TR－H－031

A Theory for Cursive Handwriting Based on the Minimization Principle

Yasuhiro WADA
Mitsuo KAWATO

1993. 9. 28

ATR人間情報通信研究所

〒619-02 京都府相楽郡精華町光台2-2　☎07749-5-1011

ATR Human Information Processing Research Laboratories

2-2, Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-02 Japan

Telephone: +81-7749-5-1011
Facsimile: +81-7749-5-1008

A Theory for Cursive Handwriting Based on the Minimization Principle

Yasuhiro WADA Mitsuo KAWATO

ATR Human Information Processing Research Laboratories,
2-2 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-02, Japan

Abstract

We propose a trajectory planning and control theory which provides explanations at the computational algorithm, representation, and hardware levels for continuous movement such as connected cursive handwriting. The hardware is based on our previously proposed forward-inverse-relaxation neural network (Wada and Kawato, 1993). Computationally, the optimization principle is the minimum torque-change criterion. At the representation level, hard constraints satisfied by a trajectory are represented as sets of via-points extracted from handwritten characters. Accordingly, we propose a via-point estimation algorithm that estimates via-points by repeating the trajectory formation of a character and the via-point extraction from the character. It is shown experimentally that for movements with a single via-point target, the via-point estimation algorithm can assign a point near the via-point target. Good quantitative agreement is found between human experimental data and the trajectories generated by the theory.

Acknowledgment:
We are grateful to Y.Koike for allowing us to use his measured human data. We would like to thank Drs. C.G.Atkeson, S. Schaal, and E.L.Saltzman for discussing and reading earlier versions of the manuscript. We wish to express our thanks to Dr. Y.Tohkura of ATR Human Information Processing Research Laboratories for his encouragement. This work was supported by a Human Frontier Science Project Grant to M.K.

1. Introduction

Handwriting production is an attractive subject for human motor control studies. When one produces cursive handwriting, in the central nervous system (CNS) a symbol that represents a character must be transformed into a motor command stream. This transformation process raises several questions. How can the CNS represent a character symbol for producing a handwritten letter? Is there an intermediate representation for handwriting? By what principle can motor planning be made or motor commands be produced?

In reaching movements, trajectory formation is an ill-posed problem because the hand can move along an infinite number of possible trajectories from the starting to the target point. However, humans can effortlessly and consistently move an arm between two targets along one of an infinite number of trajectories. Therefore, the brain should be able to compute a unique solution by imposing an appropriate criterion to the ill-posed problem. For example, a smoothness performance index can be introduced to solve this problem. Flash and Hogan (1985) proposed a mathematical model, the *minimum-jerk model*. They proposed that the trajectories followed by the subjects' arms tended to minimize the integral of the square of the jerk (rate of change of acceleration) of the hand position in Cartesian coordinate space, integrated over the entire movement. Their proposed performance index is the following quadratic measure:

$$C_J = \int_0^{t_f} \left\{ \left(\frac{d^3 X}{dt^3} \right)^2 + \left(\frac{d^3 Y}{dt^3} \right)^2 \right\} dt \tag{1}$$

Here, (X, Y) are Cartesian coordinates of the hand and t_f is the movement time. Their model is based solely on the kinematics of movement, hence, it is independent of the dynamics of the musculoskeletal system.

On the other hand, based on the idea that the objective function must be related to dynamics, Uno, Kawato and Suzuki (1989) proposed the *minimum torque-change criterion*, which accounts for both the desired trajectory determination and the dynamics of trajectory control. The computational theory postulates that the trajectory of the human arm is determined so as to minimize the time integral of the square of the rate of torque change. The following is a quadratic measure of performance:

$$C_T = \int_0^{t_f} \sum_{j=1}^{M} \left(\frac{d\tau^j}{dt} \right)^2 dt \qquad (2)$$

where τ^j is the torque generated by the j-th actuator of M actuators, and t_f is the movement time. It is apparent that this objective function is critically related to arm dynamics. For movements between a pair of targets just in front of the body, predictions by both computational models were close to the experimental data. However, the trajectories predicted by the minimum torque-change model were quite different from the minimum-jerk model in four behavioral situations, where it was found that the minimum torque-change model predicted the actual data better than the minimum-jerk model (Uno, Kawato and Suzuki 1989, see also Kawato 1994 for review of recent data).

Furthermore, Uno, Suzuki and Kawato (1989) have proposed a minimum muscle-tension-change model in which the objective function is the sum of the square of the rate of change of muscle tension, integrated over the entire movement. It is a biologically more plausible model than the minimum torque-change model. If the joint torque were to be generated by only one muscle, and if its moment arm were constant regardless of the joint angle and the same for all the different muscles, then the minimum muscle-tension-change model would be identical to the minimum torque-change model. However, joint torque is generated by a number of muscles, muscle moment arms are different, and moment arms do depend on joint angles. Thus, the minimum muscle-tension-change model is quite different from the minimum torque-change model.

Recently, the minimum muscle-tension-change model has been extended to the minimum motor-command-change model (Kawato 1992), which solves the enormous excess degree of freedom in the CNS by the smoothness principle in the state space of the CNS. It seems more plausible to impose the smoothness constraint at the CNS level than at the peripheral level.

Regarding the hardware level, the unique trajectory that yields the best performance of Eq. (1) is readily computed by applying the Euler-Lagrange equation. Several hardware models that can compute minimum-jerk trajectories have been proposed using recurrent neural networks (Jordan 1989; Massone and Bizzi 1989; Hoff and Arbib 1992). On the other hand, because the dynamics of the human arm is nonlinear, finding a unique trajectory based on the minimum torque-change model and the minimum muscle-tension-change model is a nonlinear optimization problem. Thus, it is a rather difficult optimization problem. To generate a trajectory based on the minimum torque-change model, Kawato et al. (1990) proposed the cascade neural network, which is a cascade structure of the forward dynamics model (FDM). Conversely, a neural network model for the minimum torque-change criterion that uses the inverse dynamics model (IDM) was proposed by Nakamura et al. (1990). There are several criticisms of these neural networks: (1) their spatial representation of time, (2) back propagation is essential, and (3) they require too many iterations. Therefore, we have proposed a new algorithm and hardware model, *FIRM(Forward-Inverse Relaxation Model)* for trajectory formation, which uses both a FDM and an IDM (Wada and Kawato 1993). This model can be implemented as a biologically plausible neural network.

Computational theories, algorithm and hardware models in reaching movement have been intensively studied and have become quite advanced. We wish to apply some of these ideas to handwriting. A variety of different handwriting models (Hollerbach 1981; Morasso and Mussa Ivaldi 1982; Edleman and Flash 1987) have been proposed. Hollerbach proposed a handwriting model based on oscillation theory. Coupled oscillations in horizontal and vertical directions produce letter forms. In this oscillation model, the vertical velocity zero-crossing in the velocity space diagram is crucial from the standpoint of control. Morasso and Mussa-Ivaldi proposed a trajectory formation model using a spline function, and reconstructed a handwritten character using the formation model. Edleman and Flash (1987) proposed a handwriting model based on snap (fourth derivative of position) minimization. The representation of a character was four basic strokes, and a handwritten character was regenerated by combining several strokes. An interesting point in Edleman and Flash's model is that it is based on a minimization principle, which is computationally close to our proposed handwriting model. In a sense, they extended their computational theory for reaching movement to the handwriting model. However, the minimum jerk approach of Flash and Hogan (1985) for reaching movements is different from Edleman and Flash (1987) theory for handwriting. At the representation level, the handwriting model uses the four strokes while the reaching model used a target and via-points. It seems that there was no compelling reason for selecting the strokes. We believe that the simplest representation for reaching movement is the boundary conditions in the optimization problem, that is, the starting point, final point and the motion duration.

In this paper, we propose a handwriting model whose computational theory and representation and hardware are the same as the model for reaching movements. Our proposed computational model for cursive handwriting generates a trajectory that passes through many via-points. However, it is quite difficult to determine the small number of via-points needed to reproduce a cursive handwritten character. This is not only a crucial problem for the CNS but also for engineering fields. We propose an algorithm that can determine the via-points of the handwritten character, based only on the same minimization principle and which does not use any other information such as zero-crossing velocity (Hollerbach 1981).

Edelman and Flash (1987) have pointed out the difficulties in determining via-point locations. They have argued two points: (1) "Parsimony of representation requires to restrict the number of points defining the curve to a minimum." (2) "One must have a good reason for the choice of every via point locus." The importance and difficulty of these two problems can be understood when the problem of extracting the via-points from a given character trajectory is considered in attempting to reconstruct the character trajectory. It is clear from approximation theory that the character can be regenerated very accurately if the number of extracted via-points is very large. Appropriate via-points can not be assigned according to a regular sampling rule if the sample duration is constant and long. Therefore, there is an infinite number of combinations of numbers and via-point positions in the problem of extracting via-points from a given trajectory, and a unique solution can not be found if a trajectory formation theory is not identified. That is, it is an ill-posed problem. To resolve this difficulty, we propose an algorithm to determine the boundary conditions, called the via-point estimation algorithm, which finds the via-points by iteratively computing both the formation module (FIRM) and the via-point extraction module. The basic concept of our model is that the via-point extraction (pattern perception) is made possible by identifying and directly utilizing a trajectory formation neural network (pattern generation).

In this paper, we first discuss the computational theory for cursive handwriting. Then, we present the structure of our proposed handwriting model. Finally, we show that the human movement data are reproduced well by the proposed handwriting model when via-points are extracted by the via-point estimation algorithm.

2. A computational model for cursive handwriting

When a human learns to perform a movement pattern as a skilled motion, how is the movement pattern represented in the CNS? Consider a reaching movement such as when one reaches his/her hand toward a cup (Figure 1), which involves a hierarchical structure of motion planning. Several conditions required to achieve the goal of movement are derived from the visual information. Especially, in a reaching movement, the starting point, final point and the motion duration should be specified. These are the boundary conditions for the minimization problem and can also be regarded as a central representation for reaching movement. A trajectory is planned by a minimization principle such as the minimum torque-change criterion. The FIRM provides a hardware model as well as an algorithm to generate the optimal trajectory. Finally, a joint torque or a muscle tension stream is computed to accomplish the movement. Then the movement trajectory is realized. We believe that the boundary conditions for reaching movement are the simplest among all kinds of arm movement representation in CNS. Handwriting, on the other hand, is complex to specify. We propose a handwriting model to keep the theory for the reaching movement as much as possible. Our basic hypothesis for

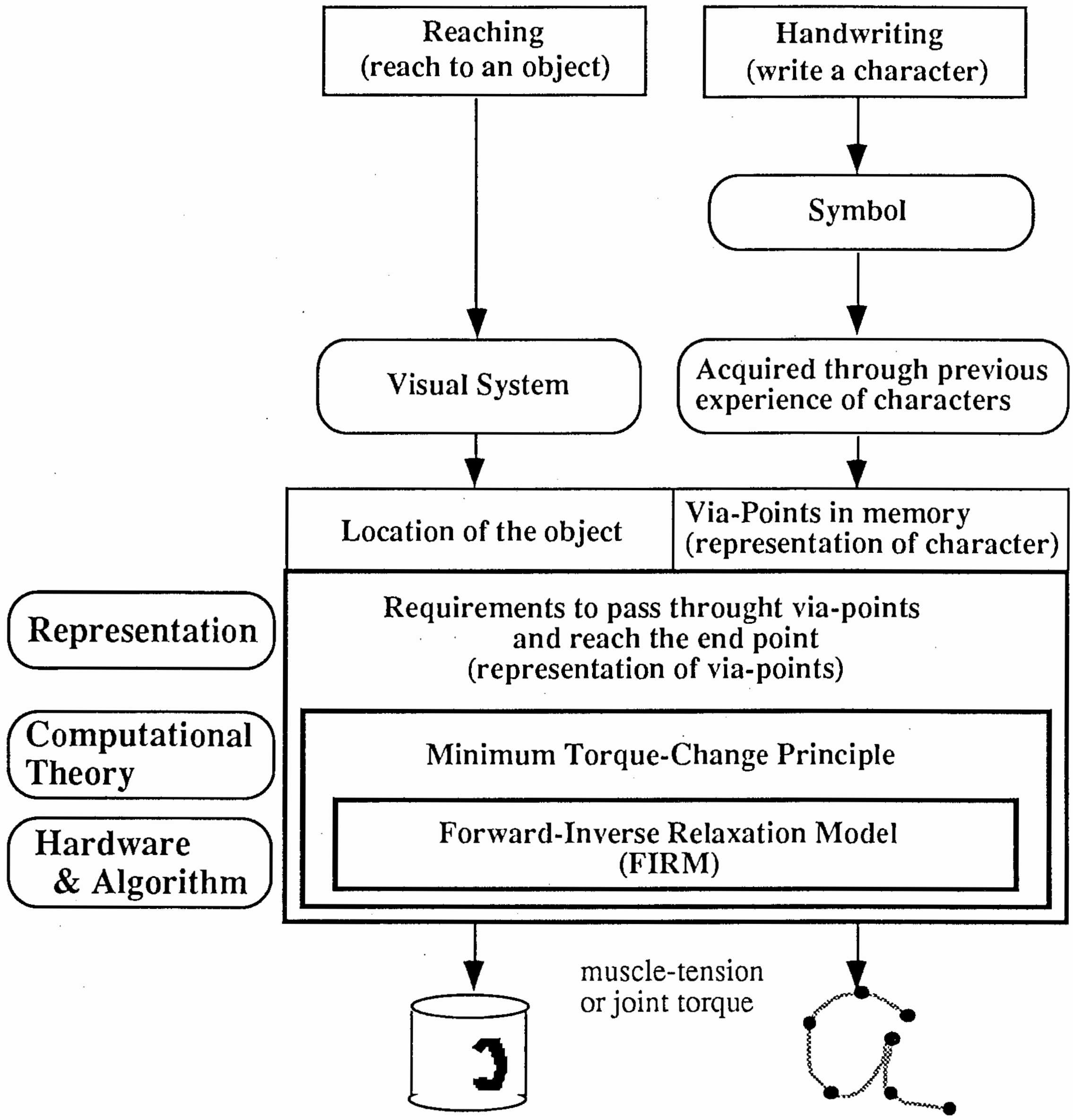


Figure 1 A handwriting model.

handwriting has three parts: (1) the computational theory is the minimization principle, particularly the minimum torque-change criterion; (2) the representation is a set of via-points that expresses the character; and (3) the hardware is based on the FIRM architecture. That is, when a human plans a cursive handwritten character, he/she should solve an optimization problem whose boundary conditions are many via-point positions instead of a few boundary conditions in the reaching movement. According to the above hypothesis, our proposed handwriting model is completely the same as the model of reaching movement in Marr's three-level understanding of the brain function (Marr 1982).

The computational theory-level, the hardware-level and the algorithm-level in the reaching movement are easily transferred to the cursive handwriting model. However, it is quite difficult to determine the via-point representation so as to reproduce a cursive handwritten character. We propose an algorithm to extract the via-points of the handwritten character. The via-point extraction problem is formulated to find the minimum number of via-points which can reproduce the given trajectory within a given error bound. Note that this via-point extraction problem is a nonlinear optimization problem. The objective function to be minimized is conceptually understood as the sum of the following two terms, the smoothness performance index of the minimum torque-change criterion and the data constraint which is defined by the error between the given trajectory and the reproduced trajectory. This objective function is minimized under a dynamic constraint given by human arm dynamics.

The via-point estimation algorithm can also be understood as a data compression algorithm that transforms a trajectory into a small number of feature points, which are the features of a symbol such as the handwritten character. Additionally, in a complex movement such as that of the Japanese toy, Kendama (stick and ball), the points extracted by the algorithm are the key to the success of the play (Kawato et al. 1994).

3. A trajectory formation neural network (FIRM)

In this section, the trajectory formation model (FIRM) shown in Figure 2 is explained. The FIRM can generate a trajectory with many via-points within a much smaller number of iterations than our previous cascade model (Kawato et al. 1990). Therefore, if the via-point information of a character estimated from an actual handwritten character is given to the FIRM, the model can provide an optimal character trajectory.

The FIRM uses a FDM, an IDM and a trajectory formation mechanism, which generates an approximate minimum torque-change trajectory. It does not require spatial representation of time or back propagation in iterative computation. The following

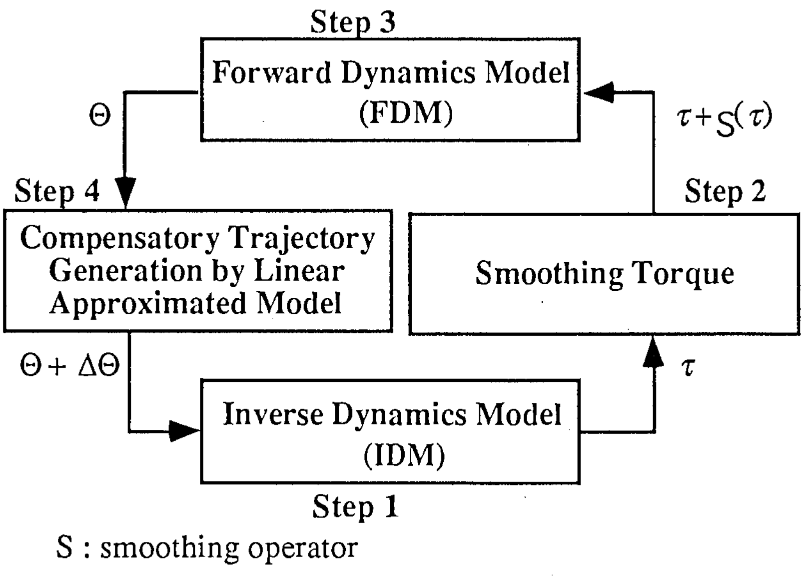


Figure 2
Trajectory formation model. (Forward-Inverse Relaxation Model). Neural network schema for arm trajectory formation using the forward dynamics model and inverse dynamics model.

algorithm is shown in Figure 2: Step 1. The torque is calculated from the joint angle trajectory that satisfies the terminal condition using the IDM, where $\Theta + \Delta\Theta$ satisfies the terminal conditions. Step 2. The torque is smoothed. Step 3. The terminal condition errors are found by generating the joint angle trajectory from the torque smoothed in Step 2 through FDM. Step 4. By finding a solution to the linear optimization problem, the compensatory trajectory $\Delta\Theta$, which cancels the terminal-condition errors, is obtained. The approximate optimal trajectory based on minimum torque-change is obtained by repeating Steps 1 to 4.

The FIRM proposed in Wada and Kawato (1993) was able to generate a trajectory with a via-point. Moreover, the algorithm of the FIRM can be easily extended to generate a trajectory with many via-points and compute an approximate optimal trajectory in several iterations. When the arm dynamics are approximated linearly as in the dynamics equation Eq. (3), there are two methods which generate the compensatory trajectory in Step 4 of the extended FIRM algorithm.

$$\tau^j = I^j \ddot{\theta}^j \qquad (j = 1, \cdots, M) \tag{3}$$

where τ^j is the torque generated by the j-th actuator. I^j and $\ddot{\theta}^j$ are the inertia of the link and the acceleration of the j-th joint angle, respectively.

The first method is that the compensatory trajectory can be generated by using the spline function because the minimum-jerk criterion is equivalent to the definition of the spline function. That is, the compensatory trajectory is formulated by the spline function minimizing the following criterion.

$$\int_0^{t_f} \left(\frac{d^3\theta^j}{dt^3} \right)^2 dt \tag{4}$$

However, an inverse matrix have to be calculated in the spline method, that is, the required spline coefficients are computed using matrix inversion. Since we believe that calculating the inverse matrix is not plausible in biological system, we use the second method in FIRM.

The second method is shown below and in Figure 3. The trajectory passing through a via-point is produced sequentially. Suppose that the position of via-points and the time passing through via-points are given.

(Step 1) A trajectory between a starting point and a final point is generated by using the minimum principle for the approximated linear dynamics such as Eq. (3).

(Step 2) The via-point V4 with the minimum value of a criterion, which is explained later, is selected. Then the trajectory Vs-V4-Vf is generated. This generated trajectory is added to the trajectory that has already been generated in step 1. The position error of the start point and the end point equal 0, since prior sub-trajectory generation has already compensated for the positional error. Thus, the boundary conditions of the generated trajectory at the start and end point become 0. The velocity and acceleration constraints at the start and end point are set to 0.

(Step 3) The via-point V5 with the minimum value in step 2 is selected. The trajectory V4-V5-Vf is generated in the same manner as step 2 and is added to the trajectory generated in step 2.

(Step 4) The selected via-point is V2. By repeating the procedure of generating and adding, a compensatory trajectory is obtained.

In calculating a trajectory passing through many via-points, the j-th actuator velocity constraint $\dot{\theta}^{j}_{via}$ and acceleration constraint $\ddot{\theta}^{j}_{via}$ at the via-point i are set by minimizing the following equation. The time of the start point t^{i}_{o} of the generated trajectory is that of the

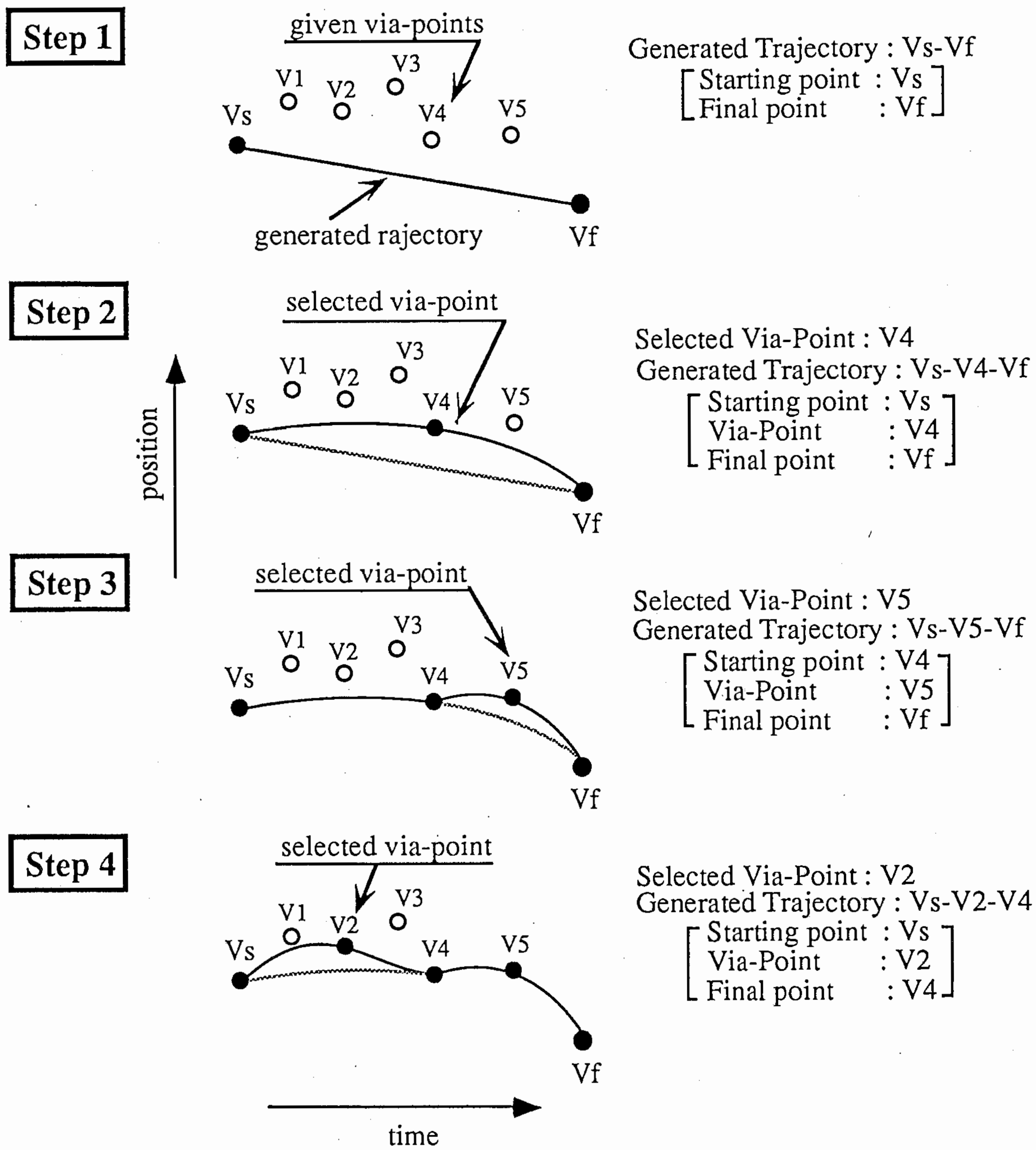


Figure 3

An algorithm for producing the compensatory trajectory in FIRM.

via-point located just before the assigned via-point i (time t^i_{via}), and the time of the end point t^i_f of the generated trajectory is that of the via-point located just after the assigned via-point i.

$$J(\dot{\theta}^j_{via}, \ddot{\theta}^j_{via}) = I^2 \left\{ \int_{t^i_0}^{t^i_{via}} \left(\frac{d\ddot{\theta}^j}{dt} \right)^2 dt + \int_{t^i_{via}}^{t^i_f} \left(\frac{d\ddot{\theta}^j}{dt} \right)^2 dt \right\} \rightarrow Min \quad (5)$$

The following velocity $\dot{\theta}^j_{via}$ and acceleration $\ddot{\theta}^j_{via}$ are found as the solution which minimizes Eq. (5):

$$\dot{\theta}^j_{via} = -\frac{5}{2} \frac{t^i_{o,via} - (t^i_f - t^i_{via})}{t^i_{o,via}(t^i_f - t^i_{via})} \theta^j_{via} \quad (6)$$

$$\ddot{\theta}^j_{via} = \frac{10\left(t^{i\,2}_{o,via} - 4t^i_{o,via} \cdot (t^i_f - t^i_{via}) + (t^i_f - t^i_{via})^2\right)}{3t^{i\,2}_{o,via} \cdot (t^i_f - t^i_{via})^2} \theta^j_{via} \quad (7)$$

where $t^i_{o,via} = t^i_{via} - t^i_o$.

Finally, we explain the criterion to select a via-point among the given via-points. As discussed in Appendix, the orthogonality of all the trajectories passing through each via-point can be proved. In the second method, the compensatory trajectory is represented as the following.

$$\theta^j \equiv c_1\varphi^j_1 + c_2\varphi^j_2 + c_3\varphi^j_3 + \cdots + c_n\varphi^j_n \quad (8)$$

where φ^j_i (i=1, 2, ⋯, n) represents a normalized orthogonal function; c_i represents a Fourier coefficient; and n represent the number of via-points. Because of the orthogonality of all the trajectories, the following equation is obtained.

$$\int_0^{t_f} \left(\frac{d^3\theta^j}{dt^3} \right)^2 dt = c_1^2 + c_2^2 + c_3^2 + \cdots + c_n^2 \qquad (9)$$

where c_i^2 is calculated by (A.14). The criterion in the second method is c_i^2. That is, the second method uses a sequential procedure to generate an trajectory by minimizing Eq. (9) by assigning every via-point. However, the method need not calculate an inverse matrix and it is expected that the generated trajectory by the second method is an approximated optimal trajectory.

4 A representation for cursive handwriting

4.1 A via-point estimation algorithm

Next, we show an algorithm that estimates via-points from the actual character trajectory. As discussed above, the following three prerequisites should be met in the via-point algorithm: (1) the number of the via-points is minimum, (2) there is a good reason for the choice of every via point locus, (3) the trajectory passing through via-points is the optimal trajectory based on the computational theory. In the following, we will show that by combination of a via-point estimation procedure and a trajectory formation procedure, our proposed algorithm can extract the approximately minimum number of via-points from a given trajectory θ_{data} with a given level of error threshold δ (the via-point estimation procedure). If a set of via-points $V = \left\{ P_1, P_2, P_3, \cdots, P_N \right\}$ are given and the arm dynamics is known, we can calculate the optimal minimum torque-change trajectory $\theta_{opt}(V)$ passing through these via-points (the trajectory formation procedure). The via-point estimation problem can be formulated to find the set V with the minimum N giving a trajectory which satisfies $\left(\theta_{data} - \theta_{opt}(V) \right)^2 < \delta$ for arbitrary time. To find the set V with the minimum N is an optimization problem and to generate the trajectory $\theta_{opt}(V)$ from V is another optimization problem. The former optimization problem is in perception domain, and the latter optimization problem is in motor-control domain. These optimization problems, which are related to the above prerequisites (1) and (2), are examined mathematically in the Appendix. The following two important properties of the proposed algorithm are proved there. First it is shown that the generated trajectory has the completeness property. That is, by increasing N, $\theta_{opt}(V)$ can approximate any θ_{data} with infinite accuracy. Second, it is shown that the algorithm finds V with approximately minimum N when θ_{data}, δ and arm dymanics are given. The above condition (3) is clearly satisfied because the trajectory is generated by the FIRM.

Our via-point estimation algorithm uses FIRM again as an important hardware module and this suggests a duality between movement pattern formation (FIRM) and movement pattern perception (via-point estimation).

4.2 A via-point extraction module

Our proposed via-point estimation algorithm (Figure 4) is essentially based on the same minimization principle as that for trajectory formation. This algorithm consists of two modules: a via-point extraction module and a trajectory formation module. The trajectory formation module is the FIRM.

In this section, the via-point estimation module is explained. In this via-point extraction module, there exists a via-point extraction procedure and a trajectory production procedure, and they are iteratively computed. Trajectory production in the module is based on the minimum-jerk model (Flash and Hogan 1985) in joint angle space, which is equivalent to the minimum torque-change model when an arm dynamics is approximated as in Eq. (3).

The procedure for via-point extraction is illustrated below and in Figure 5.

(Step 1) A trajectory between a starting point and a final point is generated by using the minimum principle for the approximated linear dynamics such as Eq. (3).

(Step 2) The point on the given trajectory with the maximum error value between the given trajectory and the generated trajectory in step 1 is selected as a via-point candidate.

(Step 3) If the maximum value of the sum of the square error is less than a threshold, the procedure above is terminated. If the maximum value of the sum of the square error is greater than the threshold, the via-point candidate is assigned as via-point i and a trajectory based on the minimization principle is generated from the starting point through the via-point i to the end point. This generated trajectory is added to the trajectory that has already been generated in step 1. By the same reason as in section 3, the boundary conditions of the generated trajectory at the

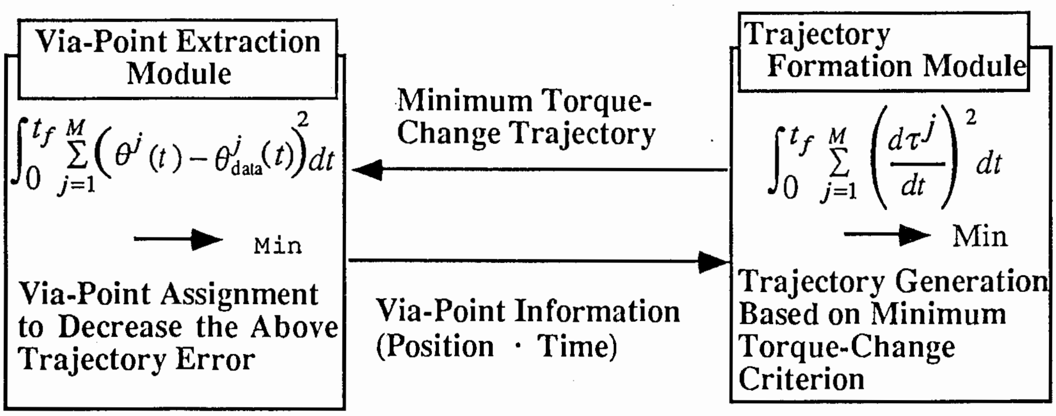


Figure 4
Via-point estimation model. τ^j is the torque generated by the j-th actuator of M actuators, and $\theta^j(t)$, $\theta^j_{data}(t)$ are the position generated by the model and the position of the given trajectory of the j-th joint angle, respectively. t_f is the movement time.

start and end point become 0. The velocity and acceleration constraints at the start and end point are set to 0. The velocity and acceleration at the via-point i are given by Eq. (6) and Eq. (7), respectively.

(Step 4) By repeating Steps 2 and 3, a set of via-points is found.

Finally, the via-points are fed to the trajectory formation module, and the trajectory based on the minimum torque change criterion is produced. The generated trajectory and the given trajectory are then compared again. If the value of the sum of the square error does not reach the threshold, the procedure above is repeated.

The algorithm of the trajectory production in the via-point extraction procedure is almost the same as that in the FIRM. A difference between FIRM and the via-point extraction procedure is how to select the via-point. The FIRM select the via-point by minimization and the via-point extraction procedure select the via-point by maximization.

The via-point estimation module minimizes the error between a given trajectory and a generated trajectory and the trajectory formation module minimizes the smoothness constraint. From this point of view, our proposed model, which minimizes the data term Eq. (10) and the smoothness term (11), is related to the standard regularization theory in computational vision (Poggio et al. 1985).

$$\int_0^{t_f} \sum_{j=1}^{M} \left(\theta^j (t) - \theta^j_{data} (t) \right)^2 dt \qquad (10)$$

$$\int_0^{t_f} \sum_{j=1}^{M} \left(\frac{d\tau^j}{dt} \right)^2 dt \qquad (11)$$

where θ^j_{data} is the position of the j-th joint angle. However, our model does not minimize the sum of the two terms simultaneously, but minimizes each term alternately. This

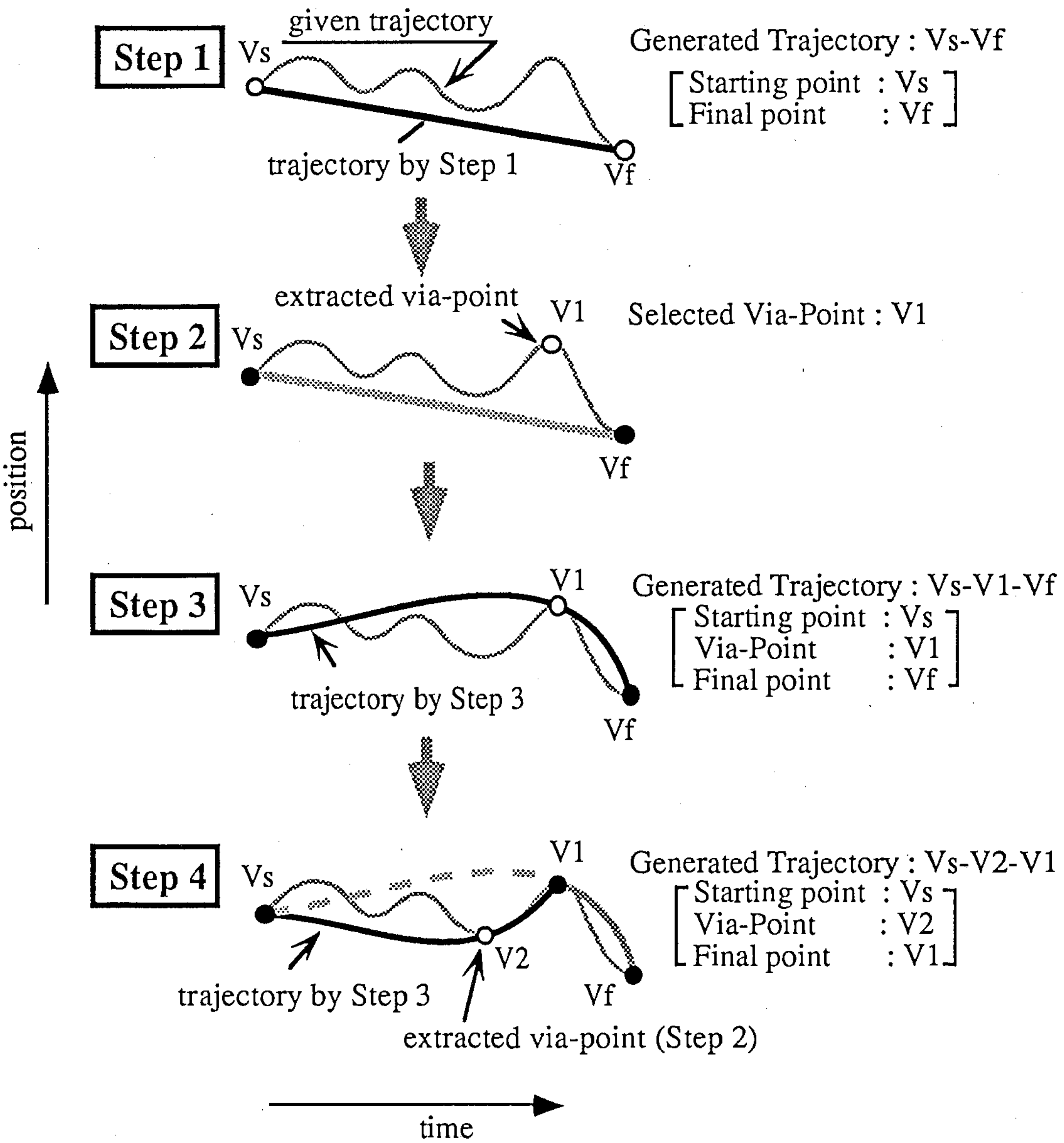


Figure 5 An algorithm for extracting via-points. Via-points are extracted so as to minimize the square of the error between the given trajectory and the generated trajectory.

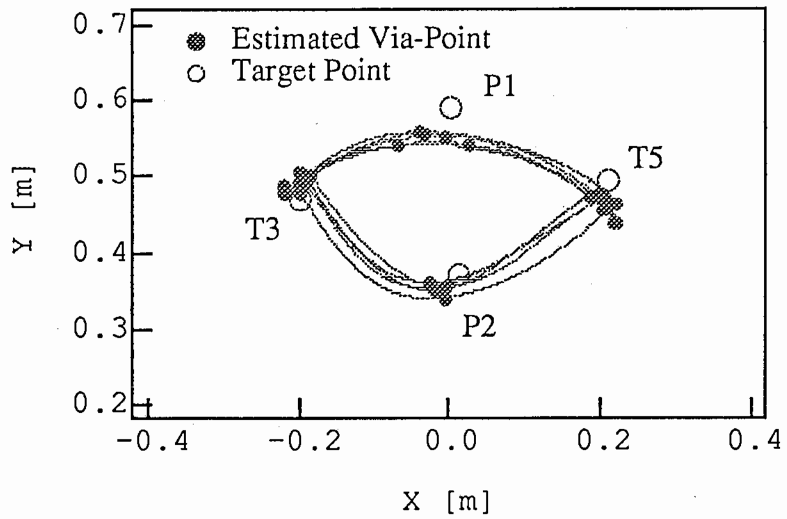
model does not find a global optimal solution but can find an approximated optimal solution with respect to the minimization of the performance index of the regularization theory.

5. Performance of the via-point estimation algorithm

In this section, we show that the via-point estimation algorithm can extract appropriate via-points by applying the algorithm to arm movement data. In the following simulation, we use the same mathematical dynamics equation as those in Uno, Kawato and Suzuki (1989) and the same physical parameter values as those in Kawato (1994) and Uno and Kawato (1994).

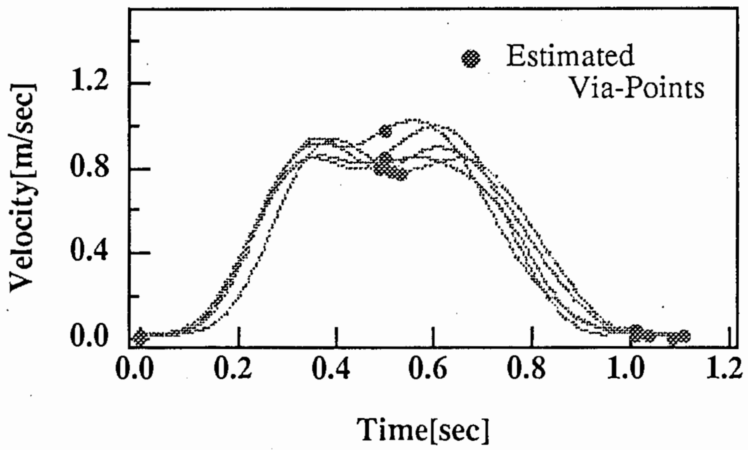
The efficacy of the via-point estimation algorithm is confirmed by reproducing human movement data with one via-point target. The start, final and via-point targets in the movement are designated to the subject (Figure 6). Two movements (T3->P1->T5 and T3->P2->T5) are examined. The target points (start, via-point, final), the human movement data measured by the OPTOTRAK system and the estimated via-point are shown in Figure 6 (a). The solid lines show measured trajectories. P1 and P2 (open circles) show target via-points. The filled circles show the via-points estimated by the algorithm. It is clear in Figure 6 (a) that the estimated via-points are close to the target via-points and that the via-point estimation algorithm can accurately estimate a given via-point in an actual human movement.

Furthermore, the tangential velocity and the curvature in movement T3->P2->T5 are shown in Figure 6 (b) and (c), respectively. The estimated via-point is not located at a peak curvature, but is estimated at a point in the direction of the final point near the peak

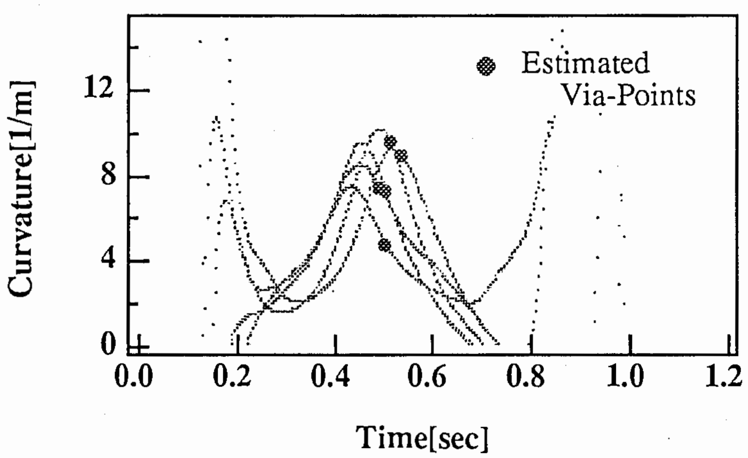


Estimated Via-Point
Target Point
P1
T5
T3
P2
0.7
0.6
0.5
0.4
0.3
0.2
-0.4
-0.2
0.0
0.2
0.4
Y [m]
X [m]

(a)



(b)



(c)

Figure 6
A result of via-point estimation in a movement with a via-point. Two movements (T3->P1->T5 and T3->P2->T5) are examined. The symbol O and the solid lines show the target points and measured trajectories, respectively. P1 and P2 show target via-points. The symbol ● shows the via-points estimated by the model. The estimated via-points were close to the target via-points.

curvature. Also, in the velocity curve, the estimated point is extracted at a point in the direction of the final point near the minimum of the velocity. Thus, the extracted via-points do not exactly coincide with kinematically definable feature points with maximum curvature or minimum velocity.

6. Performance of the handwriting model

In this section, we discuss the via-points estimated from actual handwriting trajectories and the relation between the via-points and the tangential velocity profile, and the curvature profile. Figure 7 (a) and (d) show the measured cursive handwritten character, the estimated via-points and the regenerated character trajectory. In Figure 7, (b) and (e), and (c) and (f) show the tangential velocity profile and the curvature profile, respectively. The estimated via-points are classified into two groups. The via-points in one group are extracted at near the minimum points of the velocity profile or near the maximum points of the curvature profile. Let us call this class of via-points as kinematic feature points. The via-points of the other group are assigned to positions that are independent of the above points. Let us call this second class of via-points as dynamic feature points. The via-points extracted from single via-point movements in the previous section are classified as kinematic feature points. Generally, the minimum of the velocity or the maximum of the curvature are considered to be the kinematic characteristics of the several orders of temporal derivatives of the movement trajectories.

We confirmed that a given trajectory can not be reproduced by using only the kinematic feature via-points (Figure 8). This shows that the dynamic via-points are important. That is, by using a method based on the minimization principle, which is a

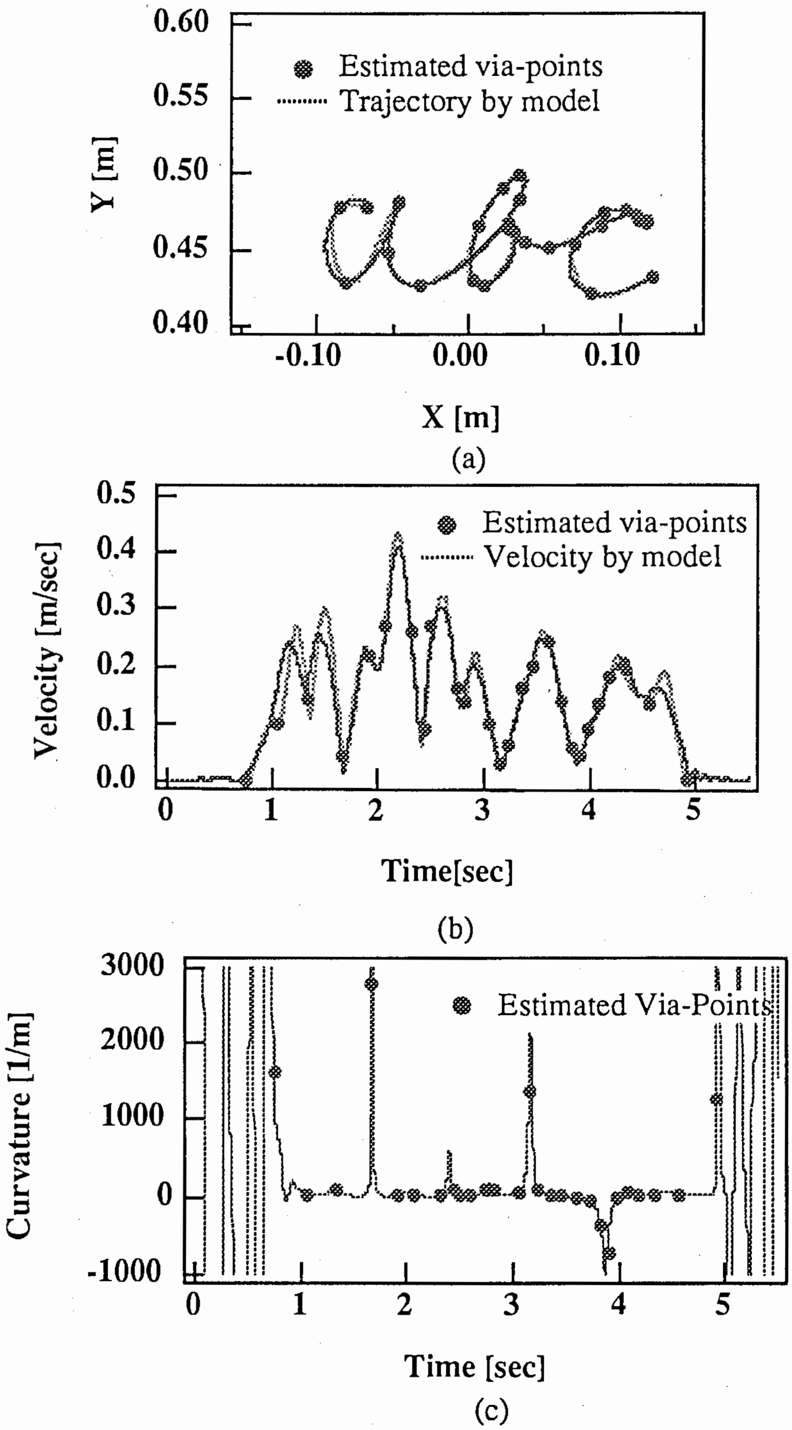


Figure 7 Estimated via-points in an example of cursive handwriting. Graphs (a), (b) and (c) show the trajectory, velocity profile and curvature profile for '*abc*'. The via-point estimation model extracts a via-point (segmentation point) between characters.

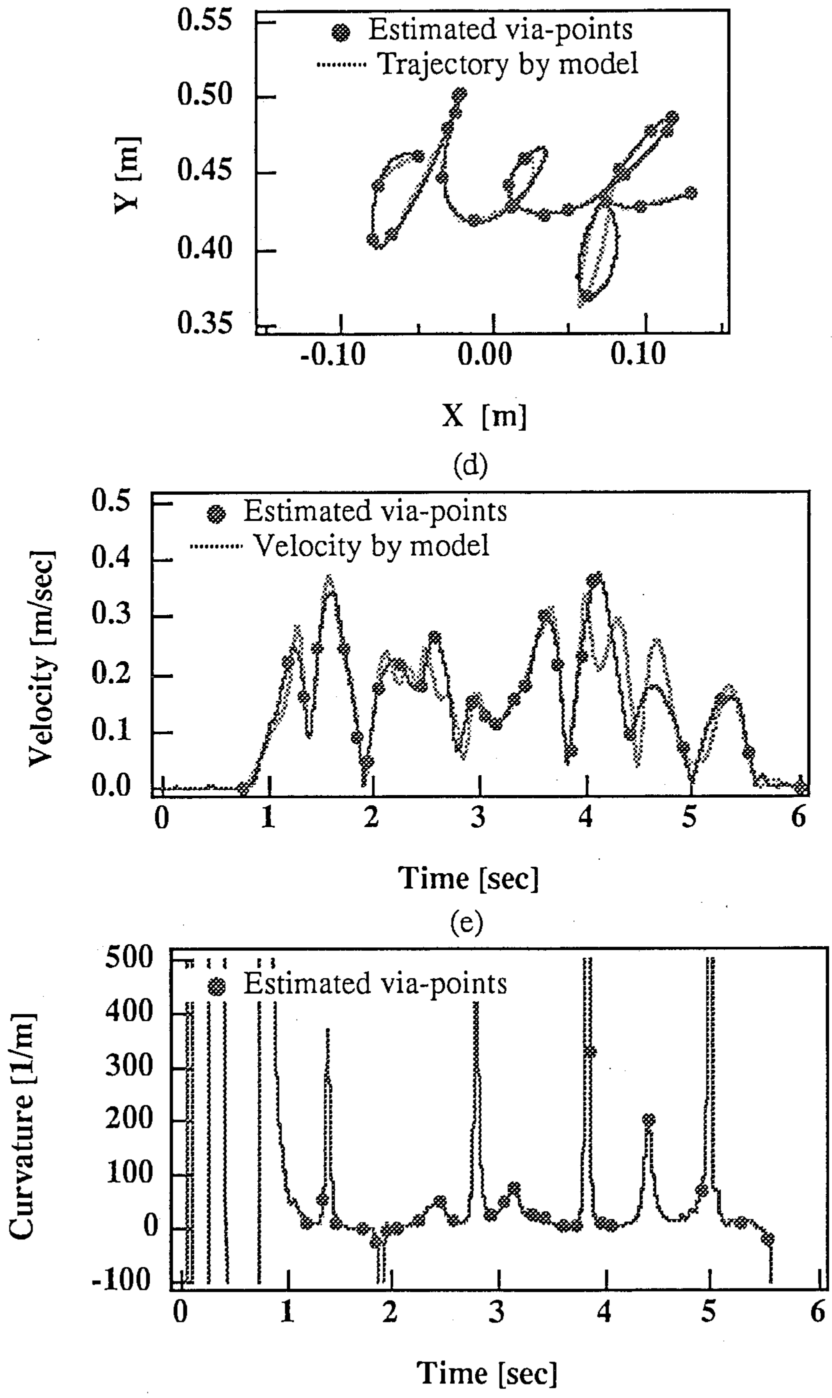
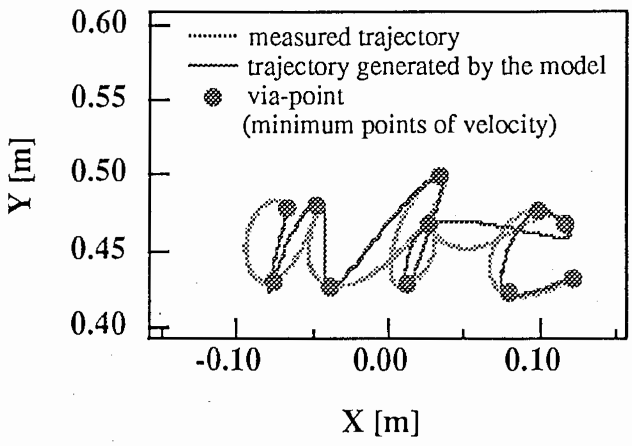
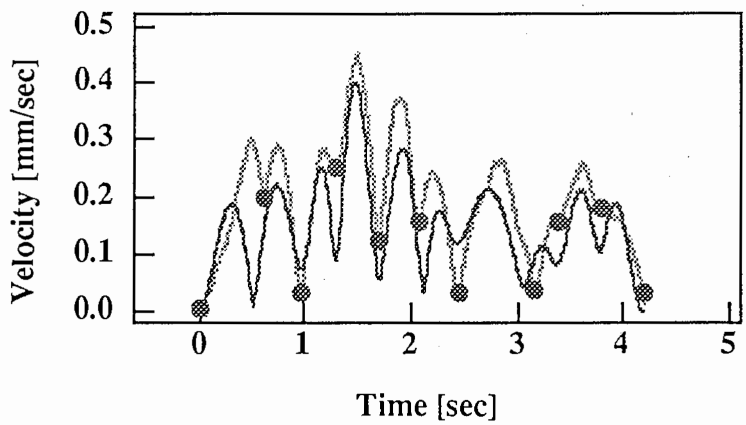


Figure 7 Estimated via-points in an example of cursive handwriting. Graphs (d), (e) and (f) show the trajectory, velocity profile and curvature profile for '*def*'. The via-point estimation model extracts a via-point (segmentation point) between characters.



(a)



(b)

Figure 8
A trajectory reproduced by using only the minimum of the velocity. Graphs (a) and (b) show the trajectory, velocity profile for '*abc*'.

quite different criterion from the minimum of the velocity or the maximum of the curvature, our proposed model can estimate points that can not be selected by any kinematic criterion. Moreover, the number of via-points is controllable by changing the regenerated trajectory error.

Finally, we point out that the via-point estimation algorithm extracts via-points between characters, that is, segmentation points, which are important in handwritten character recognition. Based on these via-points, we have already succeeded in pattern recognition of handwritten connected characters without using a word dictionary (Wada et al. 1994).

7. Discussion

Our proposed handwriting model is composed of via-point representation, the via-point estimation algorithm, and the trajectory formation model based on the minimization principle. In experiments, good qualitative and quantitative agreement was found between human data and the trajectories generated by the model. Our model is unique in that the same optimization principle and hard constraints used for reaching movement are also used for cursive handwriting. Our representation is based on the minimization principle, and does not use *a priori* knowledge. Therefore, it is quite different from other models (Hollerbach 1981; Morasso and Mussa Ivaldi 1982; Edelman and Flash 1987).

Furthermore, the actual via-points extracted by our proposed algorithm include the kinematic feature points used in Hollerbach (1981), Morasso and Mussa Ivaldi (1982), Edelman and Flash (1987). We demonstrated that features such as the minimum velocity point are important, but insufficient for reconstructing the trajectory if the same optimization principle as the reaching movement is used. Edelman et al. (1990) extracted segmentation points (called anchor points) using their handwriting model (Edelman and Flash 1987). Their method of extracting anchor points is a practically derived method, and uses motor knowledge indirectly, which is quite different from what our proposed model does.

Finally, we discuss the proposed model from a movement-pattern recognition point of view. Hoffman et al. (1993) and Rumelhart (1993) proposed a neural network recognition system, which established quite a good recognition rate for cursive handwritten characters, based on Hollerbach's handwriting model (Hollerbach 1981). Their models only make indirect use of the knowledge of motor control; that is, the feature inputs to the network are basically the vertical velocity zero crossing and the mid-point between vertical velocity zero crossing points. Rumelhart also used several other feature points. The feature points that are fed to their recognition system are almost the same as the via-point extracted by our via-point estimation algorithm. However, in the oscillation theory, the mid-points are not assigned as features of handwritten characters. On the other hand, our via-point estimation algorithm can assign both the vertical velocity zero crossing and the mid-point. This indicates that our proposed via-point algorithm could be applied to a recognition model. In the accompanying paper (Wada et al., 1994), we further explore this possibility.

REFERENCES

Edelman S, Flash T (1987). A Model of Handwriting. *Biol Cybern* **57**: 25-36

Hoff B, Arbib M.A. (1993). Models of trajectory formation and temporal interaction of reach and grasp, *Journal of Motor Behavior* **25**: 175-192

Flash T, Hogan N (1985). The coordination of arm movements; An experimentally confirmed mathematical model. *Journal of Neuroscience* **5**: 1688-1703

Hoffman, J., Skrzypek, J., Vidal, J. J. (1993). Cluster Network for Recognition of Handwritten, Cursive Script Characters. *Neural Networks*, **6**, 69-78.

Hollerbach JM (1981). An oscillation theory of handwriting. *Biol Cybern* **39**: 139-156

Jordan M.I. (1989). Indeterminate motor skill learning problems. In M. Jeannerod (Ed.), *Attention and Performance, XIII*. (pp.796-836) Cambridge, MA; MIT Press.

Kawato M, Maeda Y, Uno Y, Suzuki R (1990). Trajectory formation of arm movement by cascade neural network model based on minimum torque-change criterion. *Biol. Cybern.*, **62**: 275-288

Kawato M (1992). Optimization and learning in neural networks for formation and control of coordinated movement. In D. E. Meyer & S. Kornblum (Eds.), Attention and Performance XIV. (pp.821-849) Cambridge, MA; MIT Press.

Kawato M (1994). Unidirectional versus bi-directional theory for trajectory planning and control. *ATR Technical Reports*, TR-H-068, March 31.

Kawato M, Gandolfo F, Gomi H, Wada Y (1994). Teaching by showing in Kendama based on optimization principle. *Proceeding of International Conference on Artificial Neural Networks*, 601-606, May 26-29, Sorrento, Italy.

Marr D. (1982). Vision. New York: Freeman

Massone L, Bizzi E (1989). A neural network model for limb trajectory formation. *Biol. Cybern.,* **61**: 417-425

Morasso P, Mussa Ivaldi FA (1982). Trajectory formation and handwriting: A computational model. *Biol. Cybern.,* **45**: 131-142

Nakamura M, Uno Y, Suzuki R, Kawato M (1990). Formation of optimal trajectory in arm movement using inverse dynamics model, *Japan IEICE Technical Report,* NC89-63 (in Japanese)

Poggio T, Torre V, Koch C (1985). Computational vision and regularization theory. *Nature* **317**: 314-319

Rumelhart, D. E. (1993). Theory to practice: A case study-recognizing cursive handwriting. In E. B. Baum (Eds.), SIAM Frontier Series (pp. 177-196). Philadelphia.

Uno Y, Kawato M, Suzuki R (1989). Formation and control of optimal trajectory in human arm movement - minimum torque-change model. *Biol. Cybern.,* **61**: 89-101

Uno Y, Suzuki R, Kawato M (1989). Minimum muscle-tension-change model which reproduces human arm movement. Proceedings of the 4th Symposium on Biological and Physiological Engineering, 229-302 (in Japaneses)

Uno Y, Kawato M (1994). Dynamic performance indices for trajectory formation of human arm. *ATR Technical Reports*, TR-H-071, March 31.

Wada Y, Kawato M (1993). A neural network model for arm trajectory formation using forward and inverse dynamics models. *Neural Networks* , **6**:919-932

Wada Y, Koike Y, Vatikiotis-Bateson E, Kawato M (1994). A computational theory for movement pattern recognition based on optimal movement pattern generation. submitted to *Biol. Cybern.*

Appendix Mathematical consideration of the via-point estimation model

In this appendix, we first mathematically show that a given trajectory is approximated by a trajectory generated using the extracted via-points with infinite accuracy (completeness). Second, we demonstrate that the number of extracted via-points for a given threshold is approximately the minimum (optimality). In (1), we prove that each trajectory passing through each via-point is orthogonal to each other. Then in (2), by using the orthogonal functions, we show that the Fourier series is complete. Finally, in (3) it is shown that the number of assigned via-points is approximately minimum.

(1) Orthogonality of a trajectory passing through the via-point

The following discussion of orthogonality holds for each joint; however, the suffix denoting the joint is omitted. Mathematical relationship between the trajectories generated in Step 3 of the via-point extraction module is considered. Consider two trajectories among three generated trajectories V_s-V_f, V_s-V_1-V_f, V_s-V_2-V_1 in Figure 5. The start time and final time of one trajectory are inside those of another trajectory. When a via-point V_3 is assigned between V_s and V_2 and a via-point V_4 is assigned between V_1 and V_f, the trajectories V_s-V_3-V_2 and V_1-V_4-V_f do not overlap in time. Furthermore, trajectories that overlap in time, such as V_3-V_2-V_1 or V_2-V_1-V_4, are never generated because the via-points extracted by the via-point estimation algorithm are always extracted between two via-points that have already been assigned.

By extending the above discussion, the orthogonality of all the generated trajectories for a given data trajectory is shown. Here, $\dddot{\gamma}_1$, $\dddot{\gamma}_2$, $\dddot{\gamma}_3$, $\cdots$, $\dddot{\gamma}_l$ are the jerk of the trajectories generated by Step 3 (see section 4). The relation between the start and final point of trajectories i and j is classified into the following two cases:

(a) Trajectories i and j do not overlap in time.

(b) Trajectories i and j overlap in time. In this case, the start time or final time of trajectories i and j are the same, or the start and final times of trajectory i are included in the movement time of trajectory j.

<u>Case (a) :</u>

It is clear in case (a) that Eq. (A.1) holds. (where t_f is motion duration.)

$$\int_0^{t_f} \left(\ddot{\gamma}_i\right)\left(\ddot{\gamma}_j\right) dt = 0 \tag{A.1}$$

<u>Case (b) :</u>

Here, t_0^i, t_f^i, and t_{via}^i represent the start time, final time and via-point time of the inside trajectory i which is contained by trajectory j, respectively. γ_i^1 is a trajectory from t_0^i to t_{via}^i. γ_i^2 is a trajectory from t_{via}^i to t_f^i.

$$\int_0^{t_f} \left(\ddot{\gamma}_i\right)\left(\ddot{\gamma}_j\right) dt = \int_{t_0^i}^{t_{via}^i} \left(\ddot{\gamma}_i^1\right)\left(\ddot{\gamma}_j\right) dt + \int_{t_{via}^i}^{t_f^i} \left(\ddot{\gamma}_i^2\right)\left(\ddot{\gamma}_j\right) dt \tag{A.2}$$

Let I_1 be an integral of the first term and I_2 be an integral of the second term on the right-hand side of (A.2). By repeating the integration by parts, the following equations are calculated.

$$I_1 = \left[\frac{d^2\gamma_i^1}{dt^2}\frac{d^3\gamma_j}{dt^3}\right]_{t_0^i}^{t_{via}^i} - \left[\frac{d\gamma_i^1}{dt}\frac{d^4\gamma_j}{dt^4}\right]_{t_0^i}^{t_{via}^i} + \left[\gamma_i^1\frac{d^5\gamma_j}{dt^5}\right]_{t_0^i}^{t_{via}^i} - \int_{t_0^i}^{t_{via}^i} \gamma_i^1\left(\frac{d^6\gamma_j}{dt^6}\right) dt \tag{A.3}$$

$$I_2 = \left[\frac{d^2\gamma_i^2}{dt^2}\frac{d^3\gamma_j}{dt^3}\right]_{t_{via}^i}^{t_f^i} - \left[\frac{d\gamma_i^2}{dt}\frac{d^4\gamma_j}{dt^4}\right]_{t_{via}^i}^{t_f^i}$$

$$+\left[\gamma_i^2 \frac{d^5\gamma_j}{dt^5}\right]_{t^i_{via}}^{t^i_f} - \int_{t_{via}}^{t^i_f} \gamma_i^2 \left(\frac{d^6\gamma_j}{dt^6}\right) dt \qquad (A.4)$$

Note that when the arm dynamics is approximated by a point mass dynamics equation such as (3), the optimal solution of the minimum torque change criterion is equivalent to the minimum jerk trajectory in the joint space and is represented as a 5th order polynomial. Thus, Eq. (A.5) holds.

$$\frac{d^6\gamma_j}{dt^6} = 0 \qquad (A.5)$$

According to the boundary conditions, Eq. (A.6) holds:

$$\gamma_i^1 = \dot{\gamma}_i^1 = \ddot{\gamma}_i^1 = 0 \quad (t = t_0^i), \; \gamma_i^2 = \dot{\gamma}_i^2 = \ddot{\gamma}_i^2 = 0 \quad (t = t_f^i)$$
$$\gamma_i^1 = \gamma_i^2, \; \dot{\gamma}_i^1 = \dot{\gamma}_i^2, \; \ddot{\gamma}_i^1 = \ddot{\gamma}_i^2 \quad (t = t_{via}^i) \qquad (A.6)$$

Therefore,

$$I_1 + I_2 = 0 \qquad (A.7)$$

(2) General Fourier series expansion of the trajectory and completeness.

Next, we show that the compensatory trajectories generated in the via-point estimation algorithm constitute a complete set. Thus, the data trajectory can be expressed by the following equation.

$$\ddot{\theta}_{data} = \sum_{i=1}^{\infty} c_i \dddot{\varphi}_i \qquad (A.8)$$

where θ_{data} is the given trajectory and $\ddot{\varphi}_i$ represents a normalized orthogonal function of the compensatory trajectory $\ddot{\gamma}_i$. Also, an inner product of the function space is defined by the integral of the jerk as follows.

$$(f, g) \equiv \int_0^{t_f} (\dddot{f})(\dddot{g}) dt$$
$$\|\gamma_i\|^2 \equiv (\gamma_i, \gamma_i) = \int_0^{t_f} (\dddot{\gamma}_i)^2 dt = \int_{t_0^i}^{t_f^i} (\dddot{\gamma}_i)^2 dt$$

c_i represents a Fourier coefficient and is defined by the following equation.

$$c_i = \left(\theta_{data}, \varphi_i\right) \tag{A.9}$$

Suppose that there are n via-points and compensatory trajectories φ_i $(i = 1,\cdots,n)$ produced by using the via-points. The following partial sum of the series is defined.

$$\theta_{sub} \equiv a_1\varphi_1 + a_2\varphi_2 + a_3\varphi_3 + \cdots + a_n\varphi_n \tag{A.10}$$

The following L_2 error is considered.

$$J = \int_0^{t_f} \left(\dddot{\theta}_{data} - \dddot{\theta}_{sub}\right)^2 dt \tag{A.11}$$

J gives the minimum square error at $a_i = c_i$. Thus, supposing $a_i = c_i$, the following inequality is given by $J \geq 0$.

$$\left\|\theta_{data}\right\|^2 \geq (c_1^2 + c_2^2 + c_3^2 + \cdots + c_n^2) \tag{A.12}$$

θ_{data} is the given trajectory and $\left\|\theta_{data}\right\|^2$ is the square norm of the jerk. Thus, this is the upper bound. Therefore, it is clear that the infinite series $\sum_{i=1}^{\infty} c_i^2$ converges. Thus, c_n^2 converges to 0.

Also, the square of the Fourier coefficient is represented as (A.13), which is the integral of the square of the compensatory trajectory jerk.

$$c_i^2 = \int_0^{t_f} \left(\dddot{\gamma}_i\right)^2 dt = \int_{t_0^i}^{t_f^i} \left(\dddot{\gamma}_i\right)^2 dt \tag{A.13}$$

The following equation is the result of the integral of the square of the jerk c_i^2 calculated according to the velocity Eq. (6) and the acceleration Eq. (7).

$$c_i^2 = 20\left\{\delta\theta_i\right\}^2 \left(\frac{1}{t_{via}^i} + \frac{1}{t_f^i - t_{via}^i}\right)^5 \tag{A.14}$$

where $\left\{\delta\theta_i\right\}^2 = \underset{0<t<t_f}{Max} \left\{\theta_{data}(t) - \left\{\gamma_1(t) + \gamma_2(t) + \cdots + \gamma_{i-1}(t)\right\}\right\}^2$

t_{via}^i is t that maximizes the above equation.

Furthermore, it is found that the second component of the right-hand side of (A.14) is positive and lower bounded. Because $c_n \to 0$ as n becomes sufficiently large, $\left\{ \delta\theta_n \right\}^2$ becomes also sufficiently small. Furthermore, from the definition of t^n_{via} the following holds.

$$\begin{aligned} &\left| \theta_{data}(t) - \left\{ \gamma_1(t) + \gamma_2(t) + \cdots + \gamma_{n-1}(t) \right\} \right| \\ &\quad \le \left| \theta_{data}(t^n_{via}) - \left\{ \gamma_1(t^n_{via}) + \gamma_2(t^n_{via}) + \cdots + \gamma_{n-1}(t^n_{via}) \right\} \right| \\ &\quad = \left| \delta\theta_n \right| < \varepsilon \qquad (A.15) \end{aligned}$$

where $0 < t < t_f$

That is, Eq. (A.16) holds for arbitrary t $(0 < t < t_f)$.

$$\left| \theta_{data}(t) - \sum_{i=1}^{n-1} c_i \varphi_i(t) \right| < \varepsilon \qquad (A.16)$$

Therefore, the partial series $\sum_{i=1}^{n-1} c_i \varphi_i(t)$ uniformly converges to the given trajectory $\theta_{data}(t)$ (uniform convergence).

Next, J defined by (A.11) is estimated. The continuity of θ_{sub} is shown first. It is shown by straightforward calculation that the 3rd and 4th time derivatives of γ_i are continuous when γ_i is produced by the velocity Eq. (6) and the acceleration Eq. (7). It is also clear that the 5th time derivatives of γ_i is discontinuous at the estimated via-point time. Suppose that the 6th time derivatives of the given data trajectory exists and is continuous. The velocity and acceleration of θ_{data} and θ_{sub} at the start time and final time are equal to 0 and the position of θ_{data} and θ_{sub} are the same. Then, since θ_{sub} is a 5th polynomial in time, the 6th time derivatives of θ_{sub} is equal to 0. Accordingly, because of the continuity of θ_{data}, θ_{sub} and $\frac{d^6\theta_{sub}}{dt^6}$, J is given by (A.17) when J is divided into piecewise continuous parts in 5th derivative of γ_i and is calculated by repeating the integration by parts and by using continuous conditions.

$$J = -\int_0^{t_f} \left(\theta_{data} - \theta_{sub} \right) \frac{d^6\theta_{data}}{dt^6}\, dt \qquad (A.17)$$

As a critical step, let us assume that $\dfrac{d^5\theta_{data}}{dt^5}$ satisfies the Lipshitz condition in $0 \le t \le t_f$. That is, the Lipshitz constant L exists. From the discussion of the uniform convergence in (A.16), the following equation holds for a small value δ when n is large.

$$\left|\theta_{data} - \theta_{sub}\right| \le \delta \tag{A.18}$$

Thus, we can obtain the next inequality from L and (A.18).

$$\left|\int_0^{t_f}\left(\theta_{data} - \theta_{sub}\right)\frac{d^6\theta_{data}}{dt^6}\,dt\right| \le L\cdot\delta\cdot t_f \tag{A.19}$$

L and t_f are constant values when θ_{data} is given. Therefore, it is possible that a sufficiently small δ is obtained when n becomes large, producing the following equation.

$$0 \le \int_0^{t_f}\left(\dddot{\theta}_{data} - \dddot{\theta}_{sub}\right)^2 dt \le \varepsilon' \tag{A.20}$$

Now, we can say that the normalized orthogonal function $\dddot{\varphi}_n$ is complete. That is, it is shown that the Fourier series generated by the proposed algorithm is complete, and the given trajectory can be reproduced to an arbitrarily specified degree of closeness by using a sufficient number of via-points. For simplicity, a one-joint system was treated here; however, the same discussion can be applied to a multi-joint system and Eq. (A.18) will hold for the sum of the joints.

<u>(3) The optimality of the number of via-points</u>

We show in this section that our algorithm not only selects the complete orthogonal function, but also, by using a finite number of via-points, our method of estimating the via-points yields close to the smallest error.

Assume l via-points produced the compensatory trajectory φ_i^m $(i=1,\cdots,l)$. The following partial sum of the series is then defined, where m $(m = 1, \cdots, M)$ shows the

m-th joint.

$$\theta^{m}_{sub} \equiv c^{m}_{1}\varphi^{m}_{1} + c^{m}_{2}\varphi^{m}_{2} + c^{m}_{3}\varphi^{m}_{3} + \cdots + c^{m}_{l}\varphi^{m}_{l} \qquad (A.21)$$

The square of the error between the given trajectory jerk and the generated trajectory jerk is computed in the same manner as in (A.11).

$$\sum_{m=1}^{M}\left\{\int_{0}^{t_f}\left(\dddot{\theta}^{m}_{data}\right)^{2} dt - \left[\left(c^{m}_{1}\right)^{2} + \left(c^{m}_{2}\right)^{2} + \left(c^{m}_{3}\right)^{2} + \cdots + \left(c^{m}_{l}\right)^{2}\right]\right\} \qquad (A.22)$$

The first term inside the { } in (A.22) is constant when the trajectory is given. Therefore, to approximate the jerk of θ^{m}_{data} by a rather small number of orthogonal functions, it is critical that the second term in (A.22) be maximized, where, $\left(c^{m}_{i}\right)^{2}$ is defined in the same manner as in (A.13). Accordingly, this requires that the sum of the integral of the square of each compensatory trajectory jerk be maximized when a via-point is extracted.

However, it is quite difficult to find an optimal solution without information about the number of via-points l because the number of combinations of the number of via-points and the positions of the via-points is infinite. Our algorithm uses a sequential procedure to extract the via-points and finds an approximated solution by maximizing the square of the integral of the jerk by extracting every via-point.

By calculating the square of the integral of the jerk using the estimated via-point velocity Eq. (6) and acceleration Eq. (7), Eq. (A.23) is given as a function of via-point time t^{i}_{via}.

$$J(t^{i}_{via}) = \sum_{m=1}^{M}\left(c^{m}_{i}(t^{i}_{via})\right)^{2} = \sum_{m=1}^{M}\left\{20\left\{\delta\theta^{m}_{i}(t^{i}_{via})\right\}^{2}\left(\frac{1}{t^{i}_{via}} + \frac{1}{t^{i}_{f} - t^{i}_{via}}\right)^{5}\right\} \qquad (A.23)$$

where $\delta\theta^{m}_{i}$ is defined by the following equation, that is, a residual trajectory for the partial sum of series.

$$\delta\theta^{m}_{i}(t^{i}_{via}) = \theta^{m}_{data}(t^{i}_{via}) - \left\{\gamma^{m}_{1}(t^{i}_{via}) + \gamma^{m}_{1}(t^{i}_{via}) + \cdots + \gamma^{m}_{i-1}(t^{i}_{via})\right\}$$

Below we show that the via-point time t^i_{via} that maximizes the sum of the square of the residual trajectory $\sum_{m=1}^{M}\left\{\delta\theta_i^m(t^i_{via})\right\}^2$ is an approximated solution that maximizes Eq. (A.23). The following equation can be derived by differentiating Eq. (A.23) with respect to t^i_{via}.

$$\frac{dJ}{dt^i_{via}} = \left(\frac{1}{t^i_{via}} + \frac{1}{t^i_f - t^i_{via}}\right)^5 \cdot$$

$$\sum_{m=1}^{M} 20\,\delta\theta_i^m(t^i_{via})\left\{2\frac{d\delta\theta_i^m(t^i_{via})}{dt} + 5\,\delta\theta_i^m(t^i_{via})\left(\frac{1}{t^i_f - t^i_{via}} - \frac{1}{t^i_{via}}\right)\right\} \quad (A.24)$$

The right-hand side of (A.24) becomes 0 when the following equation holds.

$$\sum_{m=1}^{M}\left\{2\,\delta\theta_i^m(t^i_{via})\frac{d\delta\theta_i^m(t^i_{via})}{dt} + 5\left\{\delta\theta_i^m(t^i_{via})\right\}^2\left(\frac{1}{t^i_f - t^i_{via}} - \frac{1}{t^i_{via}}\right)\right\} = 0 \quad (A.25)$$

Eq. (A.25) becomes 0 when the first term becomes 0 and the via-point is located at the mid-point of the motion duration. Also, the time that maximizes $\sum_{m=1}^{M}\left\{\delta\theta_i^m(t^i_{via})\right\}^2$ is equivalent to the time when the first term on the left-hand side of (A.25) is equal to 0. One method of finding a good approximation of the solution to (A.25) is to search for the point that maximizes $\sum_{m=1}^{M}\left\{\delta\theta_i^m(t^i_{via})\right\}^2$ around the mid-point of the motion duration. In several movements, we confirmed by numerical computation that the maximum point of (A.23) and the maximum point of the sum of the square of the residual trajectories are almost the same. Accordingly, it is shown that our proposed model can extract an almost optimal number of via-points to approximate the given trajectory jerk with a finite number of orthogonal functions. Thus, we can say that the via-point estimation model can estimate the smallest number of via-points required to reproduce a character.