
imshow(gambar);

<\code>
```

Listing program 2. Program to measure the angle in RFA test.

```
%Plotmarker
% --- Executes on button press in plotPoint_Calc.
function plotPoint_Calc_Callback(hObject, eventdata, hand
les)
% hObject    handle to plotPoint_Calc (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUI
DATA)
axes(handles.edgeDisplay);
hold on;
axis on;

counter1=1;

xSTJ1=zeros(1,2);
ySTJ1=zeros(1,2);

counter2=1;
xSTJ2=zeros(1,2);
ySTJ2=zeros(1,2);

for c=1:4
    [x,y]=ginputc(1, 'Color', 'r', 'LineWidth', 0.5);
    if mod(c,2)==1
        plot(x,y, 'g.', 'MarkerSize', 15, 'LineWidth', 1);

        xSTJ1(counter1)=x;
        ySTJ1(counter1)=y;
        counter1=counter1+1;

    elseif mod(c,2)==0
            plot(x,y, 'r.', 'MarkerSize', 15, 'LineWidth', 1);
            xSTJ2(1,counter2)=x;
            ySTJ2(1,counter2)=y;

            counter2=counter2+1;
    end
end

% middle point
global Xstjleg;
global Ystjleg;
```

```
Xstjleg=times(plus(xSTJ1,xSTJ2),1/2);
Ystjleg=times(plus(ySTJ1,ySTJ2),1/2);

plot(Xstjleg,Ystjleg, 'b.', 'MarkerSize', 15, 'LineWidth', 1);

<\code>
% --------
%Calculating RFA Angle
% ----------
<code>

% --- Executes on button press in getangle.
function getangle_Callback(hObject, eventdata, handles)
% hObject    handle to getangle (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUI
DATA)
global Xstjleg;
global Ystjleg;
disp('====RFA===');
disp(Xstjleg);
disp(Ystjleg);

%Calculating gradient of each marker point then take the
average value
linemidcalc=line(Xstjleg,Ystjleg,'LineWidth',2);
meangradient_calca=mean(gradient([Xstjleg],[Ystjleg]));

%Convert gradient to degree
calca_angle=atan(meangradient_calca)*57.2957795131;

set(handles.meangradient_calca,'String',sprintf(num2str(me
angradient_calca)));
set(handles.calca_angle,'String',sprintf(num2str(calca_angle
)));

<\code>
```

Listing Program 3. Program to measure the angle in MLAA test.

```
%Plot marker for medial malleolus (MM), navicular
%tuberosity (NT), and metatarsal head (MH)

<code>

% --- Executes on button press in MH_PLOT.
function MH_PLOT_Callback(hObject, eventdata, handles)
% hObject    handle to MH_PLOT (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUI
DATA)
    axes(handles.display_image);
    hold on;
    axis on;
    global XMH;
    global YMH;

    [x,y]=ginputc(1, 'Color', 'r', 'LineWidth', 0.5);
     XMH=x;
     YMH=y;

    plot(x,y, 'b.', 'MarkerSize', 15, 'LineWidth', 1);
% --- Executes on button press in NT_PLOT.
function NT_PLOT_Callback(hObject, eventdata, handles)
% hObject    handle to NT_PLOT (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUI
DATA)
axes(handles.display_image);
    hold on;
    axis on;
    global XNT;
    global YNT;

    [x,y]=ginputc(1, 'Color', 'r', 'LineWidth', 0.5);
    XNT=x;
    YNT=y;

    plot(x,y, 'r.', 'MarkerSize', 15, 'LineWidth', 1);
% --- Executes on button press in MM_PLOT.
function MM_PLOT_Callback(hObject, eventdata, handles
)
% hObject    handle to MM_PLOT (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUI
DATA)
axes(handles.display_image);
    hold on;
    axis on;

    global XNM;
    global YNM;

    [x,y]=ginputc(1, 'Color', 'r', 'LineWidth', 0.5);
    XNM=x;
    YNM=y;

plot(x,y, 'g.', 'MarkerSize', 15, 'LineWidth', 1);

<\code>

%Calculating MLAA

<code>
```

```
% --- Executes on button press in getAngle.
function getAngle_Callback(hObject, eventdata, handles)
% hObject    handle to getAngle (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUI
DATA)
   global XNM;
   global YNM;

   global XNT;
   global YNT;

   global XMH;
   global YMH;

   %variable relation between points
   MH_COORDINATE=[XMH YMH];
   NT_COORDINATE=[XNT YNT];
   MM_COORDINATE=[XNM YNM];

   J=[MM_COORDINATE;NT_COORDINATE];
   K=[NT_COORDINATE;MH_COORDINATE];
   L=[MH_COORDINATE;MM_COORDINATE];

   %Finding eucledian distance of each point
   M=pdist(J,'euclidean');
   N=pdist(K,'euclidean');
   O=pdist(L,'euclidean');

   disp(M);
   disp(N);
   disp(O);

  %Calculating cos value of NT
   cos_NT=((M^2)+(N^2)-(O^2))/(2*M*N);
   disp(cos_NT);

   %Calculating MLAA degree
   NT_Rad=acos(cos_NT);
   NT_Deg=NT_Rad*57.2958;

<\code>
```

Listing Program 4. Program to measure AHI test.

```
%Calculating Height of Foot
<code>
% --- Executes on button press in VerDist.
function VerDist_Callback(hObject, eventdata, handles)
% hObject    handle to VerDist (see GCBO)
```

```
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUI
DATA)
axes(handles.ImOut);
hold on;
axis on;

%Plot marker for start point and end point of foot height
counter1=1;

xSTJ1=zeros(1,5);
ySTJ1=zeros(1,5);

counter2=1;
xSTJ2=zeros(1,5);
ySTJ2=zeros(1,5);

for c=1:2
   [x,y]=ginputc(1, 'Color', 'r', 'LineWidth', 0.5);
   if mod(c,2)==1
      plot(x,y, 'g.', 'MarkerSize', 15, 'LineWidth', 1);

      xSTJ1(counter1)=x;
      ySTJ1(counter1)=y;
      counter1=counter1+1;

   elseif mod(c,2)==0
         plot(x,y, 'r.', 'MarkerSize', 15, 'LineWidth', 1);
         xSTJ2(1,counter2)=x;
         ySTJ2(1,counter2)=y;

         counter2=counter2+1;
   end
end

% middle point
global XVer;
global YVer;

XVer=times(plus(xSTJ1,xSTJ2),1/2);
YVer=times(plus(ySTJ1,ySTJ2),1/2);

plot(XVer,YVer, 'b.', 'MarkerSize', 15, 'LineWidth', 1);


plot1VerCor=[xSTJ1 ySTJ1];
plot2VerCor=[xSTJ2 ySTJ2];

%Find Euclidean distance of start point and end point
VerDist=[plot1VerCor;plot2VerCor];
VerDistPix=pdist(VerDist,'euclidean');

%Calibrating distance
global VerDistCm;
```

```

b = VerDistPix * 0.026458333;
VerDistCm = b-2.29938;

set(handles.VerDistCm,'string',num2str(VerDistCm))

<\code>

Calculating Length ( Total ) of Foot

<code>

% --- Executes on button press in HorDist.
function HorDist_Callback(hObject, eventdata, handles)
% hObject    handle to HorDist (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUI
DATA)
axes(handles.ImOut);
hold on;
axis on;

%Plot marker for start point and end point of foot length
counter1=1;
xleg1=zeros(1,5);
yleg1=zeros(1,5);

counter2=1;
xleg2=zeros(1,5);
yleg2=zeros(1,5);

for c=1:2
   [x,y]=ginputc(1, 'Color', 'r', 'LineWidth', 0.5);
   if mod(c,2)==1
       plot(x,y, 'g.', 'MarkerSize', 15, 'LineWidth', 1);

       xleg1(1,counter1)=x;
       yleg1(1,counter1)=y;
       counter1=counter1+1;

   elseif mod(c,2)==0
        plot(x,y, 'r.', 'MarkerSize', 15, 'LineWidth', 1);

        xleg2(1,counter2)=x;
        yleg2(1,counter2)=y;

        counter2=counter2+1;
   end
end
% middle point
global Xhor;
global Yhor;

Xhor=times(plus(xleg1,xleg2),1/2);
```

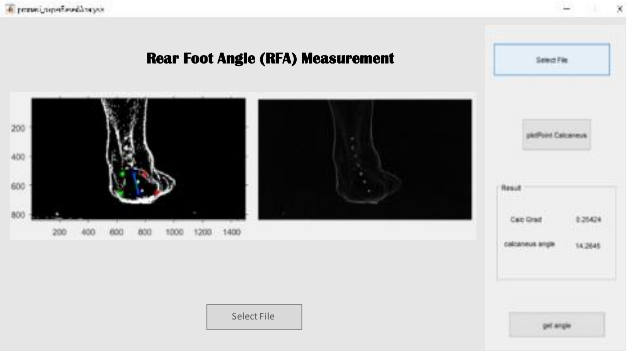
```
Yhor=times(plus(yleg1,yleg2),1/2);
plot(Xhor,Yhor, 'b.', 'MarkerSize', 15,'LineWidth', 1 );

plot1HorCor=[xleg1 yleg1];
plot2HorCor=[xleg2 yleg2];

%Find Euclidean distance of start point and end point
HorDist=[plot1HorCor;plot2HorCor];
HorDistPix=pdist(HorDist,'euclidean');

%Calibrating distance
global HorDistCm;

a=HorDistPix*0.026458333;
HorDistCm = a-2.90794;

set(handles.HorDistCm,'string',num2str(HorDistCm))

<\code>

%Calculating Length (Instep) of Foot
<code>
% --- Executes on button press in InsDist.
function InsDist_Callback(hObject, eventdata, handles)
% hObject    handle to InsDist (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUI
DATA)
axes(handles.ImOut);
hold on;
axis on;
%Plot marker for start point and end point of foot Instep
length
counter1=1;
xins1=zeros(1,5);
yins1=zeros(1,5);

counter2=1;
xins2=zeros(1,5);
yins2=zeros(1,5);

for c=1:2
   [x,y]=ginputc(1, 'Color', 'r', 'LineWidth', 0.5);
   if mod(c,2)==1
       plot(x,y, 'g.', 'MarkerSize', 15, 'LineWidth', 1);

       xins1(1,counter1)=x;
       yins1(1,counter1)=y;
       counter1=counter1+1;

   elseif mod(c,2)==0
        plot(x,y, 'r.', 'MarkerSize', 15, 'LineWidth', 1);

        xins2(1,counter2)=x;
        yins2(1,counter2)=y;

        counter2=counter2+1;
    end
end
% middle point
global XIns;
global YIns;

XIns=times(plus(xins1,xins2),1/2);
YIns=times(plus(yins1,yins2),1/2);
plot(XIns,YIns, 'b.', 'MarkerSize', 15,'LineWidth', 1 );

plot1InsCor=[xins1 yins1];
plot2InsCor=[xins2 yins2];

%Find Euclidean distance of start point and end point
InsDist=[plot1InsCor;plot2InsCor];
InsDistPix=pdist(InsDist,'euclidean');

%Calibrating distance
global InsDistCm;

a=InsDistPix*0.026458333;
InsDistCm = a-2.90794;

set(handles.InsDistCm,'string',num2str(InsDistCm))

<\code>
```

4) Anthropometric Test with RFA.

Rearfoot angle was calculated by uploading the binary image in Anthro-Posture v1.0 software, then clicked the <plotPoint Calcaneus> button to plot the four points at the upper and below edges of calcaneus line. The result of the gradient and angle of tibia and calcaneus was displayed on the screen (Fig. 9). The classification of the foot posture of all participants was determined based on the angle, as presented in Table IV (*p* for pronation, *n* for neutral and *s* for supination).

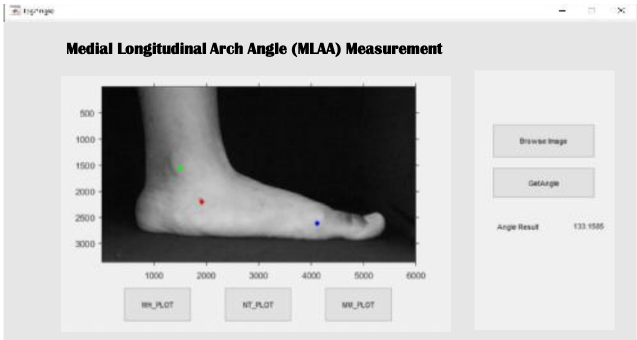
5) Anthropometric Test with MLAA.

Medial longitudinal arch angle was calculated by uploading the grayscale image in Anthro-Posture v1.0 software, then marked the three points of MH, NT, and MM on the image by clicking the buttons of <MH_PLOT>, <NT_PLOT> and <MM_PLOT> respectively. The result of MLAA was displayed on the screen after clicking the <GetAngle> button (Fig. 10). The classification of the foot posture of all participants was determined based on the angle, as presented in Table V. All participants were classified having the pronated and neutral feet based on the MLAA measurement. From the results, 70% of participants had a neutral right foot and 90% a neutral left foot.



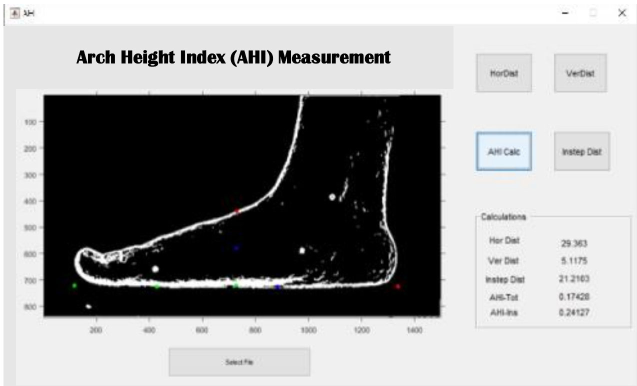
Calc Grad 0.25424
calcaneus angle 14.2645

Fig. 9. The RFA test in *Anthro-Posture v1.0* software.



Angle Result 133.1585

Fig. 10. The MLAA test in *Anthro-Posture v1.0* software.



Hor Dist 29.363
Ver Dist 5.1175
Instep Dist 21.2103
AHI-Tot 0.17428
AHI-Ins 0.24127

Fig. 11. The AHI test in *Anthro-Posture v1.0* software.

6) Anthropometric Test with AHI.

The arc height index was calculated by uploading the binary image in Anthro-Posture v1.0 software, then clicked the buttons of <HorDist> and <VerDist> to mark the two pairs points of total foot length and instep length. Whereas for the instep arc height

TABLE IV. THE CLASSIFICATION RESULT OF RFA.

Subject-n	Rearfoot angle (°)		Classification	
	Right	Left	Right	Left
1	1.65	2.91	*n*	*n*
2	6.00	1.36	*p*	*p*
3	2.81	4.66	*n*	*n*
4	15.43	8.05	*p*	*p*
5	9.05	6.11	*p*	*p*
6	3.93	10.07	*n*	*p*
7	5.41	3.20	*p*	*n*
8	3.73	2.38	*n*	*n*
9	13.34	16.13	*p*	*p*

TABLE V. THE CLASSIFICATION RESULT OF MLAA.

Subject-n	MLAA (°)		Classification	
	Right	Left	Right	Left
1	140.77	134.17	*n*	*n*
2	128.75	131.89	*p*	*n*
3	150.04	150.01	*n*	*n*
4	125.59	131.82	*p*	*n*
5	145.27	149.81	*n*	*n*
6	157.07	151.86	*p*	*p*
7	137.36	150.03	*n*	*n*
8	146.11	152.35	*n*	*n*
9	142.63	143.70	*n*	*n*
10	139.23	141.02	*n*	*n*

TABLE VI. THE CALCULATED *AHI_TOTAL* AND *AHI_INSTEP* OF THE INNER-SIDE FOOT.

Subject -n	Right foot (cm)			Left foot (cm)		
	AHI_ total	AHI_ instep	Arch group	AHI_ total	AHI_ instep	Arch group
1	0.18	0.27	*n*	0.21	0.27	*n*
2	0.15	0.22	*l*	0.18	0.22	*l*
3	0.23	0.32	*n*	0.24	0.34	*n*
4	0.18	0.25	*l*	0.24	0.26	*l*
5	0.17	0.24	*l*	0.17	0.24	*l*
6	0.23	0.34	*n*	0.22	0.32	*n*
7	0.21	0.29	*n*	0.19	0.28	*n*
8	0.21	0.29	*n*	0.24	0.36	*n*
9	0.24	0.33	*n*	0.22	0.30	*n*
10	0.19	0.28	*n*	0.19	0.27	*n*

index used the instep foot length instead of total foot length [28]. The comparison result between the instep height and the instep foot length is displayed respectively in Fig. 11, while the calculated AHI of all participants is presented in Table VI (arch group: *n* for normal, *l* for lower).

IV. DISCUSSION

The *Anthro-Posture v1.0* software has been created, validated and tested completely in this study. The classification results of all anthropometric tests could be learned more to compare them to other works.

Based on the RFA test, of all participants, 50% had a neutral right foot and 60% had a neutral left foot. According to the finding result of RFA using the goniometer device [5], the method consistency was lower compared to MLAA and other tools.

The MLAA test in this study resulted in more participants having a neutral foot. Hence, the result difference of the classification from both tools was around 20-30%. Furthermore, one could interpret that of 50% participants having a pronated right foot with RFA, 40% of them have a neutral right foot with MLAA. In the other side for the left feet, of 40% of participants having a pronated left foot with RFA, 75% of them have a neutral left foot with MLAA. These results reinforced the findings of the previous research [5] which stated that MLAA was the strongest uniplanar tool due to its higher reliability, good agreement on steps for foot classification and wider foot classification limits.

For AHI test, it is often used as supporting data for further observation. One is categorized in the high-arched group when the instep arch height index is at least 0.388 and in the low-arched group when AHI equal or less than 0.262 [29]. From all participants, it is only 30% of them included in the lower-arched group. Associated with MLAA, from the lower-arched foot participants, 66% of them have pronated feet. Meanwhile associated with RFA, 100% of participants have pronated feet. All these AHI results were quite relevant because based on the finding study [22], a low-arched may result in a pronated foot, but not all pronated feet have a low-arched type.

The test results of RFA, MLAA, and AHI have been validated by the specialists and other comparison tools using a ruler, arc and *Konivea* software in order to improve the accuracy. The error means of length and angle in this software were 6.22% and (0.26-1.74) %, respectively.

Determination of the points of tibial and calcaneus in this first algorithm was still carried out manually, the same as that done by the doctors in their manual measurement using markers, goniometers and arc [9], [13]. But in the case of DIP, this manual step was a limitation that should be eliminated in a future developing algorithm. However, compared to other DIP research conducted by Lin et. Al [14], this *Anthro-Posture v1.0* software was not only capable of measuring the angle and length of images but also providing the classification results of the foot posture.

V. CONCLUSION

The anthropometric techniques are commonly used in the classification of foot types, but the lack of carrying out the assessment with these uniplanar tools should be improved in many ways. This study has demonstrated the development algorithm in MATLAB to measure the foot posture, which is named *Anthro-Posture v1.0* software. The advantages of this technique are providing statistical medical data in a shorter time and minimizing the human error in measurement. In the future, this study can be improved to be used by doctors in obtaining large amounts of data for research needed.

REFERENCES

[1] M. N. Anas, “An instrumented insole system for gait monitoring and analysis,” Int. J. Interact. Mob. Technol., vol. 10, no. 6, pp. 30–34, 2014.

[2] S. Morioka, M. Okita, Y. Takata, S. Miyamoto, and H. Itaba, “Effects of changes of foot position on Romberg’s quotient of postural sway and leg muscles electromyographic activities in standing,” J. Japanese Phys. Ther. Assoc., vol. 3, no. 1, pp. 17–20, 2000.

[3] J. Kongcharoen, S. Pruitikanee, S. Puttinaovarat, Y. Tubtiang, and P. Chankeaw, “Gamification smartphone application for leg physical therapy,” Int. J. online Biomed. Eng., vol. 15, no. 8, pp. 31–41, 2019.

[4] A. K. Buldt, G. S. Murley, P. Levinger, H. B. Menz, C. J. Nester, and K. B. Landorf, “Are clinical measures of foot posture and mobility associated with foot kinematics when walking?,” J. Foot Ankle Res., vol. 8, no. 1, pp. 1–12, 2015.

[5] H. Menz and S. E. Munteanu, “Validity of 3 Clinical Techniques for the Measurement of Static Foot Posture in,” Journal of Orthopaedic and Sports Physical Therapy, vol. 3, no. 8, 2005.

[6] A. Keenan et al., “The Foot Posture Index : Rasch Analysis of a Novel , Foot-Specific Outcome Measure,” vol. 88, no. January, pp. 88–93, 2007.

[7] J. Burns, B. P. Hons, A. Keenan, B. Pod, and A. Redmond, “Foot Type and Overuse Injury in Triathletes,” vol. 95, no. 3, pp. 235–241, 2005.

[8] M. Razeghi and M. E. Batt, “Foot type classification: a critical review of current methods,” Gait Posture, vol. 15, no. 3, pp. 282–291, 2002.

[9] B. Langley, M. Cramp, and S. C. Morrison, “Clinical measures of static foot posture do not agree,” J. Foot Ankle Res., pp. 1–6, 2016.

[10] A. C. Redmond, J. Crosbie, and R. A. Ouvrier, “Development and validation of a novel rating system for scoring standing foot posture: The Foot Posture Index,” Clin. Biomech., vol. 21, no. 1, pp. 89–98, Jan. 2006.

[11] G. S. Murley, H. B. Menz, and K. B. Landorf, “Journal of Foot and Ankle Research A protocol for classifying normal- and flat-arched foot posture for research studies using clinical and radiographic measurements,” vol. 13, pp. 1–13, 2009.

[12] V. Erickson, A. U. Kamthe, and A. E. Cerpa, “Measuring Foot Pronation Using RFID Sensor Networks,” in Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems, 2009, pp. 325–326.

[13] J. S. Lee, K. B. Kim, J. O. Jeong, N. Y. Kwon, and S. M. Jeong, “Correlation of Foot Posture Index With Plantar Pressure and Radiographic Measurements in Pediatric Flatfoot,” Ann Rehabil Med, vol. 39, no. 1, pp. 10–17, 2015.

[14] C. H. Lin, C. C. Yeh, and Z. H. Qiu, “Assessment of Subtalar Joint Neutral Position : Study of Image Processing for Rear Foot Image,” vol. II, pp. 3–7, 2017.

[15] C. Lin, Z. Qiu, and C. Yeh, “Image processing for rear foot image evaluating leg and foot angles,” Measurement, vol. 126, no. May, pp. 168–183, 2018.

[16] A. C. Redmond, J. Crosbie, and R. A. Ouvrier, “Development and validation of a novel rating system for scoring standing foot posture : The Foot Posture Index,” vol. 21, pp. 89–98, 2006.

[17] T. Mcpoil, D. Ph, M. W. Cornwall, and D. Ph, “Relationship Between Neutral Subtalar Joint Position and Pattern of Rearfoot Motion During Walking,” Foot angkle, vol. 15, no. 3, 1994.

[18] M. Hill, R. Naemi, H. Branthwaite, and N. Chockalingam, “The relationship between arch height and foot length : Implications for size grading,” Appl. Ergon., vol. 59, pp. 243–250, 2017.

[19] C. Lin, Z. Qiu, and C. Yeh, “Image processing for rear foot image evaluating leg and foot angles,” Measurement, vol. 126, no. May, pp. 168–183, 2018.

[20] J. J. Li Tan, “Image Processing Basics,” in Digital Signal Processing (Second Edition), 2013.

[21] S. Gupta and S. G. Mazumdar, “Sobel Edge Detection Algorithm,” International Journal of Computer Science and Management Research, vol. 2, no. 2, pp. 1578–1583, 2013.

[22] M. T. Gross, “Intraexaminer Reliability , Interexaminer Reliability , and Mean values for Nine Lower Skeletal Measures in Healthy Naval Midshipmen.”

[23] X. Zhao, T. Tsujimoto, B. Kim, and K. Tanaka, “Association of arch height with ankle muscle strength and physical performance in adult men,” vol. 34, no. 2, pp. 119–126, 2017.

[24] T. G. Mcpoil et al., “Effect of using truncated versus total foot length to calculate the arch height ratio,” vol. 18, pp. 220–227, 2008.

[25] M. Saghazadeh, K. Tsunoda, and T. Okura, “Foot arch height and rigidity index associated with balance and postural sway in elderly women using a 3D foot scanner,” Foot Ankle Online J., vol. 7, no. 4, 2014.

[26] C. M. Norris, “Lower limb motion during walking, running and jumping,” in *Managing Sports Injuries (Fourth Edition)*. doi:10.1016/B978-0-7020-3473-2.00011-3, 2011.

[27] A. Puig-diví and C. Escalona-marfil, “Validity and reliability of the Kinovea program in obtaining angles and distances using coordinates in 4 perspectives,” 2019.

[28] C. Balsalobre-Ferna´Ndez, “The Concurrent Validity And Reliability Of A Low-Cost,High-Speed Camera-Based Method For Measuring The Flight Time Of Vertical Jumps,” J. Strength Cond. Res., vol. 28, no. 2, pp. 528–533, 2014.

[29] R. J. Butler, H. Hillstrom, J. Song, C. J. Richards, and I. S. Davis, “Arch Height Index Measurement System,” *J. Am. Podiatr. Med. Assoc.*, vol. 98, no. 2, pp. 102–106, Mar. 2008.

[30] M. T. H. Parash, H. Naushaba, A. Rahman, and S. C. Shimmi, “Types of Foot Arch of Adult Bangladeshi Male,” vol. 1, no. 4, pp. 52–54, 2013.