

无监督的猕猴运动皮层锋电位信号 CKF 解码

薛明龙¹ 吴海锋¹ 曾玉¹

摘要 如何通过猕猴运动皮层的神经元锋电位信号估计其手指移动位置是一神经解码问题, 现存方法解决该问题大多采用有监督训练, 需要通过训练数据得到神经元锋电位信号与手指移动位置的关系, 因此其估计性能依赖于训练数据. 本文提出了一种无监督解码方法, 该方法基于状态空间模型 (State space model, SSM), 利用神经网络得到神经元锋电位数与手指移动位置的关系权值, 再用逐次状态估计方法去估计手指移动的位置. 为减少训练的复杂度和提高估计准确度, 采用一种非线性的积分卡尔曼滤波 (Cubature Kalman filtering, CKF) 来完成神经网络的训练和手指位置的逐次状态估计. 与传统方法相比, 该方法的最大的特点是无监督, 可以由神经元锋电位簇向量直接估计手指移动位置, 而无需有监督训练. 实验结果显示, 当采用较少的有监督数据, 现存方法与本文方法相比有较大的估计误差; 当采用较多的有监督数据, 现存方法才具有与本文方法相近似的估计误差.

关键词 神经解码, 状态空间模型, 无监督训练, 积分卡尔曼滤波

引用格式 薛明龙, 吴海锋, 曾玉. 无监督的猕猴运动皮层锋电位信号 CKF 解码. 自动化学报, 2017, 43(2): 302–312

DOI 10.16383/j.aas.2017.c160065

Unsupervised CKF Decoding for Macaque Motor Cortical Spikes

XUE Ming-Long¹ WU Hai-Feng¹ ZENG Yu¹

Abstract How to estimate macaque's moving finger position through neuron spikes in his motor cortex is a problem about neural decoding. For the problem, most of existing methods use a supervised training algorithm and require supervised data to obtain the relationship between the spikes and the finger's moving position. Therefore, the performance of the existing methods depends on the training data. This paper proposes an unsupervised decoding method, which, based on a state space model (SSM), adopts neural networks to obtain the weights between the neuron spikes and the finger's moving position, and then estimates the finger's position through sequential state estimators. To reduce computational complexity and enhance estimation accuracy, a nonlinear cubature Kalman filter (CKF) is used to train the neural network and estimate the sequential moving positions. Compared with the existing methods, the proposed method's advantage is to be unsupervised. It could estimate the finger's position only through the spike vector instead of the supervised training data. Experiment results show that the existing methods have more estimation errors than the proposed method when a small amount of supervised data is adopted, and that the existing ones have similar estimation errors only when more supervised data adopted.

Key words Neural decoding, state space model (SSM), unsupervised training, cubature Kalman filter (CKF)

Citation Xue Ming-Long, Wu Hai-Feng, Zeng Yu. Unsupervised CKF decoding for macaque motor cortical spikes. *Acta Automatica Sinica*, 2017, 43(2): 302–312

神经编码是将外部世界映射到脑活动的一个过

程^[1], 外部世界经由视觉、听觉或嗅觉等感知信息传递到大脑, 大脑在得到这些信息后作出一系列反应, 这些反应体现在身体的骨骼或肌肉的活动中^[2]. 可以用神经生理记录等分析仪器采集相应的脑活动信号, 然后将骨骼或肌肉的运动与脑活动建立一一对应关系, 从而完成神经编码^[3]. 例如, 可以记录肢体运动时的神经元锋电位 (Neuron spike), 然后找到肢体运动方向、速度与神经元锋电位的对应关系^[4–8]. 神经解码是神经编码的逆过程, 它将脑活动映射到外部世界. 例如, 由运动皮层中神经元锋电位的变化去估计手指运动的方向、速度^[9–12], 通过辅助眼区的神经元锋电位估计眼睛移动的方向^[13], 通过脑磁图 (Magnetoencephalography, MEG) 信号

收稿日期 2016-01-21 录用日期 2016-05-31
Manuscript received January 21, 2016; accepted May 31, 2016
国家自然科学基金 (61262091), 云南省第 17 批中青年学术和技术带头人资助项目 (2014HB019), 云南省教育厅科学基金重点项目 (2014Z093), 云南民族大学研究生创新基金项目 (2016YJCXS03) 资助
Supported by National Natural Science Foundation of China (61262091), the 17th Batches of Young and Middle-aged Leaders in Academic and Technical Reserved Talents Project of Yunnan Province (2014HB019), the Project of Scientific Research Foundation of Yunnan Provincial Department of Education (2014Z093), and the Project of Postgraduate Innovation Foundation of Yunnan Minzu University (2016YJCXS03)
本文责任编辑 田捷
Recommended by Associate Editor TIAN Jie
1. 云南民族大学电气信息工程学院 昆明 650500
1. School of Electrical and Information Technology, Yunnan Minzu University, Kunming 650500

估计手腕的移动^[14]. 神经编码和解码的原理可应用到生物医学工程等领域, 在有听力障碍的患者耳中植入人工耳蜗, 将声音信号编码为可刺激听觉神经的数字信号, 从而使患者具有感知声音的能力^[15]; 用神经解码的方法让伤残人士通过大脑直接去控制设备的移动, 例如鼠标^[10] 或机械手臂等^[16].

本文主要研究通过猕猴运动皮层中神经元锋电位去估计其手指移动的位置, 这是一种典型的神经编码和解码问题. 较早的编码和解码方法采用线性方法^[17-18], 即神经脉冲数与运动状态的关系为成正比. 线性方法最大的优点是计算简单, 但其估计的准确性较低. 目前, 较为流行的是采用状态空间模型 (State space model, SSM) 方法解决该问题. SSM 模型已被广泛地运用到神经科学领域中, Czanner 等建立了一个广义线性 SSM 模型, 这种模型比传统模型更能准确描述神经元的活动特性^[19]; Malik 等利用 SSM 模型, 采用 EM (Expectation maximum) 算法准确估计神经元锋电位激发率 (Fire rate) 函数^[20]; Brown 等通过 SSM 把老鼠海马区的神经锋电位序列 (Spike train) 解码为位置信息^[21]; Shانهchi 等建立了一个 SSM 模型, 并运用最优反馈控制模型来解码猴子运动的状态^[22].

本文提出了一种猕猴运动皮层信号的无监督积分卡尔曼滤波解码方法 (Unsupervised cubature-Kalman-filtering decoding, UCKD). UCKD 也是基于 SSM 模型, 对多个神经元锋电位所构成的簇向量 (Population vector)^[23-24] 进行解码. 该解码方法采用神经网络得到神经元锋电位与手指移动位置的关系权值, 同时采用逐次状态估计方法^[25] 估计手指移动的位置. 为减少训练的复杂度和提高估计准确度, 采用一种非线性的积分卡尔曼滤波 (Cubature Kalman filtering, CKF)^[26] 完成神经网络的训练和手指位置的逐次状态估计. 与传统方法相比, 该方法的最大特点是无监督, 可以由记录的神经元锋电位簇向量直接估计手指移动的位置, 不需要先用有监督的训练数据对手指移动位置进行神经编码. 当有监督的数据缺失或者不完备时, 只能依靠无监督神经解码. 实验结果验证, 当采用较少的有监督数据, 现存的有监督方法与本文方法相比有较大的估计误差.

1 相关工作

由神经元锋电位簇向量估计猕猴手指移动的方法中, 较早提出的方法是线性拟合^[17-18] 可表示为

$$\mathbf{Y}_k = \mathbf{L}(\mathbf{s}_k) \quad (1)$$

其中, $\mathbf{Y}_k = [y_{1k}, y_{2k}, \dots, y_{Mk}]^T$ 为第 $k \times \Delta t$ 时刻

猕猴的手指移动信息矢量, $k = 1, 2, \dots, K$, Δt 为采样时间, y_{mk} 为猕猴第 m 个手指移动信息, $m = 1, 2, \dots, M$, 例如横坐标、纵坐标等信息, $\mathbf{s}_k = [s_{1k}, s_{2k}, \dots, s_{Nk}]^T$ 为神经元锋电位簇向量, s_{Nk} 为第 N 个电极所得到的神经元锋电位数, $\mathbf{L}(\cdot)$ 为一线性函数.

该方法先通过一组有监督的数据拟合出 $\mathbf{L}(\cdot)$, 然后将第 k 个时刻的簇向量 \mathbf{s}_k 代入 $\mathbf{L}(\cdot)$ 后估计出手指第 k 个时刻的移动信息. 线性拟合方法简单, 线性函数容易得到, 然而却存在以下问题. 通过 $\mathbf{L}(\cdot)$ 求得反函数 $\mathbf{s}_k = \mathbf{L}^{-1}(\mathbf{Y}_k)$ 时, 发现 \mathbf{s}_k 可能为负数, 但 \mathbf{s}_k 只可能为 ≥ 0 的数.

最大后验概率 (Maximum a posteriori, MAP) 估计能避免该问题, 表示为^[1]

$$\hat{\mathbf{Y}}_k = \max_{\mathbf{Y}_k \in \mathbb{Y}^M} \{P(\mathbf{s}_k|\mathbf{Y}_k)P(\mathbf{Y}_k)\} \quad (2)$$

其中, $P(\mathbf{s}_k|\mathbf{Y}_k)$ 是在条件 \mathbf{Y}_k 下 \mathbf{s}_k 的观测值的概率, $P(\mathbf{Y}_k)$ 是 \mathbf{Y}_k 的概率, \mathbb{Y}^M 为 \mathbf{Y}_k 的搜索集合. MAP 方法利用广义线性函数 (General linear model, GLM)^[1], 可以避免估计出来的 \mathbf{s}_k 为负数的情况. 然而同线形拟合一样, 最大后验概率估计将 $\mathbf{Y}_k, k = 1, 2, \dots, K$ 看作为相互独立的数据. 但实际上, 手指移动是一个具有一定速度和方向的连续过程, 特别当 Δt 较小, 例如 10^{-2} s 时, \mathbf{Y}_k 与 \mathbf{Y}_{k-1} 是相关的. 因此, 该方法的估计准确性较低.

SSM 方法采用逐次状态估计的方法表示 \mathbf{Y}_k 与 \mathbf{Y}_{k-1} 的相关性, 其状态方程和观测方程为^[1, 9-12]

$$\mathbf{Y}_k = \mathbf{h}(\mathbf{Y}_{k-1}) + \boldsymbol{\omega}_k \quad (3a)$$

$$\mathbf{s}_k = \mathbf{f}(\mathbf{Y}_k) + \mathbf{v}_k \quad (3b)$$

其中, $\mathbf{h}(\cdot)$ 表示自变量为 \mathbf{Y}_{k-1} 的函数, $\mathbf{f}(\cdot)$ 表示自变量为 \mathbf{Y}_k 的函数, $\boldsymbol{\omega}_k$ 是过程噪声矢量, \mathbf{v}_k 是观测噪声矢量.

通过式 (3a) 和 (3b) 估计 \mathbf{Y}_k , 需要知道函数 $\mathbf{h}(\cdot)$ 和 $\mathbf{f}(\cdot)$, 若均为线性函数, 可以利用卡尔曼滤波方法估计 \mathbf{Y}_k ^[11]; 若为非线性函数, 可以利用扩展卡尔曼滤波或粒子滤波方法^[9]. 但是, 无论采用何方法估计 \mathbf{Y}_k , 必须知道这两个函数. 传统的 SSM 方法需要通过有监督的训练数据来拟合得到 $\mathbf{h}(\cdot)$ 或 $\mathbf{f}(\cdot)$, 因此其估计性能必然会受到拟合的影响. 当拟合函数较精确, 其估计效果就会较好, 反之则会较差. 本文提出一种不依赖有监督数据的无监督方法, 通过神经网络训练出神经元锋电位 \mathbf{s}_k 和手指移动信息 \mathbf{Y}_k 间的关系, 然后利用 SSM 估计出手指移动信息.

2 系统模型

猕猴的手指移动估计问题描述如下: 首先对一

只猕猴进行训练,使其能够做“弹珠台”运动;一旦目标物出现,就用手指指向目标物.图1是猕猴的手指在一个长为 L 、宽为 W 的平面内的运动轨迹,猕猴的手指初始位置在 A 点,当目标物出现在 B 点,猕猴的手指就移动到 B 点,随后 B 点目标物消失, C 点出现目标物,猕猴的手指又移向 C 点,再至 D 点,反复向目标物出现的位置运动.同时给猕猴的脑部连接一组电极,在 $k \times \Delta t$ 时刻, $k = 1, 2, \dots, K$ 记录猕猴运动皮层的神经元锋电位数 s_k ,然后通过该神经元锋电位数估计猕猴手指移动信息 Y_k .采用SSM模型估计手指移动首先需要确定状态方程和观测方程.

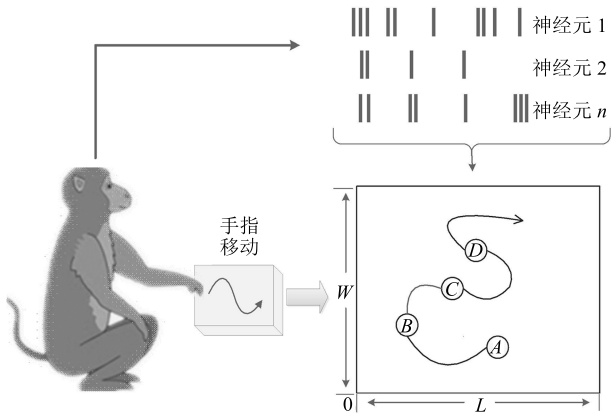


图1 猕猴手指移动轨迹图
Fig.1 An example for the trajectory of a macaque's finger

2.1 状态方程

以运动信息 Y_k 中手指移动纵坐标 y_k 为例确定状态方程(3a).用 y_k 代替 Y_k 后,状态方程变为 $y_k = h(y_{k-1}) + \omega_k$,其中 $h(y_{k-1})$ 和 ω_k 需要确定.由于纵坐标 $y_k \in [0, L]$,则两个相邻时刻纵坐标间差值 $y_k - y_{k-1} = \Delta y \in [-L, L]$.将 $[-L, L]$ 等分成 $2L$ 个区间 $\mathcal{L}_1, \mathcal{L}_2, \dots, \mathcal{L}_{2L}$,并让 Δt 足够小,例如 10^{-2} s,统计出 Δy 落在各个区间的概率 $p(\Delta y \in \mathcal{L}_i), i = 1, 2, \dots, 2L$.图2给出了其概率分布图,数据来源于公开数据(详见第3.1节系统设置),其中 $\Delta t = 70$ ms, $L = 15$ cm.由图2可以看出, Δy_k 小于 0.5 cm 的概率约为 60%,小于 2.5 cm 的概率约为 97%,其概率分布曲线近似于零均值的高斯分布,由此可得到相应的状态方程为

$$y_k = y_{k-1} + \omega_k \tag{4a}$$

其中, ω_k 是方差为 Q_ω 、均值为 0 的高斯分布噪声.

2.2 测量方程

同样, y_k 代替 Y_k 后,观测方程(3b)变为

$$s_k = f(y_k) + v_k \tag{4b}$$

其中 $f(y_k)$ 和 v_k 需要确定.由文献[1, 17-18]可知, $f(\cdot)$ 近似为一种线性函数.

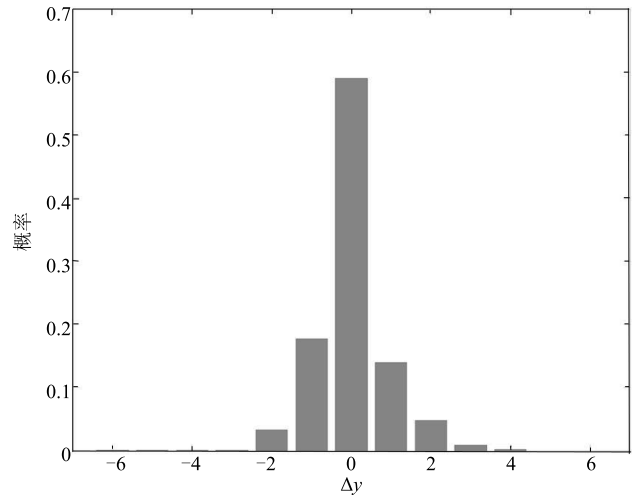


图2 相邻两时刻纵坐标间差值 Δy 的分布概率图
Fig.2 An example for the distribution of Δy

将式(4a)代入 $s_k = f(y_k) + v_k$ 中,有

$$s_k - s_{k-1} = \gamma \omega_k + v_k - v_{k-1} \tag{5}$$

其中, γ 为一常数向量.令第 n 个电极上两相邻时刻神经元锋电位数差值 $\Delta s_n = s_{nk} - s_{n(k-1)}$,则 $\Delta s_n \in [-S, S]$,其中 S 为电极所能采集到的最大锋电位数.将 $[-S, S]$ 等分成 $[2S]$ 个区间 $\mathcal{S}_1, \mathcal{S}_2, \dots, \mathcal{S}_{2S}$,统计出 Δs 落在各个区间的概率 $p(\Delta s_n \in \mathcal{S}_i), i = 0, 1, 2, \dots, 2S$.图3给出了其概率分布图,数据来源于公开数据(详见第3.1节系统设置),其中 $\Delta t = 70$ ms.从图3可以看出, Δs_n 为 0 的概率约为 30%, Δs_n 为 1 或 -1 的概率约为 80%,其概率分布曲线也近似于零均值的高斯分布.由于 ω_k 和 Δs_n 均近似为零均值高斯分布,则由式(5)可知, v_k 也是零均值的高斯分布的噪音矢量.而且,由于各电极上各神经元锋电位数相互独立,因此各电极上的观测噪声也相互独立.最终,得到观测方程 $f(y_k) + v_k$ 中,有 $s_k = f(y_k) + v_k$.其中, $f(\cdot)$ 为一线性函数, $[v_{1k}, v_{2k}, \dots, v_{Nk}]^T$ 为 $\mathbf{0}$ 均值、方差矩阵为 R_v 的高斯分布噪声矢量, v_{nk} 为零均值、方差为 σ_{nk}^2 的高斯分布噪声, $R_v = \text{diag}\{\sigma_{1k}^2, \sigma_{2k}^2, \dots, \sigma_{Nk}^2\}^T$.

通过状态方程(4a)和观测方程(4b)估计猕猴手指移动的位置 y_k ,需要知道函数 $f(\cdot)$ 的具体表达式.传统的方法通过有监督的训练方法,即知道神

经元锋电数目 s_k 和对应的猕猴手指移动位置 y_k , 然后通过该数据拟合出 $f(\cdot)$. 本文提出一种 UCKD 方法, 不需要监督数据就能估计出 y_k .

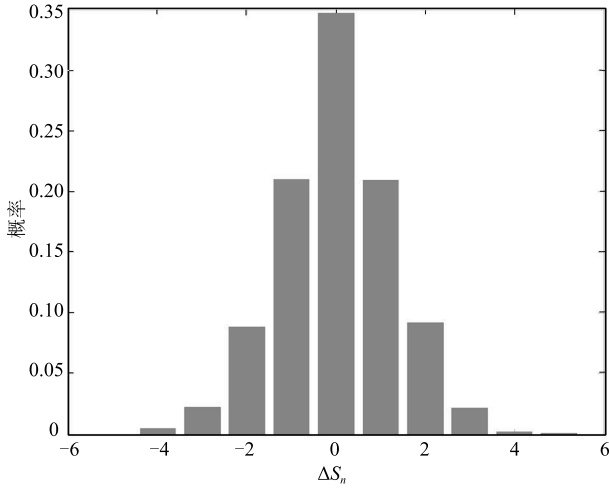


图 3 相邻两时刻纵坐标间差值 Δs 的分布概率图

Fig. 3 An example for the distribution of Δs

3 无监督的猕猴运动皮层信号的 CKF 解码

3.1 权值训练与位置估计

由上一节所述, 估计手指移动的纵坐标 y_k 需要先得到函数 $f(\cdot)$, 为此, 将式 (4b) 变为

$$\mathbf{s}_k = \mathbf{f}(\mathbf{y}_k, \mathbf{w}_k) + \mathbf{v}_k \quad (6)$$

其中, $\mathbf{a}_k = [a_{1k}, a_{2k}, \dots, a_{Nk}]^T$, $\mathbf{b}_k = [b_{1k}, b_{2k}, \dots, b_{Nk}]^T$, $\mathbf{y}_k = [y_k \ 1]^T$, $\mathbf{f}(\cdot)$ 表示 \mathbf{s}_k 与 y_k 成线性关系, 为

$$\mathbf{f}(\mathbf{y}_k, \mathbf{w}_k) = \mathbf{a}_k \times y_k + \mathbf{b}_k \quad (7)$$

将式 (7) 视为单层感知器的神经网络, 如图 4 所示, 其中, 激活函数 $\varphi(\mathbf{s}_k) = \mathbf{s}_k$. 利用 SSM 方法训练权值向量 \mathbf{w}_k , 于是给出一组关于 \mathbf{w}_k 的状态空间方程

$$\hat{\mathbf{w}}_k = \hat{\mathbf{w}}_{k-1} + \boldsymbol{\theta}_k \quad (8a)$$

$$\mathbf{s}_k = \mathbf{f}(\hat{y}_k, \hat{\mathbf{w}}_k) + \mathbf{v}_k \quad (8b)$$

其中, $\boldsymbol{\theta}_k$ 为权值噪声, 假定其均值为 $\mathbf{0}$ 、方差矩阵为 R_w , \hat{y}_k 和 $\hat{\mathbf{w}}_k$ 是 y_k 和 \mathbf{w}_k 的估计值. 得到状态空间方程 (8) 后, 采用逐次状态估计方法来训练权值 \mathbf{w}_k , 从而得到 $\mathbf{f}(\cdot)$.

值得注意的是, 通过式 (8) 训练 $\hat{\mathbf{w}}_k$ 需要知道 \hat{y}_k 的信息, 我们仍然通过一组状态空间方程求解 \hat{y}_k . 由式 (4a) 和式 (6) 构成状态方程和观测方程, 并用 \hat{y}_k 代替 y_k , 有以下状态空间方程

$$\hat{y}_k = \hat{y}_{k-1} + \omega_k \quad (9a)$$

$$\mathbf{s}_k = \mathbf{f}(\hat{y}_k, \hat{\mathbf{w}}_k) + \mathbf{v}_k \quad (9b)$$

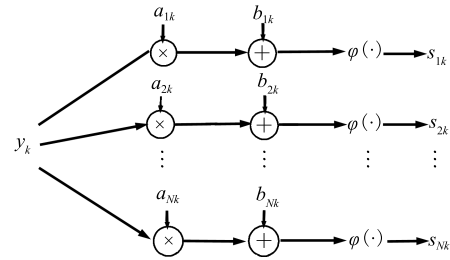


图 4 权值训练神经网络信号流

Fig. 4 Weight training neural network signals flow

同样, 由式 (9) 估计 y_k 也需要 $\hat{\mathbf{w}}_k$. UCKD 采用的卡尔曼类的逐次状态方法 (详见第 4.2 节), 其实是在高斯分布的噪声下, 使得条件概率期望值 $E[p(y_k, \mathbf{w}_k | \omega_k, \mathbf{v}_k)]$ 最大. 其过程是, 先固定状态 y_k , 求解权值 \mathbf{w}_k , 然后再固定权值 $\hat{\mathbf{w}}_k$, 求解状态 y_k , 反复迭代直到收敛, 如图 5 所示. 该过程与 EM 算法类似^[27], 由 Jensen 不等式可知, 若 $p(y_k, \mathbf{w}_k | \omega_k, \mathbf{v}_k)$ 为凸函数, 那么有 $E[p(y_k, \mathbf{w}_k | \omega_k, \mathbf{v}_k)] \geq p[E(y_k, \mathbf{w}_k | \omega_k, \mathbf{v}_k)]$, 经过反复训练, $E[p(y_k, \mathbf{w}_k | \omega_k, \mathbf{v}_k)]$ 将接近其下限值, 致使算法达到收敛.

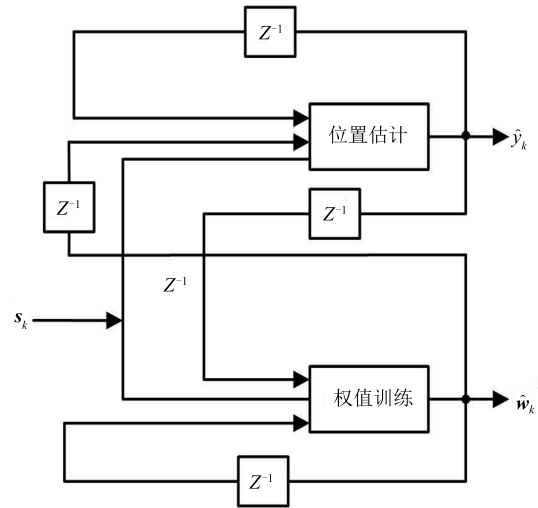


图 5 位置估计与权值训练

Fig. 5 Position estimation and weight training

3.2 积分卡尔曼滤波

通常采用逐次状态估计的方法求解权值和位置信息^[25], 例如线性卡尔曼滤波 (Kalman filter, KF)、扩展卡尔曼滤波 (Extended Kalman filter, EKF) 和粒子滤波 (Particle filter, PF) 等. 然而, KF 要求方程涉及的函数均为线性函数, 而 $\mathbf{f}(\cdot)$ 仅是关于 y_k

的线性函数而非 w_k ; EKF 可处理非线性函数, 但它是一阶泰勒展开式近似, 且需求导数, 复杂度较高; PF 也能处理非线性函数, 但需要大量粒子, 耗时较长。

本文采用 CKF^[26] 求解式 (8) 的权值训练和式 (9) 的位置估计, 同上述方法相比, CKF 不仅可以处理非线性过程, 而且保留了函数的二阶信息, 同时无需求导, 因此具有较高的近似度和较快的计算时间。CKF 基于三阶球面径向积分法则, 近似地求解高斯非线性函数的积分, 例如, 对于一个 n 维高斯分布向量 $x \in \mathbf{R}^n$, 其近似的积分规则可表示为^[26]

$$\int_{\mathbf{R}^n} h(x) \mathcal{N}(x; \mu, \Sigma) dx \approx \frac{1}{2n} \sum_{i=1}^{2n} h(\mu + \sqrt{\Sigma} \xi_i)$$

其中, $h(x)$ 表示关于向量 x 的函数, $\mathcal{N}(x; \mu, \Sigma)$ 表示为均值为 μ 、方差矩阵为 Σ 的 n 维高斯分布密度函数, $2n$ 个数值积分为

$$\xi_i = \begin{cases} \sqrt{n} e_i, & i = 1, 2, \dots, n \\ -\sqrt{n} e_{i-n}, & i = n+1, n+2, \dots, 2n \end{cases}$$

$e_i \in \mathbf{R}^n$ 表示为 n 维单位向量中的第 i 个列向量, 即除第 i 元素为 1, 其余为 0。

算法 1 和算法 2 分别给出了用 CKF 求解权值 w_k 和纵坐标 y_k 的算法过程, 均包含了时间更新和观测更新两个步骤。在算法中引入了符号 $k|k-1$ 和 $k|k$ 。 $y_{k|k-1}$ 表示给定观测值 s_1, s_2, \dots, s_{k-1} 下对 y_k 的估计值。由此, y_k 的最终估计值表示为 $y_k = y_{k|k}$ 。算法中以下参数需要确定:

- 1) 遗忘因子 λ , 取值范围 $10^{-2} \sim 10^{-5}$;
- 2) $w_{k|k}$ 的初值 w_0 , 可在 $-0.5 \sim 0.5$ 之间随机产生;
- 3) $y_{k|k}$ 的初始值 y_0 , 为 $0 \sim 1$ 之间的随机数;
- 4) 状态噪声 ω_k 的方差 Q_ω , 取值范围为 $10^{-1} \sim 10^0$;
- 5) 状态噪声 θ_k 的方差矩阵 R_w , 取值为 $\delta I_{2N \times 2N}$, 其中 δ 约为 $10^{-1} \sim 10^{-2}$;
- 6) 观测噪声矢量 v_k 的方差矩阵 R_v , 由式 (5) 可近似为 $s_k - s_{k-1}$ 的方差, 有

$$R_v \approx \text{diag} \left\{ \sum_{k=1}^{K-1} \frac{(s_{k+1} - s_k)^2}{K-1} \right\} \quad (10)$$

算法 1. 训练 w_k 的 CKF 算法

初始条件. $P_{w,0|0} = \delta I$, I 为单位矩阵, δ 约为 $10^{-1} \sim 10^{-2}$, $\hat{w}_{0|0} = \hat{w}_0$

步骤 1. 时间更新

步骤 1.1. 分解

$$P_{w,k|k} = Z_{w,k|k} Z_{w,k|k}^T$$

步骤 1.2. 计算数值积分点 ($i = 1, 2, \dots, 2n$)

$$W_{i,k|k} = Z_{w,k|k} \xi_i \cdot \lambda^{-\frac{1}{2}} + \hat{w}_{k|k}$$

步骤 1.3. 计算传播积分点 ($i = 1, 2, \dots, 2n$)

$$W_{i,k+1|k}^* = W_{i,k|k}$$

步骤 1.4. 计算预测状态

$$\hat{w}_{k+1|k} = \frac{1}{2} n \sum_{i=1}^{2n} W_{i,k+1|k}^*$$

步骤 1.5. 计算预测错误协方差

$$P_{w,k+1|k} = \frac{1}{2} n \sum_{i=1}^{2n} W_{i,k+1|k}^* W_{i,k+1|k}^{*T} - \hat{w}_{k+1|k} \hat{w}_{k+1|k}^T + R_w$$

步骤 2. 测量更新

步骤 2.1. 分解

$$P_{w,k+1|k} = Z_{w,k+1|k} Z_{w,k+1|k}^T$$

步骤 2.2. 计算数值积分点 ($i = 1, 2, \dots, 2n$)

$$W_{i,k+1|k} = Z_{w,k+1|k} \xi_i + \hat{w}_{k+1|k}$$

步骤 2.3. 计算传播积分点 ($i = 1, 2, \dots, 2n$)

$$S_{i,k+1|k} = f(Y_{k+1|k+1}, W_{i,k+1|k})$$

步骤 2.4. 计算预测观测值

$$\hat{s}_{k+1|k} = \frac{1}{2} n \sum_{i=1}^{2n} S_{i,k+1|k}$$

步骤 2.5. 计算新息协方差矩阵

$$P_{SS,k+1|k} = \frac{1}{2} n \sum_{i=1}^{2n} S_{i,k+1|k} S_{i,k+1|k}^T - \hat{s}_{k+1|k} \hat{s}_{k+1|k}^T + R_v$$

步骤 2.6. 计算互协方差

$$P_{ws,k+1|k} = \frac{1}{2} n \sum_{i=1}^{2n} W_{i,k+1|k} S_{i,k+1|k}^T - \hat{w}_{k+1|k} \hat{s}_{k+1|k}^T$$

步骤 2.7. 计算卡尔曼增益

$$G_{w,k+1} = P_{ws,k+1|k} P_{SS,k+1|k}^{-1}$$

步骤 2.8. 计算滤波状态值

$$\hat{w}_{k+1|k+1} = \hat{w}_{k+1|k} + G_{w,k+1} (s_{k+1} - s_{k+1|k})$$

步骤 2.9. 计算新息协方差矩阵

$$P_{w,k+1|k+1} = P_{w,k+1|k} - G_{w,k+1} P_{SS,k+1|k} G_{w,k+1}^T$$

算法 2. 估计 y_k 的 CKF 算法

初始条件. $P_{y,0|0} = 1$, $y_{0|0} = y_0$

步骤 1. 时间更新

步骤 1.1. 分解

$$P_{y,k|k} = Z_{y,k|k} Z_{y,k|k}^T$$

步骤 1.2. 计算数值积分点 ($i = 1, 2, \dots, 2n$)

$$\mathbf{Y}_{i,k|k} = Z_{y,k|k} \xi_i + \hat{y}_{k|k}$$

步骤 1.3. 计算传播积分点 ($i = 1, 2, \dots, 2n$)

$$\mathbf{Y}_{i,k+1|k}^* = \mathbf{Y}_{i,k|k}$$

步骤 1.4. 计算预测状态

$$\hat{y}_{k+1|k} = \frac{1}{2n} \sum_{i=1}^{2n} \mathbf{Y}_{i,k+1|k}^*$$

步骤 1.5. 计算预测错误协方差

$$P_{y,k+1|k} = \frac{1}{2n} \sum_{i=1}^{2n} \mathbf{Y}_{i,k+1|k}^* \mathbf{Y}_{i,k+1|k}^{*\top} - \hat{y}_{k+1|k} \hat{y}_{k+1|k}^T + Q_\omega$$

步骤 2. 测量更新

步骤 2.1. 分解

$$P_{y,k+1|k} = Z_{y,k+1|k} Z_{y,k+1|k}^T$$

步骤 2.2. 计算数值积分点 ($i = 1, 2, \dots, 2n$)

$$\mathbf{Y}_{i,k+1|k} = Z_{y,k+1|k} \xi_i + \hat{y}_{k+1|k}$$

步骤 2.3. 计算传播积分点 ($i = 1, 2, \dots, 2n$)

$$\mathbf{S}_{i,k+1|k} = \mathbf{f}(\mathbf{Y}_{i,k+1|k}, \mathbf{w}_{k+1|k})$$

步骤 2.4. 计算预测观测值

$$\mathbf{s}_{k+1|k} = \frac{1}{2n} \sum_{i=1}^{2n} \mathbf{S}_{i,k+1|k}$$

步骤 2.5. 计算新息协方差矩阵

$$\mathbf{P}_{SS,k+1|k} = \frac{1}{2n} \sum_{i=1}^{2n} \mathbf{S}_{i,k+1|k} \mathbf{S}_{i,k+1|k}^T - \hat{\mathbf{s}}_{k+1|k} \hat{\mathbf{s}}_{k+1|k}^T + R_v$$

步骤 2.6. 计算互协方差

$$\mathbf{P}_{ys,k+1|k} = \frac{1}{2n} \sum_{i=1}^{2n} \mathbf{Y}_{i,k+1|k} \mathbf{S}_{i,k+1|k}^T - \hat{y}_{k+1|k} \hat{\mathbf{s}}_{k+1|k}^T$$

步骤 2.7. 计算卡尔曼增益

$$\mathbf{G}_{y,k+1} = \mathbf{P}_{ys,k+1|k} \mathbf{P}_{ss,k+1|k}^{-1}$$

步骤 2.8. 计算滤波状态值

$$\hat{y}_{k+1|k+1} = \hat{y}_{k+1|k} + \mathbf{G}_{y,k+1} (\mathbf{s}_{k+1} - \hat{\mathbf{s}}_{k+1|k})$$

步骤 2.9. 计算新息协方差矩阵

$$P_{y,k+1|k+1} = P_{y,k+1|k} - \mathbf{G}_{y,k+1} \mathbf{P}_{ss,k+1|k} \mathbf{G}_{y,k+1}^T$$

3.3 正则化位置估计

通过状态空间方程 (8a), (8b) 和 (9a), (9b) 求解纵坐标 y_k 并不具有唯一解, 假定 \hat{y}_k 也为 y_k 的一个估计值, 且与期望值 y_k 成线性关系, 令

$$\hat{y}_k = c \hat{y}'_k + d \quad (11)$$

其中, c 和 d 为任意常数. 将式 (11) 代入式 (8a), 式 (8b) 和式 (9a), 式 (9b) 中, 有

$$\hat{\mathbf{w}}'_k = \hat{\mathbf{w}}'_{k-1} + \boldsymbol{\theta}_k \quad (12a)$$

$$\mathbf{s}_k = \mathbf{f}(\hat{y}'_k, \hat{\mathbf{w}}'_k) + \mathbf{v}_k \quad (12b)$$

$$\hat{y}'_k = \hat{y}'_{k-1} + \omega_k \quad (13a)$$

$$\mathbf{s}_k = \mathbf{f}(\hat{y}'_k, \hat{\mathbf{w}}'_k) + \mathbf{v}_k \quad (13b)$$

其中, $\mathbf{w}'_k = [\mathbf{a}'_k^T \ \mathbf{b}'_k^T]^T$, $\mathbf{a}'_k = c \mathbf{a}_k$, $\mathbf{b}'_k = d \mathbf{a}'_k + \mathbf{b}_k$, $\mathbf{f}(\cdot)$ 表示为

$$\mathbf{f}(\hat{y}'_k, \hat{\mathbf{w}}'_k) = \mathbf{a}'_k \times \hat{y}'_k + \mathbf{b}'_k \quad (14)$$

将新的状态空间方程 (12a), (12b) 和 (13a), (13b) 与原状态空间方程 (8a), (8b) 和 (9a), (9b) 相比, 发现两者具有相同的形式. 由此可知, 只要与期望值 y_k 成线性关系的均是其估计值, 它并不具有唯一解. 注意到式 (11)~(14) 给出了必要性的推导, 没有给出充分性, 即如果是式 (8a), 式 (8b) 和式 (9a), 式 (9b) 的解, 那么一定与期望值成线性关系. 但是, 经多次实验结果显示, 通过 UKCD 得到的解均与期望值成线性关系. 为限定估计范围, 将其规定在 $-1 \sim 1$ 之间, 这个带有约束条件的估计问题可表示为

$$\hat{y}_k = u(\mathbf{s}_k) \quad (15)$$

$$\text{s.t.} \quad -1 \leq y_k \leq 1 \quad (16)$$

其中, $u(\cdot)$ 函数表示为通过式 (8a), 式 (8b) 和式 (9a), 式 (9b) 以及算法 1 和算法 2 估计 y_k . 为满足式 (15), 在每一次估计结果 y_k 上加一个双曲正切的激活函数, 使其不会超出区间 $[-1, 1]$, 即

$$\hat{y}_k = \tanh(y_k) \quad (17)$$

由上所述, 估计值 \hat{y}_k 与期望值 y_k 成线性关系, 对 \hat{y}_k 做一线性变化可得其实际状态 \tilde{y}_k , 表示为

$$\tilde{y}_k = \frac{(y_{\max} - y_{\min})(\hat{y}_k - \hat{y}_{\min})}{\hat{y}_{\max} - \hat{y}_{\min}} + y_{\min} \quad (18)$$

其中, y_{\max} 和 y_{\min} 为猕猴手指移动位置的最大值和最小值, 由于猕猴手指的移动范围已经被限定, 因此通常为已知, 例如 $y_{\max} = L$, $y_{\min} = 0$; \hat{y}_{\max} 和 \hat{y}_{\min}

为估计值 \hat{y}_k , $k = 1, 2, \dots, K$ 中的最大值和最小值, 若 K 足够大, 则 \hat{y}_k 遍历, 那么 $\hat{y}_{\max} = 1$, $\hat{y}_{\min} = -1$.

3.4 平滑

在估计的结果中, 某些区间的数据中包含较多的高频分量. 为提高估计性能, 采用五点三次移动平均的平滑方法^[28] 消除高频分量对估计结果的影响.

3.5 UCKD 算法

算法 3. UCKD 算法

给定观测值: $\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_K$

步骤 1. 初始化 $\hat{\mathbf{w}}_{0|0} = \hat{\mathbf{w}}_0$ 和 $\hat{y}_{0|0} = \hat{y}_0$, 并确定各噪声方差 Q_ω , R_w 和 R_v .

步骤 2. 令 $k = 0, 1, \dots, K$

步骤 2.1. 由 $\hat{\mathbf{w}}_{k|k}$ 和 $\hat{y}_{k|k}$, 通过式 (9) 和算法 2 计算 $\hat{y}_{k+1|k+1}$;

步骤 2.2. 由 $\hat{y}_{k+1|k+1}$ 和 $\hat{\mathbf{w}}_{k|k}$, 通过式 (8) 和算法 1 计算 $\hat{\mathbf{w}}_{k+1|k+1}$;

步骤 2.3. 使 $\hat{y}_{k+1|k+1} = \tanh(\hat{y}_{k+1|k+1})$, 并令 $\hat{y}_{k+1} = \hat{y}_{k+1|k+1}$, $\hat{\mathbf{w}}_{k+1} = \hat{\mathbf{w}}_{k+1|k+1}$;

步骤 2.4. 由式 (17) 对 \hat{y}_{k+1} 线性变换至 \tilde{y}_{k+1} ;

步骤 2.5. 对 \tilde{y}_{k+1} 做五点三次平滑;

步骤 2.6. 若 $k = K - 1$, 结束, 否则, $k = k + 1$, 回到步骤 2.1.

4 实验结果

4.1 系统设置

实验对记录的猕猴运动皮层神经元锋电位数值及其对应手指移动位置的数据进行处理, 数据来源于文献 [1] 的公开数据¹. 具体过程为: 猕猴手指在长为 L , 宽为 W 的平面内运动, 当目标物出现, 猕猴的手指就指向目标物; 猕猴的脑部连接 N 个电极, 每隔 Δt 的采样时间就记录猕猴运动皮层的峰值电位数值, 记录时间共为 $K\Delta t$; 与此同时, 在每 Δt 时刻记录猕猴手指在移动平面内的坐标值以供验证. 实验参数 $L = 25$ cm, $W = 18$ cm; 采样周期 $\Delta t = 70$ ms; 数据长度 $K = 3000$; 每一组神经元簇向量由 $N = 42$ 个电极记录不同的神经元的峰值电位数值.

我们评价猕猴手指移动位置的估计性能采用以下方式: 从数据中随机抽取一段进行有监督的训练或拟合, 再随机抽取一段进行验证, 评价的指标采用均方根误差 (Rooted mean square error, RMSE) e_r , 定义为

$$e_r = \sqrt{\frac{1}{K} \sum_{k=1}^K (y_k - \hat{y}_k)^2} \quad (19)$$

将本文的 UCKD 方法与传统的线性拟合方法^[17-18] 和 SSM 方法进行对比, 其中 SSM 方法的逐次状态估计器采用 KF 滤波^[11-12], 涉及的参数设置:

1) 线性拟合方法

a) 训练: 将数据分为两部分, 一半用于进行有监督的拟合, 另一半用于测试;

b) 拟合: 采用最小二乘拟合得到式 (1) 的线性函数, 并用该拟合函数估计位置.

2) KF 滤波方法

a) 训练: 与线性拟合相同, 一半数据用于有监督的拟合, 另一半用于测试;

b) 拟合: 采用最小二乘拟合得到 $\mathbf{f}(y_k)$, 然后通过式 (4) 和卡尔曼滤波方法估计 y_k ;

c) 初始值 $y_0 = 0$;

d) 状态噪声 ω_k 的方差

$$Q_\omega = \frac{1}{K-1} \sum_{k=1}^K (y_{k+1} - y_k)^2$$

其中, K 为训练数据长度;

e) 观测噪声矢量 \mathbf{v}_k 的方差矩阵

$$\mathbf{R}_v = \text{diag} \left\{ \sqrt{\frac{1}{K-1} (\mathbf{s} - \hat{\mathbf{s}})(\mathbf{s} - \hat{\mathbf{s}})^T} \right\}$$

其中, $\mathbf{S} = [\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_K]$, $\hat{\mathbf{s}}_k = \mathbf{f}(y_k)$.

3) 本文方法

a) 神经网络设置: 输入层 1, 输出层 84, 激活函数 $\varphi(\mathbf{s}_k) = \mathbf{s}_k$;

b) 初始值

$$\hat{y}_0 = 0$$

$$\hat{\mathbf{w}}_{0|0} = [\text{rand}(1, 84) - 0.5] \left(\frac{84}{3} \right)^{\frac{1}{2}}$$

其中, $\text{rand}(m, n)$ 为 $0 \sim 1$ 之间均匀分布的 $m \times n$ 矩阵;

c) 各方差矩阵初始值

$$P_{y,0|0} = 1$$

$$P_{w,0|0} = 0.05 \times I_{84 \times 84}$$

d) 状态噪声 ω_k 的方差 $Q_\omega = 0.8$;

e) 观测噪声矢量 \mathbf{v}_k 的方差矩阵 \mathbf{R}_v 由式 (10) 得到;

f) 遗忘因子 $\lambda = 0.005$;

g) 权值噪声 $\boldsymbol{\theta}_k$ 的方差矩阵 $\mathbf{R}_w = 10^{-2} \times I_{84 \times 84}$.

¹<http://booksite.elsevier.com/0780123838360>

需要注意的是, 本文方法中初始值 $w_{0|0}$, 状态噪声方差, 权值噪声方差矩阵以及遗忘因子等参数的选择并不唯一, 其取值范围参见第 4.2 节, 在该取值范围内取值对算法的性能没有本质影响.

4.2 结果分析

图 6 是线性拟合方法对猕猴手指移动的纵坐标估计曲线, 随机抽取 350 个数据训练进行线性拟合 (训练数据长度采取 350 原因在于, 即使长度增加也不能显著改善算法性能), 然后对剩余数据进行测试. 图 6 给出了其中 100 个测试数据的结果图. 图中 Linear 表示线性拟合, Linerfitsmooth 表示线性拟合后再进行五点三次平滑, Actual 表示手指实际运动的纵坐标. 从图 6 可以看出, 线性拟合方法能够将猕猴手指移动位置的变化趋势估计出来, 例如, 当约第 10~20 数据点间实际轨迹向下移动, 估计曲线也向下移动; 当约第 30~40 点间实际轨迹向上移动, 估计曲线也向上移动. 但是该曲线即使经过平滑后在相邻时刻间的波动也比较大; 当约第 50~60 点间估计曲线有上下波动, 主要原因还是线性拟合把相邻时刻猕猴手指移动的过程看作独立所致.

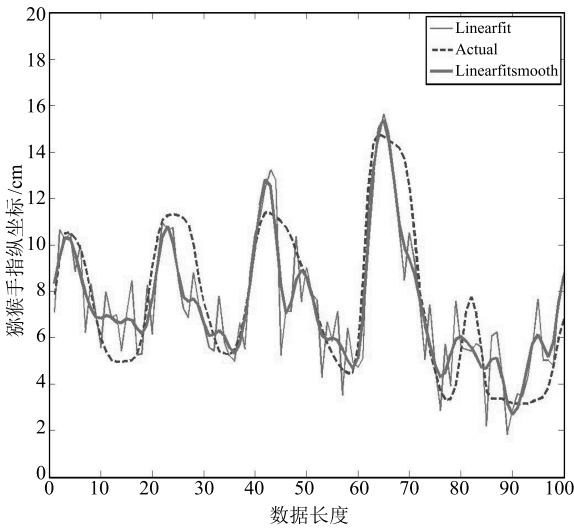


图 6 线性拟合位置估计与手指移动实际位置曲线对比
Fig.6 Linear fitting position estimation compared with fingers moving curve of actual position

图 7 给出了 KF 方法对猕猴手指移动纵坐标的估计, 其中随机抽取 350 个数据进行拟合得到线性函数, 然后再对剩余数据进行了估计, 图中仍然显示了其中 100 个点的数据. 由于该方法采用了 SSM 模型, 将手指的移动当作非独立的过程, 因此不仅能估计手指移动的趋势, 而且曲线也较线性拟合光滑. 例如, 约第 50~60 点间 KF 估计曲线的上下波动程度就比线性拟合的缓慢.

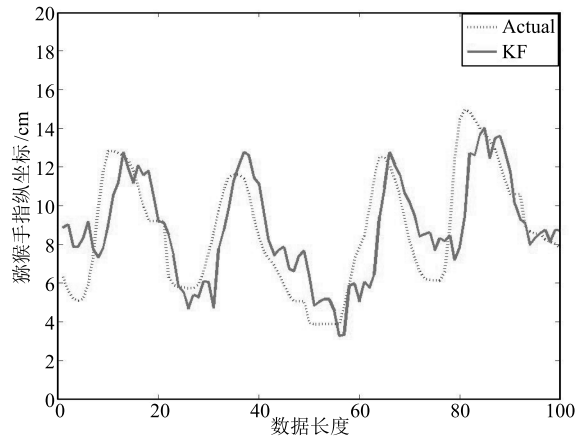


图 7 KF 位置估计与手指移动实际位置曲线对比
Fig.7 KF algorithm position estimation compared with fingers moving curve of actual position

图 8 给出了 UCKD 算法对猕猴手指移动纵坐标的估计, 由于该方法不需要有监督的数据, 因此只是随机抽取了一段长 350 的数据进行训练, 在估计过程中对该数据训练了 20 次, 然后再对该数据进行估计. 图 8 显示了数据中第 101~200 个数据的估计结果. 从图 8 可以看出, 虽然 UCKD 并没有采用有监督的数据, 但是也估计出了手指移动的趋势, 且 UCKD 方法还采用了五点三次平滑方法, 其曲线也比较平滑. 另外, 注意到在约第 50~60 点间的估计曲线有较大误差. 原因在于, 无论本文的无监督方法还是有监督的 KF 方法都是一种逐次状态估计, 即需要通过前一次的估计状态和当前的观测值去修正当前的估计状态, 它带来的效应就是当状态的趋势出现反转 (由下降突变成上升或上升变下降) 时, 估计的曲线就会出现一定的误差或震荡, 20~30 点正是趋势出现逐步改变时. 图 7 中有监督的 KF 方法在这些点上也有较大的震荡, 而且第 10 点和第 80 点附近的误差都很大, 而这些点恰恰是趋势发生改变的时候.

图 9 给出了 UCKD 算法随训练次数增加 RMSE 收敛的过程图. 图中用长度为 3000 的数据进行了训练, 训练结束后再求这 3000 个估计点的 RMSE 值. 从图 9 可以看出, 在第 2 次训练之后 RMSE 就稳定在 2.1 左右. 图 10 是在一次训练下, UCKD 随训练数据长度从 1200 变化到 3000 时, 对整个 3000 点取得估计的 RMSE 曲线. 从图 10 可以看出, RMSE 由 1200 时的 4.5 cm 降到了 3000 的 2.2 cm 左右, 说明随着数据长度增加, 估计误差逐渐减小. 从图 9 和图 10 可以发现, UCKD 算法性能与训练次数和数据的长度有关系, 训练次数或数据长度的增加均会使估计值越来越精确. 表 1 给出了 5 组交叉验证的 RMSE, 首先随机抽取长度为

500 的数据进行 UCKD 训练, 然后再随机抽取另外 500 个数据进行测试. 由表 1 可以看出, RMSE 随着训练次数的增加而减小, 并在第 6 次训练时趋向稳定.

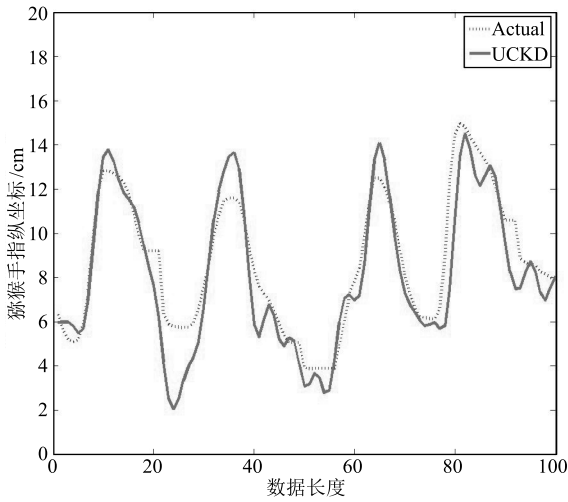


图 8 UCKD 位置估计与手指移动实际位置曲线对比
Fig.8 UCKD algorithm position estimation compared with fingers moving curve of actual position

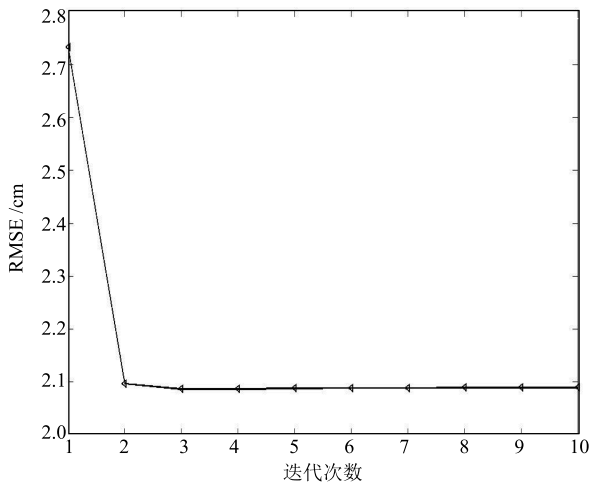


图 9 UCKD 训练次数 RMSE 变化图
Fig.9 UCKD's RMSE correlated with the number of training

为进一步评判上述三种方法的性能, 图 11 给出了它们的 RMSE 曲线, 其中线性拟合方法和 KF 方法分别抽取了长度为 50~350 的有监督数据进行函数拟合, 然后再通过测试数据进行估计, 反复进行 15 次实验, 取平均值后得到最后 RMSE 曲线. UCKD 不需要有监督数据, 对随机抽取的长度为 350 的数据分别进行 10 次和 20 次训练得到权值后, 再得到该数据估计的平均 RMSE 指标, 实验次数也是 15 次. 从图 11 可以看出, 线性拟合方法和 KF 方法的

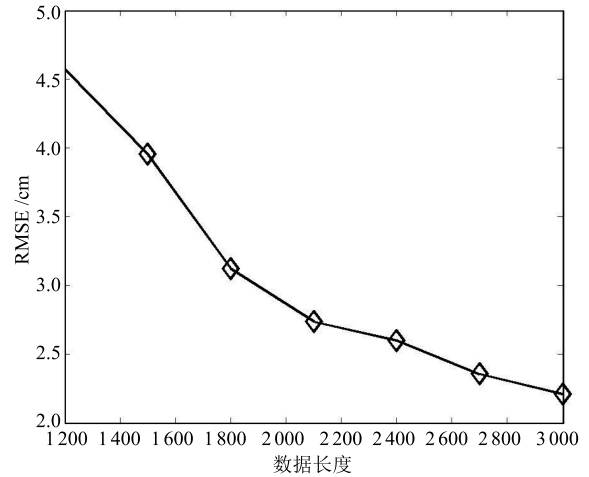


图 10 UCKD 算法随数据长度变化的位置估计 RMSE 曲线图

Fig.10 UCKD algorithm with data length change of position estimation RMSE curve

表 1 5 次训练的 RMSE (cm)

Table 1 Five times the RMSE of training (cm)

训练次数	第 1 组	第 2 组	第 3 组	第 4 组	第 5 组	均值	方差
1	3.835	4.098	4.470	3.480	4.165	4.010	0.139
2	3.207	4.154	4.293	3.045	3.708	3.681	0.307
3	2.658	4.009	3.434	2.485	2.812	3.080	0.398
4	2.203	3.144	2.493	2.277	2.248	2.473	0.153
5	2.151	2.528	2.219	2.218	2.113	2.246	0.027
6	2.144	2.347	2.173	2.194	2.080	2.188	0.010
7	2.147	2.328	2.163	2.182	2.073	2.179	0.009
8	2.153	2.342	2.158	2.178	2.065	2.179	0.010
9	2.160	2.365	2.155	2.177	2.060	2.171	0.007
10	2.166	2.389	2.153	2.179	2.058	2.170	0.007

RMSE 和训练数据的长度有关, 随着训练数据的长度由 50 增加至 150, 线性拟合和 KF 方法分别从 7 cm 减小至 2.2 cm 和从 5 cm 减小至 2 cm 左右. 这说明两种方法的估计性能依赖于有监督训练数据的长度, 当数据越多时, 拟合的函数越精确, 估计也越精确. 对于 UCKD 来说, 训练次数为 10 次和 20 次的 RMSE 分别约为 2.2 cm 和 1.9 cm 左右, RMSE 水平约等于线性拟合和 KF 方法采用有监督训练数据长度为 150 个时取得的水平. 说明 UCKD 虽然没有经过有监督数据的训练, 但是一样能取得线性拟合和 KF 方法取得的估计性能.

从图 9~11 可以看出, KF 算法采用 350 个有监督数据进行拟合, 其 RMSE 曲线就收敛到约 2 cm; 而 UCKD 需要对 350 个无监督数据训练 10

次, 或者对 3000 个数据训练一次后才得到相似的 RMSE 性能. 从时间上看, UCKD 算法收敛时间约为 KF 或线性拟合算法的 $3000/350 = 8.5$ 倍. 因此, UCKD 虽然无需有监督的数据训练, 但却以更多的无监督训练数据或训练次数为代价.

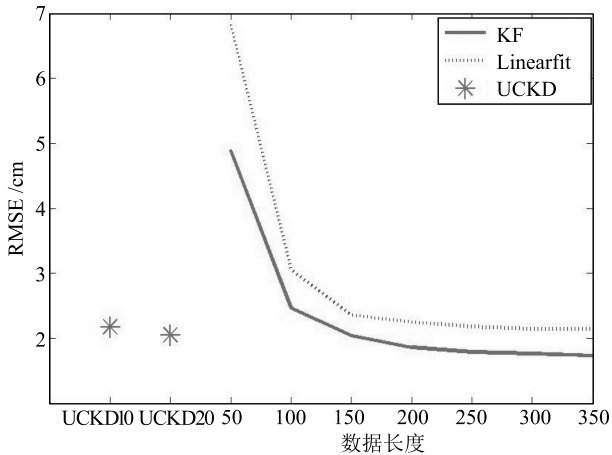


图 11 三种方法的 RMSE 变化图

Fig. 11 Three algorithm of RMSE variation

5 讨论

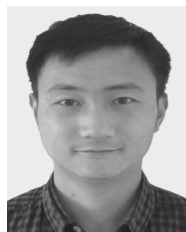
通过猕猴运动皮层神经元脉冲信号来估计其手指移动位置是一种典型的神经解码问题, 本文针对该问题提出了一种无监督积分卡尔曼滤波解码方法, 称为 UCKD 方法. 该解码方法采用神经网络得到神经元锋电位数与手指移动位置的关系权值, 然后再利用权值关系估计手指移动的位置. 为此, 构建了两组 SSM 方程, 利用逐次状态估计方法来得到权值和移动位置. 由于该 SSM 方程并不是线性, 为减少训练的复杂度和提高估计准确度, 逐次状态估计方法采用的是 CKF 滤波. 同时, 由于方程并不具有唯一解, 往往与期望值成线性关系, 还需要将该结果线性变换至手指实际移动的变化区间内, 最后, 为了提高估计精度, 采用了五点三次平滑来减少 RMSE 值. 该方法的最大的特点是无监督训练, 在只知道神经元锋电位信号的情况下依旧能够估计猕猴手指移动的位置. 相反, 现存的线性拟合以及 SSM 方法需要知道神经元锋电位信号以及与其对应的手指移动位置信息, 通过这些有监督的训练数据才能估计猕猴手指移动位置. 从实验结果看, 当采用较少的有监督数据, 现存方法与本文方法相比有较大的估计误差; 当采用较多的有监督数据, 现存方法才具有与本文的方法相近似的估计误差. 另外, 本文的 UCKD 方法是一种可同时进行训练和位置估计的方法, 随着神经元锋电位信号的长度增大, 其估计准确度会逐渐提高, 因此无需对同样一段数据进行反复训练而使

计算时间增加. 另外, 本算法需要一些先验知识. 例如, 神经元锋电位信号与猕猴手指移动之间成何关系. 我们假设其为线性关系^[2], 然后针对该假设设计单层的神经网络进行训练. 另外, 由于本文算法采用无监督学习方式, 相比于有监督算法, 其需要较多的训练次数或数据长度才能使算法收敛, 达到较低的估计误差.

References

- 1 Wallisch P, Lusignan M E, Benayoun M D, Baker T I, Dickey A S, Hatsopoulos N G. *MATLAB For Neuroscientists* (2nd edition). London: Elsevier, 2014.
- 2 Gabbiani F, Cox S J. *Mathematics for Neuroscientists*. London: Elsevier, 2010.
- 3 Kass R E, Eden U T, Brown E N. *Analysis of Neural Data*. New York: Springer, 2014.
- 4 Hatsopoulos N G, Ojakangas C L, Paninski L, Donoghue J P. Information about movement direction obtained from synchronous activity of motor cortical neurons. *Proceedings of the National Academy of Sciences of the United States of America*, 1998, **95**(26): 15706–15711
- 5 Hatsopoulos N G, Xu Q Q, Amit Y. Encoding of movement fragments in the motor cortex. *The Journal of Neuroscience*, 2007, **27**(19): 5105–5114
- 6 Moran D W, Schwartz A B. Motor cortical representation of speed and direction during reaching. *Journal of Neurophysiology*, 1999, **82**(5): 2676–2692
- 7 Reynaud-Bouret P, Rivoirard V, Grammont F, Tuleau-Malot C. Goodness-of-fit tests and nonparametric adaptive estimation for spike train analysis. *Journal of Mathematical Neuroscience*, 2014, **4**(1): 1–41
- 8 Georgopoulos A P, Kettner R E, Schwartz A B. Primate motor cortex and free arm movements to visual targets in three-dimensional space. II. Coding of the direction of movement by a neuronal population. *The Journal of Neuroscience*, 1988, **8**(8): 2928–2937
- 9 Brockwell A E, Rojas A L, Kass R E. Recursive Bayesian decoding of motor cortical signals by particle filtering. *Journal of Neurophysiology*, 2004, **91**(4): 1899–1907
- 10 Brockwell A E, Kass R E, Schwartz A B. Statistical signal processing and the motor cortex. *Proceedings of the IEEE*, 2007, **95**(5): 881–898
- 11 Wu W, Shaikhouni A, Donoghue J P, Black M J. Closed-loop neural control of cursor motion using a Kalman filter. In: *Proceedings of the 26th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. San Francisco, CA: IEEE, 2004. 4126–4129
- 12 Wu W, Gao Y, Bienenstock E, Donoghue J P, Black M J. Bayesian population decoding of motor cortical activity using a Kalman filter. *Neural Computation*, 2006, **18**(1): 80–118
- 13 Ebitz R B, Platt M L. Neuronal activity in primate dorsal anterior cingulate cortex signals task conflict and predicts adjustments in pupil-linked arousal. *Neuron*, 2015, **85**(3): 628–640

- 14 Wang W, Sudre G P, Xu Y, Kass R E, Collinger J L, Degenhart A D, Bagic A I, Weber D J. Decoding and cortical source localization for intended movement direction with MEG. *Journal of Neurophysiology*, 2010, **104**(5): 2451–2461
- 15 Campbell J, Sharma A. Visual cross-modal re-organization in children with cochlear implants. *PLoS One*, 2016, **11**(1): e0147793
- 16 Hochberg L R, Serruya M D, Friehs G M, Mukand J A, Saleh M, Caplan A H, Branner A, Chen D, Penn R D, Donoghue J P. Neuronal ensemble control of prosthetic devices by a human with tetraplegia. *Nature*, 2006, **442**(7099): 164–171
- 17 Serruya M D, Hatsopoulos N G, Paninski L, Fellows M R, Donoghue J P. Instant neural control of a movement signal. *Nature*, 2002, **416**(6877): 141–142
- 18 Warland D K, Reinagel P, Meister M. Decoding visual information from a population of retinal ganglion cells. *Journal of Neurophysiology*, 1997, **78**(5): 2336–2350
- 19 Czanner G, Eden U T, Wirth S, Yanike M, Suzuki W A, Brown E N. Analysis of between-trial and within-trial neural spiking dynamics. *Journal of Neurophysiology*, 2008, **99**(5): 2672–2693
- 20 Malik W Q, Hochberg L R, Donoghue J P, Brown E N. Modulation depth estimation and variable selection in state-space models for neural interfaces. *IEEE Transactions on Biomedical Engineering*, 2015, **62**(2): 570–581
- 21 Brown E N, Frank L M, Tang D D, Quirk M C, Wilson M A. A statistical paradigm for neural spike train decoding applied to position prediction from ensemble firing patterns of rat hippocampal place cells. *The Journal of Neuroscience*, 1998, **18**(18): 7411–7425
- 22 Shanechi M M, Wornell G W, Williams Z M, Brown E N. Feedback-controlled parallel point process filter for estimation of goal-directed movements from neural signals. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 2013, **21**(1): 129–140
- 23 Sanger T D. Neural population codes. *Current Opinion in Neurobiology*, 2003, **13**(2): 238–249
- 24 Zhang K C, Sejnowski T J. Accuracy and learning in neuronal populations. *Progress in Brain Research*, 2001, **130**: 333–342
- 25 Haykin S O. *Neural Networks and Learning Machines* (3rd edition). New Jersey: Pearson Education, Inc., 2009.
- 26 Arasaratnam I, Haykin S. Cubature Kalman filters. *IEEE Transactions on Automatic Control*, 2009, **54**(6): 1254–1269
- 27 Friedman N. The Bayesian structural EM algorithm. In: Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1998. 129–138
- 28 Chou Y L. *Statistical Analysis*. Holt International, 1975.



薛明龙 云南民族大学电气信息工程学院硕士研究生. 主要研究方向为机器学习和神经网络.

E-mail: xmlxy123@foxmail.com

(**XUE Ming-Long** Master student at the School of Electrical and Information Technology, Yunnan Minzu University. His research interest covers machine learning and neural network.)



吴海锋 云南民族大学教授. 主要研究方向为射频识别技术. 本文通信作者.

E-mail: whf5469@gmail.com

(**WU Hai-Feng** Professor at Yunnan Minzu University. His main research interest is radio frequency identification (RFID) technology. Corresponding author of this paper.)



曾玉 云南民族大学讲师. 主要研究方向为射频识别技术.

E-mail: yv.zeng@gmail.com

(**ZENG Yu** Lecturer at Yunnan Minzu University. Her main research interest is radio frequency identification (RFID) technology.)