METHODS & TECHNIQUES

Integrated Analysis of Cell Shape and Movement in Moving Frame

Yusri Dwi Heryanto1,†, Chin-Yi Cheng1, Yutaka Uchida2, Kazushi Mimura3, Masaru Ishii2, Ryo Yamada1

1Unit of Statistical Genetics, Center for Genomic Medicine, Graduate School of Medicine, Kyoto University, Kyoto, Japan 606-8507, Japan.

2Department of Immunology and Cell Biology, Graduate School of Medicine and Frontier Biosciences, Osaka University, Osaka, 565-0871, Japan.

3Graduate School of Information Sciences, Hiroshima City University, Hiroshima, 731-3194 Japan.

†Authors for correspondence: (yusri.dh@gmail.com; yamada.ryo.5u@kyoto-u.ac.jp)

ABSTRACT

The cell movement and morphological change are two interrelated cellular processes. The integrated analysis is needed to explore the relationship between them. However, it has been challenging to investigate them as a whole. The cell trajectory can be described by its speed, curvature, and torsion. On the other hand, the 3-Dimensional (3D) cell shape can be studied by using a shape descriptor such as Spherical harmonic (SH) descriptor, which is an extension of Fourier transform in 3D space.

We propose a novel method using parallel-transport (PT) to integrate these shape-movement data by using moving frames as the 3D-shape coordinate system. This moving frame is purely determined by the velocity vector. On this moving frame, the movement change will influence the coordinate system for shape analysis. By analyzing the change of the SH coefficients over time in the moving frame, we can observe the relationship between shape and movement.

We illustrate the application of our approach using simulational and real datasets in this paper.

Summary statement

We developed a novel approach to integrate the analysis of cell shape and movement using moving frame.

KEYWORDS: cell shape, cell movement, moving frame, spherical harmonics, integrated analysis
INTRODUCTION

The cell movement and morphological change are highly integrated. They share many biological mechanisms controlled by the cytoskeleton, cell membrane, membrane proteins, and extracellular matrix (Friedl and Wolf, 2009; Ridley, 2003). Almost all forms of the cell active movement need forces that are generated from dynamic shape change (Lämmermann and Sixt, 2009). Moreover, the difference in the shapes and sizes of motile cells reflects their movement pattern (Keren et al., 2008; Lämmermann and Sixt, 2009).

The recent advances in 3D cell imaging and tracking have provided us an ability to collect the cell movement and morphological data simultaneously (Nketia et al., 2017; Maška et al., 2014; Meijering et al., 2012; Wang et al., 2008). It significantly facilitated researchers to study the dynamic interplay between shape and movement of the cell. Yet, it also introduces new challenges. First, we need the ability to objectively quantify the cell shape and movement. Without this quantification, we cannot measure any standardized form of measurement that allows statistical procedures and mathematical calculations. Second, we need a novel computational method to integrate these data. This method should quantify the change of shape and the dynamic of movement simultaneously.

A variety of movement quantification can be straightforwardly computed. Given the trajectories, we can measure the total trajectory length, the distance between start and end point, the speed, and the acceleration of the cell (Meijering et al., 2012). However, the cell trajectories usually suffer from noise, which may bias the results of some analyses. The smoothing procedures such as Gaussian Process (GP) (Mchutchon and Rasmussen, 2011) are necessary to address this problem.

The shape quantification is not as simple as movement quantification. Cells are 3D objects that have arbitrary spatial positions, directions, and scales in the 3D-space. These shapes may be variations of the same shape and should be recognized as the same one. Thus, the shape objects should be normalized and put in a common frame of reference to make shape representations invariant under these isometric transformations (i.e. translation, scale, and rotation). Normalization for translation is easy to deal by translating the object so that its center of mass is at the origin. The scale normalization can be done by scaling the object in such a way that the surface area or volume is equal to one. On the other hand, the normalization for rotation is hard and a subject of many studies. The most well-known approach is the principal component analysis (PCA) based approach (Shilane et al., 2004). Another approach is to transform each shape into a function and then calculate the rotation that minimizes the distance between the two functions (Makadia and Daniilidis, 2003; Makadia et al., 2004).

We propose a more natural way to align the rotation of a moving object such as a motile cell. The cells rarely have a clear axis of orientation. However, when moving, we can define the front face of the cell as the most forward part of the cell. A frame of reference where is the object direction becomes one of its axis and moves with the object along the trajectory is called moving frame (Hanson and Ma, 1995). This moving frame is constructed primarily using the velocity vector. Therefore, by using a moving frame as the reference frame, a change in the movement pattern will also change the shape quantification. Hence, this approach can bridge the analysis of shape and movement.
After normalization, we need a shape descriptor, which is a compact numerical representation of the 3D object shape. Spherical harmonic (SH) descriptor is a widely used descriptor to study the 3D cell shape (Shen et al., 2009; Ducroz et al., 2012; Du et al., 2013; Medyukhina et al., 2020). It is a spherical analogue of the 1-Dimensional Fourier series. It considers a surface as a function on the unit sphere, which can be represented as a set of unique coefficients. In shape-movement analysis, SH was used to analyze amoeboid cell motion (Ducroz et al., 2012; Du et al., 2013) and to perform shape classification of motile cells (Medyukhina et al., 2020).

In this paper, we developed a framework that generates, standardizes, and integrates the shape and movement data of the cells. This framework comprises trajectory smoothing using Gaussian Process, translation and scale normalization, rotation normalization using moving frame, and spherical harmonic decomposition for shape analysis. To illustrate our approach, we use both of simulated and real datasets.

RESULTS

Simulational dataset

The first step is to create a smooth trajectory from the observed data. Then, we realigned the shape on each timepoint by using moving frames which were created from the smooth trajectory. We extracted the shape-movement features from these realigned shape.

To validate these, we created a simulational dataset that consists of 250 datapoints. From these 250 datapoints, we selected 26 datapoints as observed data. The smooth trajectory and the shape-movement features that were extracted from these observed data should reflect the trajectory and features from the complete original data.

The validation of the trajectory smoothing

We compared the smooth trajectory obtained from the observation data with the complete trajectory (FIG. 1A). It is difficult to see any difference due to almost perfect reconstruction (The mean squared error = 1.203). The quality of the smooth trajectory degrades around t=0 and t=250. This is probably due to the Gaussian Processes had fewer data to process around the boundary (i.e. no data at \( t < 0 \) or \( t > 250 \)).

We constructed the PT moving frames on this smooth trajectory (FIG. 1B). Orientation normalization was performed using PT frames as the canonical frame of reference (FIG. 2). After reorientation, we observed the cell protrude its pseudopod toward the movement direction (Movie 2b).

The validation of the shape-movement features extracted from the observed data

The shape-movement features of the simulational cell is shown on FIG. 3. The features from the complete original data are also shown on the graph. The pattern that we obtained from observation data are similar to the complete data, even though the magnitude of the peak is different. These differences due to the data from observational data is smaller than the original data.
On the upper part of FIG. 3, the movement behavior of the cell is shown on the speed, curvature, and torsion graph. The bimodal graph of the speed graph indicates the accelerate-decelerate-accelerate-decelerate pattern. Meanwhile, the peak on the curvature and torsion graph indicates the change of the direction between $t = 100$ and $t = 130$.

In the middle part of FIG. 3, the global deformation patterns of the cell are shown as the eccentricity of the shape. Here, $E_{xy}$ and $E_{xz}$ changed as time progressed and had values more than one in the majority of the time. In contrast, the value $E_{yz}$ did not vary a lot from one. These finding indicate that the cell had ellipsoid shape with the longer axis on the direction of the movement. Furthermore, the relation between shape and movement can be seen on lower part of the FIG. 3. In this plot, the speed, the $E_{xy}$, and $E_{xz}$ are standardized for comparison purpose. We can easily identify the similarity of the bimodal pattern on the speed, the $E_{xy}$, and $E_{xz}$ plot.

**Real data**

For each cell in the real dataset, we aggregated the features from all timepoints and calculated its descriptive statistics measurements such as median, percentile, and median absolute deviation. These measurements can be readily used as the features for classification and visualization as shown in the latter paragraphs.

To explore and visualize the high-dimensional shape-movement features, we used UMAP to plot the features of the real dataset on 2-dimensional plot. The FIG. 4A and the FIG. 4B shows the 2D plot of the features from PT moving frame and the standard basis (i.e. no realignment). Qualitatively, the features obtained on the PT moving frame separate each group better than the features from the standard basis. From the FIG. 4A, we can see that the saline group is separated from the treated groups (i.e. GM-CSF, LPS, and PMA). Each of the GM-CSF, LPS, and PMA groups are also separated in different clusters.

Quantitatively, we performed K-nearest neighbors (KNN) classifier on the features from the PT moving frame and the standard basis to show the performance of our approach. The FIG. 5A show the mean and the standard deviation of the KNN classifier accuracy. Most of the accuracy scores on the PT moving frame are superior to the standard basis. By using one sample T-test, the accuracy difference is significant (i.e. p-value < 0.05) on $k = 15, 20, 25$ (FIG. 5B).

On FIG. 6, we plot the importance of each feature on the accuracy of KNN classifier. The most importance features come from the speed features which is the number of the peak on curvature graph, the number of the peak on torsion graph, the 75th percentile and median absolute deviation of the speed. Here, the shape features play a minor role to classify the sample. We can see the relationship between each feature from the Fig 7. We found some interesting observations from it. Torsion and curvature features are negatively correlated with the speed features. It suggests that the cells that are moving faster are inclined to move straight (i.e. rarely change their direction and low magnitude of direction change). We also found that the fast cells shape were more ellipsoid and had shape change rate higher than the slower cells.
**DISCUSSION**

In this proof-of-principle study, we developed a framework for an integrated analysis of the cell shape and movement data on moving frame. We illustrated the application of our framework on the simulated and real data. Our approach can identify the relationship between shape-movement features such as the similarity between the speed and cell eccentricity pattern on the simulational dataset. Then, we can utilize the shape-movement features that are extracted by our framework for a further analysis or modelling. To illustrate this, we used shape-movement features extracted from our framework to visualize and classify neutrophils treated with some stimulants. We showed our features could separate the groups of the stimulants.

The key contribution of our approach is the capability to couple the movement information with the shape data using moving frame. The moving frame has many applications in the sciences such as computer graphics (Hanson and Ma, 1995), medicine (Patil et al., 2015), and biology (Goriely et al., 2008). The moving frame has been applied to study the movement of the organism or cells (Crenshaw, 1993; Crenshaw and Edelstein-Keshet, 1993). In this study, we use the moving frame as a canonical reference frame of the shape. It makes shape analysis task independent to the rotation transformation. Also, its construction is easy and fast. The other approach such as PCA-based approach, rotation invariant shape descriptor, and estimation from spherical images use only shape information to align the rotation (Kazhdan et al., 2003; Shilane et al., 2004; Makadia et al., 2004). We can integrate the shape-movement information because the moving frame is constructed by the velocity vector. In brief, when the object change its movement (i.e. changed its velocity vector), the moving frame is simultaneously change. The moving frame is the basis for shape descriptor. Thus, the change in moving frame will ultimately affect the shape analysis.

We need a smooth trajectory to construct moving frames. In our study, we used Gaussian process (GP) as smoothing method. However, our approach is not restricted to this. We can also apply the other smoothing procedures such as spline smoothing (Eubank, 1999) or Savitzky-Golay filter (Savitzky and Golay, 1964). We chose GP because it can capture the uncertainty in movement. This data can be useful for further movement analysis. GP also more appropriate for our data because it does not need dense training data. The application of other smoothing methods should be studied in the future.

We used spherical harmonics (SH) coefficient as shape descriptor. The alternative descriptors are distribution-based (Osada et al., 2002), neural network-based (Fang et al., 2015), and wavelet-based descriptor (Laga et al., 2006). As SH descriptor, they need shape normalization. Thus, our approach also applies for these descriptors. The performance of the descriptors should also be tested. The limitation of SH is the shape must be topologically sphere (i.e. there are no holes on it). The shape that is not topologically sphere could not be mapped onto sphere using spherical parameterization. Despite the limitation, most of the cells analyzed were topologically sphere. We excluded the cells which did not meet this requirement.

We used our approach to explore the shape-movement features of the neutrophil which were stimulated by GM-CSF, LPS, and PMA. We showed that our approach could extract and quantify the cell shape and movement information. The quantification allows us to...
perform statistical procedures, mathematical modelling, or classification. These reagents stimulate immune cells (Faurschou and Borregaard, 2003; Kutsuna et al., 2004; McAvoy et al., 2011). Thus, we expect our features can separate these groups from the saline control group. Indeed, we found that the treated groups are separated on UMAP visualization. Interestingly, each of the GM-CSF, LPS, and PMA group has different cluster. This indicate each of the stimulants has different effects to the cell shape and movement. This finding is interesting to investigate further. For classification, we used KNN algorithm to measure how well the extracted shape-movement features captured the structure of shape-movement information on the dataset. The similar data points should close to each other. We showed that the shape-movement features that were obtained on moving frame is better than on the standard basis. Medyukhina et al. used the rotation-invariant SH classify migrating cells using Support Vector Machine (SVM) classifier (Medyukhina et al., 2020). However, the information about rotation is lost in the rotation-invariant SH (Kazhdan et al., 2003). Our approach preserved the rotation information (e.g. when the cell rotated, the moving frame also rotated).

We also measured the correlation between cell protrusion and movement. It is well known that one of the cell migration modes is by generating protrusion (reviewed on Yamada and Sixt (2019)). Yet, it is difficult to quantify the correlation. It is because of no standardized way to determine the cell front. By moving frame, we can solve this problem easily. Using our approach, we found that the fast cells have ellipsoid shape with one of its principal axes on the movement direction. This ellipsoid shape might because the cell is protruding but keeping the volume constant. It agrees with the experimental observation (Yamada and Sixt, 2019).

Despite the advantages, our integrated analysis approach have limitations. This approach need a high resolution shape data. The low resolution could create inaccurate trajectory due to imprecise calculation of cell center. We also need many data points (e.g preferably more than 10) to create smooth trajectory.

Taken together, we showed that our framework can provide standardized measurement of cell shape-movement information and integrates these data with the appropriate resolution and number of data. This ability helps to gain insight into the mechanobiological process of the cell. However, some limitations may apply to this work.

**METHODS**

**Overview**

The schematic diagram of our approach is shown in FIG. 8. The trajectory of the cell was smoothed using Gaussian Process regression. From this smooth curve, we constructed moving frames at each sample point and extracted the trajectory properties such as the speed, curvature, and torsion at each sample point. For each 3D object from each timepoint, we change the coordinate basis from the standard basis to the moving frame basis. In this moving frame, the cell direction become the new x-axis. We performed the SH decomposition on this new basis to obtain SH coefficients. By analyzing the change of these coefficients over time, we can observe the dynamic of shape change when it was moving. We performed the statistical analysis of the shape and movement features to study their relationship.
### Trajectory and Moving Frame of the Mass Center

**Gaussian Process smoothing**

We obtained the trajectory by calculating the mass center of the 3D object at every timepoint. The mass center of the object is defined as the arithmetic mean of their vertices. Thus, we had \((x(t), y(t), z(t))\) as the position of the mass center at the time \(t\). Then, we defined \(s_x = (x(1), \ldots, x(n))\), \(s_y = (y(1), \ldots, y(n))\), and \(s_z = (z(1), \ldots, z(n))\). For each sequence \(s_i; i = x, y, z\), we standardized the sequence so that it has mean zero and standard deviation one. From hereafter, the subscript \(i\) refers to one of the element of \(x, y, z\), unless otherwise stated.

We modeled each of the sequence as \(s'^*_i(t) = f_i(t) + \epsilon\) where \(s'^*_i\) is a standardized sequence. The function \(f_i(t)\) is an underlying smooth function and \(\epsilon\) is a Gaussian noise with mean zero and variance \(\sigma^2\).

The function \(f_i(t)\) can be approximated by Gaussian Process (GP) Regression (Rasmussen and Williams, 2006). In the GP regression, we have mean function and kernel function as hyperparameters. The mean function is a function that calculate the mean at any point of the input space and the kernel function specify the covariance between pairs of input. As our sequence had zero mean, we chose the zero mean function. And because we need a smooth function, we used Squared-Exponential (SE) kernel function.

The SE kernel itself has two parameters: (1) the lengthscale parameter that determines how far the influence of one sample point to its neighbor points, (2) the variance parameter that determines the mean distance from the function’s mean.

We used the function \(f_i(t)\) to interpolate new data points between two consecutive elements of sequence \(s'^*_i(t)\) and \(s'^*_i(t+1)\). Thenumber of new data points determines the smoothness of the new sequence.

Next, we performed the inverse transform of standardization by multiply \(s'^*_i\) by the standard deviation of \(s_i\), then add it with the mean of \(s_i\) to produce a near-smooth sequence \(r_i\). Then, we defined \(r(t) = (r_x(t), r_y(t), r_z(t))\) as the smooth trajectory.

**Parallel transport moving frame**

One of the main ideas in our approach is a frame of reference which is moving along with the curve and telling us the main directions of the movement (Fig 9). We made this frame of reference using parallel-transport (PT) moving frame algorithm from Hanson and Ma (1995). In brief, we calculated a tangent vector \(T_i\) for each point \(i\) on the curve. Then, set an initial normal vector \(N_0\) which is perpendicular to the first tangent vector \(T_0\). For each sampled
point, we calculate the cross product of \( \mathbf{U} = \mathbf{T}_i \times \mathbf{T}_{i+1} \). If the length \( \| \mathbf{U} \| = 0 \) (i.e. both vectors are parallel) then \( \mathbf{N}_{i+1} = \mathbf{N}_i \). Otherwise, to obtain \( \mathbf{N}_{i+1} \), we rotate \( \mathbf{N}_i \) around the vector \( \mathbf{U} \) by the angle \( \theta = \mathbf{T}_i \cdot \mathbf{T}_{i+1} \) using the rotation matrix \( \text{Rot}(\mathbf{U}, \theta) \) defined on Hanson and Ma (1995). After we obtain \( \mathbf{T}_i \) and \( \mathbf{N}_i \) for each timepoint \( i \), we can calculate the vector \( \mathbf{B}_i = \mathbf{T}_i \times \mathbf{N}_i \) to construct moving frames. The complete algorithm is shown in Algorithm 1.

Algorithm 1: Parallel transport moving frame algorithms

**Input:** 1. Tangent vectors \( \mathbf{T}_i, i = 0, 1, ..., n \);  
2. An initial normal vector \( \mathbf{N}_0 \perp \mathbf{T}_0 \)  

**Output:** Vectors \( \mathbf{N}_i \) and \( \mathbf{B}_i \), \( i = 0, 1, ..., n \) where \( \mathbf{T}_i, \mathbf{N}_i, \mathbf{B}_i \) are orthogonal

for \( i = 0 : (n-1) \) do

\[ \mathbf{U} = \mathbf{T}_i \times \mathbf{T}_{i+1}; \]

if \( \| \mathbf{U} \| = 0 \) then

\[ \mathbf{N}_{i+1} = \mathbf{N}_i; \]

else

\[ \mathbf{U} = \mathbf{U} \backslash \| \mathbf{U} \|; \]

\[ \theta = \arccos(\mathbf{T}_i \cdot \mathbf{T}_{i+1}); \text{ // } 0 \leq \theta \leq \pi \]

\[ \mathbf{N}_{i+1} = \text{Rot}(\mathbf{U}, \theta) \times \mathbf{V}_i; \]

end

end

\[ \mathbf{B}_i = \mathbf{T}_i \times \mathbf{N}_i, i = 0, 1, ..., n; \]

Orientation normalization

We change the coordinate basis from the standard basis to the PT moving frame basis using simple linear algebra transformation:

\[
\mathbf{v}^* = \begin{bmatrix} T_x & N_x & B_x \\ T_y & N_y & B_y \\ T_z & N_z & B_z \end{bmatrix}^{-1} \mathbf{v}
\]

where \( \mathbf{v}^* \) and \( \mathbf{v} \) are the coordinates of the vertex in moving frame and standard basis respectively. The subscript \( x, y, \) and \( z \) here are the first, second, and third component of the vectors \( \mathbf{T}, \mathbf{N}, \) and \( \mathbf{B} \).
Speed, curvature, and torsion

From the smooth trajectory, we can extract the trajectory features such as speed, curvature, and torsion as defined on Patrikalakis and Maekawa (2009). In summary, speed is the distance traveled per unit of time. Curvature measures the failure of a trajectory to be a straight line, while torsion measures the failure of a trajectory to be planar. Taken together, the curvature and the torsion of a space curve are analogous to the curvature of a plane curve.

Spherical Harmonic Decomposition

After orientation normalization, shapes were decomposed Spherical Harmonic (SH) transform. To perform SH decomposition, we need to map the object surface to the unit sphere. Before it, we normalize the volume of all objects to one.

Spherical parameterization

We used the mean-curvature flow spherical parameterization method from Kazhdan (Kazhdan et al., 2012) to maps the object surface to a unit sphere. The result of spherical parameterization is a continuous and uniform mapping between a point on the object surface and a pair of the latitudinal-longitudinal coordinate \((\theta, \phi)\) on a unit sphere:

\[
v(\theta, \phi) = (x(\theta, \phi), y(\theta, \phi), z(\theta, \phi))
\]

where \((x(\theta, \phi), y(\theta, \phi), z(\theta, \phi))\) is the Cartesian vertex coordinates.

Spherical Harmonic expansion

On the unit sphere, each of the \(x(\theta, \phi), y(\theta, \phi), z(\theta, \phi)\) can be approximated using the real form spherical harmonic (SH) series:

\[
\begin{align*}
x(\theta, \phi) &\approx \sum_{l=0}^{l_{max}} \sum_{m=-l}^{m_{max}} C_x(l,m) Y^m_l(\theta, \phi) \\
y(\theta, \phi) &\approx \sum_{l=0}^{l_{max}} \sum_{m=-l}^{m_{max}} C_y(l,m) Y^m_l(\theta, \phi) \\
z(\theta, \phi) &\approx \sum_{l=0}^{l_{max}} \sum_{m=-l}^{m_{max}} C_z(l,m) Y^m_l(\theta, \phi)
\end{align*}
\]

\(Y^m_l(\theta, \phi) = \begin{cases} 
\sqrt{2} K^{-m}_l p^{-m}_l (\cos(\theta)) \sin(-m\phi) & \text{if } m < 0 \\
K^0_l p^0_l (\cos(\theta)) & \text{if } m = 0 \\
\sqrt{2} K^m_l p^m_l (\cos(\theta)) \cos(m\phi) & \text{if } m > 0
\end{cases}
\]
where \( P^m_l(\theta,\phi) \) is the associated Legendre polynomial and \( l_{\text{max}} \) is the maximum degree of the SH expansion we want.

The coefficient of degree \( l \), order \( m \), \( C(l,m) \) can be obtained using standard least-square estimation. Using \( x(\theta,\phi) \) as an example, assume that the number of vertices is \( n \) and \( x_i = x(\theta_i,\phi_i) \). We need to find the coefficients \( C_x = (c_1, c_2, \ldots, c_k)^T \) where \( c_j = C_x(l,m) \).

The index \( j \) for \( l \in (0,\ldots, l_{\text{max}}), m \in (-l,\ldots,l) \) is obtained from the equation \( j = l^2 + l + m + 1 \). We can obtain the coefficients by solving equation 6.

\[
\begin{pmatrix}
y_{1,1} & y_{1,2} & \cdots & y_{1,k} \\
y_{2,1} & y_{2,2} & \cdots & y_{2,k} \\
\vdots & \vdots & \ddots & \vdots \\
y_{n,1} & y_{n,2} & \cdots & y_{n,k}
\end{pmatrix}
\begin{pmatrix}
c_1 \\
c_2 \\
\vdots \\
c_k
\end{pmatrix}
= 
\begin{pmatrix}
x_1 \\
x_2 \\
\vdots \\
x_n
\end{pmatrix}
\]

(6)

where \( y_{i,j} = Y^m_j(\theta,\phi), \ j = l^2 + l + m + 1, \) and \( k = (l_{\text{max}} + 1)^2 \). After obtaining the coefficients for each of \( C_x, C_y \), and \( C_z \), we can bundle it into one feature vector of the shape, \( C = (C_x, C_y, C_z) \).

Shape characteristics measures

Each coefficient of the expansion retains a shape information corresponding to a particular spatial frequency. The increasing degree of \( l \) describes the finer scales of shape information. The direction of shape changes can be detected in each of three sets of coefficients, especially \( C_x(l=1,m=1) \) for deformation in the \( x \)-direction, \( C_y(1,0) \) for \( y \)-direction, and \( C_z(1,-1) \) for \( z \)-direction. If \( C_x(1,1) = C_y(1,0) = C_z(1,-1) \) and the other coefficients are zero then the object is a perfect sphere. Based on these three coefficients, we defined three eccentricity index:

\[
E_{xy} = \frac{C_x(1,1)}{C_y(1,0)}
\]

(7)

\[
E_{xz} = \frac{C_x(1,1)}{C_z(1,-1)}
\]

(8)

\[
E_{yz} = \frac{C_y(1,0)}{C_z(1,-1)}.
\]

(9)
$E_{xy}$, $E_{xz}$, and $E_{yz}$ are the measurements of how much the object deviates from being sphere in $xy$, $xz$, and $yz$ plane, respectively (Fig. 10). We can exclude the $E_{yz}$ because the calculation of $E_{xy}$ and $E_{xz}$ already contains all of the eccentricity information.

The difference between shape $i$ and $j$, $d(i,j)$, can be calculated using any distance metrics intended for real-valued vector spaces. The most common is the $L_2$ norm distance:

$$d(i,j) = \| C_i - C_j \|_2$$

where $C_i$ and $C_j$ are SH coefficients for shape $i$ and $j$, respectively. Here, we calculated the rate of shape change which is defined as $d(t,t+1)$ where $t,t+1$ are two consecutive timepoints.

Statistical analysis

For each cell in the real dataset, we calculated the median, median absolute deviation, 25th percentile, and 75th percentile of the speed, curvature, torsion, shape change rate, $E_{xy}$, and $E_{xz}$ from all timepoints. We also counted the number of peaks from the curvature and torsion graph. These values are used as features (26 features in total). Then, we standardized all these features to have mean zero and standard deviation one. To show the correlation from each features, we calculated Kendall rank correlation coefficient.

We use UMAP (McInnes et al., 2018) for visualization in a 2D plot. The UMAP parameters that we used: number of neighbors = 20, number of components = 2, minimum distance = 0.1, and the Euclidean distance as metric.

K-nearest neighbor (KNN) classifier was trained on the shape-movement features on the PT moving frame and on the standard basis to show the performance of our approach. We varied the hyper-parameter $k = 3, 5, 10, 15, 20, 25$. A stratified 10-fold cross-validation was used to give a set of 10 accuracy scores for each hyper-parameter $k$. The accuracy score is the fraction of the correct classifications. From each iteration of cross-validation, we calculated the difference of accuracy scores between these two approaches. The One sample T-test was performed to test whether the accuracy difference was significantly different from zero or not.

Datasets

Simulational dataset

We created a 3D cell object that moves along a path using open-source software Blender (Community, 2018). First, we create an icosphere object and a path for the object to follow. The speed of the object can be controlled in Blender by adjusting the position curves over time on the Graph Editor menu. Then, we manually protruded the pseudopodia along the object movement. The cell surface texture was produced by adding clouds texture and random noise. We rendered the 3D objects using Eevee render engine in Blender.
The dataset consists of 250 time points and a 3D object at each time point. The 3D objects were saved as triangular mesh objects. In our simulation, the cell moves in the accelerate-decelerate-accelerate-decelerate pattern. The cell starts from a near-spherical shape, protruding a pseudopod when accelerating, and back to near-spherical shape when decelerating (Movie 1, Movie 2a). The volume of the cell is fixed to be one. We chose the shape object at timepoint \( t = \{1, 10, 20, \ldots, 250\} \) as the observed data. The rest of the data were used as the holdout data. Using simulational dataset, we want to verify whether the features extracted from the complete data on can be approximated using the features extracted from the smooth trajectory of the observed data. The true trajectory of the cell is defined as the center of the mass of the cell at each of 250 timepoints. This dataset are available at this paper source code github repository.

Real dataset

The real dataset consists of 3D objects from microscope images of neutrophils. We isolated the neutrophils from the bone marrow of LysM-EGFP mice. Erythrocytes were excluded from harvested bone marrow cells using ACK lysing buffer and density gradient centrifugation (800xG for 20 minutes) using 62.5 percent percoll. The isolated neutrophils were mixed with collagen with \( 3 \times 10^4 \) cells/µl. Next, we dropped collagen-cell mixture (10 ul) on the glass bottom dish. After the collagen-cell mixture turned into a gel, we add culture medium to the dish. The dish was incubated at 37 °C for 2 hours. Before the imaging, the culture medium was replaced with the imaging medium. After the cells were stimulated with Granulocyte- Macrophage Colony-Stimulating-Factor (GM-CSF) 25 ng ml\(^{-1}\), Lipopolysaccharide (LPS) 10 µg ml\(^{-1}\), or (Phorbol 12-myristate-13-acetate (PMA) 1 µg ml\(^{-1}\), we immediately performed imaging for 90 minutes at 1-minute intervals using two-photon excitation microscope (Nikon A1R MP). We used lens Nikon X20 (Apo LWD 20X/0.95 WI \( \lambda \) S), excitation wave length 900 nm, and xy-spatial resolution 0.5 µm. The imaging was performed at 45 µm (15 stacks) in 3 µm steps in the Z-axis direction. Then, we constructed the 3D mesh shape object using a method from Cheng et al. (2020).

We analyzed the cells that were captured in a minimal 6 consecutive timepoints (i.e. 233, 398, 293, and 166 cells in saline, GM-CSF, LPS, and PMA group, respectively).

All animal experiments were carried out according to the guidelines approved by the Osaka University Institutional Animal Care and Use Committee.

Method Implementation

The above method was developed as Julia Programming Language package under MIT license (Bezanson et al., 2017). The package can be downloaded using Julia from the Github repository.

We used GaussianProcesses.jl package for the Gaussian Process smoothing (Fairbrother et al., 2018). For SH coefficient calculation, we used Julia wrapper for the GNU Scientific Library (GSL) (Galassi, 2009).

For the SE kernel parameter, we set the lengthscale= \( e^{1.0} \) and variance = 1.0. We note that the choice of the parameters is not necessarily optimal, but it gives good modeling results in our simulation. For the SH decomposition, we used \( l_{max} = 6 \).
Acknowledgements

We thank for Hironori Shigeta and Shigeto Seno of the Graduate School of Information Science and Technology, Osaka University, for the support in 3D mesh object creation.

Competing interests

The authors declare no competing or financial interests.

Contribution


Funding

This study is funded by Core Research for Evolutional Science and Technology (CREST), Japan with grant numbers JPMJCR1502 and JPMJCR15G1.

Data availability

The source code and simulational dataset are available at https://github.com/yusri-dh/MovingFrame.jl. The real dataset are available upon request.

Supplementary

Movie 1

Movie 1. The simulational cell manually created using Blender.

Movie 2a

Movie 2a. The simulational cell shape before reorientation. The centers of mass for each timepoints are normalized to origin.

Movie 2b

Movie 2b. The simulational cell shape after reorientation. The centers of mass for each timepoints are normalized to origin.
REFERENCES


**Figures**

**A: Trajectory smoothing**

**B: PT Moving Frame**

**Fig. 1.** Trajectory of simulated cell. (A) The Gaussian Process reconstructed the original trajectory from the observation. (B) The vectors $\mathbf{T}$ together with the vectors $\mathbf{N}$ and $\mathbf{B}$ construct moving frames for each point on the curve.
Fig. 2. Reorientation of shape using PT moving frame. (A) Before the reorientation, we use the standard basis (e.g. X;Y;Z) to describe the shape of the 3D object. (B) After the reorientation, the moving frame becomes a canonical coordinate system to describe the shape. The moving frame consists of the tangent vector T (red) which is parallel to the direction of the movement and two orthonormal vectors N and B (green and blue, respectively).
Fig. 3. The extracted shape and movement features from the smooth trajectory that is inferred from the observed data (label: inferred) and from the complete original data (label: original). (A,B,C) The movement features of the cell are characterized by its speed, curvature, and torsion, respectively. (D,E,F) The global shape features of the cell is represented by the ratio of spherical harmonic coefficients. The pattern of the features that is extracted from the smooth trajectory are similar to the features from the complete data. (g) The relation between shape and movement of the cell can be seen in this graph. Here, we standardized the graph of speed, $E_{xy}$, and $Exz$ for easy comparison.
Fig. 4. The Shape-Movement features embedding on 2D space using UMAP. We utilized UMAP to visualize the high-dimensional shape-movement features on (A) PT moving frame and on (B) Standard basis.
Fig. 5. The analysis of shape-movement features on PT moving frame. (A) The comparison of accuracy of KNN classifier across the 10-folds cross-validation trained on PT moving frame and on standard basis. (B) The accuracy difference of KNN classifier on PT moving frame and on standard basis. The one sample t-test showed that the difference was significantly different from zero for $k = 15; 20; 25$.
Fig. 6. The permutation importance plot that shows the importance of each feature. We found that the movement features was dominantly selected as important features.
Fig. 7. The correlation matrix of each features.
Fig. 8. The schematic diagram of integrated analysis of shape and movement in moving frame.

\[
\begin{align*}
&\sum_{l=1}^{l_{\text{max}}} \sum_{m=-l}^{l} C_x(l, m) Y_l^m(\theta, \phi) \\
&\sum_{l=1}^{l_{\text{max}}} \sum_{m=-l}^{l} C_y(l, m) Y_l^m(\theta, \phi) \\
&\sum_{l=1}^{l_{\text{max}}} \sum_{m=-l}^{l} C_z(l, m) Y_l^m(\theta, \phi)
\end{align*}
\]
Fig. 9. The construction of PT Moving Frame on each timepoints. The moving frames is moving with the curve and telling us the directions of the object movement.
Fig. 10. The shape eccentricity calculated using SH coefficients. For examples, figure (A) have eccentricity: $E_{xy} = 2; E_{xz} = 2; E_{yz} = 1$, (B): $E_{xy} = 1=2; E_{xz} = 1; E_{yz} = 2$, and (C): $E_{xy} = 1; E_{xz} = 1=2; E_{yz} = 1=2$. 
Movie 1. The simulational cell manually created using Blender.

Movie 2a. The simulational cell shape before reorientation. The centers of mass for each timepoints are normalized to origin.

Movie 2b. The simulational cell shape after reorientation. The centers of mass for each timepoints are normalized to origin.