Gait Phase Partitioning and Footprint Detection Using Mutually Constrained Piecewise Linear Approximation with Dynamic Programming

Makoto YASUKAWA†, Nonmember, Yasushi MAKIHARA†, Toshinori HOSOI††, Members, Masahiro KUBO†, Nonmember, and Yasushi YAGI†, Member

SUMMARY Human gait analysis has been widely used in medical and health fields. It is essential to extract spatio-temporal gait features (e.g., single support duration, step length, and toe angle) by partitioning the gait phase and estimating the footprint position/orientation in such fields. Therefore, we propose a method to partition the gait phase given a foot position sequence using mutually constrained piecewise linear approximation with dynamic programming, which not only represents normal gait well but also pathological gait without training data. We also propose a method to detect footprints by accumulating toe edges on the floor plane during stance phases, which enables us to detect footprints more clearly than a conventional method. Finally, we extract four spatial/temporal gait parameters for accuracy evaluation: single support duration, double support duration, toe angle, and step length. We conducted experiments to validate the proposed method using two types of gait patterns, that is, healthy and mimicked hemiplegic gait, from 10 subjects. We confirmed that the proposed method could estimate the spatial/temporal gait parameters more accurately than a conventional skeleton-based method regardless of the gait pattern.

key words: gait phase partitioning, piecewise linear approximation, dynamic programming, footprint detection, rehabilitation

1. Introduction

Human gait analysis has been widely used in many research fields, such as forensics [1]–[3], biomechanics, medical, and health research fields. There is a rich body of literature on human gait analysis, particularly in the fields of medical and health research, including research on change detection in health fields. There is a rich body of literature on gait analysis techniques can be mainly divided into invasive methods and non-invasive methods. Invasive methods often use wearable sensors, such as accelerometers [12], [13], gyroscopes [14], [15], inertial measurement units [16], [17], and motion capture systems with markers attached to body joints [18], [19]. However, for invasive methods, it takes a relatively long time to attach the wearable sensors or markers to subjects’ body parts and may also constrain subjects’ movement, which is unsuitable in real applications.

By contrast, because non-invasive methods do not require the attachment of a sensor/marker to the body, they are more suitable for real applications. A typical approach to non-invasive methods is video-based gait analysis. Generally, video-based gait analysis is further divided into two categories: appearance-based and model-based methods. Appearance-based methods extract a gait feature directly from a captured video without assuming a human body model. Various appearance-based gait representations have been proposed, including the size-normalized silhouette sequence [20], gait energy image [21], frequency-domain feature [22], and local binary pattern [23]. However, the above-mentioned appearance-based features do not explicitly represent gait parameters, such as duration of stance/swing phases, step length (SL), stride, toe angle (TA), and walking speed. Those explicit gait parameters are essential for medical/health applications. For example, the relationship between spatio-temporal gait asymmetry and balance for chronic stroke patients has been reported [24].

Model-based methods [25]–[27] assume an articulated human body model, and fit it to a walking video. They are therefore generally more suitable to medical/health applications than appearance-based methods because the fit model is useful for extracting the above-mentioned gait parameters. However, traditional model fitting approaches suffer from error-prone estimation results and high computational cost. By contrast, recent deep learning-based approaches to human pose estimation [28]–[31] have improved the situation and hence are used in gait recognition field [32]. However, most deep learning-based approaches estimate the human pose frame by frame and hence do not guarantee temporal consistency (e.g., left and right feet are sometimes falsely swapped between adjacent time frames). Moreover, the joints of detailed body parts, such as ankles and toes, are still difficult to estimate accurately using these methods. Consequently, it may not be possible to accurately extract gait parameters, such as the single support duration (SSD) and the TA, even using deep learning-based pose estimation methods.

Therefore, we propose a non-invasive method using an RGBD (i.e., color and depth) image sequence to estimate...
four gait parameters, that is, SSD, double support duration (DSD), TA, and SL, which are often used in medical/health applications [10]. For this purpose, we partition the entire image sequence into gait phases (e.g., stance and swing phases) to compute the SSD/DSD, and also extract a footprint on the ground plane to compute the TA and SL, as outlined in Fig. 1.

Regarding gait phase partitioning, we first extract the position sequences of the left and right feet using depth image analysis. Because the position remains still during the stance phase and moves during the swing phase, we cast stance/swing phase partitioning as the problem of the piecewise linear approximation of the position sequence. A straightforward application of the piecewise linear approximation to each foot may not always result in a reasonable phase transition; for example, after a swing phase of the left foot, another swing phase of the same foot may come before a swing phase of right foot, which is unlikely to occur in reality. Therefore, we propose a unified piecewise linear approximation method in which the piecewise lines for the left and right feet are constrained by each other to prevent the above-mentioned unlikely transition.

Regarding footprint extraction, we build our method on a feature-point accumulation method [33], where a series of footprints on an image plane is used as a guide for pedestrian detection. If we directly apply the original method [33] for our purpose, we may, however, suffer from blurry footprints that include accumulated feature points not only during the stance phase but also during the swing phase, which can still be a sufficient guide for pedestrian detection but would be insufficient for the accurate estimation of the TA and SL. Therefore, we extend the method to obtain a clearer footprint by accumulating features during the stance phase only, which are recognized by the aforementioned gait partitioning method. The contributions of this paper are mainly summarized as the following two points:

1. **Mutually constrained piecewise linear approximation for gait phase partitioning**

   Piecewise linear approximation has been widely studied and many approaches using dynamic programming (DP) have been proposed for a curve [34], [35], waveform [36], and polygon [37]. They aim at a single target (e.g., point series of a single closed curve); however, we aim at jointly partitioning the phases of both the left and right feet. More specifically, we design a DP-based framework of joint piecewise linear approximation on position sequences of the left and right feet, where invalid states (e.g., a state for which the left and right feet are swung simultaneously) and invalid state transitions (e.g., the direct transition from the left swing phase to the right swing phase) are prohibited.

2. **Footprint extraction using toe edge accumulation**

   The original method [33] accumulates feature points regardless of the gait phases; however, we accumulate foot edges only for the stance phase because the footprint is observed in it, which contributes to clearer footprint extraction that is free from unnecessary and irrelevant edges during swing phases. Additionally, the original method accumulates features on an image plane; however, we do this on the ground plane via depth image analysis, which enables us to obtain the TA and SL.

### 2. Related Work

#### 2.1 Gait Phase Partitioning

A gait cycle is partitioned into a different number of phases...
depending on the purpose of the research. For example, a two-phase model \cite{38, 39} includes the stance phase and swing phase. A three-phase model divides a gait cycle into two half-stance phases and a swing phase \cite{40}. The first half-stance phase ranges from the heel strike to the middle of the mid-stance, whereas the second half-stance ranges from the middle of the mid-stance to toe-off. A six-phase model is composed of the initial contact, loading response, mid-stance, terminal stance, pre-swing, and swing phase \cite{41}. In this study, we use the simple yet useful two-phase model because the two phases, that is, stance and swing phases, capture the most fundamental part of the gait, and can be observed by a vision sensor.

2.1.1 Machine Learning-Based Approaches

Machine learning-based approaches typically train a gait phase partitioning model using training data with ground-truth labels of the phases. Liu and Sarkar \cite{42} proposed a gait phase partitioning method for gait recognition using a population hidden Markov model. They divided the gait cycle into 20 phases in a temporally uniform manner, and hence the partitioned phases did not correspond to physical gait events, such as stance phase and swing phase, which are required to compute gait parameters such as the stance phase time. Tang et al. \cite{43} proposed a deep learning-based method to output the toe-off event using a consecutive silhouettes difference map generated by the frame difference. However, the toe-off event alone is insufficient for calculating gait parameters such as the stance phase time, which also requires the heel strike event.

Additionally, the necessity of a sufficient amount of training data is another shortcoming of machine learning-based approaches, and this becomes more problematic when they are applied to medical/health fields. This is because it is time-consuming to collect training data for each targeted disease in medical applications, such as when recruiting patients with a specific disease and setting up a data capturing system in a hospital or care home. Although we may construct a generic pre-trained model that is trained on healthy subjects, it may not work well for pathological gait because of significant differences in gait between healthy subjects and patients.

2.1.2 Rule-Based Approaches

Rule-based approaches partition the gait phase using predefined rules without the training data that are required for machine learning-based approaches.

Auvinet et al. \cite{44} proposed a heel strike detection algorithm using an RGBD sensor by searching the temporal local maxima of the distance between the left and right knee joints along the walking longitudinal axis. However, the heel strike event alone is insufficient for calculating gait parameters such as the stance phase time, which also requires the toe-off event. Latorre et al. \cite{45} proposed a method for detecting the heel strike and toe-off event based on the antero-posterior distance between the ankle and sacral joints, which were extracted using the Microsoft Kinect SDK. More specifically, it detects the heel strike and toe-off when the ankle joint is located most anterior and posterior to the sacrum, respectively. However, the method does not guarantee a physically plausible gait phase transition because it processes the left and right feet independently; for example, it may skip a double support phase when toe-off is accidentally detected earlier than the heel strike.

2.2 Footprint Detection

Promising approaches to footprint detection are methods that use the temporal accumulation of feature points (e.g., corners or edges) at each pixel in a gait image sequence \cite{33, 46, 47}. The intuition behind this approach is based on the fact that, during the stance phase (i.e., time duration from the heel strike to toe-off), the foot remains at the same place, and the accumulation of foot feature points over time highlights the foot position during the stance phase (i.e., heel strike or toe-off position).

Early work by Bouchrika et al. \cite{33} demonstrated that a series of heel strikes is highlighted well by the corner points voting over time, and that they are useful as a guide for pedestrian detection. Jung et al. \cite{46} further extended the previous work \cite{33} to obtain a more accurate foot position during the stance phase. They extracted key frames at the local maxima of the vertical head position time-series and composed a filtered accumulated image using the silhouette at each key frame, which more clearly highlighted the foot position at each step. Because these two methods accumulate feature points on an image plane, the position/posture of the footprint in the world coordinate cannot be estimated.

By contrast, Evans et al. \cite{47} used multiple calibrated cameras to extract footprints on the floor plane (i.e., the world coordinate). They accumulated foreground pixels over multiple views and frames (time) on the floor plane and then extracted footprints using thresholding. The method is, however, not practical in a clinical setting because it takes time to set up and calibrate multiple cameras onsite.

3. Overview

In this section, we outline the proposed method along with Fig. 1. Given background and input depth images, we extract a silhouette of a person using depth-based background subtraction \cite{48}, and then project it to the world coordinate to obtain a three-dimensional (3D) point cloud.

Next, we extract a left/right foot point cloud, compute its center of gravity (CoG), and then partition temporal sequences of the CoG into gait phases using the proposed piecewise linear approximation. Thereafter, we extract edges from RGB images and then project them to the world coordinate to obtain a 3D edge point cloud.

Subsequently, we select foot edge points based on the vertical position of the 3D edge points and then accumulate the edges during the stance phases obtained by the
above-mentioned gait phase partitioning. Finally, we compute the spatio-temporal gait parameters (e.g., SL and SSD) using the foot model fitted to the accumulated edge image on the floor plane.

4. Preprocessing

4.1 Floor Plane Detection

The floor plane is extracted from background RGB and depth images averaged over multiple frames. The area of the floor plane, where a carpet with a pre-defined color is placed, is detected using a chroma key technique (Fig. 2). The depth data in the floor plane area are converted to a 3D point cloud, and then the floor plane geometry (e.g., the surface normal) is obtained using a plane model fitted using RANSAC [49], [50].

4.2 Silhouette Extraction

Silhouettes were extracted from the background and input depth images using depth-based background subtraction [48], which considers the depth value in addition to the observation probability. An example of the input depth image and extracted silhouette is shown in (Fig. 3 (a), (b)).

4.3 World Coordinate Setting

To calculate the direction of the left and right feet, and the SL with respect to the walking direction, we define the $Z$-axis of the world coordinate to coincide with the walking direction, whereas we define the $Y$ and $X$-axes as the vertical direction (i.e., the direction of the surface normal of the floor plane) and remaining orthogonal direction (i.e., medio-lateral direction), respectively.

For this purpose, we estimate the walking direction from the whole body CoG sequence. First, we compute the CoG of the point cloud of the extracted silhouette (Fig. 3 (c)), and then apply a moving average filter to the time-series of the CoGs to obtain the whole body CoG. We then obtain the walking direction using the line segment that connects the start and end points of the whole body CoG. We perform all subsequent calculations in this world coordinate system.

4.4 Foot Position Extraction

We extract left and right feet position sequences from the 3D feet point cloud of a person, which are used as input data for gait phase partitioning. First, we extract the top and bottom of the person point cloud, and then compute the person’s height as their difference in the vertical direction. Next, we extract the left and right feet centroid positions for each frame.

We extract the feet point cloud as a subset of the point cloud that is included within 10% of the body height from the bottom (Fig. 3 (c)), and compute the centroid of the point cloud for both feet. The left and right feet point clouds (magenta and cyan points in Fig. 3 (c), respectively) are extracted by dividing the point cloud for both feet into two parts based on the medio-lateral component of the centroid for both feet. The centroid of each point cloud is defined as the position of the right and left feet (magenta and blue circles in Fig. 3 (c), respectively). The $Z$-axis component of the centroid in the $t$-th frame is set to $z_{\text{CoG}}(t)$, where $f$ means a variable that represents either the left or right foot, that is, $f \in \mathcal{F} = \{ \text{left}, \text{right} \}$.

The above-mentioned straightforward method may, however, fail to estimate the foot position when the rear foot is not sufficiently visible because of self-occlusion (e.g., when walking with the legs crossing). Specifically, when the rear foot is not visible, the method forces the point cloud of the other visible foot to be partitioned into the left and right foot, and hence the rear foot position is suddenly located near the visible front foot position.

Moreover, outliers are more clearly observed in the medio-lateral component ($X$-axis) than the walking
the mis-detection from the normal walking movement, whereas time-series. We ased to the other foot position compared with the smoothed time-series, and determine that mis-
tifying the outliers based on the medio-lateral component walking. Therefore, we detect the mis-detection by iden-
ff

\[ Z \]

inidents typically change a great deal during walking, that is, \( Z \)-axis component (\( Z \)-axis) because the \( Z \)-axis components typically change a great deal during walking, that is, it is difficult to differentiate the sudden change because of mis-detection from the normal walking movement, whereas the \( X \)-axis component remains relatively still, even during walking. Therefore, we detect the mis-detection by identifying the outliers based on the medio-lateral component (Fig. 4 (a)). More specifically, we observe both raw time-series and smoothed time-series, and determine that mis-detection takes place when the raw time-series is clearly biased to the other foot position compared with the smoothed time-series.

First, we define the medio-lateral (\( X \)-axis) component of the foot position in the \( t \)-th frame as \( x^\text{cog}_f(t) \), and its smoothed version using a moving average filter as \( \hat{x}^\text{cog}_f(t) \), (see Fig 4 (a)). Given a set of time indices defined as \( T = \{1, \ldots, T\} \), where \( T \) is the length of the time-series, that with the correct estimation (i.e., inliers) is defined as

\[ T^\text{IL} = \{ t \mid t \in T, g_f(x^\text{cog}_f(t) - \hat{x}^\text{cog}_f(t)) < x_{\text{thresh}} \}, \]

where \( x_{\text{thresh}} \) is the threshold for outlier detection and \( g_f \) is a sign indicator for foot \( f \) to judge whether the foot position is biased to the other foot, which returns +1 and -1 for the left and right feet, respectively. We then recover the \( Z \)-axis components of the outliers by applying a cubic spline interpolation obtained from the inlier set \( T^\text{IL} \), and define the interpolated \( Z \)-axis component at the \( t \)-th frame as \( \hat{z}^\text{cog}_f(t) \). The final foot position in the \( t \)-th frame is obtained as

\[ z_f(t) = \begin{cases} \hat{z}^\text{cog}_f(t) & (t \in T^\text{IL}) \\ z^\text{cog}_f(t) & \text{(otherwise)} \end{cases} \]

5. Piecewise Linear Approximation Using DP [37]

We build our gait phase partitioning method on the piecewise linear approximation using DP [37]. The original method [37] aims at piecewise linear approximation for a two-dimensional (2D) polygonal curve; however, we apply it to the foot position sequence \( \{z_f(t)\} \). In this section, we briefly describe the method [37] for our problem setting to better understand the proposed method and make this paper self-contained. Additionally, for simplicity, we omit the subscript for the foot in this section. The piecewise linear approximation for the point sequence is defined by either of the following two problems:

1. **min-\( \epsilon \) problem**: approximation by a polygonal line with a given number of line segments \( M \) so that the total approximation error is minimized; and

2. **min-# problem**: approximation by a polygonal line with the minimum number of line segments so that the approximation error does not exceed the given maximum tolerance \( \Delta_{\text{tol}} \).

First, we describe the min-\( \epsilon \) problem. We redefine the foot position sequence as a point sequence in the 2D domain of time \( t \) and space \( z(t) \) as \( \mathcal{P}_{1:T} = \{p(1), \ldots, p(T)\} = \{(1, z(1)), \ldots, (T, z(T))\} \). Assuming that the \( m \)-th line segment ends at the \( t_m \)-th frame, a subset of the 2D point sequence \( \mathcal{P}_{t_m-1:t_m} = \{p(t_{m-1} + 1), \ldots, p(t_m)\} \) belongs to the \( m \)-th line segment. Naturally, the first line segment starts at the first frame, that is, \( t_0 = 0 \), whereas the last (\( M \)-th) line segment ends at the \( T \)-th frame, that is, \( t_M = T \). Given the line equation for a line segment,

\[ \hat{z}(t; a, b) = at + b, \]

where \( a \) and \( b \) are the inclination and intercept of the line segment, respectively, the approximation error for the \( m \)-th line segment by the least square is

\[ e^2(t_{m-1}, t_m) = \min_{a,b} \sum_{t_{m-1} + 1}^{t_m} (\hat{z}(t; a, b) - z(t))^2. \]

Consequently, the total minimum accumulation error \( E(M) \) over \( M \) line segments is derived as

\[ E(M) = \sum_{m=1}^{M} e^2(t_{m-1}, t_m). \]
of the 2D point subsequence up to the \(t\)-th frame, that is, \(P_{1,t} = \{p(1), \ldots, p(t)\}\) by \(m\) line segments. In the DP framework, note that state \((T, M)\) represents the original problem and hence is regarded as the goal state, whereas the state \((0, 0)\) is the initial state. Additionally, a path from the initial state \((0, 0)\) to the goal state \((T, M)\) contains a set of pairs of an end frame and a line segment index \(Q = \{(t_m, m) \mid m = 0, \ldots, M\}\), and hence it naturally corresponds to one approach to piecewise linear approximation with \(M\) line segments (see Fig. 5).

The remaining problem is how to obtain the optimal path that satisfies Eq. (4). For this purpose, we define the accumulation error \(D(t, m)\) at state \((t, m)\), that is, the minimal error given by the optimal piecewise linear approximation of the subsequence \(P_{1,t}\) by \(m\) line segments. After initializing the accumulation error at the initial state as \(D(0,0) = 0\), we then recursively update it for each state as

\[
t'(t, m) = \operatorname{argmin}_{t' \in T(t)} [D(t', m - 1) + e^2(t', t)], \quad (6)
\]

\[
D(t, m) = D(t', m, m - 1) + e^2(t'(t, m), t), \quad (7)
\]

where \(t'(t, m)\) is the time index of the previous state that minimizes the accumulation error \(D(t, m)\) up to the current state, and \(T(t)\) is a set of time indices of possible previous states that can be transitioned to the \(t\)-th frame, and is simply defined as \(T(t) = \{t' \mid t' < t\}\). Once the update reaches the goal state \((T, M)\), we recursively obtain the optimal path by backtracking with \(t'(t, m)\); for example, given the optimal solution to the end frame of the \(m\)-th line segment \(t_m\), the previous state is simply derived as \(t_{m-1} = t'(t_m, m)\).

Second, we describe the min-\# problem. Because this problem considers the maximum error tolerance \(\Delta_{\text{tol}}\) per line segment, the set of possible previous time indices \(T\) in Eq. (6) is redefined to satisfy the condition \(T(t) = \{t' \mid t' < t, e^2(t', t) < \Delta_{\text{tol}}\}\). We then scan the state in the last frame \(\{(T, m) \mid m = 1, \ldots\}\) and find the minimum \(M^\ast\) that can be reached from the initial state \((0,0)\). In practice, we achieve this by initializing \((T, m) = (1) \forall m\), setting a set of the number of line segments with non-negative cumulative cost \(\mathcal{M} = \{m \mid D(T, m) \neq -1\}\), and finding the minimum \(M^\ast = \min \mathcal{M}\). Once we obtain the goal state \((T, M^\ast)\), we obtain the optimal path similarly to the case of the min-\(\varepsilon\) problem.

6. Mutually Constrained Piecewise Linear Approximation with DP

6.1 Definition of States and Transitions for a Gait Cycle

As discussed in the Introduction, if we directly apply the above-mentioned piecewise linear approximation to each left and right foot independently, reasonable phase transitions may not be necessarily obtained (e.g., simultaneous swing phases for both the left and right leg take place). Therefore, we extend it to a mutually constrained piecewise linear approximation to guarantee a reasonable phase transition between the left and right legs.

More specifically, we define four states \(S = \{s_1, s_2, s_3, s_4\}\) for a gait cycle, as shown in Table 1, and also consider cyclic state transitions \(\mathcal{H} = \{(s_1 \rightarrow s_2), (s_2 \rightarrow s_3), \ldots, (s_4 \rightarrow s_1)\}\), as shown in Fig. 6. We then define a 2D discrete state space for DP as \(\Omega^\text{cycle} = \{(t, s) \mid t = 0, \ldots, T, s \in S\}\), whereas the conventional method defines the 2D discrete space of time and piecewise line index \(\Omega\) for each left and right foot independently. We then obtain the state transition by finding the optimal path in the state space \(\Omega^\text{cycle}\) (see Fig. 7 (b)), whereas the conventional method does this in the state space \(\Omega\) (see Fig. 5 (b)).

6.2 Phase-Dependent Piecewise Line Segments

The conventional piecewise linear approximation does not consider constraints on the piecewise line segments; however, we consider them for a more reasonable approximation. Specifically, the foot position moves during the swing phase, whereas it essentially stays still during the stance phase. Therefore, we modify the line segment equation to enforce this property:

![Fig. 5 Example of piecewise linear approximation.](image)

![Fig. 6 Gait state transition.](image)
function from the current state $s$ to the previous state defined as $h(s) = (s - 1) \mod 4$. For example, assuming a transition from the previous state $(t', s_3)$ to the current state $(t, s_4)$ for the left foot, the line fitting error $e^2_{\text{left}}(t', t)$ between the $t''$-th frame and $t$-th frame alone does not make sense because the stance phase has already started, not in the $t'$-th frame but at the beginning of state $s_2$ (let it be the $t''$-th frame). By taking this into account, the updated line fitting error $e^2_{\text{left}}(t'', t)$ can be computed by first cancelling (subtracting) the line fitting error $e^2_{\text{left}}(t', t)$ between the $t''$-th frame and $t'$-th frame, and then adding the line fitting error $e^2_{\text{left}}(t', t)$ between the $t''$-th frame and $t$-th frame.

Consequently, we need to store not only the time index $t'(t, s)$ of the optimal previous state but also the foot-dependent time index $t''(t, s)$ of the optimal previous phase.

To summarize, the foot-dependent time index $t''(t, s)$ follows from the optimal previous state if its transition is a keeping transition:

$$t''(t, s) = \begin{cases} t'(t(t', s), h(s)) & (h(s) \rightarrow s \in \mathcal{H}^\text{keep}_f) \\ t(\text{otherwise}) \end{cases},$$

(9)

where $\mathcal{H}^\text{keep}_f$ is a set of keeping transitions for foot $f$ defined in Table 2. Finally, we can compute the updated line fitting error $\tilde{e}^2_f(t', t, s)$ as

$$\tilde{e}^2_f(t', t, s) = e^2_f(t'(t, s), t) - e^2_f(t''(t, s), t').$$

(10)

6.4 Recursive Update

We can now define the recursive update for the mutually constrained piecewise linear approximation using the DP framework. Specifically, the time index $t'(t, s)$ of the optimal previous state and the cumulative error $D_f(t, s)$ are defined as

$$t'(t, s) = \arg \min_{t' \in \mathcal{T}(t, s)} \sum_f \{D_f(t', h(s)) + \tilde{e}^2_f(t', t, s)\},$$

(11)

$$D_f(t, s) = D_f(t'(t, s), h(s)) + \tilde{e}^2_f(t'(t, s), t, s),$$

(12)

where $\mathcal{T}(t, s) = \{ t' \mid t''(t', s) \leq t' \leq t''(t, s) < \Delta_{\text{min}} \forall f \}$ is a set of time indices of possible previous states, and $t''(t)$ and $t''(t)$ are defined as

$$t''(t) = \max(t - \Delta_{\text{max}}, 0),$$

(13)

$$t''(t) = \begin{cases} T - 1 & (t = T) \\ \max(t - \Delta_{\text{min}}, 0) & (\text{otherwise}) \end{cases}.$$  

(14)

Note that $\Delta_{\text{min}}$ and $\Delta_{\text{max}}$ denote the minimum and maximum time elapsed for each state, respectively, as described above. Additionally, note that the cumulative errors at the initial states are initialized as $D_f(0, s) = 0 \forall s$. Once we complete the update, we determine the optimal terminal state:

$$s^* = \arg \min_s \sum_{f \in F} D_f(T, s),$$

(15)
and then obtain the state transition by backtracking based on the stored time indices for the optimal previous states \( t'(t, s) \) similarly to the case of the conventional piecewise linear approximation.

7. Footprint Detection

7.1 Toe Edge Extraction on the Floor Plane

As addressed in the overview section, the first step to footprint detection is toe edge extraction on the floor plane. For this purpose, we first extract an edge image from an input RGB image using an edge detection method \[52\], which outputs the edge likelihood per pixel (see Fig. 8 (a)). We then convert the edge pixels in the RGB image to those in the depth image coordinate, and obtain a 3D edge point cloud by projecting the converted edge pixels to the world coordinate based on a depth value for each edge pixel (see Fig. 8 (b)). Thereafter, we set a 3D toe ROI at the \( t \)-th frame around the foot position on the floor plane \([\hat{x}_{f}^{\text{cog}}(t), 0, z_{f}(t)]^T\) as \( X_{f}^{\text{ROI}}(t) = \{ [x, y, z] | |x - \hat{x}_{f}^{\text{cog}}(t)| < \Delta x, |y| < \Delta y, |z - z_{f}(t)| < \Delta z \} \), where we experimentally set \( \Delta x = 20 \text{ cm}, \Delta y = 3 \text{ cm}, \) and \( \Delta z = 20 \text{ cm} \).

Next, the 3D toe edges in the ROI \( X_{f}^{\text{ROI}}(t) \) for each stance phase of each foot are voted on the floor plane image whose image resolution is 1 cm\(^2\)/pixel. We set a voting weight \( w \) for the edge intensity \( I \) using a type of soft thresholding with a hyperparameter \( \alpha \) as \( w = 1 - e^{-\alpha I} \), whose range is \([0, 1]\). Once we obtain the toe voting value at the \((i, j)\)-th pixel as \( \tilde{I}^{\text{vote}}(i, j) \), we then obtain its smoothed version using max pooling with distance attenuation:

\[
\tilde{I}^{\text{vote}}(i, j) = \max_{(p, q) \in \mathcal{N}(i, j)} \left( I^{\text{vote}}(p, q) e^{-\beta \sqrt{(i-p)^2+(j-q)^2}} \right),
\]

where \( \mathcal{N}(i, j) \) is the nearest region around \((i, j)\) and \( \beta \) is a hyperparameter that controls distance attenuation.

![Fig. 8 Toe edge extraction. In (a), the brighter the pixels, the higher the edge likelihood. In (b), cyan and magenta represent the right and left toe edge points, respectively.](image)

7.2 Footprint Estimation Using Ellipse Model Fitting

We estimate the position and orientation of a footprint by fitting the ellipse model to the smoothed voting image \( \tilde{I}^{\text{vote}} \). The ellipse model can be defined by five parameters \( \theta = \{a, b, i_0, j_0, \theta\} \), where \( a \) and \( b \) are the half lengths of the major and minor axes, respectively, \((i_0, j_0)\) is the ellipse center, and \( \theta \) is the inclination angle of the major axis, and its equation is

\[
f(i, j; \theta) = \left\{ \frac{(j - j_0) \cos \theta + (i - i_0) \sin \theta}{a} \right\}^2 + \left\{ \frac{(j - j_0) \sin \theta - (i - i_0) \cos \theta}{b} \right\}^2 - 1 = 0. \tag{17}
\]

Next, we introduce a set of pixels \( \mathcal{M}(\theta) \) that belongs to the ellipse with the parameters \( \theta \), and then we obtain the optimal parameter \( \theta^* \) by maximizing the summation of the votes as

\[
\theta^* = \argmax_{\theta} \sum_{(i, j) \in \mathcal{M}(\theta)} \tilde{I}^{\text{vote}}(i, j). \tag{18}
\]

8. Gait Feature Extraction

In this section, we introduce four gait features using the proposed method that we use for accuracy evaluation in the experiments.

1. Single support duration (SSD) [s]

The SSD is the time when either the left or right foot is in the stance phase while the other foot is in the swing phase. It is specifically the time duration of state \( s_1 \) and \( s_3 \) for the left and right feet, respectively (see Table 1 and Fig. 6).

2. Double support duration (DSD) [s]

The DSD is the time when both feet are in stance phases. It is specifically \( s_2 \) or \( s_2 \) in the case in which either the left or right foot is in front, respectively (see Table 1 and Fig. 6).

3. Toe angle (TA) [deg]

The TA is the orientation of the ellipse chosen to fit the footprint. The TA is 0 [deg] when it is parallel to the walking direction. Additionally, the TA is positive if it is in the lateral direction and vice versa (see Fig. 9 (a)).

4. Step length (SL) [m]

The SL is the difference between the front ends of the
elliptic models of two temporally adjacent left and right stance phases in the walking direction (see Fig. 9(b)).

9. Experiment

9.1 Setup

We collected data from 10 subjects: eight men and two women. We asked each subject to walk along a path in a normal manner. Moreover, to investigate the effectiveness of the proposed method for a pathological gait, we also asked each subject to mimic a right-side hemiplegia patient by shortening the SSDs of his/her right foot, by dragging his/her toe during the swing phase, and by positioning his/her toe outward.

We set up a data capturing system for experimental validation. We installed an RGBD camera (Microsoft Kinect v2) and three color cameras (PointGrey Flea3 FL3-U3-13E4C) with a lens (SPACECOM 3.5 mm) on a straight walking path of 6m, as shown in Fig. 10. The RGBD camera captured an RGB image with a 1,920 × 1,080 pixel-size and a depth image with a 512 × 424 pixel size at 30 fps, which were used as input data for the proposed method. By contrast, each color camera captured an RGB image with a 1,280 × 1,024 pixel size at 60 fps, which was used to prepare the ground-truth gait features. We installed one of the three color cameras on the side of the walking path and the other two on the ceiling.

We manually detected the heel strike and toe-off event to define the ground truth of the SSD and DSD using the side view camera. By contrast, we used the cameras installed on the ceiling to obtain the ground truth of the TA and the SL. For this purpose, we converted a raw captured image (Fig. 11 (a)) to a calibrated image (Fig. 11 (b)) using lens distortion correction and homography transformation, and then obtained the ground truth of the TA and SL by manually fitting the ellipse model to the foot. For the calibration purpose, we placed white markers at 50 cm intervals, and set the image resolution after the homography transformation to 1 cm²/pixel.

The average absolute error between the estimated gait features and the ground truth gait features was evaluated for each left and right foot. As a benchmark, we adopted a method proposed by Latorre et al. [45], which is based on skeletal information detected by Kinect, and evaluated the errors in the same manner, except for the TA because Latorre et al. did not propose a method for estimating the TA. The paired t-test was used to calculate the statistical significance probability between the proposed and benchmark method. The significance level was set to 5% in each feature.

9.2 Results

Data for a total of 50 steps (26 steps to the right and 24 steps to the left) and 161 steps (81 steps to the right and 80 steps to the left) were collected and analyzed for normal gait and mimicked hemiplegic gait, respectively.

We summarize the estimation accuracy of gait features in Table 3. Regarding the temporal parameters (i.e., SSD and DSD) of a normal gait, the estimation error of the proposed method was 0.04 s at most, and the laterality was 0.01 s. Additionally, regarding the spatial parameters, the SL estimation error of the proposed method was 0.03 m at most, and the laterality was 0.01 m. The TA estimation errors were 4.6 deg and 5.5 deg for right and left feet, respectively. There was no significant difference between the proposed method and the benchmark method for SSD, DSD, and SL.

By contrast, regarding the temporal parameters of the mimicked hemiplegic gait, the estimation error of the proposed method was 0.05 s at the maximum, and the laterality was 0.01 s. Moreover, the proposed method yielded a 0.06 s and 0.04 s smaller SSD and DSD than the benchmark method for the right side, which was the paralyzed side. Regarding the spatial parameters, the SL estimation error of the proposed method was 0.03 m at most, and the proposed method yielded a 0.04 m smaller SL error than the benchmark method. The right and left side TA errors were 5.7 and 3.3 deg, respectively. Both SSD and DSD had significantly smaller errors on the paralyzed side than the benchmark method. The error of SL was significantly smaller than that of the benchmark method on both the non-paralyzed
side and the paralyzed side.

9.3 Discussion

First, we discuss the difference in the estimation accuracy between the normal and mimicked hemiplegic gait. There was no significant difference for SSD, DSD, and SL between the proposed and benchmark method in the normal gait. However, the proposed method showed significantly smaller error than the benchmark method in mimicked hemiplegic gait in SSD, DSD, and SL especially on the paralyzed side. This is because pose estimation sequences obtained using Kinect may well fit the normal gait, but not the movement of paralyzed limbs. By contrast, the proposed method, which does not require training data, fits a variety of gait types well, including the hemiplegic gait used in the experiment.

In [53], the authors proposed a method for estimating temporal parameters using the IMU, and reported errors with respect to the method using marker-based motion capture. The results were reported as 12.9 ms for the swing time and 10.3 ms for the stance time. Although higher accuracy than that achieved using our method has been reported, we consider that it is difficult to maintain the same accuracy for people with disabilities because the authors’ approach is a learning-based method in which only healthy subjects are used.

In [54], the authors proposed a method to estimate the heel strike and toe-off timings while walking on a treadmill using marker-based motion capture data, and reported the error from the method using ground reaction force (GRF) plate data. The subjects were seven healthy people and 11 impaired people, including stroke survivors. In the healthy subjects, the average error was within one frame (16.7 ms) of the GRF events for both event timings. In the impaired subjects, the average error was within two frames (33.4 ms) of the GRF events, and the estimation error increased. Because the authors did not estimate the swing time or the stance time, it was difficult to compare their results with those of the proposed method. However, we consider that our method is superior in that the estimation accuracy can be maintained, even for impaired people.

Additionally, it is generally difficult to estimate a TA using a depth sensor such as Kinect because the depth difference between the foot and the floor plane is subtle; however, the proposed method achieved 5 deg accuracy approximately by incorporating a color edge accumulation framework. We consider the TA estimation accuracy to be meaningful because a therapist generally measures the joint angle with a resolution of 5 deg [55].

10. Conclusion

We proposed a novel gait phase partitioning method and footprint detection method using one RGBD sensor. The proposed method was effective for both a healthy and mimicked hemiplegic gait, and more accurate than the conventional skeleton-based method. Some issues remain in the proposed method.

First, regarding the gait phase partitioning algorithm, we approximated the swing phase using a single line segment, assuming that the foot swing speed was constant. It is possible that some people with disabilities may drag the toe at the initial swing phase and the swing speed may not be constant. Therefore, it is necessary to improve the algorithm to approximate the swing phase for some line segments according to the change in the swing speed.

Second, regarding the footprint detection algorithm, a carpet is required to reduce the luster to avoid detecting the edges of the person reflected on the floor. It is necessary to improve the toe edge detection method, which is robust to the floor material.

Additionally, because of the characteristics of the sensor, the footprint detection accuracy changes according to the distance from the sensor. It is also necessary to evaluate the appropriate measurement range in a clinical measurement. We will develop a gait analysis system with improvements to the above issues and proceed with application experiments in clinical settings.

Acknowledgments

We thank Maxine Garcia, PhD, from Edanz Group (https://en-author-services.edanz.com/ac) for editing a draft of this manuscript.

References

[1] H. Iwama, D. Muramatsu, Y. Makihara, and Y. Yagi, “Gait verification system for criminal investigation,” IPSJ Trans. on Computer Vision and Applications, vol.5, pp.163–175, Oct. 2013.
[2] I. Bouchrika, M. Goffredo, J. Carter, and M. Nixon, “On using gait in forensic biometrics,” Journal of Forensic Sciences, vol.56, no.4, pp.882–889, 2011.
[3] N. Lynnerup and P. Larsen, “Gait as evidence,” IET Biometrics, vol.3, no.2, pp.47–54, June 2014.
[4] C. Zhou, I. Mitsugami, and Y. Yagi, “Detection of elderly gait impairment by patch-gel,” IEEE Transactions on Electrical and Electronic Engineering, vol.10, no.S1, pp.S69–S76, April 2015.
[5] Y. Barak, R.C. Wagenaar, and K.G. Holt, “Gait Characteristics of Elderly People With a History of Falls: A Dynamic Approach,” Physical Therapy, vol.86, no.11, pp.1501–1510, Nov. 2006.

[6] Y.A. Grimbergen, M.J. Knol, B.R. Bloem, B.P. Kremer, R.A. Roos, and M. Munneke, “Falls and gait disturbances in huntington’s disease,” Movement disorders: official journal of the Movement Disorder Society, vol.23, no.7, pp.970–976, 2008.

[7] R. Baker, “Gait analysis methods in rehabilitation,” Journal of neurorehabilitation and rehabilitation, vol.3, no.1, p.4, 2006.

[8] M. Gabel, R. Gilad-Bachrach, E. Renshaw, and A. Schuster, “Full body gait analysis with kinect,” 2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society, pp.1964–1967, Aug. 2012.

[9] R.A. Clark, K.I. Bower, B.F. Mentiplay, K. Paterson, and Y.-H. Pua, “Concurrent validity of the microsoft kinect for assessment of spatiotemporal gait variables,” Journal of Biomechanics, vol.46, no.15, pp.2722–2725, 2013.

[10] R. Liao, Y. Makihara, D. Muramatsu, I. Mitsugami, Y. Yagi, K. Yoshiyama, H. Kazui, and M. Takeda, “A video-based gait disturbance assessment tool for diagnosing idiopathic normal pressure hydrocephalus,” IEEE Transactions on Electrical and Electronic Engineering, vol.15, no.3, pp.433–441, Feb. 2020.

[11] K. Aoki, T.T. Ngo, I. Mitsugami, F. Okura, M. Niwa, Y. Makihara, Y. Yagi, and H. Kazui, “Early detection of lower mmse scores in elderly based on dual-task gait,” IEEE Access, vol.7, pp.40085–40094, 2019.

[12] M. Patterson and B. Caulfield, “A novel approach for assessing gait using foot mounted accelerometers,” 2011 5th International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth) and Workshops, pp.218–221, IEEE, 2011.

[13] M. Boutaayamou, C. Schwartz, J. Stamatakis, V. Denoël, D. Maquet, K. Aoki, T.T. Ngo, I. Mitsugami, F. Okura, M. Niwa, Y. Makihara, Y. Yagi, and H. Kazui, “Early detection of lower mmse scores in elderly based on dual-task gait,” IEEE Access, vol.7, pp.40085–40094, 2019.

[14] A. Mannini, V. Genovese, and A.M. Sabatini, “Online decoding of hidden markov models for gait event detection using foot-mounted gyroscopes,” IEEE J. Biomed. Health Inform., vol.18, no.4, pp.1122–1130, 2013.

[15] J. Taborri, E. Scalona, S. Rossi, E. Palermo, F. Patanè, and P. Cappa, “Real-time gait detection based on hidden markov model: is it possible to avoid training procedure?,” 2015 IEEE international symposium on medical measurements and applications (MeMeA) proceedings, pp.141–145, IEEE, 2015.

[16] T. Seel, J. Raisch, and T. Schauer, “Imu-based joint angle measurement for gait analysis,” Sensors, vol.14, no.4, pp.6891–6909, 2014.

[17] T. Gujarathi and K. Bhole, “Gait analysis using imu sensor,” 2019 10th International Conference on Computing, Communication and Networking Technologies (ICCNT), pp.1–5, IEEE, 2019.

[18] B. Carse, B. Meadows, R. Bowers, and P. Rowe, “Affordable clinical gait analysis: An assessment of the marker tracking accuracy of a new low-cost optical 3d motion analysis system,” Physiotherapy, vol.99, no.4, pp.347–351, 2013.

[19] H. Luo and J. Luo, “Evaluating the intra-limb coordination during gait in hemiplegia,” 2018 IEEE International Conference on Cyborg and Bionic Systems (CBS), pp.612–615, IEEE, 2018.

[20] S. Sarkar, P.J. Phillips, Z. Liu, J.R. Vega, P. Grother, and K.W. Bowyer, “The humanoid gait challenge problem: data sets, performance, and analysis,” IEEE Trans. Pattern Anal. Mach. Intell., vol.27, no.2, pp.162–177, Feb. 2005.

[21] J. Han and B. Bhanu, “Individual recognition using gait energy image,” IEEE Trans. Pattern Anal. Mach. Intell., vol.28, no.2, pp.316–322, Feb. 2006.

[22] Y. Makihara, R. Sagawa, Y. Mukaigawa, T. Echigo, and Y. Yagi, “Gait recognition using a view transformation model in the frequency domain,” Proc. 9th European Conf. on Computer Vision, Graz, Austria, pp.151–163, May 2006.

[23] M. Hu, Y. Wang, Z. Zhang, D. Zhang, and J.J. Little, “Incremental learning for video-based gait recognition with lbp flow,” IEEE Trans. Cybern., vol.43, no.1, pp.77–89, Feb. 2013.

[24] M.D. Lewek, C.E. Bradley, C.J. Wutzke, and S.M. Zinder, “The relationship between spatiotemporal gait asymmetry and balance in individuals with chronic stroke,” Journal of applied biomechanics, vol.30, no.1, pp.31–36, 2014.

[25] A. Bobick and A. Johnson, “Gait recognition using static activity-specific parameters,” Proc. 14th IEEE Conference on Computer Vision and Pattern Recognition, pp.423–430, 2001.

[26] H.D. Yang and S.W. Lee, “Reconstruction of 3d human body pose for gait recognition,” International Conference on Biometrics, pp.619–625, 01 2006.

[27] G. Ariyanto and M.S. Nixon, “Marionette mass-spring model for 3d biometrics,” 2012 5th IAPR International Conference on Biometrics (ICB), pp.354–359, March 2012.

[28] H.-S. Fang, S. Xie, Y.-W. Tai, and C. Lu, “Rpme: Regional multi-person pose estimation,” ICCV, 2017.

[29] Z. Cao, G. Hidalgo, T. Simon, S.E. Wei, and Y. Sheikh, “Openpose: realtime multi-person 2d pose estimation using part affinity fields,” arXiv preprint arXiv:1812.08008, 2018.

[30] J. Li, C. Wang, B. Zhu, Y. Mao, H.S. Fang, and C. Lu, “Crowdpose: Efficient crowded scenes pose estimation and a new benchmark,” Proc. IEEE/ICVF Conference on Computer Vision and Pattern Recognition, pp.10863–10872, 12 2018.

[31] K. Sun, B. Xiao, D. Liu, and J. Wang, “Deep high-resolution representation learning for human pose estimation,” Proc. IEEE Conference on Computer Vision and Pattern Recognition, pp.5693–5703, 2019.

[32] R. Liao, C. Cao, E.B. Garcia, S. Yu, and Y. Huang, “Pose-based temporal-spatial network (ptsn) for gait recognition with carrying and clothing variations,” Proc. Chinese Conference on Biometric Recognition, Cham, pp.474–483, Springer International Publishing, 2017.

[33] I. Bouchrika and M.S. Nixon, “Model-based feature extraction for gait analysis and recognition,” International Conference on Computer Vision/Computer Graphics Collaboration Techniques and Applications, pp.150–160, 03 2007.

[34] R. Bellman and B. Kotkin, “On the approximation of curves by line segments using dynamic programming: ii,” tech. rep., RAND CORP SANTA MONICA CALIF, 1962.

[35] R. Bellman and R. Roth, “Curve fitting by segmented straight lines,” Journal of the American Statistical Association, vol.64, no.327, pp.1079–1084, 1969.

[36] T. Pavlidis and S.L. Horowitz, “Segmentation of plane curves,” IEEE Trans. Comput., vol.100, no.8, pp.860–870, 1974.

[37] A. Kolesnikov and P. Fränti, “Polygonal approximation of closed contours,” Scandinavian Conference on Image Analysis, pp.778–785, 06 2003.

[38] J.M. Jasiewicz, J.H. Allum, J.W. Middleton, A. Barriskill, P. Condie, B. Purcell, and R.C.T. Li, “Gait event detection using linear accelerometers or angular velocity transducers in able-bodied and spinal-cord injured individuals,” Gait & Posture, vol.24, no.4, pp.502–509, 2006.

[39] R.W. Selles, M.A.G. Formanoy, J.B.J. Bussmann, P.J. Janssens, and H.J. Stam, “Automated estimation of initial and terminal contact timing using accelerometers; development and validation in transfibial amputees and controls,” IEEE Trans. Neural Syst. Rehabil. Eng., vol.13, no.1, pp.81–88, March 2005.

[40] J.A. Blaya and H. Herr, “Adaptive control of a variable-impedance ankle-foot orthosis to assist drop-foot gait,” IEEE Trans. Rehabil. Eng., vol.12, no.1, pp.24–31, March 2004.

[41] R.W. Selles, M.A.G. Formanoy, J.B.J. Bussmann, P.J. Janssens, and H.J. Stam, “Automated estimation of initial and terminal contact timing using accelerometers; development and validation in transfibial amputees and controls,” IEEE Trans. Neural Syst. Rehabil. Eng., vol.13, no.1, pp.81–88, March 2005.

[42] J.A. Blaya and H. Herr, “Adaptive control of a variable-impedance ankle-foot orthosis to assist drop-foot gait,” IEEE Trans. Neural Syst. Rehabil. Eng., vol.12, no.1, pp.24–31, March 2004.

[43] J. Bae and M. Tomizuka, “Gait phase analysis based on a hidden markov model,” Mechatronics, vol.21, no.6, pp.961–970, 2011.
Y. Tang, Z. Li, H. Tian, J. Ding, and B. Lin, “Detecting toe-off events utilizing a vision-based method,” Entropy, vol. 21, no. 4, p. 329, 2019.

E. Auvinet, F. Multon, C.-E. Aubin, J. Meunier, and M. Raison, “Detection of gait cycles in treadmill walking using a kinect,” Gait & Posture, vol. 41, no. 2, pp. 722–725, 2015.

J. Latorre, R. Llorens, C. Colomer, and M. Alcaláiz, “Reliability and comparison of kinect-based methods for estimating spatiotemporal gait parameters of healthy and post-stroke individuals,” Journal of Biomechanics, vol. 72, pp. 268–273, 2018.

S. Jung and M.S. Nixon, “Heel strike detection based on human walking movement for surveillance analysis,” Pattern Recognition Letters, vol. 34, no. 8, pp. 895–902, June 2013.

M. Evans, S. Colyer, D. Cosker, and A. Salo, “Foot contact timings and step length for sprint training,” 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 1652–1660, IEEE, 2018.

F. Okura, S. Ikuma, Y. Makihara, D. Muramatsu, K. Nakada, and Y. Yagi, “Rgb-d video-based individual identification of dairy cows using gait and texture analyses,” Computers and Electronics in Agriculture, vol. 165, p. 104944, 2019.

M.A. Fischler and R.C. Bolles, “Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography,” Communications of the ACM, vol. 24, no. 6, pp. 381–395, 1981.

R. Kurban, F. Skuka, and H. Bozpolat, “Plane segmentation of kinect point clouds using ransac,” The 7th international conference on information technology, pp. 545–551, 2015.

J.-C. Perez and E. Vidal, “Optimum polygonal approximation of digitized curves,” Pattern recognition letters, vol. 15, no. 8, pp. 743–750, 1994.

P. Dollár and C.L. Zitnick, “Fast edge detection using structured forests,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 37, no. 8, pp. 1558–1570, Aug. 2015.

J.-D. Sui, W.-H. Chen, T.-Y. Shiang, and T.-S. Chang, “Real-time wearable gait phase segmentation for running and walking,” 2020 IEEE International Symposium on Circuits and Systems (ISCAS), pp. 1–5, IEEE, 2020.

J. Zeni, J. Richards, and J. Higginson, “Two simple methods for determining gait events during treadmill and overground walking using kinematic data,” Gait & Posture, vol. 27, no. 4, pp. 710–714, 2008.

D.C. Boone, S.P. Azem, C.-M. Lin, C. Spence, C. Baron, and L. Lee, “Reliability of goniometric measurements,” Physical therapy, vol. 58, no. 11, pp. 1355–1360, 1978.

Makoto Yasukawa received the M.S. degree in natural science from Fukushima University in 2016. He is currently pursuing the Ph.D. degree in information science at Osaka University. He was engaged in research on medical science at Biometrics Research Laboratories, NEC Corp. His research interests lie in the boundary between computer vision and rehabilitation, gait analysis, and biomechanics.

Yasushi Makihara received the B.S., M.S., and Ph.D. degrees in engineering from Osaka University in 2001, 2002, and 2005, respectively. He was a specially appointed assistant professor (full-time), assistant professor, and associate professor at the Institute of Scientific and Industrial Research, Osaka University, in 2005, 2006, and 2014, respectively. He is currently a professor at the Institute for Advanced Co-Creation Studies, Osaka University. His research interests are computer vision, pattern recognition, and image processing, including gait recognition, pedestrian detection, morphing, and temporal super-resolution. He is a member of IPSJ, IEICE, RSJ, and JSME.

Toshinori Hosoi received the B.S. and M.S. degrees in engineering from Osaka University in 1999 and 2001, respectively. He was engaged in research on pattern recognition at Media and Information Research Laboratories and Data Science Research Laboratories, NEC Corp. He is currently the research manager of the high-speed camera vision team at Biometric Research Laboratories, NEC Corp. His research interests are computer vision, pattern recognition, and medical information processing, including high-speed vision, object tracking, and agitated patient detection from biological signals. He is a member of IEICE, JAMI, and JMAI.

Masahiro Kubo received the B.S. and M.S. degrees from Yokohama National University in 1997 and 1999, respectively. He was a visiting scientist at Harvard University from 2007 to 2008 studying biological sensing technology. He was engaged in research on biological sensing devices at Green Platform Research Laboratories, NEC Corp. He is currently Director of the medical science technology group at Biometric Research Laboratories, NEC Corp. His research interests are computer vision, medical imaging, medical information processing, and medical solutions. He is a member of JMAI.

Yasushi Yagi received his Ph.D. degree from Osaka University in 1991. In 1985, he joined the Product Development Laboratory, Mitsubishi Electric Corporation, where he worked on robotics and inspections. He became a research associate in 1990, a lecturer in 1993, an associate professor in 1996, and a professor in 2003 at Osaka University. He was also Director of the Institute of Scientific and Industrial Research, Osaka University from 2012 to 2015, and the Executive Vice-President of Osaka University from 2015 to 2019. He is currently a professor at the Institute of Scientific and Industrial Research, Osaka University.