

# 基于栈式循环神经网络的血液动力学状态估计方法

姚 垚<sup>1,2</sup> 冀俊忠<sup>1,2</sup>

**摘要** 利用 fMRI 数据准确地估计血液动力学状态,能得到一种更接近神经元层面的大脑活动的客观表示,这将促进人们对大脑运行机理的深刻理解,推动脑认知的进一步发展.迄今为止,人们已经提出了许多血液动力学状态估计方法.然而,这些方法大都只考虑了相邻时刻血液动力学状态之间的关系,忽视了更深层次的时序特征.而对模型参数先验信息的需求也使一些方法在实际应用中受到了限制.为此,本文提出了一种基于循环神经网络的血液动力学状态估计新方法.首先,利用血液动力学模型中非线性函数的反函数建立 BOLD 信号与血液动力学状态之间的映射关系,并构建模型的反演过程.然后,采用一种堆叠三个 RNN 模块的栈式神经网络结构来拟合这种映射关系,使其能够以 BOLD 信号作为输入,得到血液动力学状态的估计值.最后,在仿真数据上验证新方法的性能.实验结果表明:与一些代表算法相比,新方法能够更合理地提取 fMRI 数据中的时间特性,有效地拟合 BOLD 信号与血液动力学状态之间的动态非线性关系.

**关键词** 循环神经网络, 功能磁共振成像, 血液动力学模型, 血液动力学状态, 神经元活动

**引用格式** 姚垚, 冀俊忠. 基于栈式循环神经网络的血液动力学状态估计方法. 自动化学报, 2020, 46(5): 991–1003

**DOI** 10.16383/j.aas.2018.c170541

## Estimation of Hemodynamic States Based on Stacked Recurrent Neural Network in fMRI Time Series

YAO Yao<sup>1,2</sup> JI Jun-Zhong<sup>1,2</sup>

**Abstract** The estimation of underlying hemodynamic states from the fMRI data can provide an objective representation of the brain activity at the neuronal level, which can contribute to the understanding of the brain operation mechanism and the development of the brain cognition research. So far, many methods have been proposed for estimating hemodynamic states from the fMRI data. However, most of these methods are limited to the further consideration of the temporal characteristic in the hemodynamic model. In addition, they require the prior knowledge on hemodynamic model parameter values, which are not measurable in the actual situation. Therefore, this paper presents a new approach based on recurrent neural network (RNN) to carry out the estimation of hemodynamic states, which employs RNN to extract the temporal features inherent in fMRI time series. Firstly, the inversion process has been constructed by the inversions of nonlinear functions in the hemodynamic model, which map the BOLD signal to hemodynamic states. Then, a novel neural network architecture called stacked recurrent neural network (SRNN) is used for estimating hemodynamic states with BOLD signals by approximating the mapping relations. Finally, the experimental results on the simulated data have shown that the new approach can not only capture the temporal characteristic in the fMRI data, but also can model the nonlinear relationship between hemodynamic states dynamically.

**Key words** Recurrent neural network (RNN), functional magnetic resonance imaging (fMRI), hemodynamic model, hemodynamic state, neural activity

**Citation** Yao Yao, Ji Jun-Zhong. Estimation of hemodynamic states based on stacked recurrent neural network in fMRI time series. *Acta Automatica Sinica*, 2020, 46(5): 991–1003

近年来,脑研究已经引起了人们的广泛关注.神

经影像技术能够以图像形式表示大脑的解剖结构与功能活动,极大地推动了脑研究的发展.而在非侵入式脑成像技术中,功能磁共振成像 (Functional magnetic resonance imaging, fMRI) 因其高空间、时间分辨率被广泛应用于脑功能研究与脑疾病诊断等领域<sup>[1]</sup>.目前, fMRI 主要的成像机制是血氧水平依赖 (Blood oxygenation level dependent, BOLD) 对比: 神经元细胞激活后的供氧需求导致邻近区域内相关的血液动力学状态发生变化,而 BOLD 信号测量的则是其中脱氧血红蛋白浓度的变化<sup>[2]</sup>.并且,该过程在不同被试甚至同一被试的不同脑区上都会

收稿日期 2017-09-25 录用日期 2018-01-20  
Manuscript received September 25, 2017; accepted January 20, 2018

国家自然科学基金 (61672065, 61375059) 资助  
Supported by National Natural Science Foundation of China (61672065, 61375059)

本文责任编辑 谭莹  
Recommended by Associate Editor TAN Ying  
1. 北京工业大学信息学部多媒体与智能软件技术北京重点实验室 北京 100124 2. 北京人工智能研究院 北京 100124  
1. Beijing Municipal Key Laboratory of Multimedia and Intelligent Software Technology, Faculty of Information Technology, Beijing University of Technology, Beijing 100124 2. Beijing Artificial Intelligence Institute, Beijing 100124

有所区别<sup>[3]</sup>. 这意味着 BOLD 信号只是神经元活动的一种间接度量, 不能完全等价表示神经元活动. 而事实上, 大多数脑功能探究本质上需要的是神经元层面上的直接信息. 例如, 脑激活定位是通过神经元活动与实验任务刺激的匹配程度来判定脑区的激活程度<sup>[4]</sup>, 而脑效应连接则需要通过神经元活动之间的关系来探索它们对应脑区之间的因果关系<sup>[5]</sup>. 因此, 从 BOLD 信号中还原出真正的神经元活动能够为脑功能研究提供更为准确的大脑活动信息, 该问题也引起了脑功能领域研究者的广泛关注<sup>[6-7]</sup>.

目前, 大部分还原方法本质上都是对血液动力学模型 (Hemodynamic model, HDM)<sup>[8]</sup> 的反演计算. 血液动力学模型是由 Balloon 模型<sup>[9]</sup> 派生而来, 用于描述血液动力学过程的数学模型. 血液动力学模型使用一组非线性微分方程定量地表达了一系列血液动力学状态之间的关系, 并通过测量函数将相关的血液动力学状态映射成 BOLD 信号. 该模型不仅准确地对血液动力学过程进行了建模, 还有效地解释了 BOLD 信号中存在的非线性特征<sup>[10]</sup>. 另外, 血液动力学模型中的参数都具有实际的生理意义, 参数值在不同被试和同一被试不同脑区上的变化能够解释血液动力学过程的差异性. 血液动力学模型可以被看作一个非线性动态系统, 其输入-状态-输出形式表示如下:

$$\begin{cases} \dot{\mathbf{x}}_t = h(\mathbf{x}_t, u_t, \boldsymbol{\theta}) \\ y_t = g(\mathbf{x}_t, \boldsymbol{\theta}) \end{cases} \quad (1)$$

其中下标  $t$  表示时间,  $u$  表示系统输入, 即神经元活动,  $\mathbf{x}$  表示一系列血液动力学状态,  $\dot{\mathbf{x}}$  表示其关于时间的导数.  $\boldsymbol{\theta}$  表示血液动力学模型参数,  $y$  表示系统输出, 即 BOLD 信号.  $h$  与  $g$  均为非线性函数.

血液动力学模型使人们对血液动力学过程有了更直观的理解, 并试图利用 BOLD 信号估计相应的模型参数、血液动力学状态以及神经元活动. 2000 年, Friston 等首次尝试解决该问题, 他们使用 Volterra 核函数拟合 BOLD 信号, 并分析了 Volterra 核函数与血液动力学模型参数之间的数值关系<sup>[8]</sup>. 随后, 他们又提出一种贝叶斯估计方法, 通过计算血液动力学模型参数在高斯分布假设下的后验分布来对其进行估计<sup>[11]</sup>. Riera 等提出了一种基于局部线性滤波器 (Local linearization filter, LLF) 的方法, 对模型的参数与血液动力学状态进行估计<sup>[12]</sup>. 该方法不仅使用了 Wiener 过程对血液动力学状态的噪声进行建模, 使其具有更强的抗噪能力, 还使用了径向基函数表示神经元活动, 使其能够对任意的神经元活动进行估计. 然而, 已有方法大部分都使用线性方法拟合血液动力学过程, 而血液动力学模型中的非线性特征会使它们受到一定限制.

针对该问题, Johnston 等提出一种基于粒子滤波器 (Particle filter) 的方法, 该方法能够保留血液动力学模型中的非线性特征, 并对状态以及模型参数进行估计<sup>[13]</sup>. 类似方法还包括基于粒子滤波与平滑 (Particle filter and smoothing) 的算法<sup>[14]</sup>, 基于无迹卡尔曼滤波器 (Unscented Kalman filter, UKF) 的算法<sup>[15]</sup>. 除 Riera 等<sup>[12]</sup> 之外, 上述方法在未通过实验设计得知神经元活动的情况下, 都无法同时对神经元活动进行估计. 因此, Friston 等提出了一种基于变分贝叶斯 (Variational Bayesian) 的动态系统反演方法, 称为动态期望最大化 (Dynamic expectation maximisation, DEM) 算法, 该算法不仅能够对模型参数与血液动力学状态进行估计, 还能够估计潜在的神经元活动<sup>[16]</sup>. 随后, Havlicek 等提出了一种结合平方根容积卡尔曼滤波器 (Square-root cubature Kalman filter, SCKF) 与 Rauch-Tang-Striebel 平滑器的方法 SCKS, 用于血液动力学状态、模型参数以及神经元活动的估计<sup>[17]</sup>. 2013 年, Laleg-Kirati 等提出了一种基于观测器 (Observer-based) 的方法, 并将其应用于血液动力学模型中, 对血液动力学状态以及神经元活动进行估计<sup>[18]</sup>. 类似地, Zayane-Aissa 等提出了一种高阶滑模 (High order sliding mode, HOSM) 观测器, 来通过 fMRI 数据估计血容积与脱氧血红蛋白含量的变化<sup>[19]</sup>. Khorama 等提出了一种结合 Tikhnov 正则牛顿法与 CKF 的方法 TNM-CKF, 在一定程度上解决了 SCKS 对模型参数先验值的需求<sup>[20]</sup>. Aslan 等使用粒子平滑期望最大化 (Particle smoother expectation maximization, PSEM) 算法对血液动力学模型进行反演计算, 以达到对血液动力学状态与模型参数的联合估计<sup>[21]</sup>.

为了对 BOLD 信号与神经元活动之间的非线性关系进行有效的建模, 以上方法大都不可避免地需要使用模型参数, 而这些参数的先验信息是无法通过测量得到的, 这就使这些方法在实际应用中受到了一定的限制. 人工神经网络是一种强大的监督学习模型, 它能够通过训练自适应地学习数据中的非线性关系, 且已经成功应用在了时序预测<sup>[22]</sup> 以及模型反演<sup>[23]</sup> 等问题中. Karam 等首次将人工神经网络方法应用至血液动力学状态估计问题中, 提出了一种基于 NARX (Nonlinear auto-regressive with exogenous input) 网络的血液动力学状态估计方法<sup>[24]</sup>, 并且不需要任何模型参数的先验信息. 然而, 该方法只考虑了 BOLD 信号与血液动力学状态之间的关系, 而没有考虑血液动力学状态之间逐层递进的非线性关系. 另外, BOLD 信号是一种时间序列, 它在当前时刻的值包含了前面时刻值的一些信息, 这种时序关系是血液动力学模型内在的

一种特性, 而上述大多数非线性方法并没有充分利用这种特性. 为此, 本文提出一种基于循环神经网络 (Recurrent neural network, RNN) 的方法, 能够充分利用血液动力学模型中的时序关系, 对血液动力学状态进行逐层映射与估计. 具体来说, 首先利用血液动力学模型中非线性函数的反函数表示 BOLD 信号与血液动力学状态之间的映射关系, 构成了模型的反演过程. 然后, 采用一种新的神经网络结构, 用于拟合上述非线性映射关系, 使其能够通过 BOLD 信号估计血液动力学状态. 该结构包含三个 RNN 模块, 并通过堆栈方式连接, 称为栈式循环神经网络 (Stacked recurrent neural network, SRNN). 实验结果表明: 与代表算法相比, 新算法能够更准确、快速地对血液动力学状态进行估计, 更具有实际应用价值.

## 1 背景知识

### 1.1 血液动力学模型

血液动力学模型描述了神经元活动、血液动力学状态以及 BOLD 信号之间的动态关系. 该模型首先使用一组微分方程计算血液动力学状态的变化趋势, 然后通过一个非线性函数将血液动力学状态映射成为 BOLD 信号. 血液动力学状态在  $t$  时刻的变化趋势计算如下:

$$\begin{cases} \dot{s} = \epsilon u_t - \kappa s_t - \gamma (f_{\text{int}} - 1) \\ \dot{f}_{\text{in}} = s_t \\ \dot{v} = \tau (f_{\text{int}} - f_{\text{out}}(v_t)) \\ \dot{q} = \tau \left( f_{\text{in}} \frac{E(f_{\text{in}}, E_0)}{E_0} - f_{\text{out}}(v_t) \frac{q_t}{v_t} \right) \end{cases} \quad (2)$$

其中  $u$  表示神经元活动,  $\mathbf{x} = [s, f_{\text{in}}, v, q]^T$  表示归一化的各个血液动力学状态, 包括血脉舒张信号 (Vasodilation signal)、血流量 (Cerebral blood flow, CBF)、血容积 (Cerebral blood volume, CBV) 以及脱氧血红蛋白 (Deoxyhemoglobin, dHb),  $\dot{\mathbf{x}}$  表示其关于时刻  $t$  的导数. 另外,  $\boldsymbol{\theta} = [\epsilon, \kappa, \gamma, \tau, \alpha, E_0, V_0]^T$  表示血液动力学模型涉及到的生理参数, 这些参数的含义与作用将在下面进行详细说明.

具体来说, 当大脑中某个脑区被激活后, 该脑区中神经元活动  $u$  的产生会导致血脉舒张信号  $s$  的增强, 增强的比例由功效系数  $\epsilon$  决定. 血脉舒张信号的大小则直接决定了血液流入血管的速度  $f_{\text{in}}$ . 同时, 血脉舒张信号也会受到其自身以及血流量  $f_{\text{in}}$  的抑制, 分别由对应的衰减系数  $\kappa$  与  $\gamma$  控制.  $\tau$  为常数, 与血液穿过血管的时间相关. 伴随着血管中血液的流入与流出, 血容积  $v$  也会增大或减小. 其中血液流

出的速度由血容积与 Grubb 常数  $\alpha$  决定:

$$f_{\text{out}}(v) = v^{\frac{1}{\alpha}} \quad (3)$$

血液中脱氧血红蛋白的含量  $q$  会在神经元供氧后增加, 同时也会伴随着血液流出而降低, 其中流入的血液中的氧提取率可由下式计算:

$$E(f_{\text{in}}, E_0) = 1 - (1 - E_0)^{\frac{1}{f_{\text{in}}}} \quad (4)$$

其中  $E_0$  表示静息态下氧提取率.

最后, BOLD 信号由血容积与脱氧血红蛋白的非线性组合计算得到, 可由函数  $g$  表示如下:

$$y_t = g(v_t, q_t) = V_0 \left( k_1 (1 - q_t) + k_2 \left( 1 - \frac{q_t}{v_t} \right) + k_3 (1 - v_t) \right) \quad (5)$$

其中  $V_0$  为静息态血容积比率, 其他参数如下所示<sup>[9]</sup>:

$$\begin{cases} k_1 = 7E_0 \\ k_2 = 2 \\ k_3 = 2E_0 - 0.2 \end{cases} \quad (6)$$

血液动力学模型 (2), (5) 的连续形式可通过积分计算:

$$\begin{cases} \mathbf{x}_t = \mathbf{x}_{t-1} + \int_{t-1}^t \frac{\partial \mathbf{x}}{\partial t} dt \\ y_t = g(\mathbf{x}_t, \boldsymbol{\theta}) \end{cases} \quad (7)$$

更进一步, 该模型可以转换为离散形式, 表示如下:

$$\begin{cases} \mathbf{x}_t = \mathbf{f}(\mathbf{x}_{t-1}, u_{t-1}, \boldsymbol{\theta}) \\ y_t = g(\mathbf{x}_t, \boldsymbol{\theta}) \end{cases} \quad (8)$$

其中  $\mathbf{f}$  为函数向量, 表示一系列状态函数, 由式 (2) 可知:

$$\begin{cases} s_t = f_1(s_{t-1}, u_{t-1}, f_{\text{int}-1}, \epsilon, \kappa, \gamma) \\ f_{\text{in}t} = f_2(f_{\text{in}t-1}, s_{t-1}) \\ v_t = f_3(v_{t-1}, f_{\text{in}t-1}, \tau, \alpha) \\ q_t = f_4(q_{t-1}, f_{\text{in}t-1}, v_{t-1}, \tau, E_0, \alpha) \\ y_t = g(v_t, q_t, E_0, V_0) \end{cases} \quad (9)$$

### 1.2 循环神经网络

循环神经网络是人工神经网络的一种, 相较于传统人工神经网络, 其相邻时刻隐层神经元之间的连接使得它能够有效地记忆前面时刻的信息, 已经成功应用于手写体识别<sup>[25-26]</sup>, 语音识别<sup>[27-28]</sup> 等领域. 一种常见的 RNN 结构如图 1

所示, 由输入层、隐藏层与输出层组成. 在  $t$  时刻, 隐层状态  $\mathbf{h}_t = [h_t^1, h_t^2, \dots, h_t^{N_H}]^T$  由输入层  $\mathbf{x}_t = [x_t^1, x_t^2, \dots, x_t^{N_I}]^T$  与前一时刻隐层状态  $\mathbf{h}_{t-1}$  共同决定:

$$\mathbf{h}_t = \sigma_s (U\mathbf{x}_t + W\mathbf{h}_{t-1})$$

其中  $N_H$  和  $N_I$  分别表示隐层与输入层的神经元节点数,  $U$  表示输入层与隐层之间的连接矩阵,  $W$  表示相邻时刻隐层之间的连接矩阵,  $\sigma_s$  为隐层神经元激活函数, 常见的激活函数有 sigmoid 函数与 tanh 函数. RNN 的输出由隐藏状态变量计算得到:

$$\mathbf{o}_t = \sigma_o (V\mathbf{h}_t)$$

其中  $V$  表示隐层与输出层之间的连接矩阵,  $\sigma_o$  表示输出神经元的激活函数.

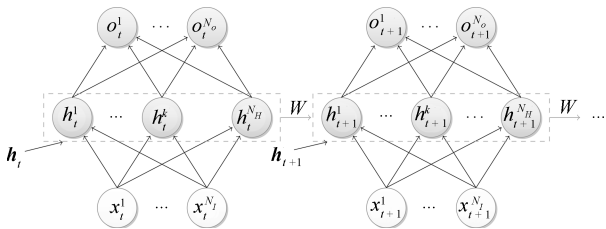


图 1 基本循环神经网络结构

Fig.1 A simple recurrent neural network

激活函数使得 RNN 能够通过建立输入与输出之间的关系  $F^{RNN}$ , 拟合任意非线性映射函数:

$$\mathbf{o}_t = F^{RNN}(\mathbf{x}_t, \mathbf{h}_{t-1}), \quad t = 1, 2, \dots, N \quad (10)$$

因此, 本文使用 RNN 对血液动力学模型中非线性函数的反函数进行拟合.

## 2 基于 SRNN 的血液动力学状态估计

为了利用 BOLD 信号准确地估计血液动力学状态, 本文提出了一种基于栈式循环神经网络的方法. 该方法分为两个步骤, 首先建立 BOLD 信号至血液动力学状态的非线性映射关系, 然后使用神经网络拟合该映射关系. 具体来说, 如式 (9) 所示, 血液动力学状态与 BOLD 信号之间的关系可以通过一系列非线性映射函数表示, 通过这些函数的反函数, 便可以建立反向映射关系. 针对式 (9) 中的非线性映射函数  $f_2, f_4$  以及  $g$ , 本文采用一种新的神经网络结构, 使用不同的 RNN 模块对它们的反函数进行拟合, 并将它们通过栈式堆叠的方式连接起来, 该结构称为 SRNN.

### 2.1 基于反函数的血液动力学模型反演

从 BOLD 信号中估计血液动力学状态的问题可以通过对血液动力学模型进行反演完成, 即通过

求解式 (8) 的反函数, 可以表示为:

$$\left\{ \begin{array}{l} \text{对于任意精度 } \varepsilon, \text{ 给定 BOLD 信号} \\ \mathbf{y} = [y_1, y_2, \dots, y_N]^T \\ \text{以及对应血液动力学状态 } X = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N], \\ \text{找到映射函数 } \hat{f} \text{ 使得:} \\ \tilde{\mathbf{x}}_t = \hat{f}(y_t, \mathbf{x}_{t-1}, \boldsymbol{\theta}), \quad t = 1, 2, \dots, N \\ \frac{1}{N} \sum_{t=1}^N \|\mathbf{x}_t - \tilde{\mathbf{x}}_t\|^2 < \varepsilon \end{array} \right. \quad (11)$$

其中  $\tilde{\mathbf{x}}_t$  表示在  $t$  时刻, 通过映射函数与 BOLD 信号得到的血液动力学状态估计值.  $N$  表示时间序列的长度.

因此, 血液动力学状态的估计可以通过求解映射函数的反函数完成. 映射函数如式 (9) 所示, 通过求解其中三个映射函数的反函数, 便可以完成由 BOLD 信号对所有血液动力学状态的估计, 其中包括测量函数  $g$  与两个状态函数  $f_2$  与  $f_4$ . 反演过程如图 2 所示, 实线部分表示反演过程中的非线性映射关系. 接下来对这三个映射函数及其反函数进行介绍.

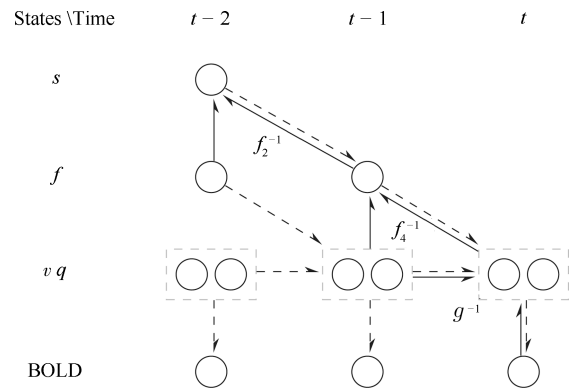


图 2 血液动力学模型反演过程

Fig.2 The inversion process of the hemodynamic model

#### 2.1.1 测量函数

测量函数  $g$  是将血容积  $v$  与脱氧血红蛋白含量  $q$  映射至 BOLD 信号的非线性函数, 其反函数  $g^{-1}$  则是由 BOLD 信号到血容积与脱氧血红蛋白含量的映射:

$$[v_t, q_t]^T = g^{-1}(y_t, \boldsymbol{\theta}), \quad t = 1, 2, \dots, N \quad (12)$$

然而, 求解二元函数的反函数是一个不适定问题, 通常不存在唯一解, 为了取得令人满意的结果, 需要在求解过程中利用其他信息.

将测量函数  $y = g(v, q, \boldsymbol{\theta})$  在  $(v_t, q_t)$  进行泰勒展开:

$$y_t = \sum_{n_1=0}^{\infty} \sum_{n_2=0}^{\infty} \frac{(v_t - v_{t-1})^{n_1} (q_t - q_{t-1})^{n_2}}{n_1! \cdot n_2!} \left( \frac{\partial^{n_1+n_2} g}{\partial v^{n_1} \cdot \partial q^{n_2}} \right) (v_{t-1}, q_{t-1}) = \bar{g}(v_t, q_t, v_{t-1}, q_{t-1}, \theta) \quad (13)$$

其中  $\frac{\partial^{n_1+n_2} g}{\partial v^{n_1} \cdot \partial q^{n_2}}$  表示函数  $g$  对  $v$  和  $q$  的  $n_1, n_2$  阶偏导数. 由测量函数的泰勒展开可知, 当前时刻的 BOLD 信号包含了当前时刻与前一时刻血容积与脱氧血红蛋白含量的信息. 因此, 利用这些信息, 我们可以建立映射关系  $\bar{g}^{-1}$ :

$$[v_t, q_t]^T = \bar{g}^{-1}(y_t, v_{t-1}, q_{t-1}), \quad t = 1, 2, \dots, N \quad (14)$$

其中,  $v_t, q_t, v_{t-1}$  与  $q_{t-1}$  分别表示当前时刻与前一时刻  $v$  和  $q$  的值.

因此, 根据映射函数  $\bar{g}^{-1}$ , 我们可以根据当前时刻的 BOLD 信号与前一时刻血容积与脱氧血红蛋白含量的估计值, 计算当前时刻的血容积与脱氧血红蛋白含量值.

### 2.1.2 状态函数

如式 (9) 所示, 状态函数  $f_3$  与  $f_4$  的反函数分别表示如下:

$$\begin{cases} f_{\text{int}-1} = f_3^{-1}(v_t, v_{t-1}) \\ f_{\text{in}t-1} = f_4^{-1}(q_t, v_{t-1}, q_{t-1}) \end{cases} \quad (15)$$

事实上, 两者在本质上并没有区别, 这是由于在使用神经网络对这些映射关系进行拟合时, 为了充分利用血液动力学模型中的时序特征, 我们使用上一个模块的隐含层作为当前模块的输入. 隐含层作为输出层的一种特征表示形式, 包含了输出层的所有信息. 这里使用  $f_4$  的反函数表示对血流量的估计, 因此, 根据映射函数  $f_4^{-1}$  可知, 我们可以使用血容积与脱氧血红蛋白含量的估计结果  $\tilde{v}$  与  $\tilde{q}$  对血流量  $f_{\text{in}}$  进行估计.

需要注意的是上述过程中涉及到血液动力学状态在时间上的不同. 其中反函数 (14) 的输出是  $t$  时刻的  $[v_t, q_t]$ , 而在式 (15) 中, 通过输入  $t$  时刻的结果后, 可以得到  $t-1$  时刻的  $f_{\text{in}}$  估计值.

同理, 通过状态函数  $f_2$  的反函数,  $t-1$  时刻的  $f_{\text{in}}$  估计值将被用于对  $t-2$  时刻的  $s$  进行估计:

$$s_{t-2} = f_2^{-1}(f_{\text{int}}, f_{\text{in}t-1}) \quad (16)$$

至此, 通过使用血液动力学模型中映射函数的反函数, 我们构建了从 BOLD 信号到所有血液动力学状态的映射关系, 从而建立了模型的反演过程, 接下来我们将使用神经网络结构对该过程进行拟合.

## 2.2 基于栈式循环神经网络的血液动力学模型反演

为了拟合上述反演过程, 本节将介绍一种新的神经网络结构. 如图 3 所示, 该结构以 RNN 为基础模块, 以时间为单元进行展开. 而 BOLD 信号与血液动力学状态也是时间序列, 因此, 该结构每个时刻的输入为 BOLD 信号在各时刻的值  $y_1, y_2, \dots, y_N$ . 输出为各血液动力学状态的估计值  $\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_N$ .

该结构由三个 RNN 模块组成, 如结构图 3 右侧所示, 每个模块可以表示为一个如式 (10) 所示的非线性映射, 用于拟合反演过程中涉及到的三个反函数 (14)~(16).

首先, 模块 1 通过拟合式 (14), 建立由 BOLD 信号到血容积与脱氧血红蛋白含量的非线性映射, 完成对这两个血液动力学状态的估计. 即模块 1 的输入为 BOLD 信号  $y$ , 输出为血容积  $v$  与脱氧血红蛋白含量  $q$ . 在时刻  $t$ , 模块 1 的隐层状态计算如下:

$$\mathbf{h}_t^{(1)} = \sigma_s^{(1)} \left( U^{(1)} y_t + W^{(1)} \mathbf{h}_{t-1}^{(1)} \right) \quad (17)$$

模块 1 的输出如下所示:

$$[\tilde{v}_t, \tilde{q}_t]^T = \sigma_o^{(1)} \left( V^{(1)} \mathbf{h}_t^{(1)} \right) \quad (18)$$

其中, 上标 (1) 表示该元素属于模块 1,  $\tilde{v}_t$  与  $\tilde{q}_t$  为由模块 1 得到的血容积与脱氧血红蛋白含量的估计值.

然后, 如式 (15) 所示, 血流量的计算需要的是血容积与脱氧血红蛋白含量的估计值, 即模块 1 的输出. 然而, 从本质上来说, 单隐层神经网络结构中隐层节点表示的是输入层与输出层的一种编码形式, 它包含了输出层的信息, 因此能够用于代替输出层作为模块 2 的输入, 即通过栈式堆叠连接两个模块, 具体计算方式如下:

$$\mathbf{h}_t^{(2)} = \sigma_s^{(2)} \left( U^{(2)} \mathbf{h}_t^{(1)} + W^{(2)} \mathbf{h}_{t-1}^{(2)} \right) \quad (19)$$

血流量  $f_{\text{in}t-1}$  的估计值可以通过下式得到:

$$\tilde{f}_{\text{in}t-1} = \sigma_o^{(2)} \left( V^{(2)} \mathbf{h}_t^{(2)} \right) \quad (20)$$

与模块 2 类似, 根据式 (16), 模块 3 对血液动力学状态  $s_{t-2}$  的估计可表示为:

$$\begin{cases} \mathbf{h}_t^{(3)} = \sigma_s^{(3)} \left( U^{(3)} \mathbf{h}_t^{(2)} + W^{(3)} \mathbf{h}_{t-1}^{(3)} \right) \\ \tilde{s}_{t-2} = \sigma_o^{(3)} \left( V^{(3)} \mathbf{h}_t^{(3)} \right) \end{cases} \quad (21)$$

最后, 通过最小化对应的均方差 (Mean square

error, MSE), 分别完成三个模块的训练.

$$\begin{cases} E_1 = \frac{1}{2N} \sum_{t=1}^N [(v_t - \tilde{v}_t)^2 + (q_t - \tilde{q}_t)^2] \\ E_2 = \frac{1}{N} \sum_{t=1}^N (f_{int} - \tilde{f}_{int})^2 \\ E_3 = \frac{1}{N} \sum_{t=1}^N (s_t - \tilde{s}_t)^2 \end{cases} \quad (22)$$

其中  $\tilde{v}_t, \tilde{q}_t, \tilde{f}_{int}$  和  $\tilde{s}_t$  分别表示各 RNN 模块的输出. 在训练神经网络的过程中, 本文使用的是 BPTT (Backpropagation through time) 算法, 该算法通过将误差反向传播到前面的时刻, 根据梯度下降的原则调整每个模块中的网络权重.

当神经网络完成训练后, 它便可以通过输入 BOLD 时间序列, 得到各个时刻血液动力学状态的估计值.

### 3 实验结果与讨论

为了对新方法的性能进行验证和分析, 本文选取了两种代表性的滤波器算法与一种神经网络方法进行比较, 分别是 DEM<sup>[16]</sup>、SCKS<sup>[17]</sup> 与 NARX 网络<sup>[24]</sup>. 本章首先介绍了数据的生成, 然后介绍了 SRNN 的训练过程, 最后从各方面对比了本文算法与代表算法的性能.

### 3.1 仿真数据

在真实 fMRI 数据中, 其对应的神经元活动与血液动力学状态是不可测量的. 因此, 通过生成仿真数据来训练 SRNN 以及验证其有效性是很有必要的. 同时, 为了保证 SRNN 能够在真实 fMRI 数据中具有实际应用价值, 需要使仿真数据能够尽可能真实地反映实际 fMRI 数据的特性. 具体来说, 生成仿真数据主要包含以下两个步骤: 首先生成神经元活动时间序列, 然后将神经元活动作为血液动力学模型的输入, 并对模型进行求解, 最后得到对应的血液动力学状态以及 BOLD 信号. 因此, 本文从神经元活动与血液动力学模型两个部分来介绍如何在生成仿真数据的同时保持其对真实 fMRI 数据的还原程度.

#### 3.1.1 神经元活动

对于血液动力学模型来说, 神经元活动  $u(t)$  是其唯一的外部输入, 在血液动力学模型参数保持不变以及不考虑系统振动的情况下, 神经元活动序列对血液动力学状态以及 BOLD 信号的变化起到了关键性的作用. 因此, 根据 fMRI 实验范式来生成对应的神经元活动是仿真数据拟合真实 fMRI 数据的关键步骤之一. 目前 fMRI 实验中, 神经元激活模式主要包括组块设计模式 (Blocked-design) 与事件相

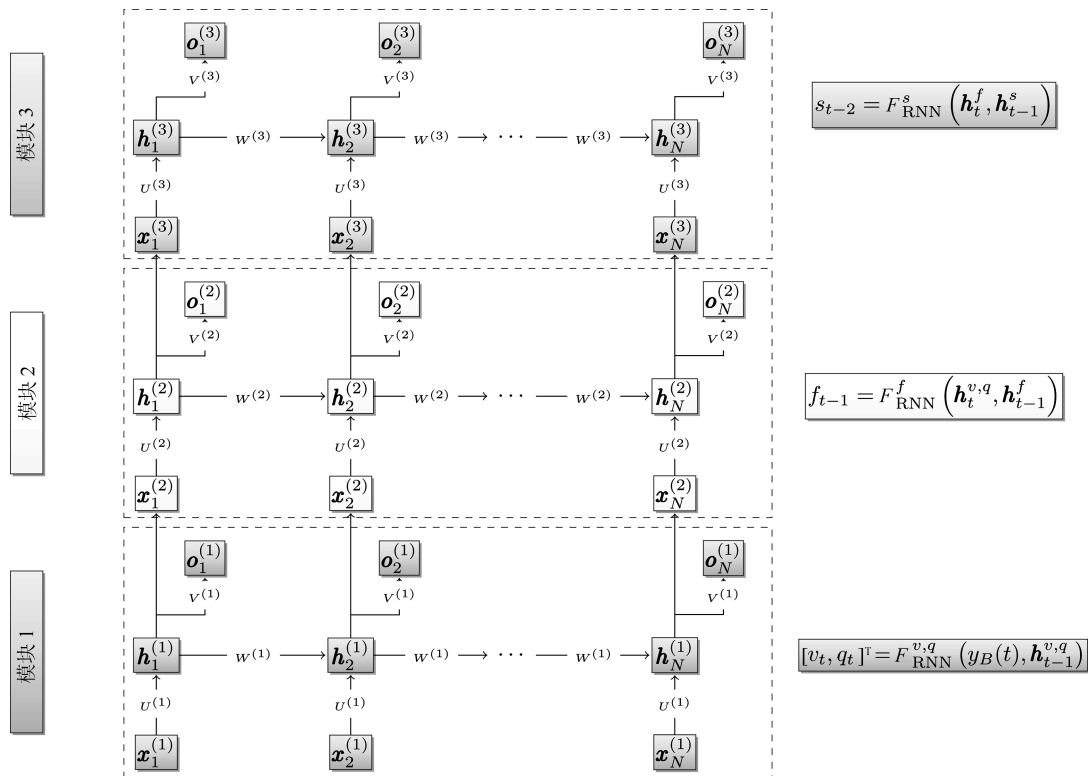


图 3 栈式循环神经网络结构  
Fig. 3 An illustration of the proposed SRNN

关 (Event-related) 模式, 分别表示对神经元的持续刺激与瞬时刺激. 其中, 事件相关模式具有更高的灵活性, 能够更真实地反映大脑的活动模式, 因此, 本文根据事件相关范式生成神经元活动.

我们使用高斯函数来表示事件相关模式中神经元的激活<sup>[17]</sup>. 首先介绍单个刺激的情况, 假设神经元在时刻  $t = a$  受到刺激, 其响应  $u(t)$  可表示为如下形式:

$$\begin{cases} v(t) = \delta(t - a) \\ m(t) = \frac{1}{\sigma} e^{-\frac{(t-\mu)^2}{4}} \\ u(t) = v(t) * m(t) \end{cases} \quad (23)$$

其中  $m(t)$  为高斯函数, 其中  $\sigma$  是用于调整最终生成的 BOLD 信号的强度的参数, 能使其与实际采集到的信号强度相近, 根据实际 fMRI 信号, 本文令  $\sigma = 8$ .  $v(t) * m(t)$  表示  $v(t)$  与  $m(t)$  的卷积, 用于描述神经元对刺激的响应与恢复静息态的过程.

为了表示更复杂的情况, 我们采用如下方式定义神经元在受到多个刺激下的时间序列:

$$v(t) = \sum_{i=1}^{N_a} \varepsilon_i \delta(t - a_i) \quad (24)$$

其中  $N_a$  表示刺激的个数,  $N_a \sim U(3, 5)$  并对其取整.  $a_i$  表示每个刺激出现的时间,  $a_1, a_2, \dots, a_{N_a}$  相互独立且有  $a_i \sim U(0, T)$ ,  $T$  为时间序列长度, 在本文中  $T = 64$ .  $\varepsilon_i$  表示每个刺激的强度,  $\varepsilon_1, \varepsilon_2, \dots, \varepsilon_{N_a}$  相互独立且有  $\varepsilon_i \sim U(0, 1)$ . 通过对上述参数  $N_a, a_i, \varepsilon_i$  进行随机采样, 便可以得到相互独立且同分布的神经元时间序列. 例如, 随机确定  $N_u = 4$ , 它们的位置为 [7; 25; 34; 56], 大小为 [1; 0.7; 0.9; 0.2], 卷积后的神经元活动如图 4 所示.

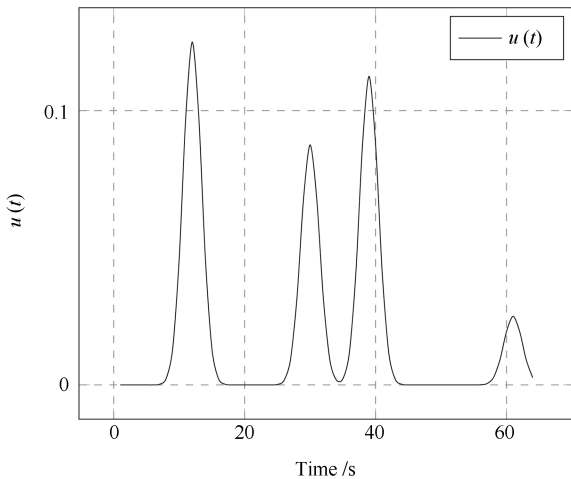


图 4 随机生成的神经元活动

Fig. 4 The random generated neural activity

通过重复采样, 便可得到各种刺激模式下神经元响应的时序.

### 3.1.2 血液动力学模型

在生成了神经元活动时间序列后, 需要选择合适的模型来生成对应的血液动力学状态以及 BOLD 信号, 用于 SRNN 的训练与验证. 血液动力学模型是描述生成过程的生理物理机制的模型, 其有效性是保证血液动力学状态及 BOLD 信号与真实 fMRI 数据的匹配程度的另一个关键因素. 血液动力学模型的具体原理已经在第 1.1 节中进行了详细的介绍, 事实上, 其对真实生理物理过程建模的有效性已经在文献 [8] 中进行了验证, 并且该模型已经在血液动力学系统反演的问题中受到了广泛的应用<sup>[11-18]</sup>. 因此, 本文通过求解以神经元活动作为外部输入的血液动力学模型, 来生成仿真数据. 另一方面, 在实际情况中, 一些血液动力学状态的值应该始终为正 (如血流量, 血容积与脱氧血红蛋白含量)<sup>[16]</sup>, 所以, 在生成数据的过程中, 我们对相应的血液动力学状态进行如下转换:

$$\begin{cases} x_1(t) = h_1(t) \\ x_i(t) = \ln h_i(t) \iff h_i(t) = \exp x_i(t), \\ i = 2, 3, 4 \end{cases}$$

因此, 血液动力学模型便转换为:

$$\begin{cases} \dot{s} = \epsilon u(t) - \kappa h_1(t) - \gamma (h_2(t) - 1) \\ \dot{f} = \frac{h_1(t)}{h_2(t)} \\ \dot{v} = \tau \frac{h_2(t) - f_{\text{out}}(h_3(t))}{h_3(t)} \\ \dot{q} = \tau \left( h_2(t) \frac{E(h_2(t), E_0)}{E_0 h_4(t)} - \frac{f_{\text{out}}(h_3(t))}{h_2(t)} \right) \\ y(t) = V_0 \left( k_1 (1 - x_4(t)) + k_2 \left( 1 - \frac{x_4(t)}{x_3(t)} \right) + k_3 (1 - x_3(t)) \right) \end{cases} \quad (25)$$

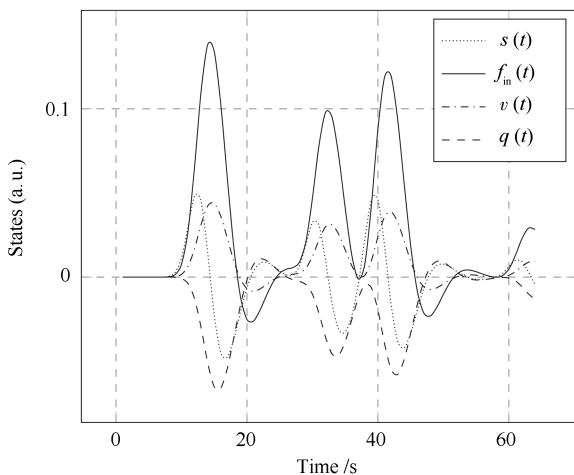
根据式 (25), 血液动力学模型可以分为两个部分: 状态转换函数与观测函数. 其中状态转换函数可以看成是一个一阶非线性微分方程组, 通过给定血液动力学状态初始值与外部输入 (神经元活动) 并对其求解, 便可得到血液动力学状态时间序列. 模型的求解使用局部线性 (Local linearization, LL) 方法<sup>[29]</sup>, 在求解血液动力学模型的过程中, 模型参数设置如表 1 所示, 最后得到的血液动力学状态如图 5 (a) 所示, 由于各状态采用不同单位, 因此表示为 arbitrary unit (a.u.).

在得到了血液动力学状态时间序列后, 通过观测函数计算对应的 BOLD 信号. 然而, 在实际 fMRI 数据采集过程中, 往往存在许多噪声, 如热效应、磁场扰动等. 因此, 根据对真实数据中噪声的估计, 在

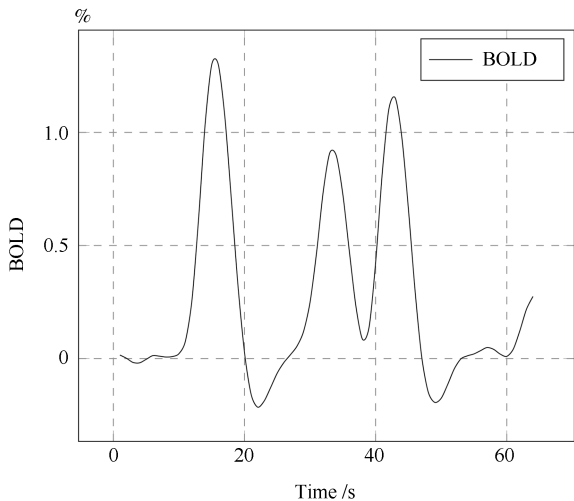
BOLD 信号中加入了方差为 0.0025 的噪声<sup>[12]</sup>.

表 1 血液动力学模型参数默认值  
Table 1 The default value of hemodynamic model parameters

Parameters	Description	Default
$\epsilon$	Neural efficiency	0.50
$\kappa$	Rate of signal decay	0.65
$\gamma$	Rate of flow dependent elimination	0.41
$\tau$	Hemodynamic transit time	0.98
$\alpha$	Grubb's exponent	0.32
$E_0$	Resting oxygen extraction fraction	0.34
$V_0$	Resting blood volume fraction	0.08



(a) 血液动力学状态  
(a) The hemodynamic states



(b) BOLD 信号  
(b) The BOLD signal

图 5 随机生成的血液动力学状态及 BOLD 信号

Fig. 5 The random generated hemodynamic states and the BOLD signal

最后, 我们便得到了仿真数据集  $u^{(k)}(t)$ ,  $x_i^{(k)}(t)$ ,  $y^{(k)}(t)$ , 其中  $t = 1, 2, \dots, 64$  表示时刻,  $i = 1, 2, 3, 4$  分别对应各个血液动力学状态变量,  $k = 1, 2, \dots, 10000$  表示生成了 10000 组样本, 其中 6000 组作为训练集, 用于 SRNN 的训练, 2000 组作为验证集, 用于对 SRNN 的参数调整, 2000 组作为测试集, 用于对 SRNN 的性能进行测试, 并与各代表算法进行比较.

### 3.2 训练神经网络

我们使用谷歌开源框架 Tensorflow 对 SRNN 进行构建与训练, 训练过程中的参数设置为: 训练批量大小为 256, 学习率为 0.1, 衰减系数为 0.96, 每迭代 200 次进行 1 次衰减, 训练过程中使用 Dropout 技术防止过拟合. 训练过程包括对三个模块的单独训练与结构调整, 其中每个模块包括参数调整与模型训练两个步骤: 1) 使用训练集进行训练, 使用验证集误差验证不同网络结构的有效性, 选择最合适的模型. 其中, 模块 1 的隐层节点数调整区间为 15 至 30 个, 模块 2 与 3 的隐层节点数调整区间为 10 至 20 个, 在保证精度接近的情况下尽可能地使用更少的节点数, 以得到更简单的模型. 2) 使用训练集对选定的模型进行训练, 直到验证集误差小于一定阈值或不再下降, 则认为该模块已经收敛. 在完成前一个模块的参数调整与训练后, 再使用前一个模块的训练结果根据第 2.2 中的规则进行下一个模块的训练.

通过验证集对模型进行调整后, 我们得到了如下网络结构: 模块 1 的隐层节点为 25, 模块 2 与模块 3 的隐层节点数为 15, 并对其进行了训练直至收敛. 最后, 使用训练得到的神经网络进行测试, 并与对比算法进行比较. 对比算法 DEM、SCKS 的参数与 SPM12 中 DEM 工具箱函数 (DEM\_demo\_hdm\_SCK.m) 相同, NARX 网络的结构与参数与原文相同<sup>[24]</sup>.

### 3.3 评估方式

为了对不同算法进行比较, 我们使用平方误差损失 (Squared error loss, SEL) 函数对算法性能进行评估, 具体如下:

$$\text{SEL}(\mathbf{x}_i) = \frac{1}{KN} \sum_{k=1}^K \sum_{t=1}^N \left( x_i^{(k)}(t) - \hat{x}_i^{(k)}(t) \right)^2, \quad i = 1, 2, 3, 4 \quad (26)$$

其中  $K$  表示测试集样本数量.  $\hat{x}_i^{(k)}(t)$  表示模型对第  $k$  个样本中第  $i$  个血液动力学状态在  $t$  时刻的估计值. 我们使用平方误差损失的对数形式  $\lg(\text{SEL})$  展示各算法的误差,  $\lg(\text{SEL})$  的值越小则表示算法的



结果越准确.

### 3.4 实验结果

#### 3.4.1 时间特性提取

在本节中, 我们测试不同的循环神经网络结构的非线性拟合能力以及对时间特性的提取能力, 包括如图 1 所示的基本 RNN 结构与长短期记忆 (Long-short term memory, LSTM) 网络<sup>[30]</sup>. RNN 与 LSTM 都属于循环神经网络, 相较于传统神经网络, 它们的优势在于能够将之前的信息应用于当前时刻的任务中. 然而, RNN 与 LSTM 的区别在于, 由于循环神经网络是以时间为单位进行展开, 误差的反向传播由当前时刻传递至前一个时刻, 梯度消失现象使得 RNN 无法学习到多个时刻之前的信息. LSTM 则通过刻意结构设计来避免这一问题, 使其能够在不需要付出额外的代价的情况下, 自动地学习到多个时刻之前的信息.

我们分别以 RNN 与 LSTM 结构为 SRNN 中的基础单元, 并使用生成的仿真数据训练集对两种 SRNN 进行训练. 在训练过程中, 两种 SRNN 对血容积  $v(t)$  估计结果的  $\lg(\text{SEL})$  随迭代过程的变化如图 6(a) 所示, 收敛后结果如图 6(b) 所示.

从收敛后的状态估计结果也可以看出, RNN 得到的结果与真实值之间的误差比 LSTM 更大. 该现象可能由以下原因导致: 1) 尽管两者使用相同的网络参数, 但是由于 LSTM 在隐层的操作更为复杂, 使其具有比 RNN 更好地非线性拟合能力, 能更好地提取血液动力学模型中的非线性特征. 2) 由血液动力学模型可知, 血液动力学状态在前后时刻存在依赖关系, BOLD 信号作为血液动力学状态的一种表现形式, 其各个时刻之间的值也存在相应的时序关系. 事实上, BOLD 响应往往在神经元被激活的数秒后到达峰值, 并且神经元的影响还会持续很长的一段时间. RNN 无法很好地处理这种长期依赖关系, 而 LSTM 是针对这种长期依赖关系刻意设计的结构, 因此能够取得更好的效果. 可以看出, 针对血液动力学模型的反演, LSTM 比 RNN 具有更好的非线性拟合能力, 并且能够更好地提取 BOLD 信号中存在的时间特性. 因此, 在接下来的实验中, 我们使用 LSTM 作为基础单元, 训练并测试 SRNN 的性能, 并与代表算法进行对比.

#### 3.4.2 血液动力学状态估计

我们使用训练集对 SRNN 进行训练, 并用测试集对比各算法的性能. 图 7 给出了 SRNN 与代表算法对各血液动力学状态在 100 个测试集样本上的平均结果, 在这里使用  $-\lg(\text{SEL})$  表示各算法的结果. 如图 8 所示, 为了更直观地展示各算法的结果, 我们在测试集的结果中, 选取了一组最接近平均值的结

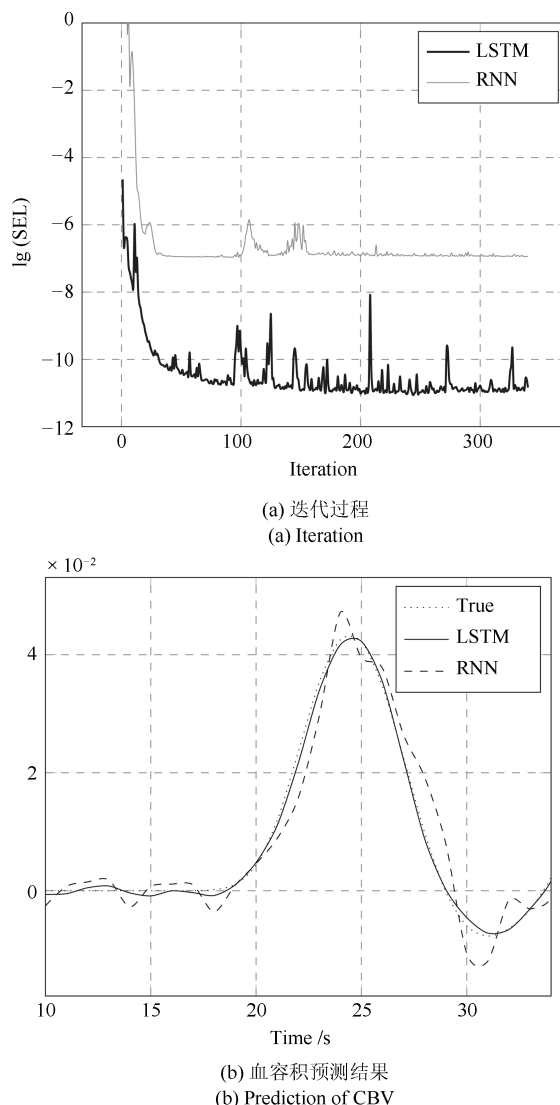


图 6 RNN 模块与 LSTM 模块结果对比  
Fig. 6 The comparison of RNN module and the LSTM module

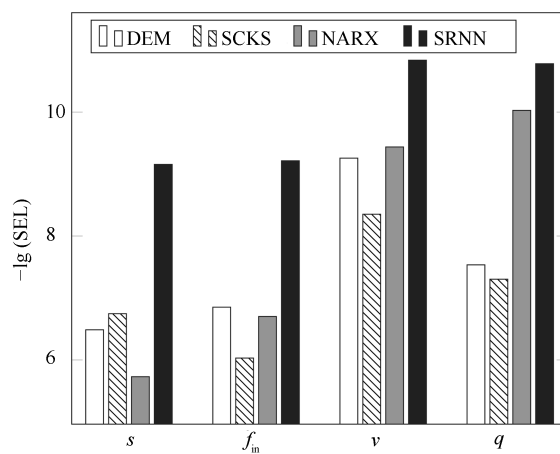


图 7 各算法对比结果  
Fig. 7 The result of different approaches

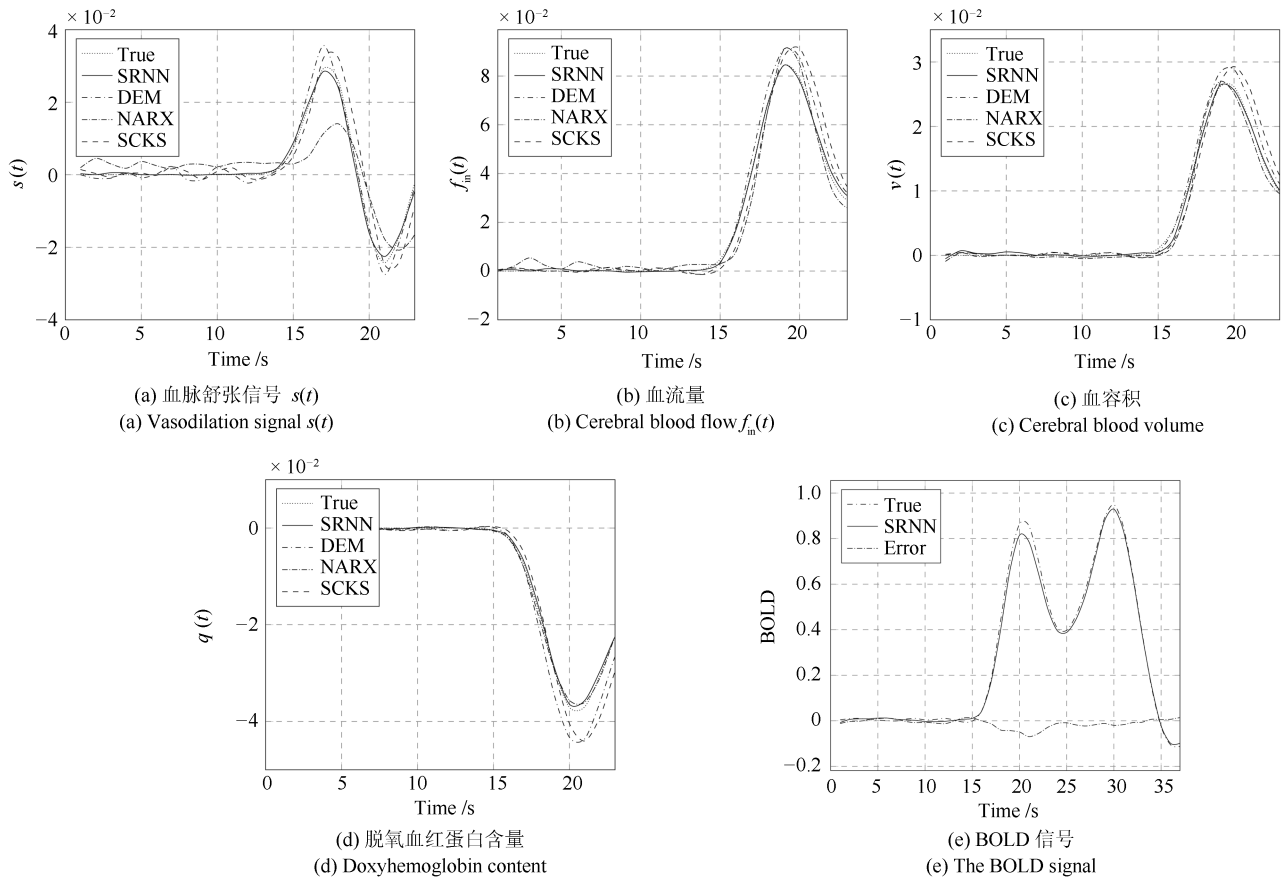


图8 DEM、SCKS 以及 SRNN 的实验结果对比

Fig. 8 The results of different approaches for estimation of hemodynamic states

果. 其中图 8(a)、8(b)、8(c)、8(d) 分别是各算法对血脉舒张信号、血流量、血容积以及脱氧血红蛋白含量的估计. 图 8(e) 表示由测量函数以及 SRNN 对血容积与脱氧血红蛋白含量的估计值计算得到的 BOLD 信号及其误差.

在对血脉舒张信号  $s(t)$ , 血流量  $f_{in}(t)$ , 血容积  $v(t)$  以及脱氧血红蛋白含量  $q(t)$  的估计中, SRNN 均取得了最好的效果, 这表明了 SRNN 能够动态地拟合血液动力学模型中的非线性特征, 而 LSTM 特殊的网络结构也能够很好地提取 BOLD 时间序列中存在的时间特性, 更准确地估计血液动力学状态.

从 NARX 网络的结果可以看出, 其对各个血液动力学状态的估计结果随着模型反演过程的递进而呈现出降低的趋势, 这是由 NARX 网络结构特性导致的. 具体来说, 它使用同一种网络用于预测不同的血液动力学状态, 输入均为 BOLD 信号. 然而, 由血液动力学模型可知, 从 BOLD 信号到不同血液动力学状态之间的非线性程度存在一定差异, 这就导致了 NARX 对血脉舒张信号与血流量的估计相对较差. SRNN 通过栈式堆叠各个模块, 用于拟合各个血液动力学状态内部的非线性关系. 从实验结果可

以看出, 其在血脉舒张信号与血流量上也取得了较好的效果, 有效地解决了这个问题.

另外, 虽然 SRNN 在血液动力学状态的估计上取得了优于代表算法的结果, 但也存在一定的误差. 由图 8 可知, 对于所有的血液动力学状态估计, 在前几个时刻的结果往往并不理想, 这是由于在训练以及测试时, 网络的隐层状态均初始化为零, 导致 SRNN 的隐层状态在前几个时刻不能得到充分的信息. 而随着时间的推移, SRNN 中隐层状态得到了足够的信息, 神经网络也趋于稳定. 这也说明了新方法能够提取血液动力学模型中的时序特征, 有助于对血液动力学状态的估计.

尽管本文并不是对神经元活动进行估计, 但准确地对相应的血液动力学状态进行估计能够为神经元活动估计提供帮助. 由状态转换函数  $f_1$  可知, 对神经元活动的估计可以由血脉舒张信号与血流量得到, 而新方法对这两个血液动力学状态的估计优于代表算法, 因此, 新方法能够得到更准确的接近神经元活动层面的信息, 为脑功能研究提供更准确的大脑活动信息, 进一步推动脑认知的发展.

### 3.4.3 时间性能比较

在本节中, 我们将对各个算法的时间性能进行比较, 并讨论它们在实际情况下的时间成本与应用价值.

首先需要说明的是, SRNN 与 NARX 算法属于监督学习算法, 需要使用大量的数据进行训练, 使神经网络能够从数据中学习出其内在的映射规律. 而 DEM 与 SCKS 则不需要进行预训练, 可以直接输入 BOLD 信号, 对血液动力学状态进行估计. 因此, 将神经网络在训练上的时间复杂度、它们的训练时间, 以及所有算法在测试集样本上运行的时间进行了比较, 如表 2 所示. 其中,  $K_D, K_{SC}, K_N, K_{SR}$  分别表示各算法的迭代次数,  $T$  表示 BOLD 时间序列的长度,  $n_s, n_p$  分别表示算法需要估计的血液动力学状态与模型参数的数量, 本文中  $n_s = 4, n_p = 7$ . 在神经网络训练时间复杂度中,  $m$  表示训练批量大小,  $n_{hn}$  与  $n_{hs}$  表示神经网络的最大隐层数.

表 2 时间性能比较

Table 2 The comparison of time performance

方法	训练复杂度	运算复杂度	训练时间 (s)	运算时间 (s)
DEM <sup>[16]</sup>	-	$O(K_D T n_s^3)$	-	16.3
SCKS <sup>[17]</sup>	-	$O(2K_{SC} T n_p^3)$	-	35.7
NARX <sup>[24]</sup>	$O(K_N T m n_{hn})$	$O(K_N T n_{hn})$	11 367	0.09
SRNN	$O(4K_{SR} T m n_{hs})$	$O(4K_{SR} T n_{hs})$	1 831	0.2

由表 2 中结果可以看出, SRNN 与 NARX 网络均需要花费一定的时间对网络进行训练, 其中 SRNN 能够以更快的速度收敛. 另一方面, 由各算法在单个样本上运行所需要的平均时间可看出, 神经网络方法具有较大的优势. 这是由于对于每一个 BOLD 时间序列, DEM 与 SCKS 方法都需要对模型参数以及血液动力学状态进行迭代估计, 每次迭代的时间与时间序列的长度以及需要估计的模型参数数量有关. 而神经网络方法在经过训练后, 将 BOLD 信号输入训练好的模型, 进行网络的前向计算便可得到其对应的血液动力学状态估计, 其运算时间则取决于神经网络的层数、每层的隐层节点数以及输入的时间序列长度等参数.

事实上, fMRI 数据是典型的高维数据, 一个被试的全脑中包含的 BOLD 时间序列个数可能高达几十万个, 经过挑选后用于功能分析的数量也仍会是一个很大的数目. 因此, 当样本数量很大时, 尽管神经网络方法需要一定的训练成本, 但是其在总体时间性能上仍然优于传统算法. 另外, 相较于 NARX 网络, SRNN 能够更快地收敛, 因此, 本文算

法在时间性能上优于其他对比算法, 具有较高的实际应用价值.

### 3.4.4 模型参数先验对代表算法的影响

本文提出的方法不需要对血液动力学模型中的模型参数进行确定, 而代表算法则会受到一定的影响. 为了探究模型参数的先验值对代表算法的限制, 我们使用不同的模型参数先验值来测试代表算法的性能. 以  $\theta$  表示模型参数的真值, 以  $\theta_D$  与  $\theta_S$  分别表示代表算法 DEM 与 SCKS 中模型参数的先验值. 我们对  $\theta_D$  与  $\theta_S$  在区间  $[0.8\theta, 1.2\theta]$  上均匀取 100 个值, 在其他参数相同的情况下运行算法, 对各血液动力学状态估计的结果如图 9 所示. 从结果可以看出, 两个代表算法均会受到其先验信息的影响, 算法的参数先验值与参数的真实值的偏差越大, 算法的误差则越大. 事实上, 不同的被试或者同一被试不同脑区上的参数都会有所区别, 这使得代表算法

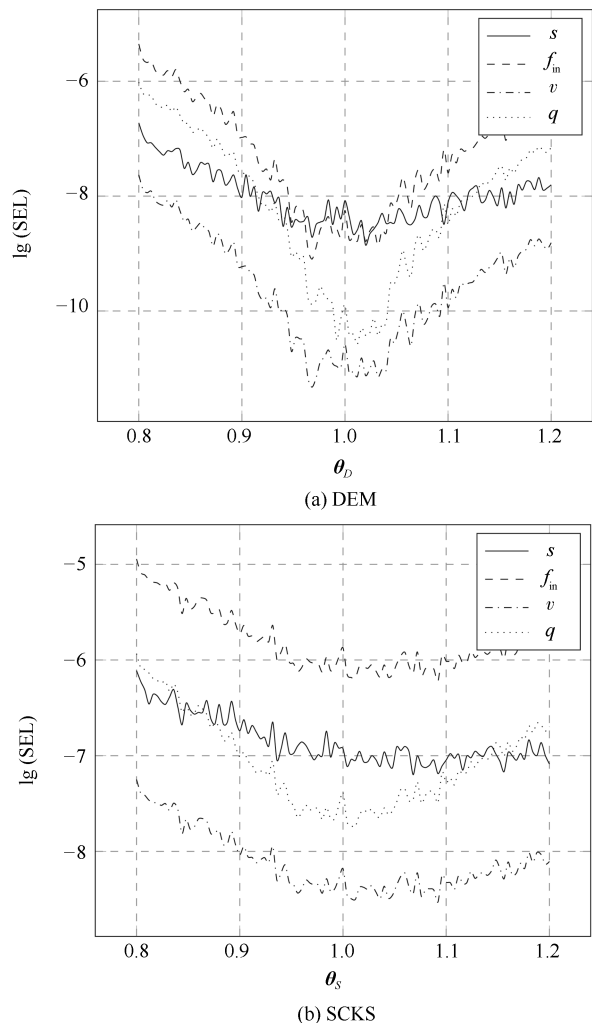


图 9 血液动力学模型参数先验值对算法 DEM 与 SCKS 性能的影响

Fig. 9 The influence of prior knowledge on model parameters on DEM and SCKS

在实际应用中受到了一定的限制. 另一方面, 由于本文算法直接通过神经网络对 BOLD 信号与血液动力学状态之间的非线性关系进行拟合, 而不需要对血液动力学模型参数进行估计, 因此能够避免先验信息带来的限制, 更具有实际应用的价值.

#### 4 结论

本文提出了一种基于栈式循环神经网络的方法, 使用 BOLD 信号对血液动力学状态进行估计. 具体来说, 首先根据血液动力学模型中映射函数的反函数建立模型的反演过程, 然后使用循环神经网络对反函数进行拟合, 并通过栈式堆叠的方式连接各 RNN 模块, 完成血液动力学状态的估计. 循环神经网络结构能够充分利用血液动力学模型中的时序关系, 有效地估计血液动力学状态. 另外, 作为人工神经网络的一种, RNN 具有强大的非线性拟合能力, 能够学习 BOLD 信号与血液动力学状态之间的非线性关系. 实验结果表明: 与代表算法相比, 新方法能够更快速、准确地估计血液动力学状态. 另外, 由于新方法不需要对血液动力学模型参数设置先验值, 因此具有更高的实际应用价值.

#### References

- 1 Logothetis N K. The underpinnings of the BOLD functional magnetic resonance imaging signal. *Journal of Neuroscience*, 2003, **23**(10): 3963–3971
- 2 Ogawa S, Lee T M, Kay A R, Tank D W. Brain magnetic resonance imaging with contrast dependent on blood oxygenation. *Proceedings of the National Academy of Sciences of the United States of America*, 1990, **87**(24): 9868–9872
- 3 Aguirre G K, Zarahn E, D'Esposito M. The variability of human, BOLD hemodynamic responses. *NeuroImage*, 1998, **8**(4): 360–369
- 4 Friston K J, Holmes A P, Worsley K J, Poline J P, Frith C D, Frackowiak R S J. Statistical parametric maps in functional imaging: a general linear approach. *Human Brain Mapping*, 1994, **2**(4): 189–210
- 5 Gitelman D R, Penny W D, Ashburner J, Friston K J. Modeling regional and psychophysiological interactions in fMRI: the importance of hemodynamic deconvolution. *NeuroImage*, 2003, **19**(1): 200–207
- 6 David O, Guillemain I, Sallet S, Reynt S, Deransart C, Segebarth C, et al. Identifying neural drivers with functional MRI: an electrophysiological validation. *PLoS Biology*, 2008, **6**(12): e315
- 7 Roebroeck A, Formisano E, Goebel R. The identification of interacting networks in the brain using fMRI: model selection, causality and deconvolution. *NeuroImage*, 2011, **58**(2): 296–302
- 8 Friston K J, Mechelli A, Turner R, Price C J. Nonlinear responses in fMRI: the Balloon model, Volterra kernels, and other hemodynamics. *NeuroImage*, 2000, **12**(4): 466–477
- 9 Buxton R B, Wong E C, Frank L R. Dynamics of blood flow and oxygenation changes during brain activation: the balloon model. *Magnetic Resonance in Medicine*, 1998, **39**(6): 855–864
- 10 Friston K J, Josephs O, Rees G, Turner R. Nonlinear event-related responses in fMRI. *Magnetic Resonance in Medicine*, 1998, **39**(1): 41–52
- 11 Friston K J. Bayesian estimation of dynamical systems: an application to fMRI. *NeuroImage*, 2002, **16**(2): 513–530
- 12 Riera J J, Watanabe J, Kazuki I, Naoki M, Aubert E, Ozaki T, et al. A state-space model of the hemodynamic approach: nonlinear filtering of BOLD signals. *NeuroImage*, 2004, **21**(2): 547–567
- 13 Johnston L A, Duff E, Mareels I, Egan G F. Nonlinear estimation of the BOLD signal. *NeuroImage*, 2008, **40**(2): 504–514
- 14 Murray L, Storkey A. Continuous time particle filtering for fMRI. In: *Proceedings of the 20th International Conference on Neural Information Processing Systems*. British Columbia, Canada: Curran Associates Inc. 2007. 1049–1056
- 15 Hu Z H, Zhao X H, Liu H F, Shi P C. Nonlinear analysis of the BOLD signal. *EURASIP Journal on Advances in Signal Processing*, 2009, **2009**: Article No. 1
- 16 Friston K J, Trujillo-Barreto N, Daunizeau J. DEM: a variational treatment of dynamic systems. *NeuroImage*, 2008, **41**(3): 849–885
- 17 Havlicek M, Friston K J, Jan J, Brazdil M, Calhoun V D. Dynamic modeling of neuronal responses in fMRI using cubature Kalman filtering. *NeuroImage*, 2011, **56**(4): 2109–2128
- 18 Laleg-Kirati T M, Arabi H, Tadjine M, Zayane C. Estimation of the neuronal activation using fMRI data: An observer-based approach. In: *Proceedings of the 2013 American Control Conference (ACC)*. Washington DC, USA: IEEE, 2013. 5457–5461
- 19 Zayane-Aissa C, Laleg-Kirati T M. A sliding mode observer for hemodynamic characterization under modeling uncertainties. In: *Proceedings of the 22nd Mediterranean Conference on Control and Automation (MED)*. Palermo, Italy: IEEE, 2014. 1488–1493
- 20 Khoram N, Zayane C, Djellouli R, Laleg-Kirati T M. A novel approach to calibrate the hemodynamic model using functional Magnetic Resonance Imaging (fMRI) measurements. *Journal of Neuroscience Methods*, 2016, **262**: 93–109
- 21 Aslan S, Cemgil A T, Akin A. Joint state and parameter estimation of the hemodynamic model by particle smoother expectation maximization method. *Journal of Neural Engineering*, 2016, **13**(4): 046010

- 22 Coyle D, Prasad G, McGinnity T M. A time-series prediction approach for feature extraction in a brain-computer interface. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 2005, **13**(4): 461–467
- 23 Junior J L, da Silva Neto A J, Neto L B, da Cunha Pires Soeiro F J, Santana C C, Velho H F C. Application of artificial neural networks and hybrid methods in the solution of inverse problems. *Artificial Neural Networks — Application*. Rijeka, Croatia: InTech, 2011. 541–566
- 24 Karam A M, Laleg-Kirati T M, Zayane C, Kashou N H. Nonlinear neural network for hemodynamic model state and input estimation using fMRI data. *Biomedical Signal Processing and Control*, 2014, **14**: 240–247
- 25 Graves A, Liwicki M, Fernández S, Bertolami R, Bunke H, Schmidhuber J. A novel connectionist system for unconstrained handwriting recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2009, **31**(5): 855–868
- 26 Jin Lian-Wen, Zhong Zhuo-Yao, Yang Zhao, Yang Wei-Xin, Xie Ze-Cheng, Sun Jun. Applications of deep learning for handwritten Chinese character recognition: a review. *Acta Automatica Sinica*, 2016, **42**(8): 1125–1141  
(金连文, 钟卓耀, 杨钊, 杨维信, 谢泽澄, 孙俊. 深度学习在手写汉字识别中的应用综述. *自动化学报*, 2016, **42**(8): 1125–1141)
- 27 Sak H, Senior A, Beaufays F. Long short-term memory recurrent neural network architectures for large scale acoustic modeling. In: Proceedings of the 2014 ISCA. Singapore, 2014. 338–342
- 28 Liu Wen-Ju, Nie Shuai, Liang Shan, Zhang Xue-Liang. Deep learning based speech separation technology and its developments. *Acta Automatica Sinica*, 2016, **42**(6): 819–833  
(刘文举, 聂帅, 梁山, 张学良. 基于深度学习语音分离技术的研究现状与进展. *自动化学报*, 2016, **42**(6): 819–833)
- 29 Jimenez J C, Shoji I, Ozaki T. Simulation of stochastic differential equations through the local linearization method.

A comparative study. *Journal of Statistical Physics*, 1999, **94**(3–4): 587–602

- 30 Hochreiter S, Schmidhuber J. Long short-term memory. *Neural Computation*, 1997, **9**(8): 1735–1780



**姚焱** 北京工业大学信息学部博士研究生。2014 年获得北京工业大学理学学士学位。主要研究方向为计算智能, 深度学习和脑科学。E-mail:

yaoyao1314@emails.bjut.edu.cn

(**YAO Yao** Ph.D. candidate at the the Faculty of Information Technology Beijing University of Technology. He

received his bachelor degree from Beijing University of Technology in 2014. His research interest covers computation intelligence, deep learning and brain science.)



**冀俊忠** 北京工业大学教授。2004 年获北京工业大学计算机应用技术专业博士学位, 2005 年和 2010 年分别在挪威科技大学、纽约州立大学布法罗分校做访问学者。主要研究方向为机器学习, 计算智能, 生物信息学和脑科学。本文通信作者。E-mail: jjz01@bjut.edu.cn

(**JI Jun-Zhong** Professor at Beijing University of Technology. He received his Ph.D. degree in computer science and application technology from Beijing University of Technology in 2004. He was a visiting scholar at Norwegian University of Science and Technology in 2005 and State University of New York at Buffalo in 2010, respectively. His research interest covers machine learning, computation intelligence, bioinformatics and brain science. Corresponding author of this paper.)