Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI-16)

Synthesizing Robotic Handwriting Motion by Learning from Human Demonstrations

Hang Yin[1,2], Patrícia Alves-Oliveira[3], Francisco S. Melo[1], Aude Billard[2] and Ana Paiva[1]
[1]GAIPS, INESC-ID and Instituto Superior Técnico, Universidade de Lisboa
[2]Learning Algorithms and Systems Laboratory, École Polytechnique Fédérale de Lausanne
[3]Instituto Universitrio de Lisboa (ISCTE-IUL), CIS-IUL, Lisboa, Portugal and INESC-ID
{hang.yin, aude.billard}@epfl.ch {patricia.alves.oliveira, fmelo, ana.paiva}@inesc-id.pt

Abstract

This paper contributes a novel framework that enables a robotic agent to efficiently learn and synthesize believable handwriting motion. We situate the framework as a foundation with the goal of allowing children to observe, correct and engage with the robot to learn themselves the handwriting skill. The framework adapts the principle behind ensemble methods - where improved performance is obtained by combining the output of multiple simple algorithms - in an inverse optimal control problem. This integration addresses the challenges of rapid extraction and representation of multiple-mode motion trajectories, with the cost forms which are transferable and interpretable in the development of the robot compliance control. It also introduces the incorporation of a human movement inspired feature, which provides intuitive motion modulation to generalize the synthesis with poor robotic written samples for children to identify and correct. We present the results on the success of synthesizing a variety of natural-looking motion samples based upon the learned cost functions. The framework is validated by a user study, where the synthesized dynamical motion is shown to be hard to distinguish from the real human handwriting.

1 Introduction

We consider a human-robot interaction scenario where a robotic agent closely interacts with children to help developing their handwriting skills. Practising handwriting skills is a time-consuming task for children, and the inclusion of robots as tools can have benefits in the long run. One of the benefits stems from the flexible roles that the robot could possibly play in comparison with human teachers. Besides demonstrating how to write a character correctly, the robot can serve as a facilitator or even as a learner, by presenting poor handwriting samples - whose error types can be customized with respect to a specific child - and gradually improving the robot performance under the help of children. These immersive scenarios allow for children to inspect, correct and engage with the robot, and as such teach themselves what is good handwriting and how to do it. This so-called *learning-by-teaching* paradigm is believed as an effective approach to motivate and engage children learners in education activities [Rohrbeck *et al.*, 2003].

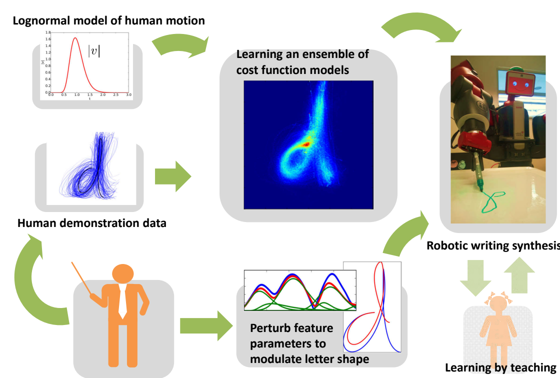


Figure 1: Pipelines of the proposed framework: a robotic agent learns from human demonstrations and then synthesizes handwriting motion by sampling in the kinematics feature space. The samples could be presented in the interaction for children to imitate or correct.

The challenge of realizing such a system lies in the endowment of proper task representations to the robot. First, the robot needs to learn a representation that encodes human-like handwriting, as well as the desired information for the derivation of a compliant motion controller to regulate the contact force. Secondly, the representation requires handwriting movement features that are straightforward to evaluate, interpret, and as such intuitive to generalize to the encodement of specific types of poor handwriting. For instance, the feature linking to the movement scale can be modulated by the robot or humans to generate or correct characters with inproportional components. Last but not the least, it is desirable to ensure rapid motion learning and synthesis, whose efficiency is of great importance in the practical human-robot interaction.

Inverse optimal control (IOC) provides a principle way for robots to achieve task learning from demonstrations (LfD), by extracting implicit cost representations which are transferable for robots to develop control commands in various contexts [Nikolaidis *et al.*, 2015][Byravan *et al.*, 2015]. In this paper, we presents a set of algorithms that efficiently learn from human data and generate rich and believable robotic writing motion, in order to support the interaction that facilitates children handwriting acquisition in a robotic companion project (Figure 1). The main contributions are summarized as:

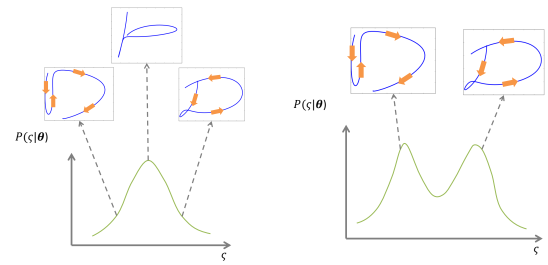
- We integrate the efficiency of local IOC with quadratic cost functions and the modeling power of ensemble methods, which are prevalent in general machine learning but less explored in the field of implicit LfD. The resulting algorithms feature rapid handwriting learning and compliant motion control synthesis, both of which are desired for the real interaction with children.
- We extend the proposed model with human-inspired curvature features to support diverse online handwriting synthesis. Remarkably, the features are explicitly associated with human understandable geometry properties. Thus they allow for generating not only legible characters for children to imitate, but also poorly written ones with believable and *controllable* deformities for exploiting the learning-by-teaching paradigm.

The paper begins by situating our work amongst related literatures in Section 2. The proposed approaches are then developed in detail in Section 3. We discuss the learning and synthesis results in Section 4, and in particular a Turing-like test to validate the human-likeness of generated motion in Section 5. Finally, Section 6 concludes the paper by summarizing the contributions and outlooking the future work.

2 Related Work

Learning with a continuous IOC formulation has widely been addressed with various techniques, such as Laplacian approximation [Levine and Koltun, 2012] or trajectory sampling [Kalakrishnan *et al.*, 2013], to evaluate the gradient approximately. The presented work avoids those expensive steps by solving a set of naive IOC problems on locally consistent data and then augmenting the results. This shares certain similarities with [Nikolaidis *et al.*, 2015] and [Cobo *et al.*, 2012] which cluster or segment demonstrated trajectories. Comparing with those, we motivate the presented work from the perspective of computing efficiency on continuous motion data which is natural for the robot control synthesis in our targeting practice.

The handwriting learning and synthesis have been investigated with various computing methods and models. In [Hood *et al.*, 2015], the PCA-based method was used to extract Gaussian distributed latent variables for letter synthesis. The framework of dynamic movement primitives was adopted in [Kulvicius *et al.*, 2012], with the aim of modeling the writing movement as well as the letter transition. Our work extends these models by relaxing the restriction on locally consistent demonstrations thus allows to learn multiple modes of writing a same letter. Such multi-modalities can also be captured by advanced probabilistic models such as factorial-HMM [Williams, 2009], recurrent density network [Graves, 2014] and probabilistic programming with proper priors [Lake *et al.*, 2015]. Our work is distinct as it learns with embedded features that are related to human movement characteristics. To this end, we can possibly bias the synthesis in a predictable way, which is advantageous in our interaction task to generate desired letter deformities.



(a) Monomodality (b) Multimodality

Figure 2: Single and multiple modalities of trajectory distributions. 2(a) is a poor model to encapsulate the diversity of style and moving direction in writing "D". Actually, the averaged trajectory is not legible, and should be assigned with low probability (high cost value) as 2(b).

3 Framework

Our problem consists of learning and synthesizing the robotic handwriting motion from a set of planar character trajectories $\mathcal{D} = \{\varsigma_i^*\}$. The central problem is to infer a parameter $\boldsymbol{\theta}$ of a cost function $\mathcal{J}$, with respect to which, the demonstrated trajectories $\mathcal{D}$ are supposed to be the (sub)optimal solutions.

From a probabilistic perspective, the human motion trajectories $\{\varsigma^i\}$ can be cast as the instances sampled from a probabilistic distribution, which is parameterized by $\boldsymbol{\theta}$ as Equation (1). This statistical model implies that the trajectories with low cost are more likely to be observed [Dvijotham and Todorov, 2010][Ziebart *et al.*, 2008].

$$P(\varsigma|\boldsymbol{\theta}) = \frac{\exp(-\mathcal{J}(\varsigma, \boldsymbol{\theta}))}{\int_{\varsigma'} \exp(-\mathcal{J}(\varsigma', \boldsymbol{\theta}))} \tag{1}$$

Thus $\boldsymbol{\theta}$ can be estimated by minimizing the negative logarithm likelihood of the observations. A simple parameterization strategy to capture the optimality of demonstrations is a quadratic function form:

$$\mathcal{J}(\varsigma, \boldsymbol{\theta}) = (\varsigma - \mu_\varsigma)^T {\Sigma_\varsigma}^{-1} (\varsigma - \mu_\varsigma) \tag{2}$$

where $\boldsymbol{\theta} = \{\mu_\varsigma, \Sigma_\varsigma\}$ encodes the reference motion and the desirability for a robotic agent to track with the end-effector. This can lead to a closed-form estimation of the partition function thus can ensure an efficient evaluation of the likelihood and the associated gradient. Moreover, the estimating parameters are closely related to the optimal control signal $\boldsymbol{u}$:

$$\boldsymbol{u} = -\boldsymbol{R}^{-1} \boldsymbol{B} \frac{\partial \mathcal{J}}{\partial \boldsymbol{x}} \tag{3}$$

where $\boldsymbol{x}$ denotes the state of trajectory ς, and $\boldsymbol{B}$ and $\boldsymbol{R}$ represent the control gain and penalty matrices which are chosen based on the dynamic model of the agent that carries out the

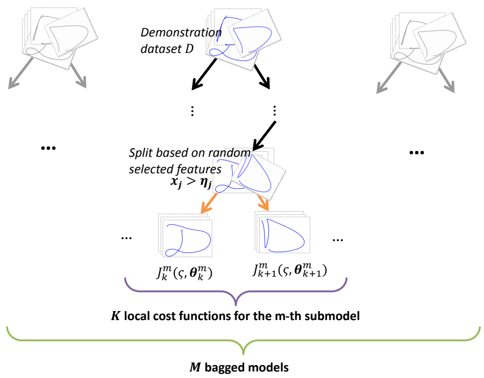


Figure 3: An ensemble of cost functions over partitioned dataset through random feature bagging.

execution. It is trivial to reveal that the control gain is proportional to the diagonal blocks of Σ_ς and the regulation point is along the mean trajectory μ_σ. Thus the estimation of the quadratic parameters actually synthesizes an impedance controller which allows the robot to accommodate the contact force, as is shown in [Yin *et al.*, 2014].

The limit of the quadratic form lies in that it assumes a globally optimal mono-mode of the trajectory distribution, whilst it is common to observe locally optimal demonstrations. This can be exemplified in human handwriting by the multiple modes of demonstrating one specific letter, see Figure 2(a) and 2(b).

3.1 An Ensemble IOC Approach

The proposed approach to address above challenges is to combine the simplicity of local quadratic cost learning with the scheme of ensemble methods. One possible way to achieve this is to apply the maximum-entropy (MaxEnt) model (1) on a fraction of dataset to construct K local model learned based on a subset of demonstrations $\mathcal{D}_k = \{\varsigma^*_{1:N_k}\}_k$, for $k = 1, ..., K$. The local data can be constructed with clustering techniques, if the number of clusters or a similarity metric is properly defined or tuned. Here we adopt a recursive random embedding scheme for the sake of computational efficiency. The idea is, at each step, the dataset is partitioned according to the criterion of decreasing the data entropy under the MaxEnt models. For instance, the N trajectories of $\mathcal{D}$ are divided into two subsets $\mathcal{D}_1$ and $\mathcal{D}_2$ according to some feature $x_j \in \varsigma$, making the entropy reduction is maximized by the partition of $\mathcal{D}_1 = \{\varsigma : x_j \geq \eta_j\}$ and $\mathcal{D}_2 = \{\varsigma : x_j < \eta_j\}$. In general, such structure can be constructed efficiently through random and greedy searching [Criminisi *et al.*, 2012]. The problem with such scheme is that the local cost function learning is sensitive to the random partition. To address this, we can introduce an ensemble of aggregated models [Breiman, 1996][Criminisi *et al.*, 2012] to obtain the estimation with a reduced variance. Figure 3 illustrates the idea of constructing the complete ensemble models.

Evaluating this ensemble model involves making decisions

Algorithm 1 RandomSubSpace - Partitioning dataset through feature bagging

Require: $\mathcal{D}$, N_d, N_D^{min}
Ensure: $\mathcal{D}^m_{k=1:K_m}$
$\mathcal{D}^m_{k=1:K_m} \leftarrow$ SPLIT($\mathcal{D}, N_x, N_D^{min}$)
function SPLIT($\mathcal{D}_{in}, N_x, N_D^{min}$)
$\{x_i\}_{i=1:N_d} \leftarrow$ RandomSelect($\{x_t\}_{t=0:T}$)
$j, \eta_j^* \leftarrow \operatorname{argmax}_{i,\eta_i} H(\mathcal{D}_{in}, \eta_i)$
if $|\mathcal{D}_{in}^{x_j \geqslant \eta_j^*}| \geqslant N_D^{min}$ and $|\mathcal{D}_{in}^{x_j < \eta_j^*}| \geqslant N_D^{min}$ **then return** Concatenate(SPLIT($\mathcal{D}_{in}^{x_j \geqslant \eta_j^*}$), SPLIT($\mathcal{D}_{in}^{x_j < \eta_j^*}$))
else return $\mathcal{D}_{in}$ ▷ Discard this split
end if
end function

from the bagged local models. One of the options is to take a weighted average over the distributions that are parameterized with the learned cost functions, as shown in Equation (4).

$$P(\varsigma|\boldsymbol{\theta}) = \frac{1}{M}\sum_{m=1}^{M}\sum_{k=1}^{K_m}\frac{|\mathcal{D}_k^m|}{|\mathcal{D}|}\frac{\exp(-\mathcal{J}_k^m(\varsigma, \boldsymbol{\theta}_k^m))}{\int_{\varsigma'}\exp(-\mathcal{J}_k^m(\varsigma', \boldsymbol{\theta}_k^m))} \quad (4)$$

where operator $|\cdot|$ denotes the cardinality of the set of data.

Note that, the proposed approach is different from [Nikolaidis *et al.*, 2015], because it is not learning local models on perfectly clustered demonstrations through methods like K-means or mixture of Gaussians. Here K_m is a natural result from the recursive suboptimal random embedding. By doing this, we avoid specifying the explicit number of clusters (e.g., how many types are there for a legible handwritten character "D"), which is unknown to the robot agent. By taking the negative logarithm of (4), the aggregated cost evaluation can be viewed as a *soft version of pointwise minimum* of a collection of cost functions. The pseudo code of the random embedding is given as Algorithm 1, where N_d and N_D^{min} trade-off the modeling power and the computational cost by controlling the size of candidate features and subset of data. A few remarks are given to highlight the advantage of the proposed approach over the previous ones:

- The model indicated by (4) allows the robot to encode complex multi-modal motion patterns, while it still can be efficiently learned because of the closed-form partition function for each local IOC model.
- The random embedding can be efficiently performed with off-shelf packages such as [Pedregosa *et al.*, 2011]. Other partitioning or clustering techniques [Criminisi *et al.*, 2012][Lee *et al.*, 2007] can also be seamlessly combined with the local cost function learning.
- Synthesis by sampling from the learned model (4) is straightforward as it resembles a mixture of Gaussians.

3.2 Human-inspired Kinematics Features

We motivate the enforcement of an informative feature by having an intuitive representation of the handwriting motion, and as such a human or the agent itself can modulate and bias

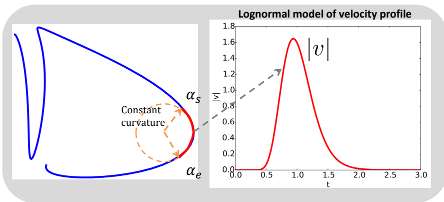


Figure 4: Modeling handwriting motion with curvature and lognormal velocity profile: the trajectory section is parameterized with a bell-shaped velocity magnitude and a constant curvature [O'Reilly and Plamondon, 2009].

the synthesis in order to generate motion with specific types of deformities.

We exploit the log-normal model [O'Reilly and Plamondon, 2009] which suggests encoding the velocity magnitude of natural human motion as an asymmetric bell-shaped lognormal profile. Moreover the path of motion stroke is constrained with a constant curvature, see Figure 4.

Here the model is adopted with slight changes on the dependent variable:

$$\varsigma(z) = \sum_{j=1}^{N} |v(z)| \begin{bmatrix} \cos(\phi_j(z)) \\ \sin(\phi_j(z)) \end{bmatrix} \quad t = Tz \tag{5}$$

$$|v(z)| = \sum_{j=1}^{N} \frac{A_j}{\sqrt{2\pi}\sigma_j(z - z_o^j)} \exp\left(-\frac{(\ln(z - z_0^j) - \mu_j)^2}{2\sigma_j^2}\right) \tag{6}$$

$$\phi_j(z) = \alpha_s^j + \frac{\alpha_e^j - \alpha_s^j}{2}\left(1 + \text{erf}\left(\frac{\ln(z - z_0^j) - \mu_j}{2\sigma_j}\right)\right) \tag{7}$$

where A, z_0, μ and σ are the parameters that regulate the velocity profile $|v|$. The curvature is parameterized by a pair of start and end angular positions α_s and α_e, with $\text{erf}(\cdot)$ denoting a Gaussian error function. The introduction of the phase variable z allows to evaluate the trajectory on a uniform interval $z \in [0, 1]$. The parameter T encodes the stroke time length, which is modeled according to the partitioned data.

It is easy to see that certain feature parameters shape the resulting motion in an explainable way. For instance, A affects the velocity magnitude; α_s and α_e impact the stroke alignment and straightness. Thus we can convert the vanilla representation $\boldsymbol{\theta}^k = \{\mu_\varsigma^k, \Sigma_\varsigma^k\}$ by extracting the statistics of kinematics features $\{A^k, z_0^k, \mu^k, \sigma^k, \alpha_s^k, \alpha_e^k\}$. This is achieved through the *RXZERO* estimation [O'Reilly and Plamondon, 2009] which estimates the parameters for the mean trajectory. Furthermore, the local variability of the kinematics parameters can be captured by

$$\Sigma_f^{k^{-1}} = (G_{\mu_\varsigma}^k)^T \Sigma_\varsigma^{k^{-1}} G_{\mu_\varsigma}^k \tag{8}$$

where $G_{\mu_\varsigma}^k$ is the Jacobian matrix that locally embeds position features into the kinematics parameter space. Algorithm 2 shows the learning routine together with the feature embedding steps. M denotes the number of aggregated models. Increasing this parameter reduces the model variance and it is chosen empirically in our experiments. Motion synthesis from this new feature space is still efficient as the agent only needs to evaluate closed-form formulas (5), (6) and (7).

Algorithm 2 Learning - Learning cost ensembles with curvature-based features

Require: $\mathcal{D} = \{\varsigma_i\}$, M, N_d, N_D^{min}
Ensure: $\mathcal{D}_{k=1:K_m}^m$, $\hat{\boldsymbol{\theta}}_k^m = \{\mu_{f_k}^m, \Sigma_{f_k}^m, \mu_{T_k}^m, \Sigma_{T_k}^m\}$, $k = 1, ..., K_m$, $m = 1, ..., M$
  for all m in $1{:}M$ **do**
    $\mathcal{D}_{k=1:K_m}^m \leftarrow$ RandomSubSpace$(\mathcal{D}, N_x, N_D^{min})$ ▷ Partitioning dataset $\mathcal{D}$ through random subspace
    for all k in $1{:}K_m$ **do**
      $\boldsymbol{\theta}_k^m = \{\mu_{\varsigma_k}^m, \Sigma_{\varsigma_k}^m\} \leftarrow \underset{\boldsymbol{\theta}}{\text{argmax}} \sum_{i=1}^{|\mathcal{D}_k^m|} \log P(\varsigma_i^* | \boldsymbol{\theta})$
      $\mu_{f_k}^m \leftarrow$ RXZERO$(\mu_{\varsigma_k}^m)$, $\Sigma_{f_k}^m \leftarrow$ Equation (8)
    end for
  end for

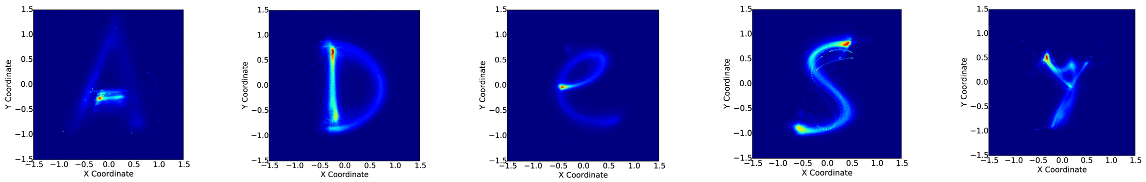
4 Results of Motion Learning and Synthesis

This section applies the proposed methods on adult handwriting motion learning and synthesis, with the aim of first evaluating the entire system on adult data before deploying it with children experiments. We will demonstrate the richness of the represented motion patterns and the computational efficiency comparing with alternative IOC methods.

4.1 Learning Handwriting from Human Data

The dataset employed was the UJI Pen Characters repository [Llorens *et al.*, 2008], which contains online handwriting samples collected from 60 adult subjects. We focused on alphabetical instances with either single or multiple strokes, for which each stroke was assumed to be independent. Yet this is by no means true as the strokes are correlated temporally and is possible to be captured by introducing extra conditional models [Lake *et al.*, 2015]. We adopted this independence assumption here to focus on the ensemble method itself, and such simplification turned out to work well in practice to synthesize reasonable motion trajectories.

The results are depicted as Figure 5(a) to 5(e). The most obvious observation is that the learned models successfully capture the legible shapes for either single or multiple-stroke characters. The variabilities of the stroke heating magnitude can be explained by the inconsistency of forming the specific trajectory sections. For some strokes, human behavior tends to be comparatively consistent, such as the short straight strokes in Figure 5(a), and 5(b) or the overall shape of "S" in 5(d). The variability of this consistency implies multiple modes in writing a specific letter. The encodement of such diversity can be best illustrated as Figure 5(e), which explicitly resembles the superimposition of two distinctive ways of forming a legible "y" in the Cartesian space. Note that the number of these patterns is not explicitly enforced beforehand but emerged from the ensemble of models which assign cost functions on random subsets of data.



(a) ”A” (b) ”D” (c) ”e” (d) ”S” (e) ”y”

Figure 5: Illustration of the learned cost ensembles that encapsulate the patterns of character profile. This is demonstrated in the Cartesian space but not the feature space for the illustrative purpose. The statistics of the curvature-based features is captured by taking samples and convert them to the original planar Cartesian space. The heat value of a point in the Cartesian space is evaluated by folding the learned cost function along the time horizon and counting the occurrences of the coordinates in the trajectory samples.

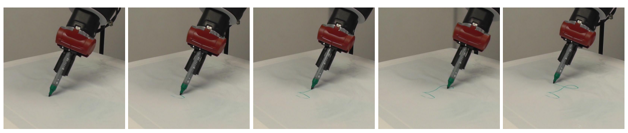


Figure 6: A 7-DOFs Baxter manipulator writes a ”g” with motion synthesized from the learned models.

4.2 Synthesis from Learned Cost Functions

With the learned model, robotic handwriting motion could be realized through an synthesized impedance controller (Equation (3)). Figure 6 shows snapshots of an example of the concrete robot motion resulted from a sample of synthetic ”g”.

The diversity of encoded motion patterns can be further demonstrated by synthesizing letter instances from the learned models. We give a few typical sampling results, again for either single or multiple strokes, as Figure 7. The synthesis samples illustrate rich writing patterns that are diversified in the aspects of size, orientation, and most importantly, the style. For instance, the ”d” that is constituted by a circle and a straight stroke, are successfully detected and encoded. We further the validation by generating poorly written characters with perturbed lognormal features. Figure 8 shows synthesized samples with an increased feature variance, which leads to various types of deformities such as inappropriate component proportion, misalignment or jerkiness in stroke transition. This demonstrates the potential of the framework to generate various good or poor handwriting motion for the children to imitate and correct.

4.3 Comparison of Computational Cost

We compared the proposed approach with alternative state-of-the-art IOC algorithms in terms of computational efficiency, especially for the letter synthesis as our targeting interaction task demands time critical performance. The results (Table 1) present the desired time for training 100 demonstrations and synthesizing one motion sample on a modern laptop. Note that training time of the proposed framework includes the overhead of RXZERO feature extraction, which eventually performs a non-convex optimization. The GPIRL [Levine *et al.*, 2011] is built upon Gaussian Process and

Table 1: Time Cost of Tested Algorithms

Algorithm	Training Time	Synthesis Time
GPIRL	$> 30h$	$22.2 \pm 5.2s$
MaxEnt	$3365.8 \pm 236.7s$	$5.8 \pm 1.2s$
EnsembleIOC	$381.7 \pm 5.7s$	$(6.4 \pm 3.7)e^{-4}s$

kernel methods. Thus it is not surprising that it does not scale well for the size of the dataset. The MaxEnt model with a popular RBF parameterization exploited the Laplacian approximation for gradient evaluation [Ziebart *et al.*, 2008][Levine and Koltun, 2012], which significantly mitigates the computing cost. However, the synthesis efficiency cannot be guaranteed because of the expensive MCMC sampling. Though there is definitely room for optimizing the implementation of all competing algorithms, it still serve as a proof-of-concept that why the presented approach is superior for the practical robot trajectory encoding and sampling, where both modeling power and efficiency are concerned.

5 Evaluating the Human-likeness of the Synthesized Motion

This section presents an online user study to examine how humans perceive the synthesized motion, as the robot needs to provide convince-looking handwriting motion to human users. Due to the obscurity of ”human-likeness”, the presented study was performed in a form of Turing-like test, where the participants were presented with a mixture of human and artificial dynamic motion, without showing the physical body of both the robot or of the human. The participants were instructed to choose among these motion samples the one they believe was generated by the algorithm. We were also interested in the confidence of the humans on their decisions to have a scale measurement of the fidelity from the subjective perspective.

5.1 Study Hypothesis

H1. *By observing the dynamic motion of the characters, the participants cannot distinguish between the agent synthetic and human written character samples. The classification performance is close to a random guess.* We

Figure 7: Synthesized motion samples from the learned cost ensemble models for different characters. The diverse modes and styles illustrate the multi-modal motion patterns encoded by the aggregation of simple cost functions

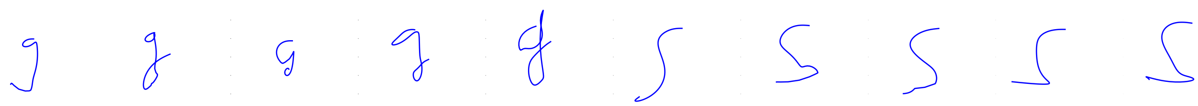


Figure 8: Synthesized motion of poor written samples by sampling from the learned model with random perturbations. The deformities can be intuitively controlled by modulating the local proportion, alignment and curvature of a specific component, as well as the continuity between the components.

expect the samples from learned ensemble models possess believable variabilities that are consistent with natural human handwriting. Thus most sampled motion parameters should result in characters which are hard to be identified from the mix up of synthesized and human samples. Quantitatively, this hypothesis implies an equivalence which can be numerically expressed as

$$\|\hat{c} - c\| \leq \delta \tag{9}$$

where $\hat{c}$ and c denote the classification performance from the experiment estimation and the random guess respectively. δ is a threshold quantifying the equivalence of the two tested values. The selection of δ will be presented in the results and analysis section.

H2. *Participants will not detain high confidence levels towards their choice*. This hypothesis checks the indistinguishability from a subjective perspective of the humans. It is expected to see the quantified confidence is lower than a certain level. We are also interested in the relation between the human confidence and concrete performance.

5.2 Study Procedure

The Turing-like test was carried out in the form of an online questionnaire. Concretely, the participants were instructed to evaluate 20 characters, containing both synthesized and human handwritten ones, by accessing web pages anonymously. They were explicitly instructed that there was only one synthesized sample for each character question. They could neither skip character pages nor browse back to the past ones to modify the previous responses. Their evaluation was based on two questions for each character:

Q1. *Which letter do you believe is written by a robot*?

To answer to this question, participants were presented with five dynamic handwriting motion for the character. The animation could be intuitively resumed or stopped by moving the cursor on or off the images. The participants were allowed to replay the motion as many times as they wanted before they made the decisions.

Q2. *How confident are you about your choice*?

The second question could be answered in a five-point type-Likert scale ranging from 1 to 5: 1-very low; 2-low; 3-neutral; 4-high; 5-very high

The sequences of characters and answer options were randomized to counter balance ordering effects. Moreover, demographic information was also collected. Participants were notified not to respond the questionnaire for multiple times at the beginning of the web page. Each individual questionnaire took about 10 to 15 minutes to complete.

5.3 Study Analysis and Results

The participants were recruited through the mailing lists within a university. A total of 68 participants completed the online questionnaire. Our sample ranges from 18 to 60 years old (*M* = 28.7; *SD* = 8.7).

In order to test **H1**, We chose $c = 0.2$ as there were five options in each character question. δ was defined according to the deviation of random classification performance $\delta = \sigma \approx 0.089$, if the number of correct classification was subject to a Binomial distribution. Our analysis showed that on average the participants achieved $\hat{c} = 0.226 \pm 0.086$ classification performance, which was close to the random guess $c = 0.2$. A further analysis showed that the null hypotheses of **H1**, $\hat{c} > c + \delta$ and $\hat{c} < c - \delta$, were both rejected by the corresponding one-sided t-test ($t_1(67) = 6.04$; $t_2(67) = 11.03$; $p < 0.01$). Therefore, the results showed statistically significant equivalence between the performance from empirical data and a theoretical value from a random guess, thus **H1** was strongly supported which suggests that participants were not able to distinguish between the character motion (synthesized versus human handwritten), wherein their choices translate the same as the random guess.

For **H2**, the averaged confidence level was 2.71 ± 0.70. One sided t-test concluded that this value was significantly below the neutral confidence level [$t(67) = 3.38$; $p < 0.01$], which also supported **H2**. We also analyzed that there was indeed a small fraction of participants who exhibited high confidence levels, however, such high confidence was not

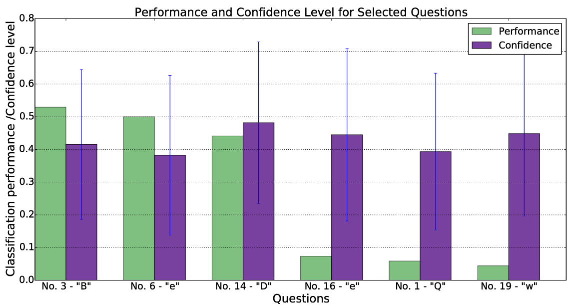


Figure 9: Classification performance and confidence levels for the selected characters on which the participants performed best and worst. The characters are sorted according to the performance, while the confidence levels are comparatively consistent. The overall performance 0.226 ± 0.086 is close to the random guess ($p < 0.01$).

necessarily related to good classification performance. A selection of the performance and confidence for the most contrasting results regarding the selected characters are shown in Figure 9, where it is obvious that the confidence levels are relatively consistent across characters and are not complying the performance trend. We also examined the confidence level associated to correct answers. The level turned out to be 2.71 ± 0.98, which is not significantly different from the overall confidence level (considering a threshold of 0.2; $t_1 = 4.63; t_2 = 3.79; p < 0.01$). A further analysis yields a rather weak Pearson's correlation ($\rho = 0.126$) between the performance and confidence level. Therefore the participants are indeed uncertain about their answers, even for the ones that happen to be correct.

6 Conclusions

In this paper, we demonstrate approaches that enable robotic agents to generate human-like handwriting motion by learning from human demonstration data. The proposed ensemble inverse optimal control approach yields highly efficient algorithms for learning a variety of types of motion trajectories, in contrast with the prior works that resort to expensive numerical optimization. The curvature-based features extend the representation to allow intuitive feature perturbation for synthesizing deformed handwriting samples. A human user study has provided a quantitative evaluation of the proposed approaches and the developed system, showing the participants were not able to confidently distinguish synthesized motion from the natural ones and their performance was close to a random guess. These contributions have laid a foundation to deploy a robotic agent in the real human robot interactions which are our future targets.

It worths noting that the proposed ensemble inverse optimal control framework can be easily extended to other general robotic tasks. In our ongoing work, we are adapting the framework to realize online robotic trajectory adaptation by estimating the task mode with the learned cost functions. An informative feature structure can benefit such general tasks by providing more compact representations and boostraping the synthesis and adaptation. Thus it would also be interesting to explore other task-dependent feature expressions or computing frameworks to learn these representations to parameterize the cost functions.

7 Acknowledgments

This work is partially funded by Swiss National Center of Robotics Research and national funds through Fundação para a Ciência e a Tecnologia (FCT) with reference UID/CEC/50021/2013 and the doctoral grant (ref. SFRH/BD/51933/2012) under IST-EPFL Joint Doctoral Initiative. P. Alves-Oliveira acknowledges a FCT grant (ref. SFRH/BD/110223/2015).

References

[Breiman, 1996] Leo Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, August 1996.

[Byravan *et al.*, 2015] Arunkumar Byravan, Mathew Monfort, Brian Ziebart, Byron Boots, and Dieter Fox. Graph-based inverse optimal control for robot manipulation. In *Proceedings of International Joint Conference on Artificial Intelligence*, 2015.

[Cobo *et al.*, 2012] Luis C. Cobo, Charles L. Isbell, and Andrea L. Thomaz. Automatic task decomposition and state abstraction from demonstration. In *Proceedings of the International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, Valencia, Spain, 2012.

[Criminisi *et al.*, 2012] Antonio Criminisi, Jamie Shotton, and Ender Konukoglu. Decision forests: A unified framework for classification, regression, density estimation, manifold learning and semi-supervised learning. *Foundations and Trends in Computer Graphics and Vision: Vol. 7: No 2-3, pp 81-227*, 2012.

[Dvijotham and Todorov, 2010] Krishnamurthy Dvijotham and Emanuel Todorov. Inverse optimal control with linearly-solvable mdps. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 335–342, 2010.

[Graves, 2014] Alex Graves. Generating sequences with recurrent neural networks. *CoRR*, abs/1308.0850, 2014.

[Hood *et al.*, 2015] Deanna Hood, Severin Lemaignan, and Pierre Dillenbourg. When children teach a robot to write: An autonomous teachable humanoid which uses simulated handwriting. In *Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, Portland, USA, 2015.

[Kalakrishnan *et al.*, 2013] M. Kalakrishnan, P. Pastor, L. Righetti, and S. Schaal. Learning objective functions for manipulation. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 1331–1336, May 2013.

[Kulvicius *et al.*, 2012] T. Kulvicius, K. Ning, M. Tamosiunaite, and F. Worgotter. Joining movement sequences:

Modified dynamic movement primitives for robotics applications exemplified on handwriting. *Robotics, IEEE Transactions on*, 28(1):145–157, Feb 2012.

[Lake *et al.*, 2015] Brenden M. Lake, Ruslan Salakhutdinov, and Joshua B. Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015.

[Lee *et al.*, 2007] Jae-Gil Lee, Jiawei Han, and Kyu-Young Whang. Trajectory clustering: A partition-and-group framework. In *Proceedings of the 2007 ACM SIGMOD International Conference on Management of Data*, SIGMOD '07, pages 593–604, New York, NY, USA, 2007. ACM.

[Levine and Koltun, 2012] Sergey Levine and Vladlen Koltun. Continuous inverse optimal control with locally optimal examples. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2012.

[Levine *et al.*, 2011] Sergey Levine, Zoran Popovic, and Vladlen Koltun. Nonlinear Inverse Reinforcement Learning with Gaussian Processes. In *NIPS*, 2011.

[Llorens *et al.*, 2008] D Llorens, F Prat, A Marzal, JM Vilar, MJ Castro, JC Amengual, S Barrachina, A Castellanos, S Espaa, JA Gmez, J Gorbe, A Gordo, V Palazn, G Peris, R Ramos-Garijo, and F Zamora. The ujipenchars database: a pen-based database of isolated handwritten characters. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco, may 2008. European Language Resources Association (ELRA).

[Nikolaidis *et al.*, 2015] Stefanos Nikolaidis, Ramya Ramakrishnan, Keren Gu, and Julie Shah. Efficient model learning from joint-action demonstrations for human-robot collaborative tasks. In *Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, HRI '15, pages 189–196, New York, NY, USA, 2015. ACM.

[O'Reilly and Plamondon, 2009] Christian O'Reilly and Rjean Plamondon. Development of a sigma-lognormal representation for on-line signatures. *Pattern Recognition*, 42(12):3324 – 3337, 2009. New Frontiers in Handwriting Recognition.

[Pedregosa *et al.*, 2011] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[Rohrbeck *et al.*, 2003] Cynthia A. Rohrbeck, Marika D. Ginsburg-Block, John W. Fantuzzo, and Traci R. Miller. Peer-assisted learning interventions with elementary school students: A meta-analytic review. *Journal of Educational Psychology*, 95(2):240–257, 2003.

[Williams, 2009] Ben Williams. Extracting motion primitives from natural handwriting data. PhD Thesis, U Edinburgh, 2009.

[Yin *et al.*, 2014] H. Yin, A. Paiva, and A Billard. Learning cost function and trajectory for robotic writing motion. In *Proceedings of IEEE International Conference on Humanoid Robots (Humanoids)*, Madrid, Spain, 2014.

[Ziebart *et al.*, 2008] Brian D Ziebart, Andrew L Maas, J Andrew Bagnell, and Anind K Dey. Maximum entropy inverse reinforcement learning. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, pages 1433–1438, 2008.