自动化学报
*Acta Automatica Sinica*
ISSN 0254-4156,CN 11-2109/TP

# 《自动化学报》网络首发论文

题目：深度强化学习联合回归目标定位
作者：姚红革，张玮，杨浩琪，喻钧
DOI：10.16383/j.aas.c200045
收稿日期：2020-01-20
网络首发日期：2020-09-25
引用格式：姚红革，张玮，杨浩琪，喻钧．深度强化学习联合回归目标定位．自动化学报. https://doi.org/10.16383/j.aas.c200045

自 动 化 学 报
ACTA AUTOMATICA SINICA

# 深度强化学习联合回归目标定位

姚红革 [1] 张玮 [1] 杨浩琪 [1] 喻钧 [1]

**摘 要：**为了模拟人眼的视觉注意机制,快速、高效地搜索和定位图像目标,本文提出了一种基于循环神经网络的联合回归深度强化学习目标定位模型,该模型将历史观测信息与当前时刻的观测信息融合并做出综合分析,以训练 Agent(智能体)快速定位目标,并联合回归器对 Agent 所定位的目标包围框进行精细调整.实验表明,所提出的模型能够在少数时间步内快速、准确地定位目标.

**关键词** 视觉注意机制, 循环神经网络, 深度强化学习, 目标定位

**引用格式** 姚红革, 张玮, 杨浩琪, 喻钧. 深度强化学习联合回归目标定位. 自动化学报,2020,**XX**(X):X-X

**DOI** 10.16383/j.aas.c200045

## Joint regression object localization based on deep reinforcement learning

YAO Hong-Ge[1] ZHANG Wei[1] YANG Hao-Qi[1] YU Jun[1]

**Abstract** To simulate the visual attention mechanism of the human eye, search and locate image objection quickly and efficiently. This paper proposes a joint-regression deep reinforcement learning object localization model based on recurrent neural network, which fuses the historical observation information with the observation information at the current time , then makes a comprehensive analysis to train the Agent to quickly locate the object, and combine with the regressor to fine-tune the object bounding box positioned by the Agent. Experiments show that the proposed model can accurately and rapidly locate the object in a few time steps.

**Keyword** Visual attention mechanism, recurrent neural network, deep reinforcement learning, object detection

**Citation** YAO Hong-Ge, ZHANG Wei, YANG Hao-Qi, YU Jun. Joint regression object localization based on deep reinforcement learning. *Acta Automatica Sinica*, 2020,**XX**(X):X-X

人眼视觉在观察客观事物时,不是关注所有信息,而是会选择性地关注所感兴趣的那一部分,同时忽略其它可见的信息,然后再将注意力转移到下一个位置进行观察,最后汇总所有信息得到结论.这种注意力机制[1]涉及两方面的问题：一是历史信息的获取和应用；二是将历史信息和当前信息融合,使用融合信息确定新的关注位置.

对于历史信息的获取与应用,循环神经网络[2](Recurrent Neural Network, RNN)在这方面具有优势,它的输入不仅包含当前的观察,还包含之前感知到的“历史”,使得当前信息和历史信息相互融合,这种融合体现了对输入信息更全面的描述.对于新的关注位置的决策,深度强化学习[3-4]中的 DQN[5-6](Deep Q-Network)能够模拟人脑对环境状态的感知能力,并对较为复杂的决策做出判断,这就为位置决策问题提供了解决思路.

鉴于此,本文将深度强化学习与 RNN 相结合,提出了一种基于 RNN 的联合回归深度强化学习目标定位模型(UR-DRQN 检测模型).它将历史经验运用

收稿日期 2020-01-20 录用日期 2020-09-07
Manuscript received January 20, 2020; accepted September 7, 2020
本文责任编委 张军平
Recommended by Associate Editor ZHANG Jun-Ping
1. 西安工业大学计算机科学与工程学院 西安 710021
1. School of Computer Science and Engineering, Xi'an Technological University, Xi'an 710021

到新场景的观察中来,即用 RNN 获取的历史信息与最新的观测信息融合,使模型能在较短的时间内找到符合要求的包围框,以缩短检测时间.同时,还设计了一个回归器对最终的定位包围框进行精调,以期进一步提高定位精度.实验表明,该模型能够较好地平衡定位的速度和精度.其特点如下:

- 融合历史信息的粗定位.使用 RNN 将当前时刻新的观测信息和过往的历史信息相融合,避免传统穷举搜索候选区域来确定目标位置的做法,并对融合信息进行分析做出动作决策,实现对图像潜在目标区域的粗定位,提升了网络的检测效率.
- 粗定位后的细调整.设计了一个回归网络对由粗定位获得的包围框进行精细调整,进一步提升网络的检测精度.
- 使用 IoU(Intersection over Union)[7]作为动作执行的奖励评判标准.IoU 使预测区域与标签区域进行直接比对,提升了强化学习方法在进行目标搜索时动作选择的准确度.
- 动态调整搜索动作.在动作网络中,不同于一般强化学习所采用的预定义搜索动作的方式,以及目标包围框尺度固定的方式,而是让搜索动作与包围框尺度都能够随环境状态的改变而改变.

## 1 相关工作

**IoU 定位** 目标定位中的很多算法,例如 R-CNN[8]、Fast R-CNN[9]、Faster R-CNN[10]等都是基于 IoU 的,它们都取得了非常好的检测效果.IoU 比值越高说明预测的包围框越准确.当满足设定阈值的预测包围框产生后,使用回归器对该包围框坐标进行精调,以达到精确定位目标位置的目的.本文不是将 IoU 用来衡量定位的准确度,而是用 IoU 值的变化来刻画动作执行效果:通过 IoU 的变化给予动作以不同的奖励,以此来学习做出可以获得更高 IoU 值的动作.当动作调整使得 IoU 达到设定的阈值时,通过回归网络对可视区域的边界框精细调整,使其能够紧凑地包围要寻找的目标.

**RNN 与信息融合** RNN 具有特殊网络结构,使得它能够将历史信息与新的输入信息相融合,这与人眼的视觉注意机制非常类似.2014 年,Volodymyr 等人[11]为了模拟人眼视觉进行目标检测,提出了一种基于 RNN 的视觉注意模型,该模型逐次地处理所观测图像的一部分,并递增地组合来自这些观测的信息以建立场景的动态内部表示,再利用网络的内部状态来选择下一个要关注的位置,并以此循环学习,最终通过对融合信息的汇总分析完成目标的定位与识别,该模型在 MNIST 数据集上的检测效果非常好.本文使用 RNN 模拟人眼的视觉注意机制,将当前时刻新的观测信息与过往的历史信息相融合,并在此基础上设计动作网络和回归网络对融合信息进行分析做出动作决策,对图像潜在目标区域进行粗定位.

**DQN 与目标检测** DQN 在决策控制领域的表现优异,许多研究人员将其应用于目标定位领域.如 2015 年,Caicedo 等人[12]提出了一种基于 DQN 的目标定位算法,采用自上而下的搜索策略分析整个目标场景,通过对初始的大边界框执行一系列形变动作,最终将目标以较小的边界框包围起来达到定位目的,但是该算法每次都需要对目标场景进行特征计算,计算资源消耗较大.2016 年,Bueno 等人[13]提出了一种层次化的目标定位模型,该模型通过将注意力不断地聚焦在包含更多信息的区域,学习不同的动作选择策略从预定义的五个子区域中选择最有可能包含目标的区域,不断迭代以缩小目标的包围框范围,这样虽然可以大大减少所需要的操作数量,提高检测的速度,但该方法检测精度不高,还有可能错过一些具有更好位置的边界框.2017 年,Hara 等人[14]提出了一种基于注意机制的视觉目标定位方法,该网络在图像的不同位置自适应的进行一系列不同形状的"瞥见",然后从这些"瞥见"中提取相关信息进行融合来估计物体的类别和边界框坐标,实验表明该网络的注意力机制性能优于一般目标定位网络.基于以上研究,本文使用 DQN 将 Agent 所感知的环境信息与 RNN 所融合的以往历史信息进行综合分析,给出每一种动作执行的置信概率,在循环迭代的过程中学习做出最优的动作完成最终的定位检测任务.

## 2 UR-DRQN 目标定位模型

### 2.1 目标定位动态决策过程

受人眼视觉注意机制的启发,目标定位问题可以视为 Agent 通过序贯决策与可视环境不断交互进而定位目标的过程,这个过程可以用一个马尔科夫决策过程[15-16](Markov Decision Process, MDP)来表达.在每个时间步,Agent 都会获得一个当前环境的状态(State),并根据这个状态做出相应的动作(Action)

以改变可视区域的大小,这个动作的效果将会由一个正向的或负向的奖励值(Reward)来衡量.然后,由上一个动作所引起的环境变化产生新的状态,Agent就在这样的不断循环中逐步学习做出更好的决策来找到目标.

- **状态**

人眼在观察客观事物时,总是不断地将目光所聚焦的视觉内容与大脑记忆的历史信息相结合,当累积的信息足够丰富时便可以得到对事物的认识.状态就是对这一过程的程序化模拟,它是历史信息和图片可视区域信息的融合表示,将来自过去的观察状况和动作执行情况与当前的关注信息相结合,以指导 Agent 做出下一步的决策,状态信息的融合过程如图 1 所示.其中,图片可视区域信息包括关注区域的中心位置信息和可视区域的范围信息,这些信息形成了 Agent 的观测区域,属于类似人眼的视觉内容,称为“类眼视觉内容”.历史信息则是对过去关注区域信息以及所采取的动作信息的汇总,历史信息的传递有助于 Agent 获得更多的经验,基于新的信息做出更好的决策,类似人脑存储的以往视觉信息,称为“类脑历史信息”.这种类眼与类脑信息融合与传递的过程由 RNN 实现.

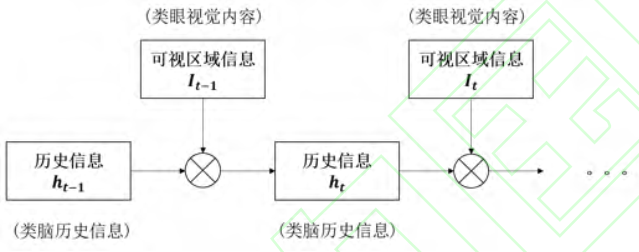


图 1 状态信息融合表示

Fig.1 Fusion representation of state information

- **动作**

Agent 所采取的动作分为两类:调整动作和终止动作.调整动作用来对可视区域的范围进行变换,包括方位变换、尺度变换和纵横比变换.终止动作用来标识目标定位过程的结束,终止动作一旦出现,则表示目标已经找到并且寻找过程结束.两类动作的描述如图 2 所示.

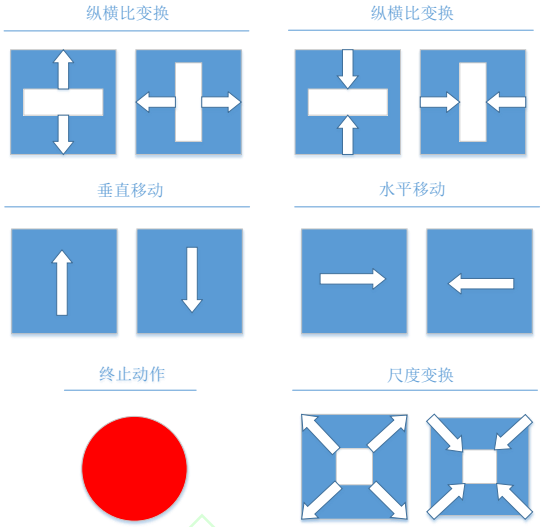


图 2 动作示意图

Fig.2 Schematic diagram of action

动作的选择通过 DQN 实现,在这个过程中采取 $\varepsilon-greedy$ 策略[17]兼顾探索与利用[18],见式(1),$\varepsilon-greedy$ 策略既保证了对新动作的不断探索和尝试,也兼顾了对最优动作的持续利用.

$$\pi(a|s)\leftarrow\begin{cases}1-\varepsilon+\dfrac{\varepsilon}{|A(s)|}, & \text{if } a=\operatorname{argmax}_a Q(s,a)\\ \dfrac{\varepsilon}{|A(s)|}, & \text{if } a\neq\operatorname{argmax}_a Q(s,a)\end{cases} \tag{1}$$

其中,$s$ 是 Agent 当前所处的状态,$a$ 是 Agent 基于当前状态所采取的动作,$A(s)$ 是 Agent 在状态 $s$ 时可以选择的动作集合,$|A(s)|$ 表示可以选择的动作数量,$\varepsilon\in[0,1]$ 是探索因子,$\pi(a|s)$ 是一个策略,它表示了给定状态 $s$ 时,动作集上的一个分布.如果策略 $\pi$ 是确定性的,那么策略 $\pi$ 在每个状态 $s$ 下会指定一个确定的动作 $a$.

- **奖励**

奖励函数直观地反映了在当前状态 Agent 所采取的区域调整动作的效果.对于动作的效果刻画,使用目标真实标记区域与 Agent 可视区域的 IoU 来衡量.记 $b$ 为 Agent 的可视区域,$g$ 为目标物体的真实标记区域,则它们之间的 IoU 由式(2)定义.

$$\mathrm{IoU}(b,g)=\frac{\mathrm{area}(b\cap g)}{\mathrm{area}(b\cup g)} \tag{2}$$

在每一个状态,Agent 都会获得一个可视区域 $b$,然后与真实标记区域 $g$ 计算 IoU 值,当 Agent 采取调整动作 $a$ 使得所处状态从 $s$ 跃迁为 $s'$ 时,就可以通过

IoU值的变化获得一个奖励值Reward,该奖励值由式(3)所示奖励函数$R_a(s,s')$定义.

$$R_a(s,s') = \text{sign}(\text{IoU}(b',g) - \text{IoU}(b,g)) \tag{3}$$

直观上看,该函数表明了当状态从$s$跃迁为$s'$时,如果IoU值有所提升则奖励为"+1",相反,如果IoU值有所损失则奖励为"-1".之所以使用sign函数是因为IoU值的差异可能十分微小,以至于Agent无法判断出这个动作所带来的效果,通过对奖励函数值的二元划分可以更明确的指导Agent将目标包含在可视区域范围内.特别地,将IoU值大于设定阈值时的动作定义为终止动作,意味着已经找到目标而不再产生新的状态变化.由此,终止动作的奖励函数可由式(4)描述.

$$R_t(s,s') = +\eta, \text{当且仅当} \text{IoU}(b,g) \geq \tau \tag{4}$$

其中,$t$为终止动作,$\eta$为终止动作的奖励值,$\tau$为判定阈值,IoU大于等于$\tau$表明可视区域较好地包含了目标区域.

- **状态转移概率与$Q$值更新**

在MDP中,当新的动作产生时,环境变化得到新的状态,这样的状态改变叫做状态转移,与不同的状态转移相关的概率叫做状态转移概率,可由式(5)描述.

$$P_{ss'}^{a} = P[S_{t+1} = s' | S_t = s, A_t = a] \tag{5}$$

其中,$P$是状态转移概率,$S$是所有状态$s$的有限集合,$A$是所有动作$a$的有限集合,$t$是不同的时刻,该式表明了当Agent在状态$s$时采取动作$a$使得状态跃迁为$s'$时的状态转移概率.

在DQN中,由于状态转移概率是未知的,所以使用$Q$值来刻画状态转移概率,当Agent处在某一状态时,会对下一个可能到达的状态计算其$Q$值,并选择使得$Q$值最大的动作去执行,继而进入新的状态.本文所提出的UR-DRQN模型使用DQN来决策Agent在某一状态下应采取的动作,网络的输入是Agent所处的状态,即历史信息与图片可视区域信息的融合表示,输出为通过Q-learning学习算法估计出的$Q(s,a)$值,基于$Q(s,a)$值,Agent便可以决策出更有可能取得高奖励值的动作.在训练过程中,$Q(s,a)$值通过式(6)所示的贝尔曼方程[19]进行迭代更新.

$$Q(s,a) = r + \gamma \max_{a'} Q(s',a') \tag{6}$$

其中,$s$和$s'$分别是Agent在当前时间步和下一时间步所处的状态,$a$和$a'$分别是Agent基于当前状态和下一状态所采取的动作,$r$是Agent获得的即时奖励,$\max_{a'} Q(s',a')$是Agent所能获得的最大未来奖励,$\gamma$是奖励折扣因子.

## 2.2 网络结构

本文的UR-DRQN网络结构如图3所示,整体使用RNN循环神经网络进行实现.其中,$I_t$是图片的注意位置信息和可视区域范围信息的综合探测信息,$h_t$是探测信息与网络历史信息的融合,$f_c(\theta_c)$、$f_a(\theta_a)$、$f_l(\theta_l)$、$f_g(\theta_g)$分别表示融合网络、动作网络、位置网络和回归网络.

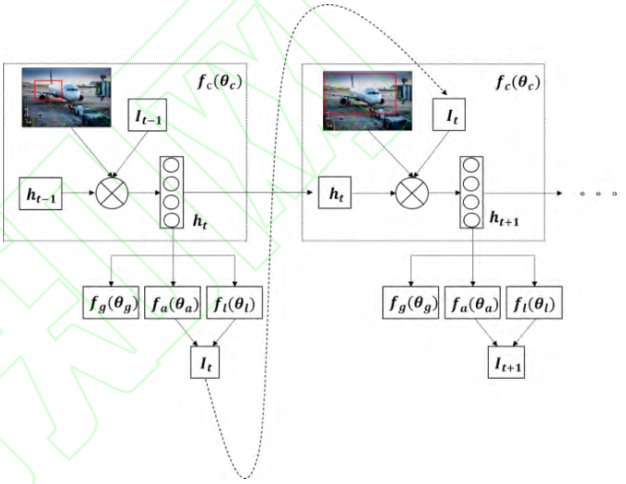


图3 模型整体结构图

Fig.3 Overall structure of the model

- **融合网络$f_c(\theta_c)$:** 在每一轮循环学习中,融合网络都会将接收到的历史信息和一组新的关注位置的坐标信息拼接并输入全连接网络fc(132×128)进行融合,随后将融合信息分别馈送至动作网络和位置网络中以产生新的动作,调整可视区域的大小并获得新的关注位置,重复这样的循环学习至最后一个epoch(轮次)时,同时将融合信息馈送至回归网络中完成对目标包围框的精细调整,如图4所示.

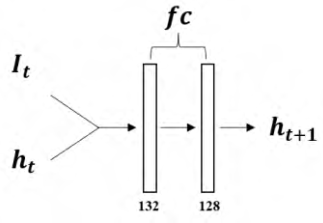


图 4 融合网络 $f_c(\theta_c)$

Fig.4 Integration network $f_c(\theta_c)$

● **动作网络$f_a(\theta_a)$：**动作网络是个结构完全相同的双并行全连接网络 fc(128×128×10),一个产生“预测值”,一个产生“目标值”,只是两者的网络参数是不同的,计算“目标值”时的网络参数是若干时间步前计算“预测值”时的网络参数.在训练阶段,计算“目标值”用来辅助网络参数的学习,在测试阶段,并不计算“目标值”,而是在接收到融合信息$h_t$时,借助$\varepsilon-greedy$策略以概率$\varepsilon$选取一个随机动作,或以概率$1-\varepsilon$通过“预测分支”得到$Q$值确定一个动作,如图 5 所示.

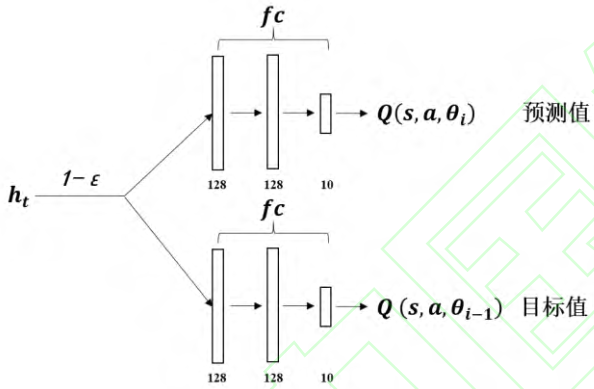


图 5 动作网络 $f_a(\theta_a)$

Fig.5 Action network $f_a(\theta_a)$

● **位置网络$f_l(\theta_l)$：**位置网络则将融合信息$h_t$输入到全连接网络 fc(128×128×1)中,网络结果可以视为一个位置均值$\mu$,然后设置一个固定方差$\sigma^2$的高斯函数$f(\mu,\sigma^2)$,下一个 epoch 在这个高斯函数中进行采样得到位置坐标$l_t$随后这个位置信息和它对应的可视区域范围信息作为输入被送入网络的下一次定位循环中,如图 6 所示.

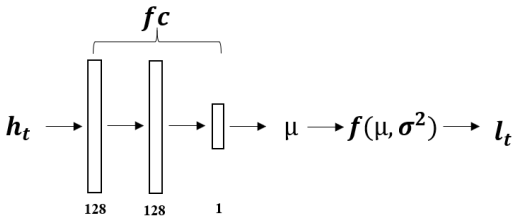


图6 位置网络 $f_l(\theta_l)$

Fig.6 Location network $f_l(\theta_l)$

● **定位与奖励：**当动作网络和位置网络完成决策时,在确定的位置$l_t$按照动作网络$f_a(\theta_a)$的结果 Action 对可视区域范围进行定位.随后,计算新的包围框与目标真实标记区域之间的 IoU 值,并依据 IoU 值得到相应的 Reward.

● **回归网络$f_g(\theta_g)$：**回归网络也是一个全连接网络 fc(128×128×4),当一个 epoch 的学习中产生了终止动作,即可视区域与真实标记区域的 IoU 大于 0.6 时,该网络会对当前的可视区域进行坐标精调,得到包围框调整偏移量$(\Delta x,\Delta y,\Delta h,\Delta w)$,如图 7 所示.

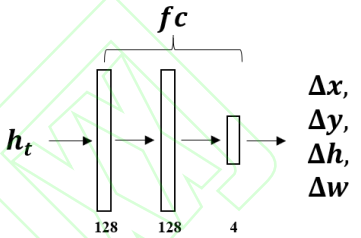


图 7 回归网络 $f_g(\theta_g)$

Fig.7 Regression network $f_g(\theta_g)$

## 2.3 网络训练

### 2.3.1 损失函数

UR-DRQN 模型的综合损失包括动作网络损失和回归网络损失,其损失函数见式(7).

$$Loss = L_{action} + \lambda L_{regression} \tag{7}$$

其中,

$$L_{action} = \frac{1}{N_{action}} \sum [(y_i - Q(s,a;\theta_i))^2] \tag{8}$$

$$L_{regression} = \sum_j \mathrm{S}(t_j - t_j^*) \tag{9}$$

在双并行动作网络的损失$L_{action}$中,$y_i$是动作网络“目标分支”得出的目标值,$Q(s,a;\theta_i)$是动作网络“预测分支”得出的预测值,$N_{action}$是动作网络的执行次数,其值等于目标定位所使用的时间步数,这是因为在定位中会多次执行动作网络,最终执行一次回归网络,故对动作网络误差取其平均值,即$1/N_{action}$.其中$N_{action} \in [1,15]$,这是因为我们从实验中发现,UR-DRQN 模型一般可在 15 个时间步之内找到目标,超过此步数一般失败,故而设定其最多进行 15 个时间步的定位尝试.

在回归网络损失$L_{regression}$中,$t_j=\{t_x,t_y,t_h,t_w\}$表示第$j$个样本的回归包围框的坐标,由回归网络得到.$t_j^*=\{t_x^*,t_y^*,t_h^*,t_w^*\}$是已知的第$j$个样本的训练标签,表示包围目标真实标记区域的包围框坐标,与$t_j$维度相同.其中,$t_x$、$t_y$分别是回归网络得到的回归包围框的中心坐标,$t_h$、$t_w$是其对应的高和宽,$t_x^*$、$t_y^*$分别是包围目标真实标记区域的包围框中心坐标,$t_h^*$、$t_w^*$是其对应的高和宽.S是$\mathrm{Smooth}_{\mathrm{L1}}$函数,见式(10).

$$\mathrm{Smooth}_{\mathrm{L1}}(x)=\begin{cases}0.5x^2,\text{if } |x|<1\\ |x|-0.5,\text{otherwise}\end{cases} \tag{10}$$

$\lambda$是损失平衡系数,用来平衡动作网络和回归网络误差计算在数量级上的差异,实验表明当$\lambda=0.1$时网络效果最佳.

### 2.3.2 训练过程

图 8 至图 10 分别描述了动作网络、回归网络和位置网络的训练更新过程.动作网络的训练使用形如$(s,a,r,s')$的经验元组[6],将该元组馈送到 Q 网络中进行计算,根据贝尔曼方程迭代更新,最后计算损失值$L_{action}$,见式(8).在训练时,只需要训练“预测分支”的网络参数,不需要训练“目标分支”的网络参数.

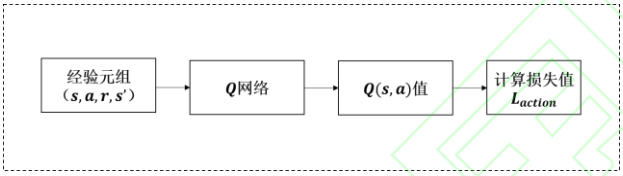


图 8 动作网络训练图

Fig.8 Action network training chart

回归网络根据经验元组$(s,b,g)$计算回归偏移量,最后计算回归预测坐标与目标真实标记区域坐标之间的损失值$L_{regression}$,见式(9).动作网络与回归网络的损失值直接进行累加并通过反向传播算法更新整个模型的权重参数.

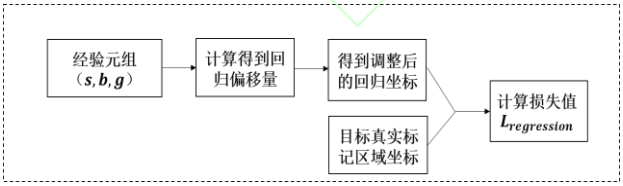


图 9 回归网络训练图

Fig.9 Regression network training chart

位置网络则根据每次得到的融合历史信息动态改变参数化的高斯函数,以此采样新的关注位置,并不参与反向训练.

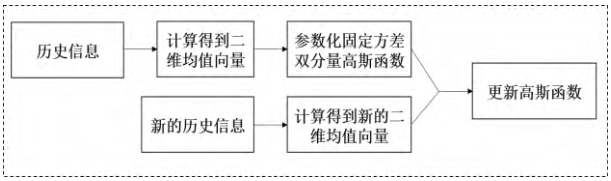


图 10 位置网络训练图

Fig.10 Location network training chart

## 3 实验

实验分别使用了 PASCAL VOC 数据集和 COCO 数据集中的图像和标注数据.在 PASCAL VOC 数据集中,采用 VOC 2007+VOC 2012 的联合训练集训练,测试在 VOC 2007 的测试集中进行.在 COCO 数据集中,采用 COCO 2014 的训练集训练,测试在 COCO 2014 的测试集中进行.

### 3.1 网络训练

#### 3.1.1 参数设置

使用标准正态分布来初始化网络的参数,对模型进行 80 个 epoch 的训练.每个 epoch,网络都进行 15 次循环学习,使用 Adam 优化器进行优化,学习率设置为 1e-6.主要参数确定如下:

1. $\varepsilon$值的确定

在动作网络的训练中,使用$\varepsilon-greedy$策略来对新动作进行探索和尝试,也兼顾对最优动作的持续利用.在实验中,探索因子$\varepsilon$初始设置为 1,并随着时间的推移以步长 0.1 不断递减至 0.1 为止.因此,在初始阶段 Agent 会采取一个随机动作执行,并在往后的每个 epoch 学习中,随着$\varepsilon$的不断减小更多地依赖已经学习到的策略做出动作决策.同时,为了帮助 Agent 更快地学习“终止动作”,将 IoU 值大于判定阈值时的动作视作终止动作.

2.Reward 折扣因子$\gamma$的确定

强化学习的目标是获得更高的累积奖励值,因此,不仅要考虑当下的奖励回报,还要考虑未来的奖励回报.由于环境是随机的,执行特定的动作不一定得到特定的状态,因此未来的奖励所占的“权重”要有

所衰减.在实验中,将贝尔曼方程中的 Reward 折扣因子$\gamma$设置为 0.9.

3.IoU 判定阈值$\tau$的确定

IoU 判定阈值表明了可视区域较好包含目标真实标记区域的最低 IoU 水平,只有当可视区域与目标真实标记区域之间的差距较小时,才可以认为二者之间的变换是一种线性变换,此时就可以使用回归模型对可视区域进行微调.若判定阈值设定较小,则二者间的关系已不满足线性关系,回归模型的效果会变差,甚至不能工作,反之,若判定阈值设定较大,则终止动作的产生需要经历更多的步骤,Agent 的学习负担加重.经过多次测试,在实验中将 IoU 判定阈值$\tau$设置为 0.6 效果较好.

4.Agent 经验数据的确定

通常,将形如$(s,a,r,s')$的元组称作 Agent 学习过程中的经验.DQN 网络一般通过设置经验回放缓冲池[6]将若干学习得到的经验元组存入其中,再随机取出一定数量的经验元组送入 DQN 网络中用作网络训练,这样做的目的是为了打破因数据关联性而导致的网络学习不收敛的状况.在 UR-DRQN 模型中,将经验元组的形式定义为$(s,a,r,s',b,g)$,其中$(s,a,r,s')$用于训练 DQN 网络,$(s,b,g)$用来训练回归网络.

5.可视区域的确定

在每一个时间步,Agent 都会获得一个关注位置信息,并基于此位置信息生成一个可视区域,这个可视区域的初始大小设置为原始图像的 1/3,若可视区域超出图像边界则超出边界的区域为无效区域.

### 3.1.2 损失函数曲线

图 11 展示了在 VOC 2007+VOC 2012 联合训练集中 UR-DRQN 模型的损失函数曲线图.从图中可以看出,随着网络训练迭代的深入,模型的综合损失不断下降,最终能够达到收敛的状态,在这个过程中网络的损失呈现波动式的跳跃是因为动作网络学习使用了强化学习的方法,Agent 在学习过程中的经验优劣不一,而这一特点也正是 Agent 学会区分动作效果的优势所在.

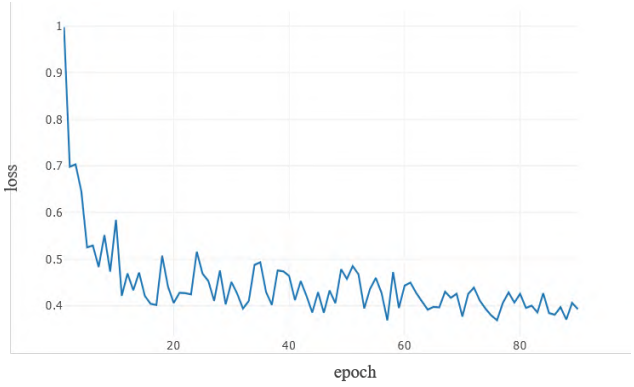


图 11 模型训练损失曲线图

Fig.11 Model training loss diagram

## 3.2 实验结果及分析

### 3.2.1 实验效果图

图 12 至图 15 所示为 Agent 在 VOC 2007 测试集 aero 类别中进行目标定位的过程,图 16 和图 17 所示为 Agent 在 COCO 2014 测试集 aero 类别中进行目标定位的过程.定位过程包括包围框的粗调和精调,红色终止动作之前都是对包围框的粗调,终止动作之后由回归网络对包围框进行精调.其中,绿色包围框内是目标真实标记区域,红色包围框内是上一时间步的可视区域,黄色包围框内是执行相应动作后的可视区域范围,图片的下方标明了 Agent 在特定时间步根据所学到的策略做出的动作选择,“reg”表示结束动作之后的回归精调.

由以下六组实验结果可以看到这样三个阶段：第一阶段聚焦,Agent 将注意力聚焦在感兴趣区域附近,分别由每个示例第一张图片的红色包围框中心表示；第二阶段粗调整,通过调整动作来动态改变可视区域的范围,在若干时间步内就可以较为成功地定位到目标大致位置和范围,分别由每个示例除图示第一张和最后一张的中间部分构成；第三阶段精细调整,由回归网络执行“reg”操作对包围框精调得到精确的位置和范围,分别由每个示例的 reg 表示.

（一） 简单背景下的测试结果

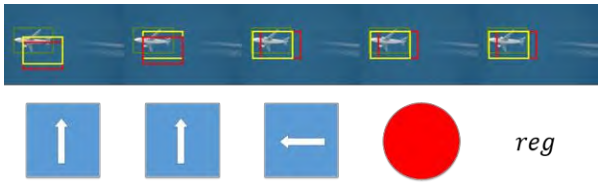


图 12 测试结果示例一

Fig.12 Test result example 1

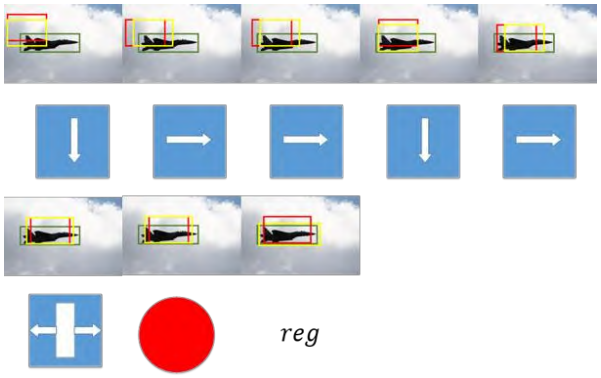


图 13 测试结果示例二

Fig.13 Test result example 2

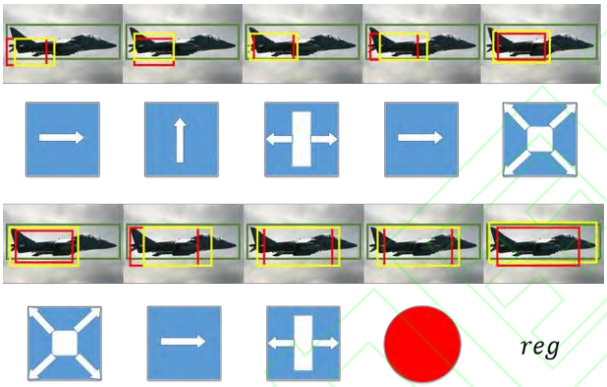


图 14 测试结果示例三

Fig.14 Test result example 3

（二） 复杂背景下的测试结果

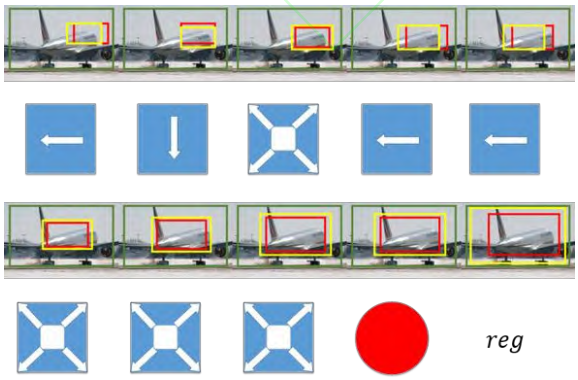


图 15 测试结果示例四

Fig.15 Test result example 4

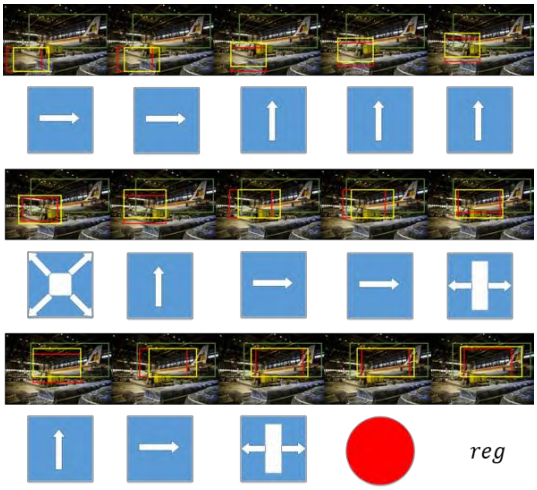


图 16 测试结果示例五

Fig.16 Test result example 5

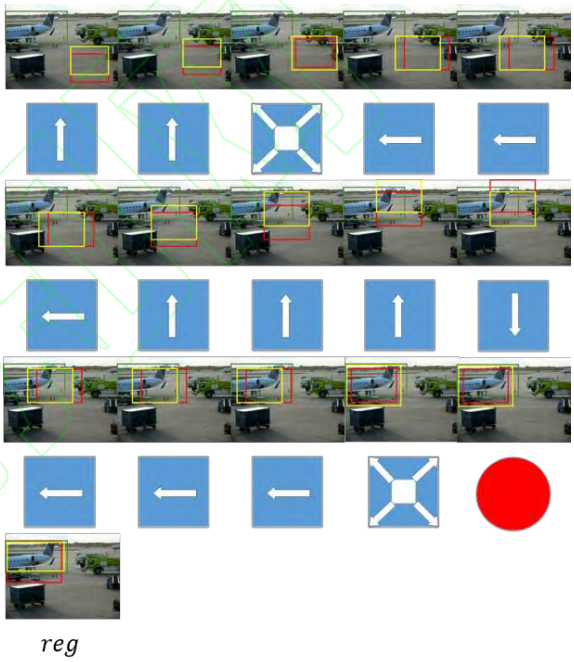


图 17 测试结果示例六

Fig.17 Test result example 6

### 3.2.2 实验结果分析

针对 3.2.1 节所显示的六个 Agent 定位序列,图 18 显示了 Agent 在这六个示例序列定位过程中执行特定动作、转换边界框时,各测试结果示例的 IoU 变化趋势.从图中可以看出,正确的检测过程通常能够通过少数的时间步来实现,这些时间步随着 IoU 的迅速增长而显著减少.图中振荡低于最小可接受阈值(0.6)的点表示目标搜索过程中对 Agent 来说比较困难或容易混淆的阶段,这种情况下 Agent 会根据以往的经验进行多次尝试来改变困局.

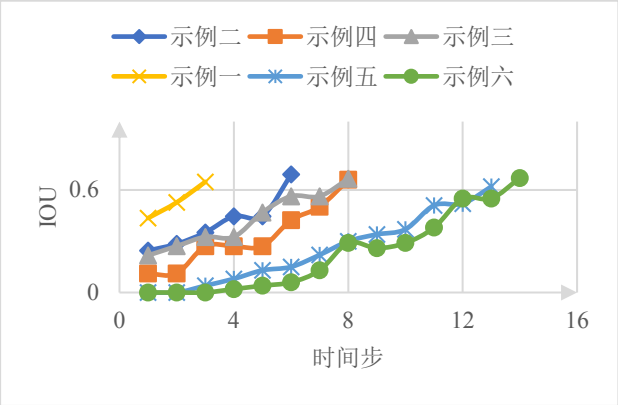


图 18 测试结果示例 IoU 变化趋势示意图

Fig.18 Schematic diagram of variation trend of IOU test result

从结果来看,示例一成功定位到目标所用的时间步较其他示例明显偏少,这是因为示例四中图片背景较其他示例更加纯净,云雾、房屋以及树木等干扰背景因素少,定位目标轮廓清晰可见,因此 Agent 在定位时显得较为容易.示例二与示例三由于背景存在大量云雾, Agent 进行了较多次的尝试,但相较而言示例三中的定位目标比示例二中的定位目标在全图的占比更大,受初始包围框大小和动作调整幅度的影响,示例三的定位步数更多一些.示例四中定位目标背景略显复杂,但主要制约因素同样是定位目标在全图中的占比尺寸较大,受初始包围框大小和动作调整幅度的影响,需要较多步才能定位目标.示例五与示例六中目标定位所使用的时间步最长,从图中可以看出,示例五相较其他示例的背景更为复杂且目标轮廓有遮挡,这些因素对 Agent 具有迷惑性,因而需要更多次的探索和尝试.示例六的背景也稍显复杂且定位目标整体不完整,位置偏向图片边缘,因而在探索中也进行了多次尝试,最终定位准确.

图 19 显示的是,当 IoU 达到 0.6 时经回归器精调边界框后的可视区域范围(浅蓝色阴影表示),可以看出,UR-DRQN 模型在简单背景和复杂背景下均能得到较好的定位效果.

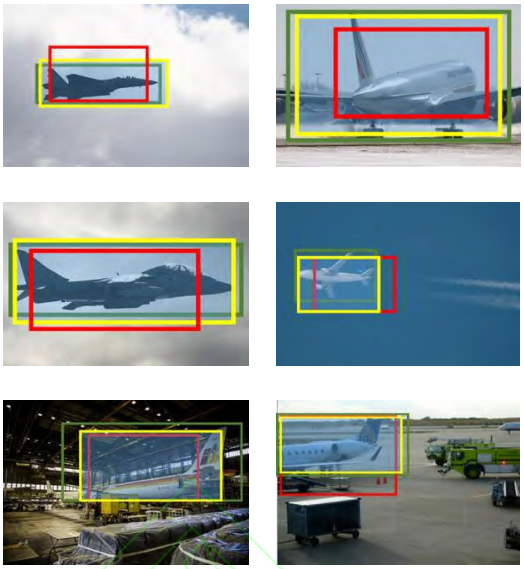

图 19 回归器精调后 IoU 交叠区域示意图

Fig.19 Schematic diagram of IOU overlapping area after fine adjustment of regressor

### 3.3 模型对比及分析

为了更好地展现本文所提出的 UR-DRQN 模型的定位效果,我们与其它三个较有影响的目标定位算法进行了对比实验.本文所进行的模型对比实验均在相同配置环境的实验设备（搭载 Titan X 的 Linux 仿真环境）下进行,并使用相同的训练集（VOC 2007+VOC 2012）和测试集（VOC 2007）进行训练和测试.其中,Fater R-CNN 在此代表传统目标定位的典型算法,Caicedo、Bueno 以及本文所提出的算法是应用深度强化学习方法进行目标定位的算法,各算法在 VOC 2007 测试集上的定位精度表现如表 1 所示,在相同实验设备下的定位耗时如表 2 所示.由于 Caicedo 及 Bueno 未对其算法模型命名,这里使用作者姓名代称其算法.

表 1 不同算法在 VOC 2007 测试集上的定位精度表现(节选部分种类)

Table 1 Positioning accuracy performance of different algorithms on VOC 2007 test set (category of excerpts)

| 种类 / 算法 | aero | bike | bird | boat | bottle | bus | car | cat | mAP |
|---|---|---|---|---|---|---|---|---|---|
| Faster R-CNN | 86.5 | 81.6 | 77.2 | 58.0 | 51.0 | 78.6 | 76.6 | 93.2 | 75.3 |
| Caicedo | 57.9 | 56.7 | 38.4 | 33.0 | 17.5 | 51.1 | 52.7 | 53.0 | 45.0 |
| Bueno | 56.1 | 52.0 | 42.2 | 38.4 | 22.1 | 46.7 | 42.2 | 52.6 | 44.0 |
| UR-DRQN | 59.4 | 58.7 | 44.6 | 36.1 | 28.3 | 55.3 | 48.4 | 52.4 | 47.9 |

表 2 不同算法平均每个 epoch 的定位耗时

Table 2 The average location time of each epoch in different algorithms

| 算法 | Faster R-CNN | Caicedo | Bueno | UR-DRQN |
|---|---|---|---|---|
| 定位耗时（s/epoch） | 372 | 271 | 251 | 219 |

实验结果表明,在应用深度强化学习进行目标定位的方法中,本文所提出的 UR-DRQN 模型在 VOC 2007 测试集上的平均定位精度与 Caicedo 和 Bueno 的方法相近,但耗时较少.与传统目标定位模型 Faster R-CNN 相比,虽然定位精度略低,但检测耗时能够约减 40%左右,在具有一定精度的情况下,速度优势明显.

## 4 结论

为模拟人眼的视觉注意与搜索机制,本文提出了一种基于 RNN 的联合回归深度强化学习目标定位模型 UR-DRQN,该方法通过 RNN 提取历史信息作为历史经验并指导 Agent 在新的观测区域进行搜索；通过回归方法对所搜索到的区域再进行精细调整.实验表明 UR-DRQN 模型可以快速地在若干步内对目标进行定位,并且较好地平衡了定位的速度和精度.该模型可以作为一个视觉注意与搜索模式应用于多数图像视觉搜索场景中.

UR-DRQN 模型在定位速度上仍有一定提升空间,受初始包围框大小和固定的动作调整幅度的影响,定位速度还有望进一步提升,并且当前主要面向单目标搜索与定位.在未来的研究中,定位速度的提升和复杂场景下的多目标搜索与定位是该方法的主要研究方向.

## Reference


1 WANG Ya-Shen, HUANG He-Yan, FENG Chong, ZHOU Qiang. Conceptual Sentence Embeddings Based on Attention Mechanism. *Acta Automatica Sinica*, 2020, **46**(7): 1390-1400.
(王亚珅, 黄河燕, 冯冲, 周强. 基于注意力机制的概念化句嵌入研究. 自动化学报, 2020, **46**(7): 1390-1400)

2 Sherstinsky A. Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network. *Physica D: Nonlinear Phenomena*, 2020, 404: 132306..

3 Sun Chang-Yin, Mu Chao-Xu. Important scientific problems of multi-agent deep reinforcement learning. *Acta Automatica Sinica*, 2020, **46**(7): 1301−1312.
(孙长银, 穆朝絮. 多智能体深度强化学习的若干关键科学问题. 自动化学报, 2020, **46**(7): 1301−1312)

4 Hasselt H, Guez A, Silver D. Deep reinforcement learning with double Q-Learning. **In: Proceedings of** the Thirtieth AAAI conference on Artificial Intelligence. Arizona, USA: AAAI, 2016. 2094-2100.

5 Mnih V, Kavukcuoglu K, Silver D, et al. Playing atari with deep reinforcement learning. arXiv preprint arXiv:1312.5602, 2013.

6 Mnih V, Kavukcuoglu K, Silver D, et al. Human-level control through deep reinforcement learning. *Nature,* 2015, **518**(7540): 529.

7 Rahman M A, Wang Y. Optimizing intersection-over-union in deep neural networks for image segmentation. **In: Proceedings of** the International Symposium on Visual Computing. Springer, Cham, Switzerland, 2016. 234-244.

8 Girshick R, Donahue J, Darrell T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation. **In: Proceedings of** the IEEE conference on Computer Vision and Pattern Recognition. Columbus, Ohio, USA: IEEE, 2014. 580-587.

9 Girshick R. Fast r-cnn. **In: Proceedings of** the IEEE International Conference on Computer Vision. Santiago, Chile, USA: IEEE, 2015. 1440-1448.

10 Ren S, He K, Girshick R, et al. Faster r-cnn: Towards real-time object detection with region proposal networks. **In: Proceedings of** the Advances in Neural Information Processing Systems. Vancouver, Canada: MIT Press. 2015. 91-99.

11 Mnih V, Heess N, Graves A. Recurrent models of visual attention. **In: Proceedings of** the Advances in Neural Information Processing Systems. Vancouver, Canada: MIT Press, 2014. 2204-2212.

12 Caicedo J C, Lazebnik S. Active object localization with deep reinforcement learning. **In: Proceedings of** the IEEE International Conference on Computer Vision. Santiago, Chile, USA: IEEE, 2015. 2488-2496.

13 Bueno M B, Giró-i-Nieto X, Marqués F, et al. Hierarchical object detection with deep reinforcement learning. *Deep Learning for Image Processing Applications*, 2017, **31**(164): 3.

14 Hara K, Liu M Y, Tuzel O, et al. Attentional network for visual object detection. arXiv preprint arXiv:1702.01478, 2017.

15 Shah S M, Borkar V S. Q-learning for Markov decision processes with a satisfiability criterion. *Systems & Control Letters*, 2018. 113: 45-51.

16 Garcia F, Thomas P S. A meta-mdp approach to exploration for lifelong reinforcement learning**. In: Proceedings of** the Advances in Neural Information Processing Systems. Vancouver, Canada: MIT Press, 2019. 5691-5700.

17 Sutton R S, Barto A G. *Reinforcement Learning: An Introduction*.

Canada: MIT Press, 2018.

18 March J G. Exploration and exploitation in organizational learning. *Organization Science*, 1991, **2**(1): 71-87.

19 Bertsekas D P, Bertsekas D P, Bertsekas D P, et al. *Dynamic Programming And Optimal Control*. Belmont, MA: Athena Scientific, 1995.

**姚红革** 西安工业大学计算机科学与工程学院副教授.主要研究方向为机器学习、计算机视觉.

E-mail: yaohongge@xatu.edu.cn

(**YAO Hong-ge** Associate professor at the School of Computer Science and Engineering, Xi 'an University of Technology. His research interest covers machine learning and computer vision.)

**张玮** 西安工业大学计算机科学与工程学院硕士研究生.研究方向为计算机视觉、机器学习.本文通信作者.

E-mail: weivanity@gmail.com

(**ZHANG Wei** Postgraduate at the School of Computer Science and Engineering, Xi 'an University of Technology. His research interest covers machine learning and computer vision. Corresponding author of this paper.)

**杨浩琪** 西安工业大学计算机科学与工程学院硕士研究生.主要研究方向为目标检测、胶囊网络、模型量化.

E-mail: curioyhq@gmail.com

(**YANG Hao-Qi** Postgraduate at the School of Computer Science and Engineering, Xi 'an University of Technology. His research interest covers object detection, capsule network and model quantification.)

**喻钧** 西安工业大学计算机学院教授.主要研究方向为图像处理、模式识别.

E-mail: yujun@xatu.edu.cn

(**YU JUN** Professor at the School of Computer Science and Engineering, Xi 'an University of Technology. Her research interest covers image processing and pattern recognition.)