

# 基于混合特征提取的流数据概念漂移处理方法

郭虎升<sup>1,2</sup> 刘艳杰<sup>1</sup> 王文剑<sup>1,2</sup>

<sup>1</sup>(山西大学计算机与信息技术学院 太原 030006)

<sup>2</sup>(计算智能与中文信息处理教育部重点实验室(山西大学) 太原 030006)

## Concept Drift Processing Method of Streaming Data Based on Mixed Feature Extraction

Guo Husheng<sup>1,2</sup>, Liu Yanjie<sup>1</sup>, and Wang Wenjian<sup>1,2</sup>

<sup>1</sup>(School of Computer and Information Technology, Shanxi University, Taiyuan 030006)

<sup>2</sup>(Key Laboratory of Computational Intelligence and Chinese Information Processing (Shanxi University), Ministry of Education, Taiyuan 030006)

**Abstract** In the era of big data, more and more data are generated in the form of data streams, which makes concept drift an important but difficult problem in streaming data mining due to its fast, infinite, unstable and dynamically changing characteristics. Most of the current concept drift processing methods have limited information extraction capability and do not fully consider the temporal features of streaming data. To address these problems, a concept drift processing method of streaming data based on mixed feature extraction (MFECD) is proposed. The method first uses convolutional kernels of different scales to model the data to construct splicing features, and uses a gating mechanism to fuse shallow inputs and splicing features for adaptive integration as different network level inputs to obtain data features that can take into account both detailed and semantic information. Based on this, attention mechanism and similarity calculation are used to evaluate the importance of stream data at different moments in order to enhance the temporal features of key site of the data stream. The experimental results show that our method can effectively extract the complex data features and temporal features contained in the streaming data, and improve the processing capability of concept drift in the data stream.

**Key words** streaming data; concept drift; feature fusion; attention mechanism; sample feature; temporal feature

**摘要** 大数据时代,越来越多的数据以数据流的形式产生,由于其具有快速、无限、不稳定及动态变化等特性,使得概念漂移成为流数据挖掘中一个重要但困难的问题。目前多数概念漂移处理方法存在信息提取能力有限且未充分考虑流数据的时序特性等问题。针对这些问题,提出一种基于混合特征提取的流数据概念漂移处理方法 (concept drift processing method of streaming data based on mixed feature extraction, MFECD)。该方法首先采用不同尺度的卷积核对数据进行建模以构建拼接特征,采用门控机制将浅层输入和拼接特征融合,作为不同网络层次输入进行自适应集成,以获得能够兼顾细节信息和语义信息的数据特性。在此基础上,采用注意力机制和相似度计算评估流数据不同时刻的重要性,以增强数据流关键位点的时序特性。实验结果表明,该方法能有效提取流数据中包含的复杂数据特征和时序特征,提高了数据流中概念漂移的处理能力。

收稿日期: 2023-03-23; 修回日期: 2023-07-18

基金项目: 国家自然科学基金项目(62276157, U21A20513, 62076154, 61503229); 山西省重点研发计划项目(202202020101003)

This work was supported by the National Natural Science Foundation of China (62276157, U21A20513, 62076154, 61503229) and the Key Research and Development Program of Shanxi Province (202202020101003).

通信作者: 王文剑([wjwang@sxu.edu.cn](mailto:wjwang@sxu.edu.cn))

关键词 流数据; 概念漂移; 特征融合; 注意力机制; 样本特征; 时序特征

中图法分类号 TP18

随着科学技术的快速发展,许多数据以数据流的形式快速产生,例如传感器产生的实时数据、用户的网络社交数据和某个地区的天气数据等,这些源源不断的数据构成了流数据(streaming data)。与静态离线数据相比,流数据具有快速、无限、不稳定及动态变化等特性<sup>[1-3]</sup>。

在流数据挖掘任务中,实际目标场景往往处于不断变化的状态,例如传感器数据会随着设备老化而发生缓慢变化,用户的聊天主题会随着用户喜好而发生改变,天气会随着季节更替而发生周期性变化,这些都导致所产生的流数据具有不稳定性,即不同时间段数据流分布特征不再满足独立同分布条件,此时数据流的联合概率分布会随着时间推移而发生变化,这种现象叫“概念漂移”<sup>[4-5]</sup>,其中,“概念”可以理解为某一时刻所有样本的分布情况。要学习的概念或者函数被称为目标概念,可以用数据的联合概率分布  $P(x, y)$  表示,其中  $x$  表示样本的  $d$  维特征向量,  $y$  表示对应样本的标签。若当前时刻  $t$  发生了概念漂移,可以将其形式化表示为:  $\exists x: P_{t-1}(x, y) \neq P_t(x, y)$ 。概念漂移是流数据挖掘在现实世界中的一个重要特性,也是流数据分析挖掘中不可避免的难点问题,打破了传统机器学习中类别标签分布固定的假设,其典型特征是实时数据分布与训练数据分布不一致,并已受到越来越多的关注和研究。

传统的概念漂移处理方法大都基于简单的非线性分类器,如支持向量机、决策树和朴素贝叶斯等。简单分类器对于复杂流数据的拟合能力较差,且无法快速适应概念漂移发生后流数据的分布变化。近年来,神经网络技术在图像分割、目标检测和文本翻译等领域得到了广泛应用。与传统方法相比,神经网络具备较强的学习能力,并且神经网络层数很多、宽度很广,对复杂数据变化的适应性较强,因此神经网络是处理流数据概念漂移的一种有效方法。

为提高流数据信息提取能力且充分考虑流数据所包含的时序特性,本文提出了一种基于混合特征提取的流数据概念漂移处理方法(concept drift processing method of streaming data based on mixed feature extraction, MFECD)。该方法首先采用不同尺度的卷积核对流数据进行建模以获得拼接特征,采用门控机制融合浅层输入与拼接特征,然后自适应集成不同层次的数据信息以获得复杂的数据特征。在

此基础上,通过时间注意力机制和相似度计算,获得对流数据关键位点增强的时序特性。本文的主要贡献有2点:

1) 提取不同尺度和不同层次的数据特性,提高了对数据流演化特性的学习能力和对分布变化的适应能力。

2) 使用时间注意力机制和相似度计算评估流数据不同时刻的重要性,增强了关键位点的时序特性,提高了模型对漂移的处理能力。

## 1 相关工作

目前针对流数据概念漂移处理的常见策略大体上可分为基于主动检测的方法和基于被动自适应的方法<sup>[6]</sup>。

基于主动检测的概念漂移处理方法通过引入概念漂移检测机制,对流数据分布的稳定性进行检测或者通过模型实时性能指标的变化来判断是否有概念漂移发生。当检测到流数据中发生概念漂移时,通过触发相应的概念漂移处理机制适应新的数据分布。常见的主动检测方法包括基于滑动窗口的处理方法、基于模型性能的处理方法和基于数据分布的处理方法等。基于滑动窗口的方法采用单个或多个滑动窗口来存储处理数据,使当前滑动窗口中容纳最新数据分布的样本,通过不断向前滑动窗口与旧分布数据相比判断是否有概念漂移发生。典型的如:基于自适应窗口的方法<sup>[7]</sup>、使用自适应滑动窗口上的熵方法<sup>[8]</sup>、基于多窗口协同滑动的方法<sup>[9-10]</sup>。基于模型性能的概念漂移处理方法大都基于模型的分类性能,当检测到模型性能发生较大程度下降时,表明流数据中发生了概念漂移,进而调整模型适应新的数据分布<sup>[11-14]</sup>。基于数据分布的概念漂移处理方法使用统计学方法检验最新样本与过去时刻样本的统计学差异性,如果零假设(两者之间数据分布是相同的)被拒绝,则表明当前时刻存在概念漂移<sup>[15-16]</sup>。

基于被动自适应的概念漂移处理方法并不使用漂移检测机制来判断流数据中是否发生概念漂移,而是通过不断调整模型以适应新的数据分布。现有的被动自适应方法绝大多数基于集成学习思想,按照学习单元大小可分为基于数据块的集成和基于单个数据的集成。基于数据块的集成方法为每个新到达

的数据块建立一个基分类器,通过不断添加到集成模型中替换性能最差的基分类器,以适应最新的数据分布<sup>[17-19]</sup>. 基于单个数据的集成方法首先初始化一组基分类器,每次仅处理1个样本,并对集成模型进行在线更新,同时对基分类器进行加权组合<sup>[20-21]</sup>.

虽然文献[6-21]方法在处理概念漂移方面发挥了重要作用,但这些方法大多采用线性或简单的非线性分类器,表示能力较差,无法处理复杂的非线性数据流挖掘问题. 目前,已有研究采用深度学习解决复杂分布式流数据中的概念漂移问题. 根据网络结构是否可以动态调整,现有的方法可以分为基于动态调整的在线深度学习和基于稳定结构的在线深度学习. 前者不断调整模型结构以适应最新的数据分布,并根据模型在最新实例上的表现控制模型的自适应率,典型的方法主要有ADL<sup>[22]</sup>, Spiking-NN<sup>[23]</sup>, SEOA<sup>[24]</sup>. 后者不直接调整模型结构,而是采用集成学习,不断调整基分类器的权值,以适应带有概念漂移的流数据,典型的方法主要有HBP<sup>[25]</sup>和NN-Patching<sup>[26]</sup>. 这些方法利用深度学习处理流数据,可以处理复杂的非线性分布数据,但在概念漂移发生时,容易受到神经网络固有缺陷的影响,对复杂流数据的信息提取能力有限,无法快速适应漂移发生时的最新数据分布,且在概念漂移处理过程中忽略了流数据所包含的时序特性.

本文提出了MFECD方法,与传统方法相比,该方法首先自适应融合不同尺度和不同层次的数据特征,弥补概念漂移丢失的信息;又通过时间注意力和相似度计算获取流数据包含的时序特征,充分考虑流数据中包含的时间上下文信息,利用流数据中的混合特征来解决流数据中的概念漂移问题,提高概念漂移的处理能力.

## 2 基于混合特征提取的流数据概念漂移处理

本文提出的MFECD方法首先采用不同尺度的卷积核提取到样本的拼接特征,利用门控机制将浅层输入与拼接特征融合得到样本的多尺度融合特征,并将其作为网络不同层次的输入进行在线自适应集成,最终得到复杂数据特征. 在此基础上,使用时间注意力机制和相似度计算,获得流数据时序特征的同时对其中重要时刻的特征进行有效增强. 利用复杂数据特征和时序特征来提高流数据的信息提取能力和概念漂移处理能力. MFECD方法的整体结构如图1所示.

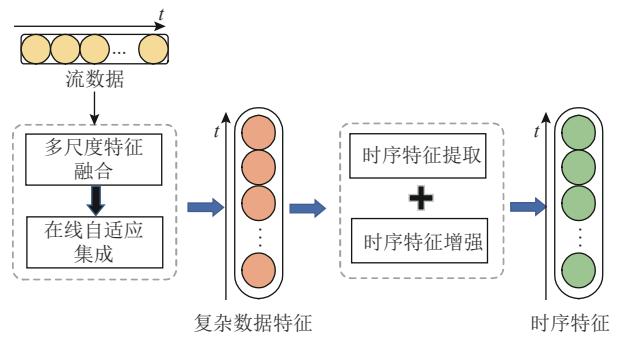


Fig. 1 Overall structure diagram of MFECD method

图1 MFECD方法整体结构图

### 2.1 基于自适应集成的数据特征融合

神经网络在许多领域已经得到了广泛应用. 虽然可以使用神经网络作为基本模型处理流数据,但神经网络的特征提取能力由固定个数的神经元决定,神经网络的每一层仅能提取固定尺度的特征,导致平稳状态下对复杂流数据的特征提取能力有限. 同时,在概念漂移发生之后,不同层次的神经网络集成时无法快速收敛适应新的数据分布,导致对概念漂移问题的处理性能较差.

首先采用不同尺度的卷积核对样本数据进行特征提取得到拼接特征,使用注意力机制评估不同尺度特征的重要性,并采用门控机制控制不同信息流的输入比例. 在此基础上在线自适应集成不同层次的数据特征. 这样不仅有效提高了神经网络在平稳阶段对流数据的特征信息提取能力,而且能够依据当前时刻的数据分布变化自适应更新融合权重,从而避免不同层次神经网络集成时存在的无法快速收敛问题.

如图2所示,对于当前时刻的输入数据 $\mathbf{x}$ ,首先将其输入3个大小分别为1, 3, 5的深度可分离卷积层<sup>[27]</sup>,得到3个不同尺度的特征 $\mathbf{x}_1, \mathbf{x}_3, \mathbf{x}_5$ ,将得到的3个特征拼接得到拼接特征 $\mathbf{m}$ :

$$\mathbf{m} = \text{Concat}(\mathbf{x}_1, \mathbf{x}_3, \mathbf{x}_5), \quad (1)$$

其中 $\text{Concat}$ 表示按通道拼接. 为了减少参数计算量,对拼接特征 $\mathbf{m}$ 进行卷积运算,将拼接特征的通道进行压缩,得到压缩特征 $\mathbf{c}$ .

本文引入通道注意力机制<sup>[28]</sup>处理不同通道的资源分配关系,由于每个通道都被认为是一个特征检测器. 对于压缩特征 $\mathbf{c}$ ,每个通道对应数据的不同尺度信息,因此需要对重要通道进行关注,向其分配更多的资源. 对压缩特征 $\mathbf{c}$ 使用通道注意力,利用全局最大池化和全局平均池化进行权重分析,在通道维度上对压缩特征进行权重分配,将得到的通道权重

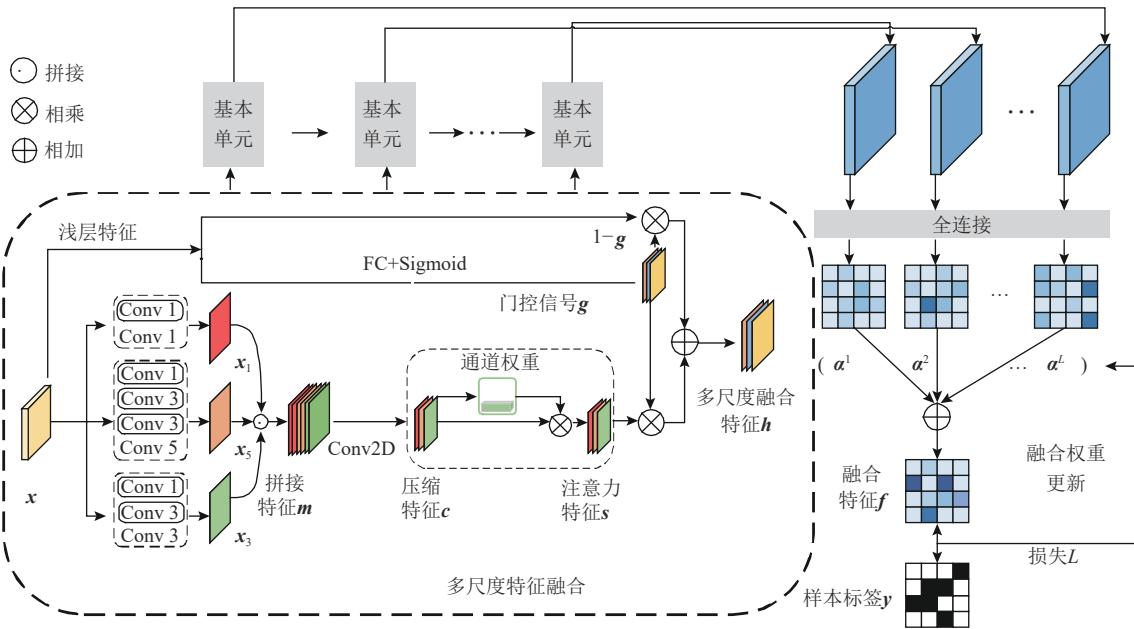


Fig. 2 Data feature fusion structure diagram based on adaptive integration

图2 基于自适应集成的数据特征融合结构图

与压缩特征相乘得到通道注意力特征:

$$s = c \times \sigma(MLP(\text{AvgPool}(c)) + MLP(\text{MaxPool}(c))), \quad (2)$$

其中  $\sigma$  表示 Sigmoid 激活函数,  $MLP$  代表多层感知机网络.

然后使用门控机制控制不同信息流的输入比例, 利用给定的浅层输入  $x$  作为门控信号, 经过一个全连接层(FC)和 Sigmoid 激活函数得到权值向量  $g$ , 将浅层输入和通道注意力特征分别乘以对应的权值并相加得到多尺度融合特征  $h$ :

$$h = s \times g + x \times (1 - g), \quad (3)$$

$$g = \sigma(Fc(x)) = \sigma(w_g x + b_g). \quad (4)$$

当概念漂移发生后, 为了能够及时适应新的数据分布, 本文引入在线学习, 将多尺度融合特征的提取结构作为基本单元, 并将若干个基本单元按照层次进行在线自适应集成. 由  $L$  个基本单元进行在线自适应集成, 其向前传播过程为

$$h^l = s \times g + h^{l-1} \times (1 - g), \quad l = 2, 3, \dots, L, \quad (5)$$

其中  $h^l$  表示第  $l$  个基本单元的输出,  $h^1 = x$ .

在线集成时,  $h_t^l$  表示时刻  $t$  第  $l$  个基本单元的输出, 每个基本单元均有一个融合权值  $\alpha_t^l$  与之对应, 在初始阶段, 融合权值均匀分布, 即  $\alpha_{t=1}^l = \frac{1}{L}$ , 则时刻  $t$  的输出为融合特征  $f_t$ :

$$f_t = \sum_{l=1}^L \alpha_t^l \times h_t^l. \quad (6)$$

根据每个基本单元的输出和对应的标签计算损失, 根据损失采用对冲方法<sup>[29]</sup>更新下一时刻各基本单元对应的融合权值:

$$\alpha_{t+1}^l = \alpha_t^l \times \beta^{L_t^l(h_t^l, y_t^l)}, \quad (7)$$

其中  $\beta$  是在线缩放因子,  $y_t^l$  表示时刻  $t$  第  $l$  个基本单元输出所对应的标签,  $L_t^l(h_t^l, y_t^l)$  表示时刻  $t$  第  $l$  个基本单元的损失.

## 2.2 时序特征增强

流数据中的数据元都是随着时间的顺序出现的, 数据之间具有时间序列相关性. 在以往的流数据处理方法中, 忽略了不同时刻数据之间的相互关系. 当概念漂移发生时, 提取流数据包含的时序特征, 有助于概念漂移问题的处理; 同时使用重要时刻的特征信息可以进一步提高概念漂移的处理能力, 因为并不是每个时刻的数据都对概念漂移处理起积极作用, 起消极作用的数据反而会加剧概念漂移.

在流数据中, 连续时刻的数据通常变化平稳, 具有高度的时间依赖性. 首先将得到的不同时刻的复杂数据特征按时间顺序组成序列特征:

$$u = (f_{t-n}, f_{t-n-1}, \dots, f_{t-2}, f_{t-1}, f_t). \quad (8)$$

在此基础上, 本文通过构建时间注意力机制模块, 对不同时刻的数据特征建模, 如图 3 所示.

首先使用挤压(squeeze)操作, 将每个时刻的数据特征进行全局平均池化计算, 通过挤压操作, 将跨空间维度  $H \times W$  的特征映射进行聚合, 全局信息压缩到 1 维特征  $z$  中, 使得这些一维特征可以被其他层利用:

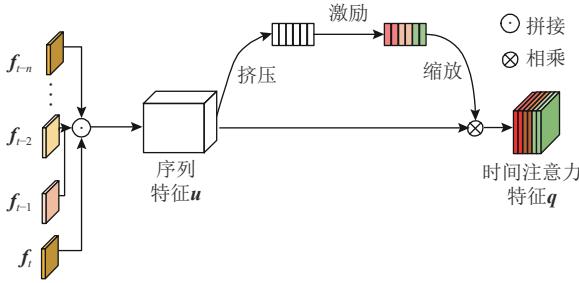


Fig. 3 Time attention mechanism diagram

图3 时间注意力机制图

$$z = F_{sq}(u) = \frac{1}{H \times W} \sum_{i=1}^H \sum_{j=1}^W u(i, j). \quad (9)$$

为了利用挤压操作的信息和利用通道间的信息依赖,接下来进行激励(excitation)操作。为了降低模型复杂度以及提升泛化能力,采用2个全连接层,其中:第1个FC层起到降维的作用,采用ReLU函数激活;第2个FC层恢复原始的维度,对每个通道的重要性进行预测,为每个特征通道生成权重。然后使用缩放(reweight)操作,将激励操作后输出的权重 $v$ 通过乘法逐通道加权到序列特征上,得到时间注意力特征 $q$ :

$$v = F_{ex}(z) = \sigma(FC(z, W)) = \sigma(W_2 \delta(W_1 z)), \quad (10)$$

$$q = F_{re}(u, v) = u \cdot v, \quad (11)$$

其中 $\delta$ 表示ReLU函数。

在流数据中,不同时刻的数据包含不同的信息。当发生概念漂移时,部分时刻的数据可能包含干扰信息或者噪声,会加剧概念漂移。因此选择对概念漂移处理有积极作用的关键数据是非常有必要的。本文将每个时刻的数据特征与初始数据特征进行相似度计算,相似度越高,证明两者的重叠信息越多,对概念漂移处理越有积极作用,则赋予其较大的权重,最终将序列特征与相似度权重相乘融合得到相似度特征:

$$p = \sum_{i=t-n}^t f_i (f_i * X_{t-1}), \quad (12)$$

其中 $*$ 表示相似度计算, $X_{t-1}$ 表示初始时刻的数据特征, $n$ 为选取的流数据时间长度。

使用注意力机制得到时间注意力特征 $q$ ,并将每个时刻的数据与初始数据进行相似度计算得到相似度特征 $p$ ,将两者融合得到每个时刻的时序特征 $r$ ,时序特征增强结构如图4所示。

### 2.3 MFECD 方法

MFECD方法首先提取不同卷积尺度的样本特征,然后对不同层次的数据特征进行在线自适应集成得到复杂数据特征。在此基础上,通过注意力机制

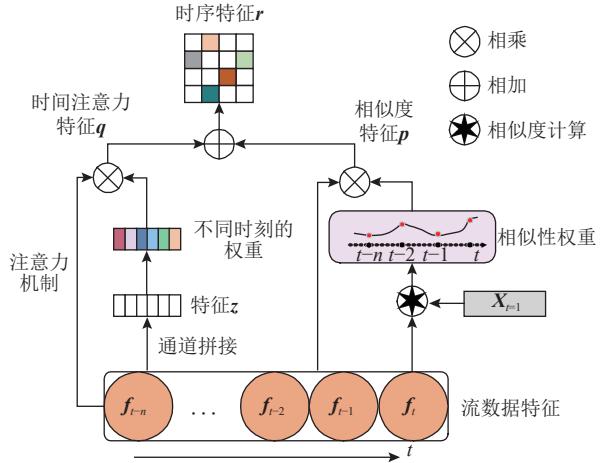


Fig. 4 Temporal feature enhancement structure diagram

图4 时序特征增强结构图

和相似度计算提取流数据的时序特征。利用混合特征提高对流数据信息的提取能力和对概念漂移问题的处理能力。基于混合特征提取的流数据概念漂移处理方法的过程具体如算法1所示。

**算法1.** 基于混合特征提取的概念漂移处理方法.

输入: 数据流 $SD$ , 缩放因子 $\beta$ , 选取时间长度 $n$ , 一个 $L$ 层的神经网络;

输出: 每个时刻的复杂数据特征 $f$ 和时序特征 $r$ .

- ① for  $t=1, 2, \dots, T$  do
- ② 从数据流 $SD$ 中获取当前数据 $x$ , 对其采用通道注意力和门控机制得到多尺度融合特征:

$$h = s \times g + x \times (1 - g);$$

- ③ 将不同层数据特征进行自适应集成得到复杂数据特征 $f$ :

$$f_t = \sum_{l=1}^L \alpha_t^l \times h_t^l;$$

- ④ 根据当前损失对下一时刻的集成权重 $\alpha$ 进行更新:

$$\alpha_{t+1}^l = \alpha_t^l \times \beta^{L_t^l(h_t^l, y_t^l)};$$

- ⑤ 根据时间长度 $n$ 对当前时刻的数据特征构建序列特征:

$$u = (f_{t-n}, f_{t-n-1}, \dots, f_{t-2}, f_{t-1}, f_t);$$

- ⑥ 分别经过映射到1维空间(squeeze)、预测通道权值(excitation)和通道加权(reweight)操作,得到时间注意力特征 $q$ :

squeeze:

$$z = F_{sq}(u) = \frac{1}{H \times W} \sum_{i=1}^H \sum_{j=1}^W u(i, j);$$

excitation:

$$\mathbf{v} = \sigma(FC(\mathbf{z}, \mathbf{W})) = \sigma(\mathbf{W}_2 \delta(\mathbf{W}_1 \mathbf{z}));$$

reweight:  $\mathbf{q} = F_{\text{re}}(\mathbf{u}, \mathbf{v}) = \mathbf{u}_c \mathbf{v}_c;$

- ⑦ 计算不同时刻数据与初始数据的相似度，得到相似度特征  $\mathbf{p}$ :

$$\mathbf{p} = \sum_{i=t-n}^t f_i(f_i * X_{t=1});$$

- ⑧ 时间注意力特征与相似度特征融合得到每个时刻的时序特征  $\mathbf{r}$ :

$$\mathbf{r} = \mathbf{q} + \mathbf{p};$$

- ⑨ 将时序特征与样本特征融合, 提高模型的概念漂移处理能力;

⑩ end for

### 3 实验与性能分析

视频数据作为一种典型的流数据, 目前受到了广泛关注, 视频目标跟踪的任务是对目标进行精准跟踪, 但由于多种因素影响, 跟踪过程中会出现多种概念漂移问题, 例如遮挡、快速运动、背景杂乱、平面内(外)旋转等, 都对目标的跟踪造成了干扰. 因此为了验证所提方法的性能, 本文在多个含概念漂移场景的视频目标跟踪数据集上进行了验证.

实验硬件环境为 NVIDIA-A100, 软件使用 Python 和 Pytorch. 训练集包含 ImageNet DET<sup>[30]</sup>, COCO<sup>[31]</sup>, ImageNet VID, YouTube-BB<sup>[32]</sup>, LaSOT<sup>[33]</sup>. 为了保持方法稳定性和动态性之间的平衡, 设置视频序列中时间长度  $n=4$ , 缩放因子  $\beta=0.99$ .

#### 3.1 数据集及评估标准

1) OTB100<sup>[34]</sup> 数据集. OTB100 数据集由 22 个对象类别的 100 个视频组成, 包含有 11 种漂移属性, 分别是背景杂乱(background clutters)、形变(deformation)、平面内旋转(in-plane rotation)、平面外旋转(out-of-plane rotation)、光照变化(illumination variation)、快速运动(fast motion)、运动模糊(motion blur)、遮挡(occlusion)、低分辨率(low resolution)、视野外(out-of-view)和尺度变化(scale variation). 每个视频序列包含不同的属性, OTB100 数据集的平均分辨率为 356×530, 视频长度在 71~3 872 帧之间.

OTB100 数据集评估基于 2 个指标: 精确率和成功率. 跟踪精确率计算跟踪方法预测的目标位置中心点与真实标注的中心点之间的距离小于给定阈值的视频帧所占的百分比. 跟踪成功率反映跟踪方法

预测的目标位置与真实标注位置之间的重合程度, 当某一帧的重合程度大于设定的阈值时, 则该帧被视为成功的, 成功帧的总数占所有帧的百分比即为跟踪成功率.

2) UAV123<sup>[35]</sup> 数据集. UAV123 是一个新的航拍视频数据集, 包含 123 个由 9 个不同对象类别组成的序列, 最小帧数为 109, 最大帧数为 3 085, 所有序列都用垂直边框完全注释. 漂移属性包含: 纵横比变化、背景杂波、相机运动、快速运动、完全遮挡、光照变化、低分辨率、视野外、部分遮挡、相似目标、尺度变化和视角变化. 与 OTB100 数据集一样, UAV123 数据集评价基于 2 个指标: 精确率和成功率.

3) LaSOT 数据集. 这是一个用于长期跟踪的大规模、高质量、注释密集的数据集. 为了减少潜在的类偏差, 它为每个类别提供相同数量的序列. LaSOT 具有较长的序列, 平均每个序列有 2 500 帧, 甚至更多, 比短期跟踪数据集更具挑战性. LaSOT 使用精确率、成功率和归一化精度评估跟踪性能. 精确率和成功率评估标准与 OTB100 数据集一致. 由于精度指标对目标尺寸和图像分辨率敏感, 因此 LaSOT 使用归一化的精度方法, 计算相对于目标大小的误差, 而不是考虑绝对距离, 归一化精度用于对跟踪器进行排序.

#### 3.2 在 OTB100 数据集上的实验结果与分析

将本文提出的 MFECD 方法与 SiamCAR<sup>[36]</sup>, SiamRPN++<sup>[37]</sup>, Ocean-online<sup>[38]</sup>, SiamR-CNN<sup>[39]</sup>, ATOM<sup>[40]</sup>, SiamFC<sup>[41]</sup>, SRDCF<sup>[42]</sup>, MDNet<sup>[43]</sup>, KCF<sup>[44]</sup> 这 9 种方法进行比较. 测试实验中, 设置参数  $L=9$ . 如图 5 所示, 成功率曲线图例中的数字表示该曲线下的面积, 其作为该跟踪方法的成功率值, 精确率曲线图例中的数字表示跟踪方法位置错误阈值取 20 时对应的精确率值, 其作为该跟踪方法的精确率值对于这个被广泛研究的数据集 OTB100, 本文方法成功率达到 0.699, 精确率达到 0.913.

图 6 和图 7 为本文方法与其他方法在 OTB100 数据集上部分漂移属性的成功率和精确率对比. 本文方法在快速运动、光照变化、平面内旋转、运动模糊和遮挡等漂移属性下结果都取得了最优, 这是由于本文方法充分利用提取到的复杂数据特征和时序特征, 在概念漂移发生时, 有效应对目标的多种变化, 从而提高目标跟踪的成功率和精确率.

为进一步说明本文方法的有效性, 本文选取了不同的典型视频序列进行实际效果展示, 如图 8 所示. 在 Basketball, Girl 2, Jump 视频数据中, 分别存在相似物干扰、遮挡、旋转和形变等典型的漂移属性.

从结果可以看出, ATOM, SiamFC, Ocean-online 方法均在漂移发生后出现了目标跟踪失败, 而 MFECD 方法由于融合了流数据的复杂样本特征和时间特征, 可以有效应对不同的分布变化, 从而解决数据流中

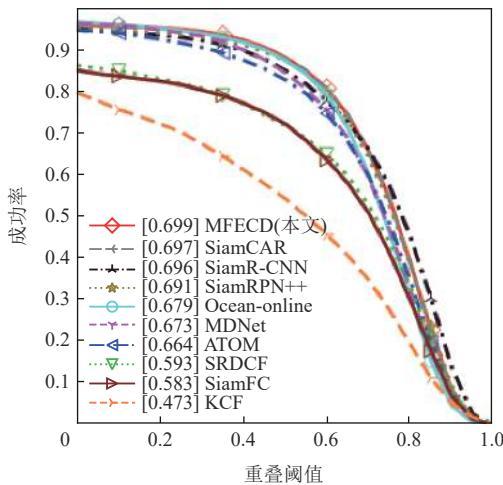
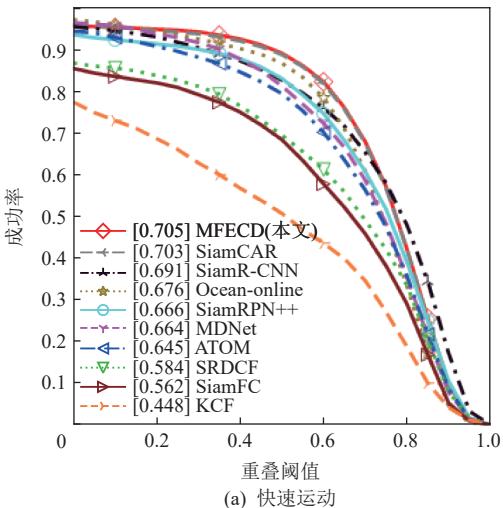
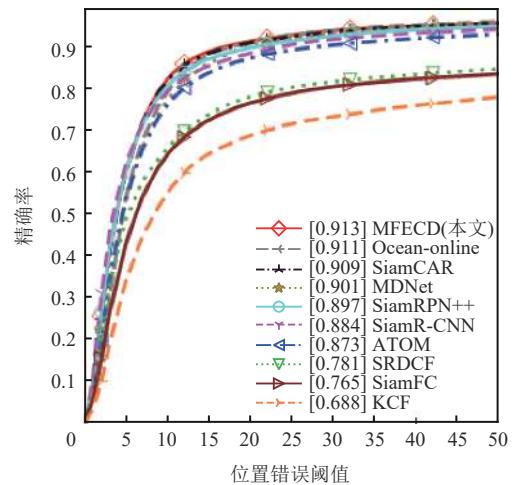
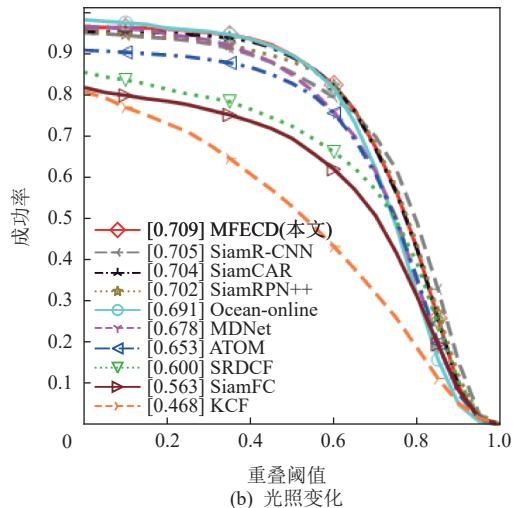


Fig. 5 Overall experimental results on OTB100 dataset

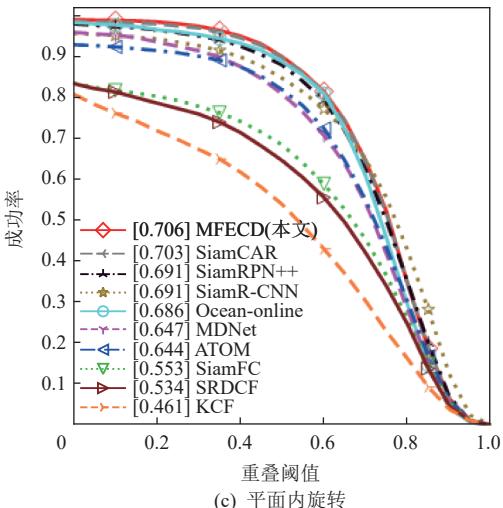
图 5 OTB100 数据集上的整体实验结果



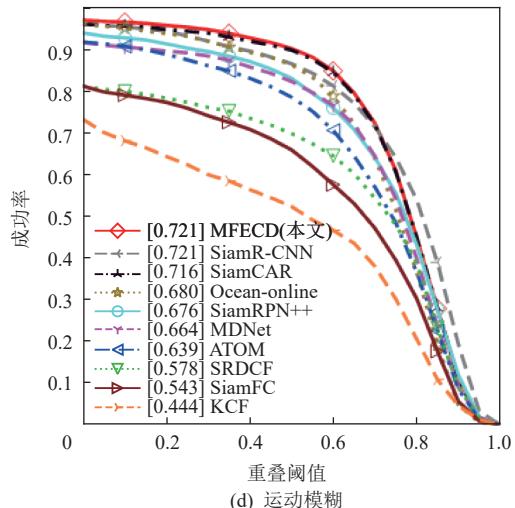
(a) 快速运动



(b) 光照变化



(c) 平面内旋转



(d) 运动模糊

存在的概念漂移问题。

### 3.3 在 UAV123 数据集上的实验结果与分析

在 UAV123 上, 将 MFECD 方法与 SiamCAR, SiamRPN++, SiamFC, ECO<sup>[45]</sup>, SAMF<sup>[46]</sup>, HiFT<sup>[47]</sup>, MEEM<sup>[48]</sup>,

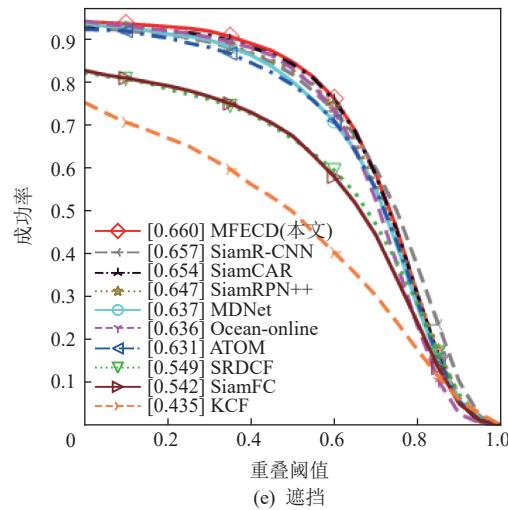


Fig. 6 Comparison results of success ratio of partial drift features on OTB100 dataset

图 6 OTB100 数据集上部分漂移属性的成功率对比结果

SiamDW<sup>[49]</sup>等多种跟踪方法进行比较,实验中参数  $L=5$ 。从图 9 可以看出,MFECD 方法成功率达到 0.62,精确率达到 0.81,这 2 个评价指标上都优于其他跟踪方法。

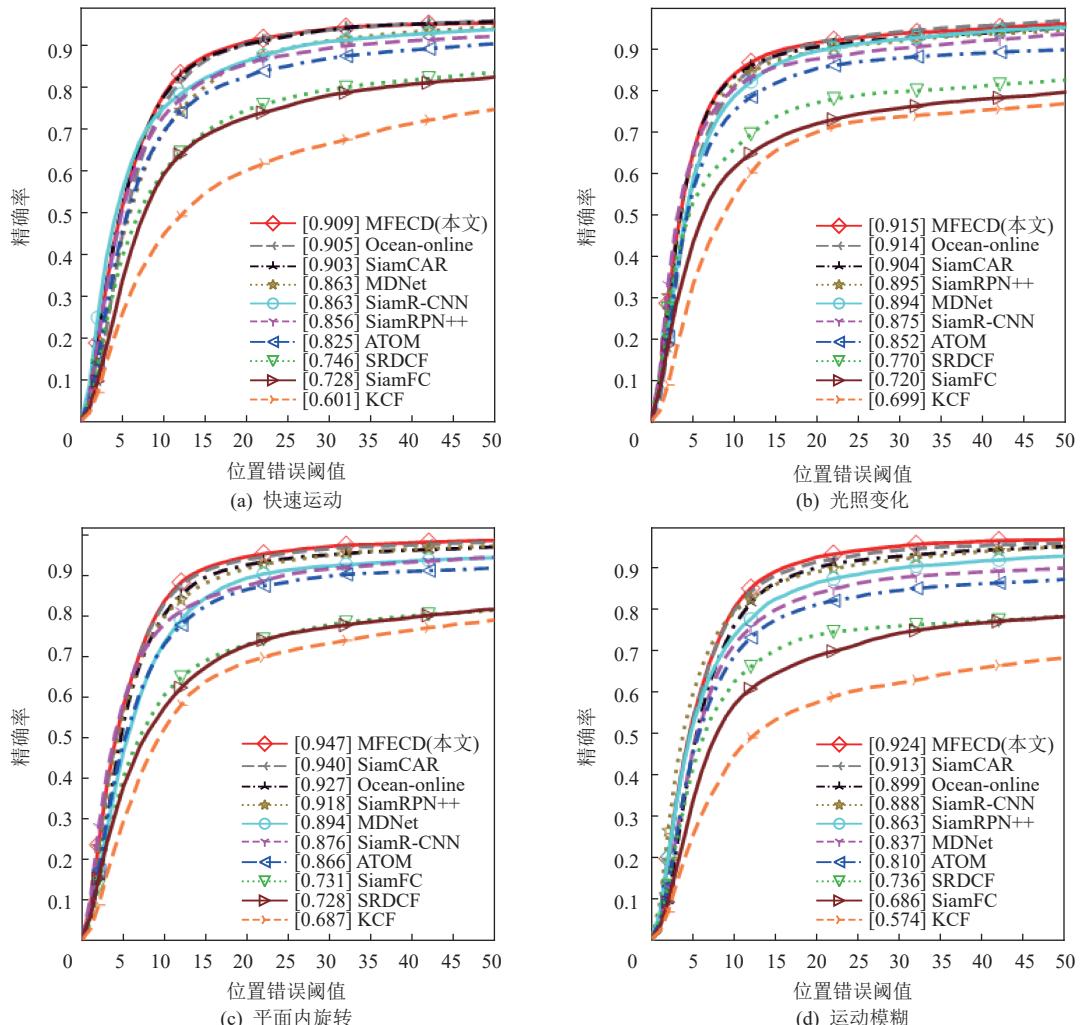


图 10 和图 11 为 MFECD 方法与其他方法在多种漂移属性上的成功率和精确率对比结果。可以看出,在纵横比变化、快速运动、完全遮挡、光照变化和尺度变化 5 种漂移场景下,MFECD 方法的结果都取得

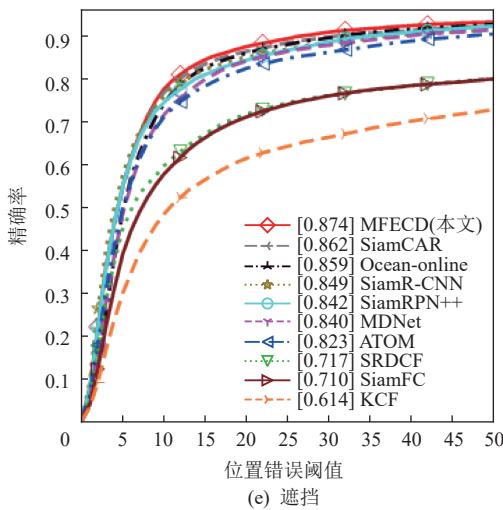


Fig. 7 Comparison results of precision of partial drift features on OTB100 dataset

图 7 OTB100 数据集在部分漂移属性的精确率对比结果

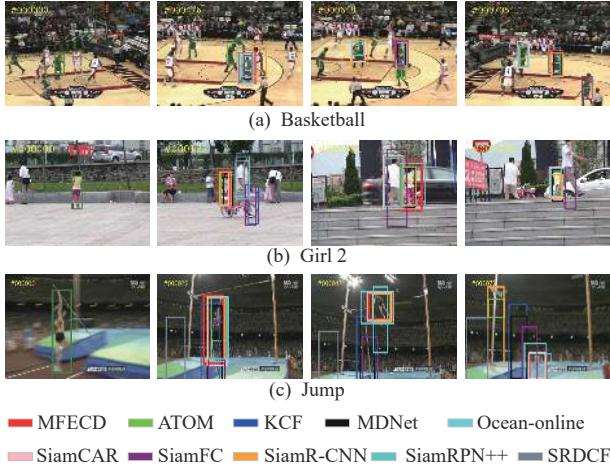


Fig. 8 Comparison of actual tracking effect

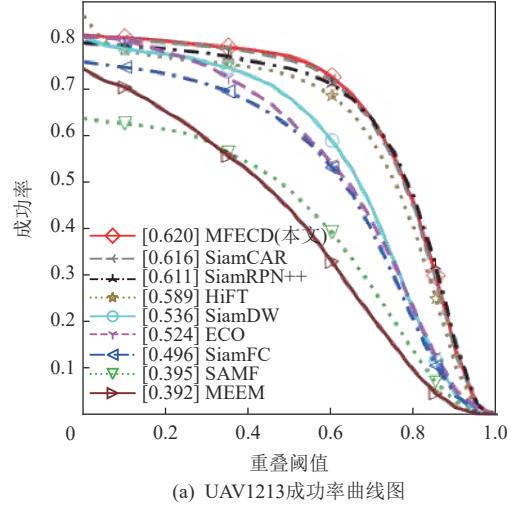
图 8 实际跟踪效果比较

了最优，充分表明 MFECD 方法在多种漂移场景下具有较强的鲁棒性，可以有效解决目标跟踪任务中出现的概念漂移问题，提高目标跟踪的成功率和精确率。

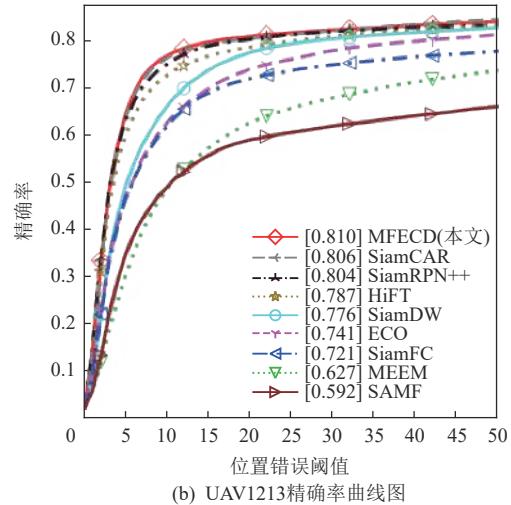
### 3.4 LaSOT 实验结果与分析

在 LaSOT 数据集上，将 MFECD 方法与 SiamCAR, SiamBAN<sup>[50]</sup>, ATOM, CLNet<sup>[51]</sup>, SiamRPN++, SPLT<sup>[52]</sup>, SiamFC, DSiam<sup>[53]</sup>, ASRCF<sup>[54]</sup>, SINT<sup>[55]</sup>, STRCF<sup>[56]</sup>, ECO, CFNet<sup>[57]</sup>, Staple<sup>[58]</sup> 等多种跟踪方法进行比较，设置参数  $L=3$ 。<sup>表 1</sup> 为在 LaSOT 数据集上测试得到的实验结果。

可以看出，MFECD 方法的归一化精度达到 0.620，精确率达到 0.532，成功率达到 0.525，结果均为最优。与经典在线跟踪方法 ATOM 相比，归一化精度提高了 0.04，这是由于 MFECD 方法可根据数据的实时变化对模型及时更新，利用流数据的混合特征，有效解决目标跟踪中出现的概念漂移。



(a) UAV123成功率曲线图



(b) UAV123精确率曲线图

Fig. 9 Overall experimental results on UAV123 dataset

图 9 UAV123 数据集上的整体实验结果

### 3.5 不同参数的实验结果

为了更好地验证不同参数对 MFECD 方法的影

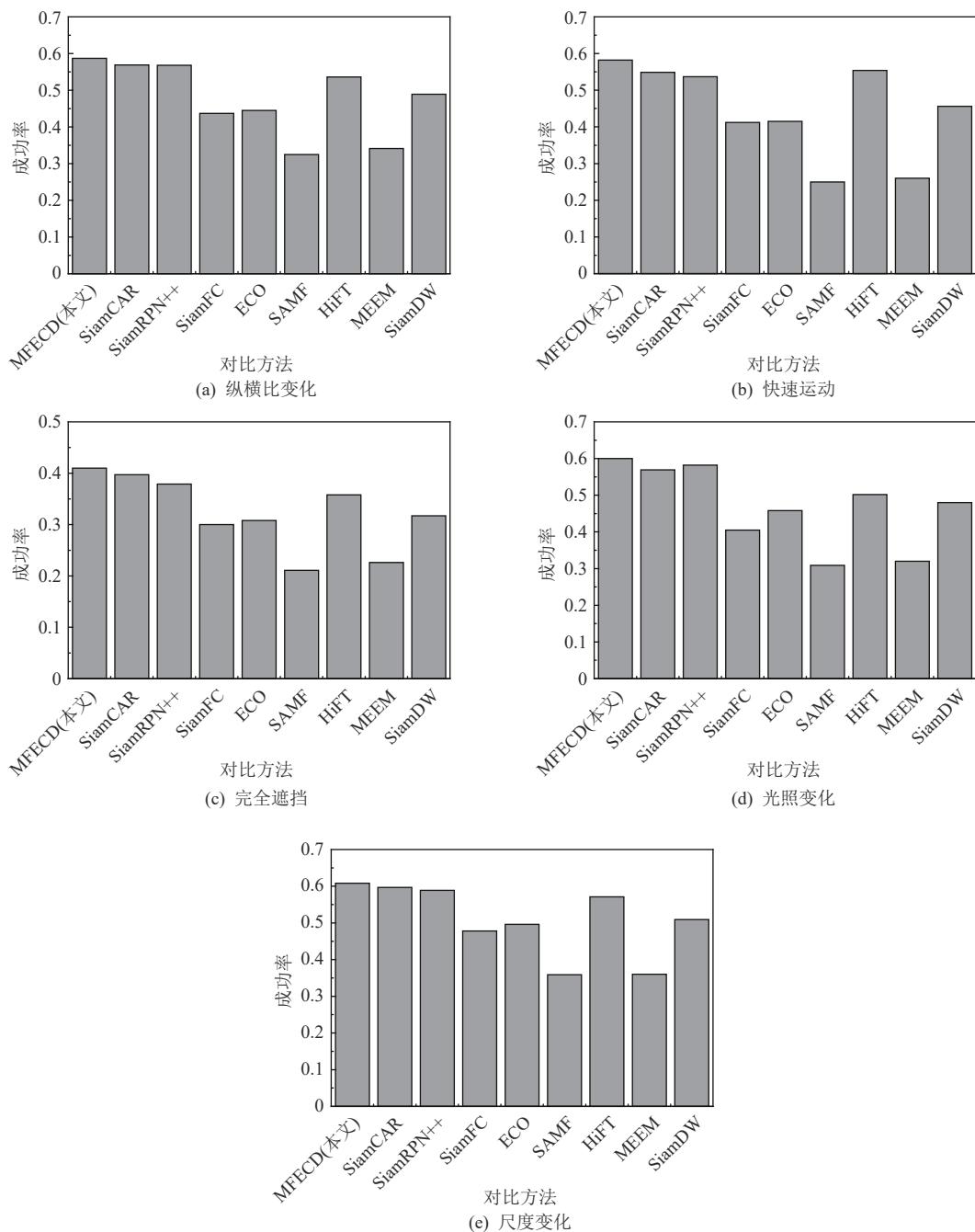


Fig. 10 Comparison results of success ratio of partial drift features on UAV123 dataset

图 10 UAV123 数据集上部分漂移属性的成功率对比结果

Table 1 Experimental Results on LaSOT Dataset

表 1 LaSOT 数据集上的实验结果

方法	归一化精度	精确率	成功率	方法	归一化精度	精确率	成功率
MFECD	0.620	0.532	0.525	DSiam	0.405	0.322	0.333
SiamCAR	0.605	0.520	0.512	ASRCF	0.391	0.337	0.359
SiamBAN	0.598	0.521	0.514	SINT	0.354	0.295	0.314
ATOM	0.576	0.505	0.515	STRCF	0.340	0.298	0.308
CLNet	0.574	0.494	0.499	ECO	0.338	0.301	0.324
SiamRPN++	0.569	0.491	0.496	CFNet	0.312	0.259	0.275
SPLT	0.494	0.396	0.426	Staple	0.278	0.239	0.243
SiamFC	0.420	0.339	0.336				

响, 本文在 3 个测试集上采用不同的参数进行实验, 参数包含基本单元的数量  $L$  和时间长度  $n$ .

从表 2 可以看出, OTB100, UAV123, LaSOT 这 3 个测试集分别在  $L=9$ ,  $L=5$ ,  $L=3$  取得了最优结果. 实验结果说明对于不同长度的视频序列, 视频帧数越多, 采用越少的基本单元数量, 方法更容易收敛以及达到最优的结果.

从表 3 可以看出, 在 3 个测试集上, 当  $n=4$  时, 跟踪结果都达到了最高. 实验结果说明结合适当数量

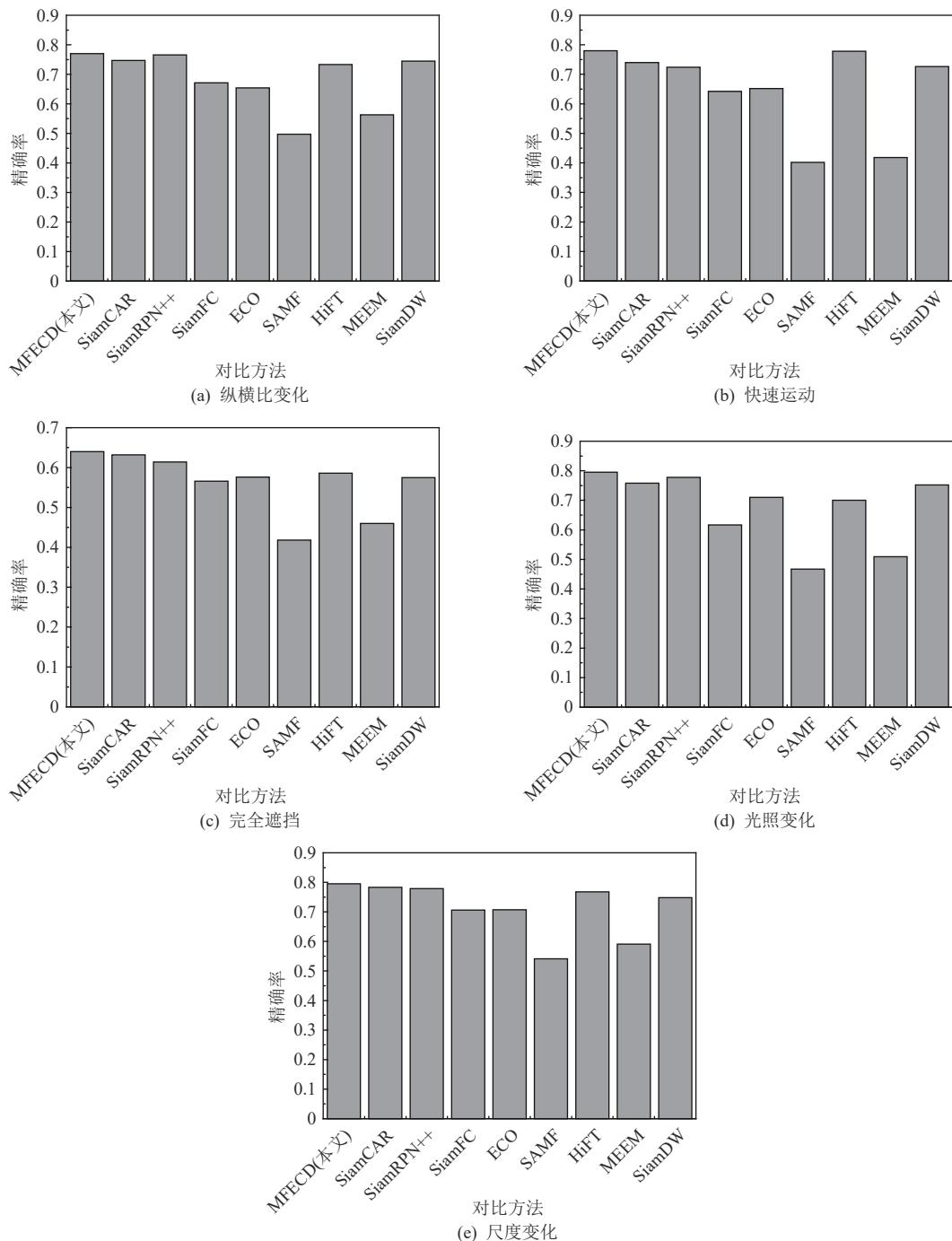


Fig. 11 Comparison results of precision of partial drift features on UAV123 dataset

图 11 UAV123 数据集上部分漂移属性的精确率对比结果

**Table 2 Experimental Results Under Different Numbers of Basic Units**

表 2 不同基本单元数量下的实验结果

<i>L</i>	OTB100		UAV123		LaSOT	
	成功率	精确率	成功率	精确率	成功率	精确率
3	0.695	0.905	0.613	0.804	<b>0.525</b>	<b>0.532</b>
5	0.698	0.912	<b>0.62</b>	<b>0.81</b>	0.516	0.525
7	0.694	0.905	0.616	0.805	0.515	0.522
9	<b>0.699</b>	<b>0.913</b>	0.619	0.81	0.519	0.528

注：加粗数字表示最优结果。

**Table 3 Experimental Results Under Different Time Lengths**

表 3 不同时间长度下的实验结果

<i>n</i>	OTB100		UAV123		LaSOT	
	成功率	精确率	成功率	精确率	成功率	精确率
2	0.697	0.909	0.602	0.786	0.517	0.526
4	<b>0.699</b>	<b>0.913</b>	<b>0.62</b>	<b>0.81</b>	<b>0.525</b>	<b>0.532</b>
6	0.695	0.907	0.612	0.801	0.522	0.532
8	0.695	0.907	0.612	0.8	0.514	0.522

注：加粗数字表示最优结果。

的历史帧,既可以排除视频序列中所包含的干扰信息,又可以充分考虑时间上下文信息,利用有效的时间信息提高目标的跟踪成功率和精确率。

## 4 结束语

针对概念漂移处理过程中存在的信息提取能力有限且未充分考虑流数据包含的时序特性等问题,本文提出一种基于混合特征提取的流数据概念漂移处理方法 MFECD。该方法首先采用不同尺度的卷积核构建拼接特征,采用门控机制将浅层输入和拼接特征进行融合,并且自适应集成不同层次的数据信息以获得复杂数据特征。在此基础上,采用时间注意力和相似度计算获得流数据的时序特征,并对关键位点进行有效增强。该方法利用复杂数据特征与时序特征,提高了神经网络的信息提取能力和对流数据概念漂移的处理能力。

**作者贡献声明:** 郭虎升提出设计思想、方法设计,撰写和修改论文;刘艳杰负责代码实现、数据测试、论文撰写及修改;王文剑负责写作指导、修改审定。

## 参 考 文 献

- [1] Lughofer E, Pratama M. Online active learning in data stream regression using uncertainty sampling based on evolving generalized fuzzy models[J]. *IEEE Transactions on Fuzzy Systems*, 2018, 26(1): 292–309
- [2] Zhai Tingting, Gao Yang, Zhu Junwu. Survey of online learning algorithms for streaming data classification[J]. *Journal of Software*, 2020, 31(4): 912–931 (in Chinese)  
(翟婷婷,高阳,朱俊武.面向流数据分类的在线学习综述[J].软件学报,2020,31(4):912–931)
- [3] Du Hangyuan, Wang Wenjian, Bai Liang. A novel evolving data stream clustering method based on optimization model[J]. *SCIENTIA SINICA Informationis*, 2017, 47(11): 1464–1482 (in Chinese)  
(杜航原,王文剑,白亮.一种基于优化模型的演化数据流聚类方法[J].中国科学:信息科学,2017,47(11):1464–1482)
- [4] Lu Jie, Liu Anjin, Dong Fan, et al. Learning under concept drift: A review[J]. *IEEE Transactions on Knowledge and Data Engineering*, 2019, 31(12): 2346–2363
- [5] Tennant M, Stahl F T, Rana O F, et al. Scalable real-time classification of data streams with concept drift[J]. *Future Generation Computer Systems*, 2017, 75: 187–199
- [6] Sergio R G, Krawczyk B, Garca S, et al. A survey on data preprocessing for data stream mining: Current status and future directions[J]. *Neurocomputing*, 2017, 239: 39–57
- [7] Bifet A, Gavalda R. Learning from time-changing data with adaptive windowing[C]//Proc of the 7th SIAM Int Conf on Data Mining. Philadelphia, PA: SIAM, 2007: 443–448
- [8] Du Lei, Song Qinbao, Jia Xiaolin. Detecting concept drift: An information entropy based method using an adaptive sliding window[J]. *Intelligent Data Analysis*, 2014, 18(3): 337–364
- [9] Guo Husheng, Ren Qiaoyan, Wang Wenjian. Concept drift class detection based on time window[J]. *Journal of Computer Research and Development*, 2022, 59(1): 127–143 (in Chinese)  
(郭虎升,任巧燕,王文剑.基于时序窗口的概念漂移类别检测[J].计算机研究与发展,2022,59(1):127–143)
- [10] Guo Husheng, Li Hai, Ren Qiaoyan, et al. Concept drift type identification based on multi-sliding windows[J]. *Information Sciences*, 2022, 585: 1–23
- [11] Baena-García M, Campo-Ávila R J, Fidalgo Del, et al. Early drift detection method[C]//Proc of the 17th ECML PKDD Int Workshop on Knowledge Discovery from Data Streams. Berlin: Springer, 2006: 77–86
- [12] Alippi C, Boracchi G, Roveri M. Just-in-time classifiers for recurrent concepts[J]. *IEEE Transactions on Neural Networks and Learning Systems*, 2013, 24(4): 620–634
- [13] Wen Yimin, Tang Shiqi, Feng Chao, et al. Online transfer learning for mining recurring concept in data stream classification[J]. *Journal of Computer Research and Development*, 2016, 53(8): 1781–1791 (in Chinese)  
(文益民,唐诗淇,冯超,等.基于在线迁移学习的重现概念漂移数据流分类[J].计算机研究与发展,2016,53(8):1781–1791)
- [14] Guo Husheng, Zhang Aijuan, Wang Wenjian. Concept drift detection method based on online performance test[J]. *Journal of Software*, 2020, 31(4): 932–947 (in Chinese)  
(郭虎升,张爱娟,王文剑.基于在线性能测试的概念漂移检测方法[J].软件学报,2020,31(4):932–947)
- [15] Nishida K, Yamauchi K. Detecting concept drift using statistical testing[C]//Proc of the 10th Int Conf on Discovery Science. Berlin: Springer, 2007: 264–269
- [16] Pears R, Sakthithasan S, Koh Y S. Detecting concept change in dynamic data streams: A sequential approach based on reservoir sampling[J]. *Machine Learning*, 2014, 97: 259–293
- [17] Brzezinski D, Stefanowski J. Reacting to different types of concept drift: The accuracy updated ensemble algorithm[J]. *IEEE Transactions on Neural Networks and Learning Systems*, 2014, 25(1): 81–94
- [18] Junior J R. Graph embedded rules for explainable predictions in data streams[J]. *Neural Networks*, 2020, 129: 174–192
- [19] Zhao Peng, Zhou Zhihua. Learning from distribution-changing data streams via decision tree model reuse[J]. *SCIENTIA SINICA Informationis*, 2021, 51(1): 1–12
- [20] Cai Huan, Lu Kezhong, Wu Qirong, et al. Adaptive classification algorithm for concept drift data streams[J]. *Journal of Computer Research and Development*, 2022, 59(3): 633–646 (in Chinese)  
(蔡桓,陆克中,伍启荣,等.面向概念漂移数据流的自适应分类方法[J].计算机研究与发展,2022,59(3):633–646)
- [21] Liang Bin, Li Guanghui, Dai Chenglong. G-mean weight classification

- method for imbalanced data stream with concept drift[J]. *Journal of Computer Research and Development*, 2022, 59(12): 2844–2857 (in Chinese)
- (梁斌, 李光辉, 代成龙. 面向概念漂移且不平衡数据流的 G-mean 加权分类方法[J]. *计算机研究与发展*, 2022, 59(12): 2844–2857)
- [22] Ashfahani A, Pratama M. Autonomous deep learning: Continual learning approach for dynamic environments[C]//Proc of the 2019 SIAM Int Conf on Data Mining. Philadelphia, PA: SIAM, 2019: 666–674
- [23] Lobo J L, Laña I, Del Ser J, et al. Evolving spiking neural networks for online learning over drifting data streams[J]. *Neural Networks*, 2018, 108: 1–19
- [24] Guo Husheng, Zhang Shuai, Wang Wenjian. Selective ensemble-based online adaptive deep neural networks for streaming data with concept drift[J]. *Neural Networks*, 2021, 142: 437–456
- [25] Sahoo D, Pham Q, Lu Jing, et al. Online deep learning: Learning deep neural networks on the fly [J]. arXiv preprint, arXiv: 1711.03705, 2017
- [26] Kauschke S, Lehmann D H, Fürnkranz J. Patching deep neural networks for nonstationary environments[C]//Proc of the 2019 Int Joint Conf on Neural Networks. Piscataway, NJ: IEEE, 2019: 1–8
- [27] Chollet F. Xception: Deep learning with depthwise separable convolutions [C] //Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2017: 1251–1258
- [28] Woo S, Park J, Lee J Y, et al. Cbam: Convolutional block attention module [C] //Proc of the European Conf on Computer Vision. Berlin: Springer, 2018: 3–19
- [29] Freund Y, Schapire R E. A decision-theoretic generalization of on-line learning and an application to boosting[J]. *Journal of Computer and System Sciences*, 1997, 55(1): 119–139
- [30] Russakovsky O, Jia Deng, Hao Su, et al. ImageNet large scale visual recognition challenge[J]. *International Journal of Computer Vision*, 2015, 115: 211–252
- [31] Lin T Y, Maire M, Belongie S, et al. Microsoft COCO: Common objects in context[C]//Proc of the 13th European Conf on Computer Vision. Berlin: Springer, 2014: 740–755
- [32] Real E, Shlens J, Mazzocchi S, et al. YouTube-boundingboxes: A large high-precision human-annotated data set for object detection in video[C]//Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2017: 5296–5305
- [33] Fan Heng, Lin Liting, Yang Fan, et al. LaSOT: A high-quality benchmark for large-scale single object tracking[C]//Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2019: 5374–5383
- [34] Wu Yi, Lim Jongwoo, Yang M H. Online object tracking: A benchmark[C]//Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2013: 2411–2418
- [35] Mueller M, Smith N, Ghanem B. A benchmark and simulator for UAV tracking[C]//Proc of the 14th European Conf on Computer Vision. Berlin: Springer, 2016: 445–461
- [36] Guo Dongyan, Wang Jun, Cui Ying, et al. SiamCAR: Siamese fully convolutional classification and regression for visual tracking[C]// Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2020: 6269–6277
- [37] Li Bo, Wu Wei, Wang Qiang, et al. SiamRPN++: Evolution of Siamese visual tracking with very deep networks[C]//Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2019: 4282–4291
- [38] Zhang Zhipeng, Peng Houwen, Fu Jianlong, et al. Ocean: Object-aware anchor-free tracking[C]//Proc of the 16th European Conf on Computer Vision. Berlin: Springer, 2020: 771–787
- [39] Voigtlaender P, Luitjen J, Torr P H S, et al. SiamR-CNN: Visual tracking by re-detection[C]//Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2020: 6578–6588
- [40] Danelljan M, Bhat G, Khan F S, et al. ATOM: Accurate tracking by overlap maximization[C]//Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2019: 4660–4669
- [41] Bertinetto L, Valmadre J, Henriques J F, et al. Fully-convolutional siamese networks for object tracking[C]//Proc of the European Conf on Computer Vision. Berlin: Springer, 2016: 850–865
- [42] Danelljan M, Hager G, Shahbaz Khan F, et al. Learning spatially regularized correlation filters for visual tracking[C]//Proc of the IEEE Int Conf on Computer Vision. Piscataway, NJ: IEEE, 2015: 4310–4318
- [43] Nam H, Han B. Learning multi-domain convolutional neural networks for visual tracking[C]//Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2016: 4293–4302
- [44] Henriques J F, Caseiro R, Martins P, et al. High-speed tracking with kernelized correlation filters[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2014, 37(3): 583–596
- [45] Danelljan M, Bhat G, Shahbaz Khan F, et al. ECO: Efficient convolution operators for tracking[C]//Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2017: 6638–6646
- [46] Li Yang, Zhu Jianke. A scale adaptive kernel correlation filter tracker with feature integration [G] //LNCS 8926: Proc of the European Conf on Computer Vision. Berlin: Springer, 2014: 254–265
- [47] Cao Ziang, Fu Changhong, Ye Junjie, et al. HiFT: Hierarchical feature transformer for aerial tracking[C]//Proc of the IEEE Int Conf on Computer Vision. Piscataway, NJ: IEEE, 2021: 15457–15466
- [48] Zhang Jianming, Ma Shugao, Sclaroff S. MEEM: Robust tracking via multiple experts using entropy minimization[C]//Proc of the 13th European Conf on Computer Vision. Berlin: Springer, 2014: 188–203
- [49] Zhang Zhipeng, Peng Houwen. Deeper and wider Siamese networks for real-time visual tracking[C]//Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2019: 4591–4600
- [50] Chen Zedu, Zhong Bineng, Li Guorong, et al. Siamese box adaptive network for visual tracking[C]//Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2020: 6668–6677
- [51] Dong Xingping, Shen Jianbing, Shao Ling, et al. CLNet: A compact latent network for fast adjusting Siamese trackers[C]//Proc of the European Conf on Computer Vision. Berlin: Springer 2020: 378–395
- [52] Yan Bin, Zhao Haojie, Wang Dong, et al. Skimming-perusal tracking:

- A framework for real-time and robust long-term tracking[C]//Proc of the IEEE Int Conf on Computer Vision. Piscataway, NJ: IEEE, 2019: 2385–2393
- [53] Guo Qing, Feng Wei, Zhou Ce, et al. Learning dynamic siamese network for visual object tracking[C]//Proc of the IEEE Int Conf on Computer Vision. Piscataway, NJ: IEEE, 2017: 1763–1771
- [54] Dai Kenan, Wang Dong, Lu Huchuan, et al. Visual tracking via adaptive spatially-regularized correlation filters[C]//Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2019: 4670–4679
- [55] Tao R, Gavves E, Smeulders A W M. Siamese instance search for tracking[C]//Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2016: 1420–1429
- [56] Li Feng, Tian Cheng, Zuo Wangmeng, et al. Learning spatial-temporal regularized correlation filters for visual tracking[C]//Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2018: 4904–4913
- [57] Valmadre J, Bertinetto L, Henriques J, et al. End-to-end representation learning for correlation filter based tracking[C]//Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2017: 2805–2813
- [58] Bertinetto L, Valmadre J, Golodetz S, et al. Staple: Complementary learners for real-time tracking[C]//Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2016: 1401–1409



**Guo Husheng**, born in 1986. PhD, professor, PhD supervisor. Senior member of CCF. His main research interests include data mining, machine learning, and computational intelligence.

郭虎升, 1986年生. 博士, 教授, 博士生导师. CCF高级会员. 主要研究方向为数据挖掘、机器学习、计算智能.



**Liu Yanjie**, born in 1995. Master candidate. His main research interests include streaming data mining and online machine learning.

刘艳杰, 1995年生. 硕士研究生. 主要研究方向为流数据挖掘、在线机器学习.



**Wang Wenjian**, born in 1968. PhD, professor, PhD supervisor. Distinguished member of CCF. Her main research interests include machine learning, data mining, and computational intelligence.

王文剑, 1968年生. 博士, 教授, 博士生导师. CCF杰出会员. 主要研究方向为机器学习、数据挖掘、计算智能.