

文章编号 1004-924X(2023)22-3345-12

针对目标遮挡的自适应特征匹配网络

毛 琳, 苏宏阳*, 杨大伟
(大连民族大学 机电工程学院, 辽宁 大连 116600)

摘要:针对目标跟踪中常见的目标遮挡问题,提出一种自适应特征匹配网络。该网络通过计算查询帧与记忆帧像素级相似度,将目标和背景相似度关系进行编码,获得像素级相似度矩阵,并通过将查询帧与记忆帧分头的方式,实现多维度相似性计算,以关注查询帧中更多区域,并通过计算的相似度矩阵,对记忆帧进行自适应特征加权,以此来提高目标跟踪的精度和鲁棒性。此外,特征记忆网络可以对记忆帧进行挑选和保存,为特征匹配提供额外表现信息,使网络隐性学习目标运动趋势,进而实现更好的跟踪结果。实验结果表明,该方法在 GOT-10k, LaSOT 等数据集上表现良好,在 GOT-10k 数据集上,本文所提出的算法与 STMTrack 算法相比, AO 值提升 1.8%。根据可视化结果显示,本文算法在目标遮挡、消失等挑战中,具有更强的鲁棒性。

关键词:目标遮挡;自适应;特征匹配;记忆网络

中图分类号:TP391.41 **文献标识码:**A **doi:**10.37188/OPE.20233122.3345

Adaptive feature matching network for object occlusion

MAO Lin, SU Hongyang*, YANG Dawei

(School of Electromechanical Engineering, Dalian Minzu University, Dalian 116600, China)

* Corresponding author, E-mail: wxhxhwdn0725@163.com

Abstract: An adaptive feature-matching network is proposed to solve the common problem of object occlusion in object tracking. By calculating the pixel-level similarity between the query and memory frames, the network encodes the similarity relationship between an object and its background and obtains a pixel-level similarity matrix. By separating the query and memory frames, the network calculates the multi-dimensional similarity to focus on more areas in the query frame and adaptively weighs the memory frame through the calculated similarity matrix to improve the accuracy and robustness of object tracking. Additionally, the feature memory network selects and saves the memory frames, provides additional apparent information for feature matching, and allows the network to implicitly learn the moving trend of an object to achieve better tracking results. Experimental results show that this method performs well on GOT-10k, LaSOT, and other datasets. On GOT-10k datasets, compared with the STMTrack algorithm, the value of the proposed algorithm is improved by 1.8%. The visualization results show that the proposed algorithm is more robust in meeting the challenges of object occlusion and disappearance.

Key words: object occlusion; self-adaption; feature matching; memory network

收稿日期:2023-04-26;修订日期:2023-06-06.

基金项目:国家自然科学基金资助项目(No. 61673084);辽宁省自然科学基金资助项目(No. 20170540192, No. 20180550866, No. 2020-MZLH-24)

1 引言

目标跟踪是计算机视觉领域中一个重要研究方向,其任务是准确跟踪给定视频序列中的标注目标,如行人、车辆、动物等。目标跟踪应用非常广泛,如视频监控、自动驾驶、机器人视觉等,但由于目标在视频中普遍存在遮挡、消失等现象,会极大降低现有算法的准确性和鲁棒性,使算法在实际应用中可靠性降低。

目标遮挡是目标跟踪中常见挑战之一,传统目标跟踪算法往往无法应对目标遮挡带来的问题,导致跟踪效果下降。近年来,研究者们开始使用深度学习来解决遮挡问题。一类基于深度学习的目标跟踪算法利用强化学习思想,在目标遮挡的情况下,动态调整跟踪器行为,以适应目标的变化。例如,ATOM^[1](Accurate Tracking Overlap Maximization)通过设计两个模块,一个模块进行训练,不断增强检测框与真实结果重叠率,另一个模块则是分类模块,用于前景背景判别,以此获得更高的跟踪精度。PrDiMP^[2](Probabilistic Regression for Visual Tracking)则在ATOM基础上,通过融合置信度回归方法,进一步优化概率回归模型,取得更加准确的回归结果。另一类基于深度学习的目标跟踪算法则采用孪生神经网络结构,以SiamRPN^[3]为基础的许多目标跟踪算法将目标跟踪看作一个相似度匹配问题,通过学习模板和搜索区域的相似度来进行目标定位,配合复杂的特征后处理来应对目标遮挡、目标消失等挑战。如通过级联区域建议网络(Region Proposal Network, RPN)^[4, 5],来生成更多的区域建议框,以达到更准确的跟踪。此外,还有一些目标跟踪算法利用分割技术,在原有的分类和回归分支基础上,增加预测目标分割掩码的分支^[6-7],通过分割算法将目标分割出来,将跟踪问题转化为一个前背景分类问题,实现跟踪和分割的互补,提高了跟踪的准确性。以上算法均取得不错成绩,但由于引入复杂的特征后处理,导致更多的计算,并且更加依赖超参的设置。总的来说,基于深度学习的目标跟踪算法已经取得了一定进展,但面对目标完全遮挡时,算法很难找到目标位置,往往需要设计繁琐的手工特征和复杂的特征后处理。

在最近,Fei Xie等人提出EoC^[8](Extract or Correlation)模块,EoC模块中集合了自注意力和交叉注意力(Self-Attention, SA; Cross-Attention, CA),作者以EoC模块构建骨干网络,通过在特征提取过程中不断计算相关性,获得良好的跟踪性能。而Yutao Cui等人则在MixFormer^[9]中提出了一种混合注意模块(Mixed Attention Module, MAM),可以同时进行特征提取和特征融合,获得优异的算法性能。

同时,在目标分割领域,针对时间信息利用不充分的问题,Oh Seoung Wug等人提出了一种基于时空记忆网络的视频对象分割方法^[10],通过在时间和空间维度上建立记忆单元来捕捉视频中对象的运动和形状信息。Xie Haozhe等人提出了一种高效的区域记忆网络^[11],通过学习目标区域和背景区域之间的相互关系来提高分割精度和效率。Paul Matthieu等人提出了一种局部记忆注意力网络^[12],利用局部上下文信息来提高分割效果,并加速模型推理。Wang Hao等人提出了一种时序记忆注意力网络^[13],通过在时间维度上建立记忆单元来建模视频序列的长期依赖关系,从而提高视频对象分割的精度。

综上所述,为解决目标跟踪中常见的目标遮挡问题,本文算法结合相关滤波和MixFormer的思想,提出一种自适应特征匹配网络,该网络通过目标特征互相关计算,对目标特征进行自适应加权,提高特征匹配准确度。同时,本文借鉴分割领域中记忆网络的思想,构建一个特征记忆网络,用于缓解目标跟踪中时间信息利用不充分问题。通过利用特征记忆网络,算法能够更好地适应跟踪目标在时间序列中的运动和变化,即使在目标完全遮挡时也能自适应推断目标位置,从而提高跟踪的精度和稳定性。在数据集上的实验结果表明,本文算法在解决目标遮挡问题方面具有更好的表现,能够更准确地跟踪目标,并且在完全遮挡情况下也能保持较高的跟踪精度。

2 自适应特征匹配网络

针对目标遮挡的自适应特征匹配网络算法框架在文献[14]的基础上构建,如图1所示,可将其分为四个部分:骨干网络、自适应特征匹配网络(Adept Feature Match Network, AFMN)、特征

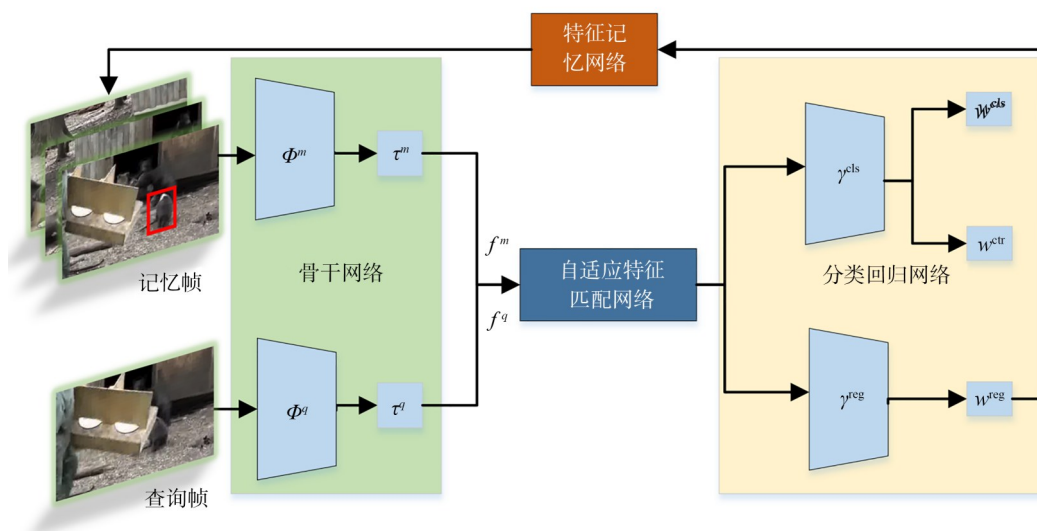


图1 整体网络框架

Fig. 1 Overall network framework

记忆网络 (Feature Memory Network, FMN)、分类回归网络。

2.1 骨干网络

在运动目标的处理过程中,常常会遭遇尺度变化和遮挡等问题,这时无法获取完整的目标特征,只能获得局部特征,这就可能导致跟踪错误。因此,本文旨在研究如何有效利用局部特征。为实现这一目标,本文采用 Inception V3 作为骨干网络,并进行迁移学习。Inception V3 同时使用多个尺度的卷积核来提取特征,这种设计使得模型能够适应不同尺度的图像,提高了模型的鲁棒性。此外,该模型还采用 1×1 卷积核来减少模型的数量和计算复杂度,同时增加了模型的非线性和表达能力。通过卷积层、池化层、批量归一化等操作,该模型将输入图像转换为高维特征向量,并逐渐提取越来越抽象的特征,以区分不同的目标,为后续处理和跟踪任务提供支持。

在训练过程中,使用相同的结构来构建查询帧和记忆帧的骨干网络,但不共享它们之间的权重,将其分别表示为 Φ^q, Φ^m 。在骨干网络提取特征完成后,分别使用自适应卷积层 τ^q, τ^m 对骨干网络的输出进行处理,以适应后续的特征处理操作,具体可用公式表示为:

$$\begin{cases} f^q = \tau^q(\Phi^q(f_{\text{input}}^q)) \\ f^m = \tau^m(\Phi^m(f_{\text{input}}^m)) \end{cases} \quad (1)$$

其中: f_{input}^q 和 f_{input}^m 代表输入, Φ^q 和 Φ^m 是提取特征

的基础网络, τ^q 和 τ^m 为单个 3×3 卷积,用于通道数的调整,以确保查询帧和记忆帧的特征在后续处理中能够对齐和匹配。这种方式可以优化特征处理过程,以提高模型的性能和效果。最终将 Φ^q 和 Φ^m 处理的特征 f^q 和 f^m 作为 AFMN 的输入进行自适应特征加权。

2.2 AFMN

在目标跟踪中,大多数算法通常依赖目标区域特征来进行跟踪。然而,当目标发生遮挡时,无法获取目标完整信息,导致算法在前背景分类和检测框回归时出现偏差,使得预测位置偏离实际轨迹,就会产生跟踪漂移的问题,如图2所示。

针对目标遮挡产生的跟踪漂移问题,分析其产生原因,以图2为例,由分数响应图可以看出,

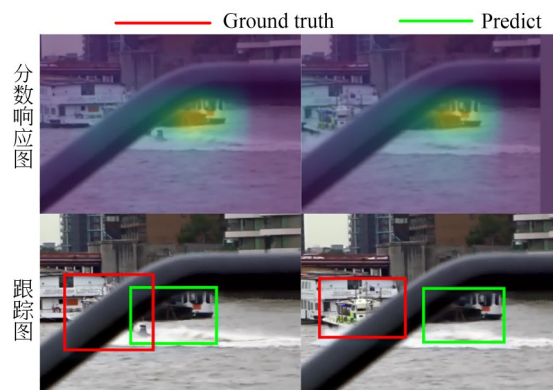


图2 跟踪漂移示意图

Fig. 2 Schematic diagram of tracking drift

在分类阶段,由于目标被遮挡,目标区域得分急剧降低,又因为存在与目标相似的背景,导致算法错误的给背景赋予较高分数,而将目标分类为背景。由跟踪图可以看出,在回归阶段,受前背景分类的影响,导致回归精度大大降低。进一步,在进行下一帧跟踪时,由于受到前一帧跟踪结果的影响,算法将上一帧跟踪结果进行特征匹配,使算法误把背景作为目标进行跟踪,造成了跟踪漂移的问题。

针对上述问题,本文在实验中发现,获得骨干网络提取的特征后,其多通道的特征图中存在许多相似或重复的特征,这可能会对特征匹配造成干扰,尤其在目标遮挡时更为明显。

为解决这个问题,如图3所示,对骨干网络的输出 f^q 和 f^m ,将其按通道拆分为 h 份,拆分过程可表示为:

$$\begin{cases} \text{Concat} \{ f_k^q \} (k=1, 2, \dots, h) = f^q \\ \text{Concat} \{ f_k^m \} (k=1, 2, \dots, h) = f^m \end{cases}, \quad (2)$$

其中: $f_k^q \in R^{(C/h, H, W)}$ 和 $f_k^m \in R^{(C/h, T, H, W)}$ 分别为3维和4维矩阵,其中 $H, W, C/h$ 分别表示矩阵 f_k^q 和 f_k^m 的行数、列数、以及分为多头后的通道数, T 表示矩阵 f^m 选取的记忆帧数。

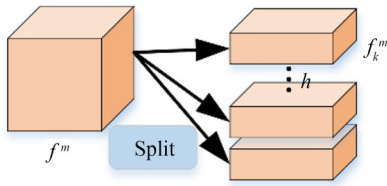


图3 特征图拆分示意图

Fig. 3 Schematic diagram of feature map splitting

将二者进行分头后,不仅可以解决相似和重复特征对跟踪造成的影响,还可以通过多次计算 f^q 和 f^m 之间的相似度,对二者的相似性关系进行多维度观察,提高特征的利用效率,并进行自适应的特征检索,为特征匹配提供更多参考,减小误判的可能。如图4所示,在进行特征图拆分后,通过多次计算相似性关系,生成多个目标响应,为算法提供更多选择,通过训练,最终实现正确的跟踪。

因此,为了更准确地确定目标位置,本文将

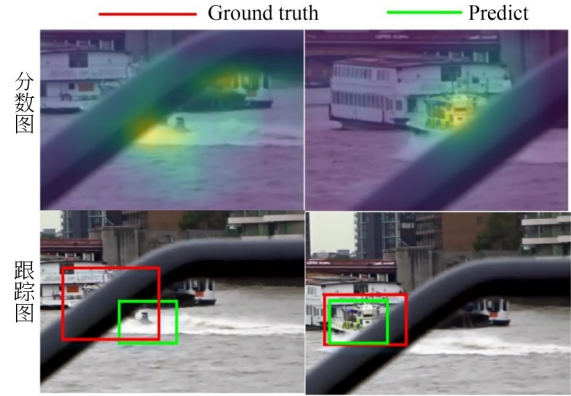


图4 多响应分数图

Fig. 4 Multi-response fractional graph

骨干网络提取的特征进行分头,提出一种基于相关滤波和注意力的方法,计算目标逐像素的相似度,通过被遮挡目标暴露部分进行局部到局部的特征匹配,从而显著提高目标定位精度。该方法将特征图转换为一维向量,并通过计算特征图向量像素间的点积相似度,判断不同特征之间的互相关程度。然后,将这些点积相似度作为权重系数组成权重矩阵,对记忆帧特征图进行加权,筛选目标特征,并与查询帧特征拼接,最终进行分类回归操作,实现自适应的特征匹配。这种方法可以更好地解决目标遮挡问题,提高跟踪算法的精度和鲁棒性。整体结构图如图5所示。

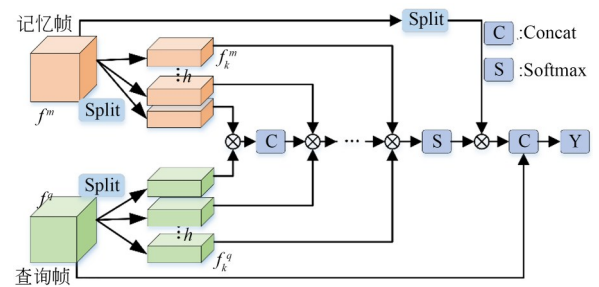


图5 AFMN结构图

Fig. 5 AFMN block diagram

在完成分头后,为计算 f_k^q 和 f_k^m 任意特征图之间像素级的相似性,将 f_k^q 和 f_k^m 每一通道的特征图降维展为向量,将其分别表示为 $f_k^q \in R^{(C/h, H \times W)}$, $f_k^m \in R^{(C/h, T \times H \times W)}$,随后,进一步将每个特征图的同一像素点组成向量,这些向量可以表示为列向量和行向量,如式(3)所示:

$$\begin{cases} \mathbf{f}_k^q = [q_1, q_2, \dots, q_i], (i = HW) \\ \mathbf{f}_k^{m'} = (\mathbf{f}_k^m)^T = \begin{bmatrix} m_1, \\ m_2, \\ \vdots \\ m_j \end{bmatrix}, (j = THW) \end{cases}, \quad (3)$$

式中,将 $\mathbf{f}_k^m$ 进行转置获得 $\mathbf{f}_k^{m'}$ 。 $\mathbf{f}_k^q \in R^{(C/h, H \times W)}$ 和 $\mathbf{f}_k^{m'} \in R^{(T \times H \times W, C/h)}$ 皆为2维矩阵, q_i 和 m_j 分别为 $\mathbf{f}_k^q$ 和 $\mathbf{f}_k^{m'}$ 按像素进行切分,长度为 C/h 的向量,切分过程如图6所示。

随后,通过矩阵乘法计算向量 q_i 和 m_j 之间的

$$\mathbf{w}_k = \mathbf{f}_k^{m'} \otimes \mathbf{f}_k^q = \begin{bmatrix} m_1, \\ m_2, \\ \dots, \\ m_j \end{bmatrix} \otimes [q_1, q_2, \dots, q_i] = \begin{bmatrix} m_1 \odot q_1, m_1 \odot q_2, \dots, m_1 \odot q_i, \\ m_2 \odot q_1, m_2 \odot q_2, \dots, m_2 \odot q_i, \\ \vdots \\ m_j \odot q_1, m_j \odot q_2, \dots, m_j \odot q_i \end{bmatrix}, \begin{cases} i = HW \\ j = THW \end{cases}, \quad (4)$$

$$\mathbf{w}_k | (\mathbf{w}_k)_{ji} = \frac{\exp(|\mathbf{m}_j| |\mathbf{q}_i| \cos \theta) / \sqrt{C/h}}{\sum_{i=1}^{HW} \sum_{j=1}^{THW} \exp(|\mathbf{m}_j| |\mathbf{q}_i| \cos \theta) / \sqrt{C/h}}, \begin{cases} i = 1, 2, \dots, HW \\ j = 1, 2, \dots, THW \end{cases}, \quad (5)$$

式中,通过矩阵乘法的方式,实现了计算不同特征图像素之间的点积相似度, $\mathbf{w}_k \in R^{(T \times H \times W, H \times W)}$ 为计算获得的2维矩阵,其中每个元素代表了 $\mathbf{f}_k^q$ 和 $\mathbf{f}_k^{m'}$ 特征图任意两个像素点之间的相似程度。随后,使用 soft max 函数对其进行归一化处理,为权重矩阵不同元素赋予相似度分数,将其表达为式(5)的形式,式中, $\exp$ 为指数操作, $\sqrt{C/h}$ 作为

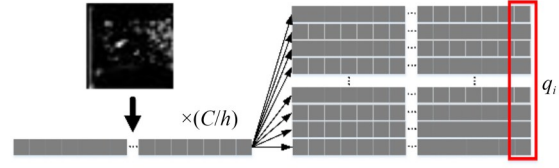


图6 特征图向量示意图

Fig. 6 Schematic diagram of feature map vector

点积相似度,由点积的几何意义可知,两个向量之间的点积结果越大,则代表两个向量越相似,以此获得权重矩阵 $\mathbf{w}_k$,可表达为如式(4)所示:

除数防止指数操作过程中数值溢出, θ 为 q_i 和 m_j 的相似度夹角。可以得出, $\mathbf{w}_k$ 经过 soft max 函数归一化处理后,其中每一元素 $(\mathbf{w}_k)_{ji}$ 都根据相似程度被赋予不同的相似度分数,将其对 $\mathbf{f}_k^m$ 进行特征加权,便可对 $\mathbf{f}_k^m$ 中不同目标特征实现自适应的注意操作,可表达为如式(6)所示:

$$\mathbf{M}_k = \mathbf{f}_k^m \otimes \mathbf{w}_k = ((\mathbf{w}_k)^T \otimes \mathbf{f}_k^{m'})^T = \left(\text{soft max} \left(\begin{bmatrix} m_1 \odot q_1, m_2 \odot q_1, \dots, m_j \odot q_1, \\ m_1 \odot q_2, m_2 \odot q_2, \dots, m_j \odot q_2, \\ \vdots \\ m_1 \odot q_i, m_2 \odot q_i, \dots, m_j \odot q_i \end{bmatrix} \otimes \begin{bmatrix} m_1, \\ m_2, \\ \vdots \\ m_j \end{bmatrix} \right) \right)^T, \quad (6)$$

$$\mathbf{M}_k | (\mathbf{M}_k)_{iv} = \left(\sum_{j=1}^{THW} \left| \frac{\exp(|\mathbf{m}_j| |\mathbf{q}_i| \cos \theta) / \sqrt{C/h}}{\sum_{i=1}^{HW} \sum_{j=1}^{THW} \exp(|\mathbf{m}_j| |\mathbf{q}_i| \cos \theta) / \sqrt{C/h}} \right| |\mathbf{m}_j|_v \cos \beta \right)^T, \begin{cases} (i = 1, 2, \dots, HW) \\ (j = 1, 2, \dots, THW) \\ (v = 1, 2, \dots, C/h) \end{cases}, \quad (7)$$

式中,为方便计算,将权重矩阵 $\mathbf{w}_k$ 进行转置,表达为 $(\mathbf{w}_k)^T$,并与 $\mathbf{f}_k^{m'}$ 进行矩阵乘法计算,其中, $(\mathbf{w}_k)^T \in R^{(H \times W, T \times H \times W)}$,包含了 $\mathbf{f}_k^q$ 和 $\mathbf{f}_k^{m'}$ 任意两个特征图像素点的点积相似度, $\mathbf{f}_k^{m'} \in R^{(T \times H \times W, C/h)}$ 则包含记忆帧 C/h 个通道的特征图像素点。使用权重矩阵 $\mathbf{w}_k$ 对 $\mathbf{f}_k^{m'}$ 的每一通道、每一像素点进行自适应加权,以关注 $\mathbf{f}_k^m$ 的不同区域,并获得概率矩阵 $\mathbf{M}_k$,并将其简化为式(7)。式中, $\cos \theta, \cos \beta$

为向量点积之间的夹角, i, j 分别为 q_i 和 m_j 的索引, v 为 m_j 像素点的索引。当 $i, v = 1$ 时, $(\mathbf{M}_k)_{iv}$ 代表 $\mathbf{f}_k^q$ 特征图第1个像素点与 $\mathbf{f}_k^m$ 第1个通道特征图的全部像素点相似度总分。当 $i = HW, v = C/h$ 时, $(\mathbf{M}_k)_{iv}$ 代表 $\mathbf{f}_k^q$ 特征图第 HW 个像素点与 $\mathbf{f}_k^m$ 第 C/h 个通道特征图的全部像素点相似度总分。

经过最终计算, $\mathbf{M}_k \in R^{(H \times W, C/h)}$ 为2维矩阵,

其中每一列代表特征筛选后的特征图,每一行代表每个通道的像素点值,将其转置并重新展开,获得 $M_k' = (M_k)^T \in R^{(C/h, H, W)}$, 整个自适应特征加权过程可简化为图 7 所示。

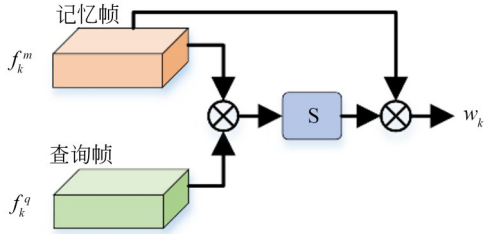


图 7 点积相似度计算

Fig. 7 Calculation of point-product similarity

对于 f_k^q 和 f_k^m 分出的每一个头,重复上面的操作,通过多次对记忆帧加权,以多角度对 f_k^m 进行关注,通过这种方式,减小错误关注对整个跟踪过程的影响,将多个 M_k' 按照通道维度重新拼接,可表示为:

$$M = \text{Concat}_{k=1}^h \{M_k'\} (k=1, 2 \dots h) \in R^{(C, H, W)}, \quad (8)$$

其中: h 为所分头数, k 为头数的索引, C, H, W 分别表示矩阵的行数、列数和通道数。

通过式(8)计算得到的概率矩阵涵盖记忆帧中所有可能目标特征,即使目标被部分遮挡也能自适应检索目标的局部特征,并通过局部特征匹配来完成跟踪过程。将该概率矩阵与查询帧特征矩阵进行拼接,可得到 AFMN 模型的最终输出 Y 。具体可表达为:

$$Y = \text{concat}(M, f^q). \quad (9)$$

该输出能够指导后续的分类和跟踪任务。具体来说,上述方法获得的输出能够有效关注目标特征信息,在进行特征匹配时给予目标所在区域更高的评分,从而实现更精确的跟踪结果。

2.3 FMN

为有效利用历史帧提供的目标信息,本文算法将回归跟踪框后的图像帧存储在特征记忆网络中。在每次跟踪新的查询帧时,从 FMN 中选择相应的记忆图像,并通过骨干网络进行特征提取。由于孪生神经网络的特性,骨干网络可以提取与查询帧目标相似的特征。将记忆特征与查询帧特征一起输入到 AFMN 中,即使目标被遮

挡,由于 AFMN 是逐像素计算相似度,因此也可以实现局部到局部的特征匹配。本文认为初始帧包含最准确的目标特征,而与查询帧相近的记忆帧包含与查询帧最相似的特征,因此记忆帧包含第一帧和查询帧的前一帧,余下 $T-2$ 帧通过以下公式来进行选取:

$$idx = \sum_{l=1}^{T-2} (P-1) // 2^l, (l=1, 2, \dots, T-2), \quad (10)$$

其中: T 为要选取的帧数, P 为当前帧的索引, l 为间隔系数,即隔多少帧选取一张记忆帧, idx 即为最终的记忆帧索引。通过记忆帧的选取,整个算法通过对多帧目标位置进行学习,在目标被完全遮挡时,也可以自适应推测目标存在位置,并具有较高的精度。

2.4 分类回归网络

根据文献[15]阅读发现,无锚检测器在检测框的标定和回归方面表现更佳且参数更少,因此采用无锚头网络进行分类回归任务。如图 8 所示,该网络包含分类分支、中心度分支和回归分支,每个分支分别使用轻量级网络 γ^{cls} , γ^{reg} 中的七个卷积层对 AFMN 传入的数据进行处理,然后使用单个卷积层 w^{cls} , w^{ctr} , w^{reg} 将其降维以进行分类回归。分类分支用于目标背景分类,中心度分支使用 FCOS 的中心度公式来加强预测框回归效果,回归分支直接估计目标位置并进行检测框标定。

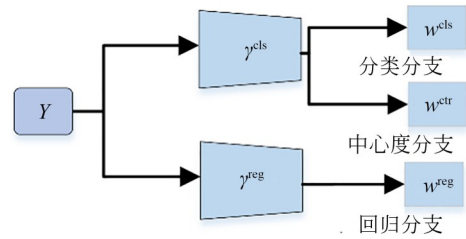


图 8 分类回归网络

Fig. 8 Classification regression network

在本任务中,为解决目标遮挡问题,采用一系列损失函数。首先,分类分支采用 FocalLoss 损失函数,可用公式表示为:

$$Loss = \frac{1}{N} \sum_{i=1}^N \sum_{j=1}^{H \times W} (-\alpha(1-p_{ij})^\gamma \log(p_{ij})) \cdot m_{ij}, \quad (11)$$

其中: $Loss$ 表示损失函数值, N 表示批次大小, H 和 W 分别表示预测结果的高度和宽度, p_{ij} 表示预测结果中位置 i, j 的概率, m_{ij} 表示掩码, α 和 γ 是

可调节的超参数,用于表示损失的重要程度。该损失函数能够有效应对样本不均衡的情况。通过调整难易样本的权重,FocalLoss使模型更关注难以跟踪的目标,从而提高对少数类别目标的学习能力,进一步提升跟踪准确性。

其次,中心度分支使用交叉熵损失函数,并结合特征图像素点与目标中心点的距离进行权重赋值。可用公式表示为:

$$Loss = \frac{1}{N} \sum_{i=1}^N (-y_i \log(\hat{y}_i) - (1 - y_i) \log(1 - \hat{y}_i)) + \frac{\lambda}{N} \sum_{i=1}^N (y_i \log(y_i) - (1 - y_i) \log(1 - y_i)), \quad (12)$$

其中: y_i 是目标中心度的标签, $\hat{y}_i$ 是预测的中心度值, N 是样本的数量, λ 是权重参数,用于调节损失值权重。通过交叉熵损失函数对比预测的中心度值和目标中心度的标签值,来衡量模型的中心度预测准确程度,并对背景标签进行了排除。最终计算所有样本损失的平均值,并根据权重参数进行损失调节。这种距离加权策略能够抑制远离目标中心点的像素得分,使模型更加关注目标的中心区域。

对于回归分支,采用IOU损失函数来拟合更准确的预测框坐标。可用公式表示为:

$$Loss = -\log(IOU), \quad (13)$$

式中, IOU 代表预测框与标签的交并比。

最后,将上述损失函数作为子损失函数,采用多元交叉熵损失进行最终的损失计算,权重比例为0.2:0.2:0.6。这样的设置在文献[14]中得到了论证,能够综合考虑分类、中心度和回归任务,使模型更好地适应目标跟踪任务的特性,提升整体性能。

3 实验结果分析

3.1 实验设计

本文网络使用GOT-10k,COCO和LaSOT数据集进行训练。骨干网络采用预训练的Inception V3模型,将AFMN的 h 和FMN的 T 都设为3,训练过程采用SGD优化器,整个训练过程包括20个周期,每个周期包含38 000个数据,初始学习率设为 1×10^{-6} ,在第一个周期,采用线性学习率,使其增长至 6×10^{-3} ,随后两个周期,采用余弦退火学习率,学习率从 6×10^{-3} 呈余弦变化下降至 1×10^{-6} ,并在后续所有周期中保持 1×10^{-6} 不变。

3.2 算法对比

为了全面地验证本文算法的有效性,在多个目标跟踪数据集上进行了测试,包括OTB-2015,VOT2018,GOT-10k和LaSOT数据集。这些数据集具有挑战性和多样性,能够评估算法在不同场景下的鲁棒性和准确性。

在测试中,本文算法表现出良好的跟踪效果。如表1所示,算法在OTB-2015数据集上表现出色,跟踪精度优于大多数的跟踪器。但在VOT2018数据集中,如表2所示,跟踪准确度低于绝大多数跟踪器,鲁棒性却优于其他跟踪器,这是因为VOT2018评估工具中所提供的跟踪框是旋转的,而本文算法回归的跟踪框是平行于 x 轴和 y 轴的,因此在VOT2018数据集中的表现会受到一定影响,但这也侧面印证了本文算法具有良好的鲁棒性。此外,如表3所示,在GOT-10k数据集上,算法也取得很好的跟踪效果,相比于STMTTrack算法,AO提高了1.8%, $SR_{0.5}$ 提高了

表1 在OTB-2015数据集上,AFMN与其他跟踪器的比较

Tab. 1 On the OTB-2015 dataset, AFMN compares to other trackers

跟踪器	Success	Precision	跟踪器	Success	Precision
AFMN	0.715	0.928	ToMP-50 ^[16]	0.701	—
STMTTrack ^[14]	0.719	0.934	MixFormer-1k ^[9]	0.696	0.911
SBT large ^[8]	0.719	0.924	SiamRPN++ ^[5]	0.696	0.914
SAOT ^[17]	0.714	0.926	KYS ^[18]	0.695	—
SiamAttn ^[19]	0.712	0.926	Ocean ^[20]	0.684	0.899
UPDT ^[21]	0.702	0.919	SiamFC++ ^[15]	0.683	—

表 2 在 VOT2018 数据集上,AFMN 与其他跟踪器的比较
Tab. 2 AFMN compares to other trackers on VOT2018 dataset

跟踪器	$EAO \uparrow$	$A \uparrow$	$R \downarrow$
AFMN	0.452	0.593	0.156
STMTTrack ^[14]	0.447	0.590	0.159
D3S ^[7]	0.489	0.640	0.150
Ocean ^[20]	0.489	0.592	0.117
SiamAttn ^[19]	0.470	0.630	0.160
KYS ^[18]	0.462	0.609	0.143
SiamBAN ^[22]	0.452	0.597	0.178
PrDiMP-50 ^[23]	0.442	0.618	0.165
DiMP-50 ^[24]	0.440	0.597	0.153
Siam R-CNN ^[25]	0.408	0.609	0.220
SiamFC++ ^[15]	0.426	0.587	0.183
SiamRPN++ ^[5]	0.414	0.600	0.234

2.4%, $SR_{0.75}$ 提高了 1.9%, 并且超过了平均性能水平。在 LaSOT 数据集上, 本文算法也表现出

了竞争力, 能够在跟踪任务中取得良好的结果。总的来说, 本文算法在多个数据集上的测试结果都表现出了较好的鲁棒性和准确性。同时, 针对目标遮挡、目标消失和背景干扰等情况时, 具有更高的鲁棒性。面对复杂场景时的跟踪效果在可视化分析部分进行展示。

3.3 可视化分析

为了验证本文算法在目标遮挡场景下的性能, 本节使用包含遮挡目标的视频序列进行算法性能的评估和分析, 将本算法与 STMTTrack 算法进行对比, 如图 9 所示, 本文算法在面对目标遮挡和目标消失时, 仍然可以准确对目标进行定位和跟踪, 在图 9(c) 中, 目标被完全遮挡, 由于本文算法是多记忆帧计算目标相似度, 网络可以隐性的学习目标运动趋势, 进而对目标所在位置进行估计, 所以实现了更准确的跟踪。这些结果表明, 本文算法可以有效地解决遮挡问题, 提高跟踪精度和鲁棒性。进一步证明了本文算法在目标遮挡状态下的有效性。

表 3 在 GOT-10k 和 LaSOT 数据集上,AFMN 与其他跟踪器的比较
Tab. 3 AFMN compares to other trackers on GOT-10k and LaSOT dataset

跟踪器	GOT-10k			LaSOT		
	$AO \uparrow$	$SR_{0.5} \uparrow$	$SR_{0.75} \uparrow$	$AUC \uparrow$	$P_{Norm} \uparrow$	$P \uparrow$
AFMN	0.660	0.761	0.594	0.608	0.696	0.636
STMTTrack ^[14]	0.642	0.737	0.575	0.606	0.693	0.633
MixFormer-1k ^[9]	0.732	0.832	0.702	0.679	0.773	0.739
STARK ^[26]	0.688	0.781	0.641	0.671	0.770	—
SBT large ^[8]	0.704	0.808	0.647	0.667	—	0.717
TrDiMP ^[23]	0.671	0.777	0.583	0.639	—	0.614
AutoMatch ^[27]	0.652	0.766	0.543	0.582	—	0.599
SAOT ^[17]	0.640	0.759	—	0.616	0.708	—
Siam R-CNN ^[25]	0.649	0.728	0.597	0.648	0.722	—
Ocean ^[20]	0.611	0.721	0.473	0.560	0.651	0.566
D3S ^[7]	0.597	0.676	0.462	—	—	—
SiamFC++ ^[15]	0.595	0.695	0.479	0.544	0.623	0.547
SiamRPN++ ^[5]	0.517	0.616	0.325	0.496	0.569	0.491

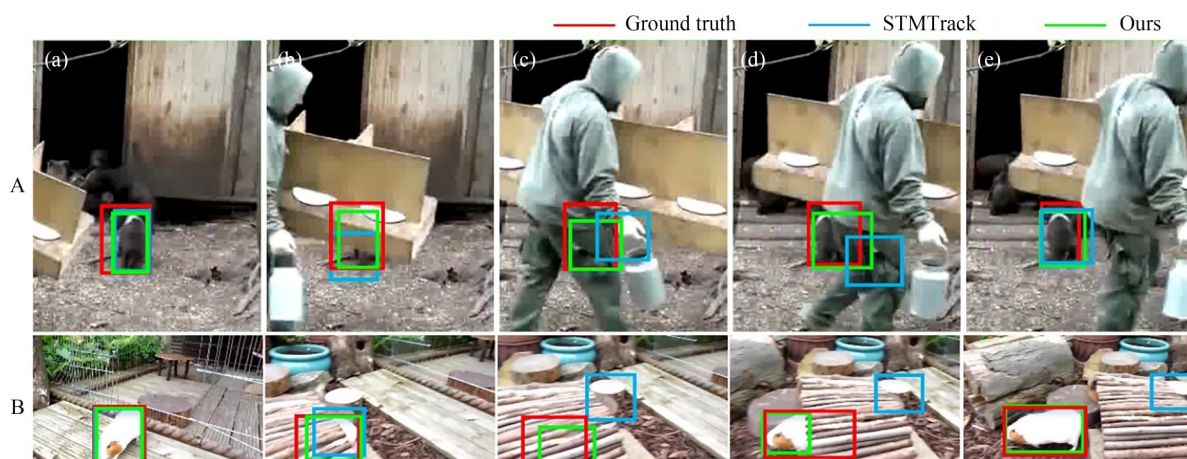


图 9 可视化对比

Fig. 9 Visual comparison



图 10 目标遮挡场景下的可视化结果

Fig. 10 Visualization results in the object occlusion scenario

同时,图 10 展示了本文算法在目标遮挡场景下更多的跟踪效果,在图 10(a),图 10(c)和图 10(d)中,目标被部分遮挡,均实现了准确跟踪,在图 10(b)中,目标短时间内完全消失,本文算法依旧自适应推断出了目标位置。实验结果表明,本文算法在目标遮挡状态下具有良好的鲁棒性和稳定性。

3.4 消融实验

为验证 AFMN 中特征图拆份数对跟踪结

果的影响,仅在 GOT-10k 数据集上进行训练和测试。该数据集包含超过 10 000 个视频序列,其中涵盖了大量目标遮挡场景,因此能够有效验证本文算法的有效性。实验结果如表 4 所示。

根据表 4 可以得出结论:当特征图被拆分为 3 份时,跟踪效果最佳。这是因为特征图中存在许多重复的特征通道,将其分成多份后可以进行多个角度的相似度比较。但由于特征图通道数有限,若份数过多,则每份所包含的目标特征就

表 4 h 对 AO 的影响Tab. 4 Influence of h on AO

h 数量	AO	$SR_{0.5}$	$SR_{0.75}$
1	0.642	0.737	0.575
2	0.656	0.755	0.586
3	0.660	0.761	0.594
4	0.649	0.746	0.577
5	0.642	0.740	0.570
8	0.639	0.737	0.561

不足以充分表达目标特性,会忽略部分特征,从而导致跟踪精度降低。

为验证 FMN 中记忆帧数对跟踪结果的影响,同样在 GOT-10k 数据集上进行训练和测试,并得到实验结果,如表 5 所示。

根据表 5 的数据,当记忆帧数为 3 时,跟踪效果最佳。这是因为过少的记忆帧无法提供足够的目标表观信息,从而导致跟踪结果不佳。

表 5 T 对 AO 的影响Tab. 5 Influence of T on AO

T 数量	AO	$SR_{0.5}$	$SR_{0.75}$
1	0.598	0.638	0.437
2	0.645	0.743	0.573
3	0.660	0.761	0.594
4	0.634	0.728	0.567
5	0.628	0.728	0.548
6	0.638	0.738	0.566

而当记忆帧过多时,如在目标遮挡的场景中,跟踪器会更倾向于匹配相似度最高的区域,而忽略被遮挡的目标,从而导致跟踪精度降低。因此,选择适当的记忆帧数可以提高跟踪器的性能。

4 结 论

本文针对目标遮挡问题提出了一种自适应特征匹配网络,该网络通过一个有效的模块 AFMN 对骨干网络提取的特征进行处理,并将目标遮挡问题转化为背景估计和目标位置估计的联合优化问题,通过计算记忆帧与查询帧的像素级相似度,将目标和背景分别编码,进而确定某一区域属于背景或目标,以此来提高目标跟踪的精度和鲁棒性。并且,通过特征记忆网络对记忆帧进行挑选和保存,为特征匹配提供额外的表观信息,同时使网络隐性的学习目标运动趋势,进而实现更好的跟踪结果。在 Got-10k 数据集上的实验结果表明,本文所提出的算法与 STMTrack 算法相比, AO 值提升 1.8%, $SR_{0.5}$ 提升 2.4%, $SR_{0.75}$ 提升 1.9%, 在使用一张 NVIDIA 1080Ti 显卡时,运行速度可达 21 FPS。并且在处理目标遮挡问题时具有良好的性能表现,与当前流行的目标跟踪算法相比具有更高的精度和更强的鲁棒性。

参考文献:

- [1] DANELLJAN M, BHAT G, KHAN F S, *et al.* ATOM: accurate tracking by overlap maximization [C]. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 15-20, 2019, Long Beach, CA, USA. IEEE, 2020: 4655-4664.
- [2] DANELLJAN M, VAN GOOL L, TIMOFTE R. Probabilistic regression for visual tracking [C]. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 13-19, 2020, Seattle, WA, USA. IEEE, 2020: 7181-7190.
- [3] LI B, YAN J J, WU W, *et al.* High performance visual tracking with siamese region proposal network [C]. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. 18-23, 2018, Salt Lake City, UT, USA. IEEE, 2018: 8971-8980.
- [4] FAN H, LING H B. Siamese cascaded region proposal networks for Real-Time visual tracking [C]. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 15-20, 2019, Long Beach, CA, USA. IEEE, 2020: 7944-7953.
- [5] LI B, WU W, WANG Q, *et al.* SiamRPN: evolution of siamese visual tracking with very deep networks [C]. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 15-20, 2019, Long Beach, CA, USA. IEEE, 2020: 4277-4286.
- [6] WANG Q, ZHANG L, BERTINETTO L, *et al.* Fast online object tracking and segmentation: a uni-

- fyng approach [C]. 2019 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 15-20, 2019, Long Beach, CA, USA. IEEE, 2020: 1328-1338.
- [7] LUKEŽIĆ A, MATAS J, KRISTAN M. D3S-A discriminative single shot segmentation tracker [C]. 2020 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 13-19, 2020, Seattle, WA, USA. IEEE, 2020: 7131-7140.
- [8] XIE F, WANG C Y, WANG G T, *et al.* Correlation-Aware deep tracking [C]. 2022 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 18-24, 2022, New Orleans, LA, USA. IEEE, 2022: 8741-8750.
- [9] CUI Y T, JIANG C, WANG L M, *et al.* MixFormer: End-To-End tracking with iterative mixed attention [C]. 2022 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 18-24, 2022, New Orleans, LA, USA. IEEE, 2022: 13598-13608.
- [10] OH S W, LEE J Y, XU N, *et al.* Video object segmentation using Space-Time memory networks [C]. 2019 *IEEE/CVF International Conference on Computer Vision (ICCV)*. October 27 - November 2, 2019, Seoul, Korea (South). IEEE, 2020: 9225-9234.
- [11] XIE H Z, YAO H X, ZHOU S C, *et al.* Efficient regional memory network for video object segmentation [C]. 2021 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 20-25, 2021, Nashville, TN, USA. IEEE, 2021: 1286-1295.
- [12] PAUL M, DANELLJAN M, VAN GOOL L, *et al.* Local memory attention for fast video semantic segmentation [C]. 2021 *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. September 27-October 1, 2021, Prague, Czech Republic. IEEE, 2021: 1102-1109.
- [13] WANG H, WANG W N, LIU J. Temporal memory attention for video semantic segmentation [C]. 2021 *IEEE International Conference on Image Processing (ICIP)*. 19-22, 2021, Anchorage, AK, USA. IEEE, 2021: 2254-2258.
- [14] FU Z H, LIU Q J, FU Z H, *et al.* STMTTrack: Template-Free visual tracking with Space-Time memory networks [C]. 2021 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 20-25, 2021, Nashville, TN, USA. IEEE, 2021: 13769-13778.
- [15] XU Y D, WANG Z Y, LI Z X, *et al.* Siam-FC++: towards robust and accurate visual tracking with target estimation guidelines [J]. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020, 34(7): 12549-12556.
- [16] MAYER C, DANELLJAN M, BHAT G, *et al.* Transforming model prediction for tracking [C]. 2022 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 18-24, 2022, New Orleans, LA, USA. IEEE, 2022: 8721-8730.
- [17] ZHOU Z K, PEI W J, LI X, *et al.* Saliency-Associated object tracking [C]. 2021 *IEEE/CVF International Conference on Computer Vision (ICCV)*. 10-17, 2021, Montreal, QC, Canada. IEEE, 2022: 9846-9855.
- [18] BHAT G, DANELLJAN M, VAN GOOL L, *et al.* Know Your Surroundings: Exploiting Scene Information for object Tracking [M]. Computer Vision - ECCV 2020. Cham: Springer International Publishing, 2020: 205-221.
- [19] YU Y C, XIONG Y L, HUANG W L, *et al.* Deformable siamese attention networks for visual object tracking [C]. 2020 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 13-19, 2020, Seattle, WA, USA. IEEE, 2020: 6727-6736.
- [20] ZHANG Z P, PENG H W, FU J L, *et al.* Ocean: Object-Aware Anchor-Free Tracking [M]. Computer Vision - ECCV 2020. Cham: Springer International Publishing, 2020: 771-787.
- [21] BHAT G, JOHNANDER J, DANELLJAN M, *et al.* Unveiling the Power of Deep Tracking [M]. Computer Vision-ECCV 2018. Cham: Springer International Publishing, 2018: 493-509.
- [22] CHEN Z D, ZHONG B N, LI G R, *et al.* SiamBAN: target-aware tracking with Siamese box adaptive network [J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022: 1-17.
- [23] WANG N, ZHOU W G, WANG J, *et al.* Transformer meets tracker: exploiting temporal context for robust visual tracking [C]. 2021 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 20-25, 2021, Nashville, TN,

- USA. IEEE, 2021: 1571-1580.
- [24] BHAT G, DANELLJAN M, VAN GOOL L, *et al.* Learning discriminative model prediction for tracking[C]. 2019 *IEEE/CVF International Conference on Computer Vision (ICCV)*. October 27 - November 2, 2019, Seoul, Korea (South). IEEE, 2020: 6181-6190.
- [25] VOIGTLAENDER P, LUITEN J, TORR P H S, *et al.* Siam R-CNN: visual tracking by re-detection[C]. 2020 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 13-19, 2020, Seattle, WA, USA. IEEE, 2020: 6577-6587.
- [26] YAN B, PENG H W, FU J L, *et al.* Learning Spatio-Temporal transformer for visual tracking [C]. 2021 *IEEE/CVF International Conference on Computer Vision (ICCV)*. 10-17, 2021, Montreal, QC, Canada. IEEE, 2022: 10428-10437.
- [27] ZHANG Z P, LIU Y H, WANG X, *et al.* Learn to Match: automatic matching network design for visual tracking[C]. 2021 *IEEE/CVF International Conference on Computer Vision (ICCV)*. 10-17, 2021, Montreal, QC, Canada. IEEE, 2022: 13319-13328.

作者简介:



毛 琳(1977—),女,山东省荣成人,博士,副教授,硕士生导师,2005年于黑龙江大学获得硕士学位,2011年于哈尔滨工程大学获得博士学位,主要从事机器视觉目标跟踪与多传感器信息融合的研究。E-mail:maolin@dlnu.edu.cn

通讯作者:



苏宏阳(2000—),男,内蒙古兴安盟人,硕士研究生,2021年于大连民族大学获得学士学位,主要从事计算机视觉和目标跟踪研究。E-mail:wxhxxh-wdn0725@163.com