



(12) 发明专利

(10) 授权公告号 CN 113128432 B

(45) 授权公告日 2022. 09. 06

(21) 申请号 202110449860.3

(22) 申请日 2021.04.25

(65) 同一申请的已公布的文献号
申请公布号 CN 113128432 A

(43) 申请公布日 2021.07.16

(73) 专利权人 四川大学
地址 610064 四川省成都市武侯区一环路
南一段24号

(72) 发明人 孙亚楠 吴杰

(74) 专利代理机构 北京正华智诚专利代理事务
所(普通合伙) 11870
专利代理师 代维凡

(51) Int. Cl.
G06V 20/56 (2022.01)
G06V 10/80 (2022.01)
G06V 10/82 (2022.01)
G06N 3/04 (2006.01)
G06N 3/08 (2006.01)

(56) 对比文件

CN 111328400 A, 2020.06.23
CN 112258486 A, 2021.01.22
CN 112508104 A, 2021.03.16
CN 111325356 A, 2020.06.23
CN 110443364 A, 2019.11.12
CN 112561039 A, 2021.03.26
CN 110689127 A, 2020.01.14
US 2020401899 A1, 2020.12.24
US 2021019599 A1, 2021.01.21
孟子尧 等. 深度神经架构搜索综述. 《计算机研究与发展》. 2021, (第(2021)01期),
张爱军 等. 自动化机器学习中的超参调优方法. 《中国科学: 数学》. 2020, (第(2020)05期),
(续)

审查员 高寒

权利要求书3页 说明书9页 附图2页

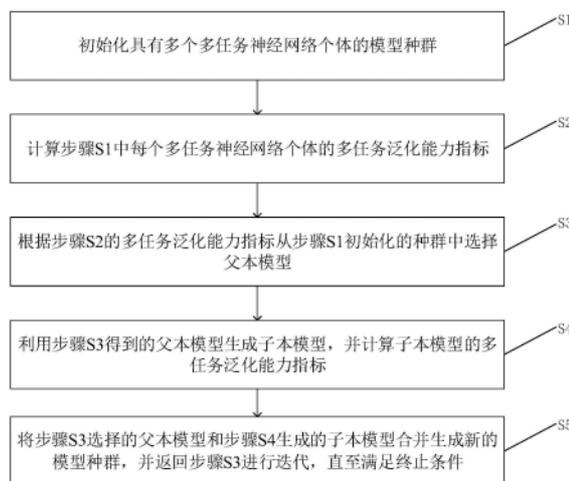
(54) 发明名称

基于演化计算的机器视觉多任务神经网络架构搜索方法

(57) 摘要

本发明公开了一种基于演化计算的机器视觉多任务神经网络架构搜索方法, 首先初始化种群, 对种群中的个体的多任务泛化能力进行评估, 然后通过二进制锦标赛选择算法随机获得两个染色体, 并比较两个染色体多任务泛化表现, 选择表现更好的一个作为父本, 再选择两个父本进行交叉、变异操作, 产生两个子代, 对子代的多任务泛化表现进行评估, 之后合并子代和父代, 根据评估结果进行环境选择, 生成新的种群, 进行新一轮的演化, 直到达到预定的终止条件, 输出多任务泛化能力最好的个体。本发明使用遗传算法来优化多任务网络模型体系结构, 在无需人工参与的情况下, 能够自动搜索到适合于多任务学习的神经网络模型, 提高多任务网络的跨任务

信息融合能力。



CN 113128432 B

[接上页]

(56) 对比文件

Phanomphon Yotchon 等.Hybrid Multi-population Evolution based on Genetic Algorithm and Regularized Evolution for Neural Architecture Search.《2020 17th International Joint Conference on Computer Science and Software Engineering (JCSSE)》.2020,

JingJiang等.Efficient network

architecture search via multiobjective particle swarm optimization based on decomposition.《Neural Networks》.2019,第123卷

Jie Wu, Yanan Sun.Evolving deep parallel neural networks for multi-task learning.《Algorithm and Architecture for Parallel Processing》.2022,

1. 一种基于演化计算的机器视觉多任务神经网络架构搜索方法,其特征在于,包括如下步骤:

S1、初始化执行场景语义分割任务和物体表面法线估计任务的神经网络,利用执行场景语义分割任务的神经网络模型和物体表面法线任务的神经网络模型所提取的图像特征特作为特征融合点,将执行场景分割任务和物体表面法线任务的神经网络模型进行拼接,其拼接方式为,在执行场景语义分割任务和物体表面法线估计任务的神经网络中的任意一种神经网络里,其每一层主干网络随机连接至另一种神经网络任一层主干网络所提取的特征融合点,表示为矩阵 D_a 和 D_b ,

$$D_a \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, D_b \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

其中, D_a 和 D_b 分别为场景语义分割任务和物体表面法线估计任务,为场景语义分割任务和物体表面法线估计任务,两者互为任务A和任务B,矩阵 D_a 表示任务B骨干网络中各层与任务A特征融合点的连接情况,矩阵 D_a 的第i行表示任务A的第i个特征融合点;若位置(i, j)为1,表示存在从任务B的第j层指向任务A的第i个特征融合点的连接,而0表示没有连接;矩阵 D_b 表示任务A的主干网络到任务B特征融合点的连接情况;

形成神经网络模型种群;

S2、计算步骤S1中初始化执行场景语义分割任务和物体表面法线估计任务的神经网络在执行对应图像分析任务时的评价指标,具体方式为:

对场景语义分割任务采用联合平均交集和像素精度评价执行语义分割任务时的性能,并采用Softmax交叉熵作为损失函数;对表面法线估计任务,采用角度距离的均值和中值误差作为执行表面法线任务时的性能,并使用余弦损失表示与预测和真实图像的角度差;

S3、根据步骤S2的评价指标,以设定比例从步骤S1初始化的模型种群中选择父本模型;

S4、利用步骤S3得到的父本模型生成子本模型,并计算子本模型的多任务泛化能力指标;

S5、将步骤S3选择的父本模型和步骤S4生成的子本模型合并生成新的模型种群,并返回步骤S3进行迭代,直至满足终止条件。

2. 根据权利要求1所述的一种基于演化计算的机器视觉多任务神经网络架构搜索方法,其特征在于,所述步骤S1具体包括:

S11、初始化两个多任务神经网络的编码矩阵,并将矩阵中的数据置零;

S12、在0-1之间生成一个随机数,分别遍历矩阵中的每一行,若所遍历的位置生成的随机数大于融合概率阈值,则将两个神经网络进行特征融合连接,其中,该位置的连接点为特征融合点,该行为特征融合边;

S13、重复执行步骤S11-S12直至生成具有多个多任务神经网络个体的模型种群。

3. 根据权利要求2所述的一种基于演化计算的机器视觉多任务神经网络架构搜索方法,其特征在于,所述步骤S2中多任务泛化能力指标的计算过程为:

S21、根据步骤S1中多任务神经网络的不同任务,为每个任务选择量级不同的损失函

数;

S22、根据步骤S21中不同量级定义对应损失函数的权重;

S23、对步骤S21和S22中的损失函数和权重使用加权算法作为对应多任务神经网络的多任务泛化能力指标,表示为:

$$\sum_{i=1}^k w_i L_i;$$

其中, w_i 为第*i*个任务的损失函数, L_i 为第*i*任务的损失函数对应的权重, k 为任务总数。

4.根据权利要求3所述的一种基于演化计算的机器视觉多任务神经网络架构搜索方法,其特征在于,所述步骤S3具体包括:

S31、从步骤S1初始化的模型种群中随机选择两个多任务神经网络个体模型;

S32、比较所选择的两个多任务神经网络个体模型的多任务泛化能力指标,将多任务泛化能力指标更高的个体模型作为一个父本模型;

S33、重复步骤S31-S32直至父本模型数量与步骤S1初始化后的模型种群中多任务神经网络个体数量相同。

5.根据权利要求4所述的一种基于演化计算的机器视觉多任务神经网络架构搜索方法,其特征在于,所述步骤S4中具体包括:

S41、对步骤S3所选择的父本模型进行交叉操作生成其后代模型;

S42、对步骤S41所生成的后代模型进行变异操作生成子本模型;

S43、重复步骤S41-S42直至子本模型的数量达到步骤S3生成的父本模型的数量。

6.根据权利要求5所述的一种基于演化计算的机器视觉多任务神经网络架构搜索方法,其特征在于,所述步骤S41具体为:

S411、随机选择一个特征融合点,并生成一个0-1之间的随机数;

S412、判断该随机数是否大于特征融合点的交叉率;

S413、若是,则对所选择的父本模型中的特征融合点处执行交叉,若不是则返回步骤S411对下一个特征融合点进行判断。

7.根据权利要求6所述的一种基于演化计算的机器视觉多任务神经网络架构搜索方法,其特征在于,所述步骤S42的变异操作具体为:

S421、对没有特征融合边连接的位置生成一个0-1的随机数,若该数大于变异概率阈值,则增加一条特征融合边的连接;

S422、对有特征融合边连接的位置生成一个0-1的随机数,若该数大于变异概率阈值,则删除当前边的连接。

8.根据权利要求7所述的一种基于演化计算的机器视觉多任务神经网络架构搜索方法,其特征在于,所述步骤S5具体包括:

S51、将父本模型和子本模型合并形成下一代种群,并根据多任务泛化能力指标的大小将其中的模型按照降序排列;

S52、利用精英策略在步骤S1形成的下一代种群中按照精英百分比率选择多任务泛化能力指标最优的多个个体;

S53、利用二进制锦标赛选择算法从剩余的个体中选择个体,直至个体数量与步骤S1初始化后的种群模型中的模型数量相同;

S54、合并步骤S52和S53所选择的个体形成新的模型种群,并返回步骤S3进行迭代,直

至满足终止条件。

9. 根据权利要求8所述的一种基于演化计算的机器视觉多任务神经网络架构搜索方法,其特征在於,所述终止条件为:当演化代数达到20代时,演化搜索随即自动停止,得到多任务泛化能力最优的多任务神经网络模型。

基于演化计算的机器视觉多任务神经网络架构搜索方法

技术领域

[0001] 本发明涉及神经网络领域,具体涉及一种基于演化计算的机器视觉多任务神经网络架构搜索方法。

背景技术

[0002] 目前,在针对单机器视觉任务,比如,图像分类、目标识别方面的相关技术,无论是传统的非神经网络方法还是如今基于卷积神经网络的深度学习方法都已经发展的较为成熟。然而,现实场景中往往需要同时处理多个任务。比如:无人驾驶,需要同时进行道路目标识别、目标深度估计等,以保障无人驾驶汽车能够安全的运行。多任务学习旨在充分利用多个相关联任务之间的信息,提高模型的泛化能力和单个任务的表现。

[0003] 多任务学习技术从传统方法不断发展到如今的基于卷积神经网络的深度学习方法。传统方法主要分为两类:一类是通过正则化的方法,比如L1、L2正则化,来构建任务之间的稀疏关系。该类方法预先假设多任务学习只需要利用到少量的任务之间信息,但是若任务的相关联度不高会导致模型的整体表现较差;另一类方法是通过聚类的方法,比如贝叶斯优化、支持向量机,来建模任务之间的关系,将相似的任务聚类在一起,在相似任务之间学习共享的信息。然而以上的传统方法都是假设数据和目标是线性关系,并不适用于如今复杂的实际应用环境。

[0004] 近年来,得益于深度学习的快速发展,其灵活的层次化表示结构以及强大的特征提取能力,受到多任务学习研究人员的广泛关注。当前基于神经网络的方法主要分为两种:硬参数共享和软参数共享,如图2所示。硬参数共享,如图2(a),是多个任务共享网络的特征提取层,然后在网络的末端设计各自的输出层。该类方法虽然可以降低任务的过拟合风险,但是由于多个任务共享同一个特征提取层,所以该类方法对任务的相关联程度也有很高的要求。如果任务之间的关联度较为松散,将会极大影响模型的最终表现。而在多任务的实际应用场景中,任务的相关联程度往往不是很高。

[0005] 因此,近年来最新的多任务学习模型主要是基于软参数共享的方法。在软参数共享(如图2(b))中,每个任务都有一个自己特定的主干网络,不同的主干网络通过共享连接学习有用的信息(特征),以促进所有任务的表现。软参数共享重点关注于怎样融合特征,以及在模型的哪个位置融合特征。目前大多数方法都是关注如何进行特征的融合,最新的方法有:1. 十字交叉网络(Cross Stitch Network)提出的线性融合方法,线性融合单元将不同的任务的特征进行加权线性组合。2. 神经判别降维网络(Neural Discriminative Dimensionality Reduction, NDDR-Network)将不同任务的特征在通道维度上进行拼接后,通过1x1卷积操作进行特征的学习,再降维以满足下一层的输入要求。

[0006] 虽然以上基于软参数共享的方法,提高了多任务学习模型的性能,但目前手工设计的网络在特征融合位置的选择上往往带有很强的主观性。比如最新的NDDR网络将特征融合操作简单的放在了VGG-16的每个卷积层之后,以融合相同层级特征。这种方式可能会导致融合一些无用的信息,从而影响到模型的性能。

发明内容

[0007] 针对现有技术中的上述不足,本发明提供了一种基于演化计算的机器视觉多任务神经网络架构搜索方法。

[0008] 为了达到上述发明目的,本发明采用的技术方案为:

[0009] 一种基于演化计算的机器视觉多任务神经网络架构搜索方法,包括如下步骤:

[0010] S1、初始化具有多个多任务神经网络个体的模型种群;

[0011] S2、计算步骤S1中每个多任务神经网络个体的多任务泛化能力指标;

[0012] S3、根据步骤S2的多任务泛化能力指标从步骤S1初始化的种群中选择父本模型;

[0013] S4、利用步骤S3得到的父本模型生成子本模型,并计算子本模型的多任务泛化能力指标;

[0014] S5、将步骤S3选择的父本模型和步骤S4生成的子本模型合并生成新的模型种群,并返回步骤S3进行迭代,直至满足终止条件。

[0015] 上述方案的有益效果是,采用演化计算中的遗传算法来对模型体系架构进行优化,通过设计的一系列演化算子,不断演化找到一个多任务泛化能力强的神经网络,最大程度的减少人的干预,设计了基于矩阵编码的交叉、变异算子,能够在搜索过程中,增强全局搜索能力,避免陷入局部最优,以此找到多任务泛化能力优异的网络架构,在环境选择环节,先进行精英选择,种群余下的个体进行两组适应度的比较然后选择,既保证了精英性又具有多样性。

[0016] 进一步的,所述步骤S1具体包括:

[0017] S11、初始化两个大小为多任务神经网络矩阵,并将矩阵中的数据置零;

[0018] S12、在0-1之间生成一个随机数,分别遍历矩阵中的每一行,若所遍历的位置生成的随机数小于融合概率阈值,则将该处位置置为1,将两个神经网络矩阵进行特征融合连接,其中,该连接点为特征融合点,该行为特征融合边;

[0019] S13、重复执行步骤S11-S12直至生成具有多个多任务神经网络个体的模型种群。

[0020] 上述进一步方案的有益效果是,通过对种群中多任务神经网络的特征融合进行随机初始化,保证了神经网络结构间的差异性,为后续探索具备高泛化能力的多任务神经网络提供了基本搜索空间。

[0021] 进一步的,所述步骤S2中多任务泛化能力指标的计算过程为:

[0022] S21、根据步骤S1中多任务神经网络的不同任务,为每个任务选择量级不同的损失函数;

[0023] S22、根据步骤S21中不同量级定义对应损失函数的权重;

[0024] S23、对步骤S21和S22中的损失函数和权重使用加权算法作为对应多任务神经网络的多任务泛化能力指标,表示为:

$$[0025] \quad \sum_{i=1}^k w_i L_i;$$

[0026] 其中, w_i 为第*i*个任务的损失函数, L_i 为第*i*任务的损失函数对应的权重, k 为任务总数。

[0027] 上述进一步方案的有益效果是,解决了不同任务的评价指标多、量级差异大,难以选择某单一指标作为判断模型性能的问题,从而实现了多任务泛化能力评价指标的统一。

[0028] 进一步的,所述步骤S3具体包括:

- [0029] S31、根据步骤S2的计算结果,随机选择两个多任务神经网络个体模型;
- [0030] S32、比较所选择的两个多任务神经网络个体模型的多任务泛化能力指标,将多任务泛化能力指标更高的个体模型作为一个父本模型;
- [0031] S33、重复步骤S31-S32直至父本模型数量与步骤S1初始化后的模型种群中多任务神经网络个体数量相同。
- [0032] 上述进一步方案的有益效果是,选择种群中表现相对较好的个体作为父本,有利于后续演化出多任务泛化能力更高的神经网络。
- [0033] 进一步的,所述步骤S4中具体包括:
- [0034] S41、对步骤S3所选择的父本模型进行交叉操作生成其后代模型;
- [0035] S42、对步骤S41所生成的后代模型进行变异操作生成子本模型;
- [0036] S43、重复步骤S41-S42直至子本模型的数量达到步骤S3生成的父本模型的数量。
- [0037] 上述进一步方案的有益效果是,通过交叉、变异等操作,能够在搜索过程中,保留泛化能力高的多任务网络架构,同时避免陷入某个局部最优解,以此找到多任务泛化能力优异的网络架构。
- [0038] 进一步的,所述步骤S41具体为:
- [0039] S411、随机选择一个特征融合点,并生成一个0-1之间的随机数;
- [0040] S412、判断该随机数是否大于特征融合点的交叉率;
- [0041] S413、若是,则对所选择的父本模型中的特征融合点执行交叉,若不是则返回步骤S411对下一个特征融合点进行判断。
- [0042] 上述进一步方案的有益效果是,通过交叉操作能够将父本中优异的神经网络结构继承到子代个体,组建具有良好多任务泛化能力的神经网络。
- [0043] 进一步的,所述步骤S42的变异具体为:
- [0044] S421、对没有特征融合边连接的位置生成一个0-1的随机数,若该数大于变异概率阈值,则增加一条特征融合边的连接;
- [0045] S422、对有特征融合边连接的位置生成一个0-1的随机数,若该数大于变异概率阈值,则删除变异概率阈值,则删除当前边的连接。
- [0046] 上述进一步方案的有益效果是,通过变异操作能够防止神经网络陷入局部最优解,探索新颖的神经网络结构,保证演化过程中神经网络的多样性,进而增强算法的全局搜索能力。
- [0047] 进一步的,所述步骤S5具体包括:
- [0048] S51、将父本模型和子本模型合并形成下一代种群,并根据多任务泛化能力指标的大小步其中的模型按照降序排列,;
- [0049] S52、利用精英策略在步骤S1形成的下一代种群中按照精英百分比率选择多任务泛化能力指标最优的多个个体;
- [0050] S53、利用二进制锦标赛选择算法从剩余的个体中选择个体,直至个体数量与步骤S1初始化后的种群模型中的模型数量相同;
- [0051] S54、合并步骤S52和S53所选择的个体形成新的模型种群,并返回步骤S3进行迭代,直至满足终止条件。
- [0052] 上述进一步方案的有益效果是,既通过精英策略保证当前性能最优的神经网络直

接进入下一代,同时又保证了整个种群神经网络结构组成的多样性,有利于后续演化过程中,搜索到新颖的多任务神经网络。

[0053] 进一步的,所述终止条件为:当演化代数达到20代时,演化搜索随即自动停止,得到多任务泛化能力最优的多任务神经网络模型。

[0054] 上述进一步方案的有益效果是,由于搜索空间巨大,而演化神经架构搜索往往会消耗较大的计算资源,因此在实践中很难实现对所有可能解的搜索,通过预先设定最大演化代数,实现在特定计算资源情况下的网络架构搜索。

附图说明

[0055] 图1为本发明基于演化计算的机器视觉多任务神经网络架构搜索方法流程图。

[0056] 图2为现有技术基于神经网络的共享连接方式,其中a为硬参数共享,b为软参数共享。

[0057] 图3为本发明实施例多任务神经网络模型个体示意图。

[0058] 图4为本发明实施例父本模型交叉生成子代模型的示意图。

具体实施方式

[0059] 下面对本发明的具体实施方式进行描述,以便于本技术领域的技术人员理解本发明,但应该清楚,本发明不限于具体实施方式的范围,对本技术领域的普通技术人员来讲,只要各种变化在所附的权利要求限定和确定的本发明的精神和范围内,这些变化是显而易见的,一切利用本发明构思的发明创造均在保护之列。

[0060] 一种基于演化计算的机器视觉多任务神经网络架构搜索方法,如图1所示,包括如下步骤:

[0061] S1、初始化具有多个多任务神经网络个体的模型种群;

[0062] 具体而言,在实施例里,为了更好地表示不同骨干网络之间特征融合的连接关系,采用矩阵基因编码策略来对提出的算法中的个体进行编码。如图3所示,以两个任务(任务A、任务B)为例。矩形分别表示任务A和任务B的每一层,而虚线箭头表示任务A和B之间的连接,圆圈表示特征融合点。种群中的一个个体(称为D)可以由矩阵 D_a 和 D_b 编码。具体来说,矩阵 D_a 表示任务B骨干网络中各层与任务A特征融合点的连接情况。矩阵 D_a 的第i行表示任务A的第i个特征融合点。如果位置(i,j)为1,表示存在从任务B的第j层指向任务A的第i个特征融合点的连接,而0表示没有连接。同理,矩阵 D_b 表示任务A的主干网络到任务B特征融合点的连接情况。在此示例中,矩阵 D_a 和矩阵 D_b 由公式(1)表示,

$$[0063] \quad D_a \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, D_b \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} \quad (1)$$

[0064] 基于上述个体表示,初始化种群的步骤为:

[0065] S11、初始化两个大小为多任务神经网络矩阵,并将矩阵中的数据置零;

[0066] S12、在0-1之间生成一个随机数,分别遍历矩阵中的每一行,若所遍历的位置生成的随机数小于融合概率阈值,则将该处位置置为1,将两个神经网络矩阵进行特征融合连接,其中,该连接点为特征融合点,该行为特征融合边;

[0067] S13、重复执行步骤S11-S12直至生成具有多个多任务神经网络个体的模型种群;

[0068] 基于矩阵基因编码策略,基于矩阵基因编码策略,第一个亲本个体的融合情况可以用两个矩阵表示,即 P_{1A} 和 P_{1B} ,如公式2所示。

$$[0069] \quad P_{1A} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, P_{1B} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2)$$

[0070] 同样,第二个父母个体的矩阵 P_{2A} 和 P_{2B} 如公式3所示。

$$[0071] \quad P_{2A} = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, P_{2B} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 1 \end{bmatrix} \quad (3)$$

[0072] 特别的,在本实施例里,为了防止网络中产生有向闭环结合,我们在初始化个体的过程中将 j 严格限制为不大于 i 。

[0073] S2、计算步骤S1中每个多任务神经网络个体的多任务泛化能力指标;

[0074] 个体多任务泛化能力指标计算即是计算个体的适应度,适应度是为父本选择和环境选择提供参考指标。在多任务学习中,每个任务都有其相应的评价指标,若选择其中一个任务的评价指标作为适应度,会导致一个任务表现的好,而其他的任务表现差。为了获得一个能够提升每个任务性能(即多任务泛化能力强)的模型,本方案选择验证数据集上的模型加权损失作为适应度。该过程首先是在训练数据集上对个体(模型)的参数进行学习,然后在验证数据集上对训练好的模型进行验证,具体的计算方式为:

[0075] S21、根据步骤S1中多任务神经网络的不同任务,为每个任务选择量级不同的损失函数;

[0076] S22、根据步骤S21中不同量级定义对应损失函数的权重;

[0077] S23、对步骤S21和S22中的损失函数和权重使用加权算法作为对应多任务神经网络的多任务泛化能力指标,表示为:

$$[0078] \quad \sum_{i=1}^k w_i L_i;$$

[0079] 其中, w_i 为第 i 个任务的损失函数, L_i 为第 i 任务的损失函数对应的权重, k 为任务总数。

[0080] S3、根据步骤S2的多任务泛化能力指标从步骤S1初始化的种群中选择父本模型;

[0081] 在本实施例里,具体包括:

[0082] S31、根据步骤S2的计算结果,随机选择两个多任务神经网络个体模型;

[0083] S32、比较所选择的两个多任务神经网络个体模型的多任务泛化能力指标,将多任务泛化能力指标更高的个体模型作为一个父本模型;

[0084] S33、重复步骤S31-S32直至父本模型数量与步骤S1初始化后的模型种群中多任务神经网络个体数量相同。

[0085] 从种群中随机选择两个个体,随机选择保证了父本的多样性,比较两个个体的适应度大小,适应度大的个体将作为一个父本,迭代执行父本选择,最终父本的总数达到种群

大小等于N。

[0086] S4、利用步骤S3得到的父本模型生成子本模型,并对计算子本模型的多任务泛化能力指标;

[0087] 在实施例里,在初始化时,为了更灵活的表示多任务网络中各个主干网络的对应特征融合连接关系,采用了矩阵编码策略。为了保留优秀父本的结构,使用对应主干网络连接相互交叉的方法。图4中给出了交叉操作的示例,具体包括如下步骤:

[0088] S41、对步骤S3所选择的父本模型进行交叉操作生成其后代模型;

[0089] 具体的交叉操作为:

[0090] S411、随机选择一个特征融合点,并生成一个0-1之间的随机数;

[0091] S412、判断该随机数是否大于特征融合点的交叉率;

[0092] S413、若是,则对所选择的父本模型中的特征融合点执行交叉,若不是则返回步骤S411对下一个特征融合点进行判断。

[0093] 以图3为例,对于任务A,父本1的第一、二个特征融合点和父本2的第一、二特征融合点彼此交叉(即 P_{1A} 中的第一、二行和 P_{2A} 中的第一、二行对应交换),对于任务B,父本1的第三个特征融合点和父本2的第三个特征融合点彼此进行交叉(即 P_{1B} 和 P_{2B} 中的第三行互换)。交叉后,两个对应的后代个体可以表示为 O_{1A}, O_{1B}, O_{2A} 和 O_{2B} ,如公式4和5所示,

$$[0094] \quad O_{1A} = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix}, O_{1B} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix} \quad (4)$$

$$[0095] \quad O_{2A} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, O_{2B} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (5)$$

[0096] 这样两个染色体实现了交叉操作。N个父本进行交叉产生N个子代,合并后种群数目变为2N。

[0097] S42、对步骤S321所生成的后代模型进行变异操作生成子本模型;

[0098] 变异操作可以在染色体上的任意位置发生,染色体上一个或多个位置可以同时发生变异。对于突变操作有增加、删除一条连接边,每个操作概率相同。在没有特征融合边的位置随机生成一个在0到1之间的数,若大于0.5则增加一条连接边;在有特征融合边的位置同样随机生成一个在0到1之间的数,若小于0.5则删除当前连接边。这样就完成对子代的变异操作。

[0099] S43、重复步骤S41-S42直至子本模型的数量达到步骤S3生成的父本模型的数量。

[0100] 经过交叉和变异生成了子代种群之后,按照步骤S2中的方法再对子代种群中的每个个体进行多任务泛化能力指标的计算。

[0101] S5、将步骤S3选择的父本模型和步骤S4生成的子本模型合并生成新的模型种群,并返回步骤S3进行迭代,直至满足终止条件。

[0102] 具体而言,在本实施例里,包括如下:

[0103] S51、将父本模型和子本模型合并形成下一代种群,并根据多任务泛化能力指标的大小步其中的模型按照降序排列,;

[0104] S52、利用精英策略在步骤S1形成的下一代种群中按照精英百分比率选择多任务

泛化能力指标最优的多个个体；

[0105] S53、利用二进制锦标赛选择算法从剩余的个体中选择多任务泛化能力指标高于设定阈值的个体，直至个体数量与步骤S1初始化后的种群模型中的模型数量相同；

[0106] S54、合并步骤S52和S53所选择的个体形成新的模型种群。

[0107] 环境选择过程中需要考虑精英性和多样性。在交叉之后，种群由父本和子代组成，变为原来的二倍。首先根据精英策略，选出一部分多任务泛化能力最佳的个体，然后再次利用二进制锦标赛选择算法从剩余的个体中选择多任务泛化能力较好的个体。最后合并两次选择的个体总共N个，以此组成下一代的种群。

[0108] 经过上述方式形成了新的模型种群，在本实施例里，以新形成的模型种群为基础，返回步骤S3进行迭代演化，当演化代数达到20代时，演化搜索随即自动停止，得到多任务泛化能力最优的多任务神经网络模型。

[0109] 实验验证

[0110] 为了验证基于演化神经网络的多任务神经网络的泛化能力，我们在机器视觉多任务学习中广泛应用的数据集NYUD-v2上进行场景语义分割和物体表面法线估计两个任务，通过演化计算自动选择出的网络模型与现有最新手工精心设计和自动化设计的多任务模型进行比较。

[0111] A. 数据集构建：

[0112] 为了验证演化出的多任务神经网络的多任务泛化能力，本方案选取了多任务机器视觉领域广泛应用的数据集NYUD-v2。NYUD-v2数据集上主要涉及室内场景，其中包含795个训练图像和654个测试图像，用于表面法线估计，语义分割等任务。

[0113] B. 任务主干网络：

[0114] 使用Deeplab-VGG-16作为任务的主干网络。Deeplab-VGG-16基于VGG-16网络，是专门为机器视觉像素级任务（如语义分割）设计的。Deeplab-VGG-16提出了空洞卷积和条件随机场。在本次实验中，仅采用空洞卷积来提高处理像素级任务的性能。具体而言，在Deeplab-VGG-16中，在最后三个卷积层中使用了dilatation参数，并修改了最后池化层中的stride以实现空洞卷积。

[0115] C. 任务评价指标

[0116] 对于语义分割，采用联合平均交集（mIoU）和像素精度（PAcc）两个指标来评估语义分割的性能。对于表面法线估计，采用角度距离的均值（Mean）和中值（Median）误差作为评价指标。此外，我们还测量了与标签图片在 11.25° 角度内（Within）的像素百分比。

[0117] D. 损失函数选取

[0118] 本实验中的两个任务语义分割、表面法线估计，分别选取不同的损失函数。对于语义分割选择Softmax交叉熵作为损失函数，而对于表面法线估计，我们使用余弦损失，以表示与预测和真实图像的角度差。由于法线表面估计的损失值比语义分割的损失小得多，因此在本方案中，我们以20:1的比例对它们进行加权作为整个模型的损失以平衡任务之间不平衡。

[0119] E. 算法参数设置

[0120] 进化阶段的所有参数设置均遵循遗传算法的惯例。具体来说，种群总数和演化代数均设置为20。交叉和突变的概率分别指定为0.9和0.2。在环境选择中，根据帕累托原理将

精英率指定为20%。为了选择在两个任务上都能达到最佳性能的多任务网络架构,在验证数据集上选择加权损失作为适应度,即语义分割和表面法线估计的损失比为20:1。

[0121] 对于经过演化过程后选出的最优多任务神经网络模型。我们使用momentum为0.9, weight decay为0.00025的随机梯度下降 (stochastic gradient descent,SGD) 训练模型。初始学习速率 (learning rare) 设置为0.0005,Deeplab-VGG-16主干网络使用power为0.9的polylearning rate。在NYUDv2数据集上,batch size设置为3,并且对演化出的最佳模型进行20,000次迭代以进行最终训练。为了公平地比较,所有比较的算法都使用相同的训练参数设置。

[0122] 在多任务学习领域中,基于软参数共享的深度学习方法因其强大的特征提取能力和灵活的层次化表示形式,成为现在主要的研究方法。但是人工搭建一个有很好效果的多任务学习模型很困难。演化神经网络可以通过一系列的演化操作在初始网络基础上找到有更好效果的网络模型,减少对人工经验的依赖。将演化神经网络应用于多任务学习模型的设计,既能解决人工设计的耗时、经验不足等问题,又能反过来启发人工设计模型架构。

[0123] 结果比较表1所示:

[0124] 表1

	表面法线估计			语义分割		算法类别
	Errors (越小越好)		Within (越大越好)	(%) (越大越好)		
	Mean	Median	11.25°	mIou	PAcc	
Single	17.04	14.72	36.59	34.4	64.4	手工精心设计
Shared	16.46	13.07	44.05	34.6	64.6	
Cross Stitch Network	16.33	12.91	44.47	34.7	64.9	
NDDR-CNN	16.21	12.64	45.46	35.5	65.6	
MTL-NAS	15.76	12.32	46.37	34.8	65.1	自动化搜索
EVO-MTL (ours)	15.92	12.31	46.47	35.7	66.0	

[0125] 其中,本方案的方法为(EVO-MTL),对比算法为单任务模型(Single)、硬参数多任务共享模型(Shared)、最新的手工多任务学习模型(Cross Stitch Network和NDDR-CNN)、最新的多任务自动搜索模型(MTL-NAS)。

[0126] 对于手工精心设计的算法比较,就所有五个指标而言,所提出的算法都超过了所有手工设计的方法,包括最新的多任务方法,即,NDDR-CNN和Cross Stitch Network。结果表明,本方案的方法(EVO-MTL)在泛化能力和提高各项任务性能方面具有卓越的表现。具体来说,在语义分割任务上,提出的EVO-MTL算法在mIou和PAcc指标方面达到最高,并且是唯一在PAcc达到66的算法。类似地,在表面法线估计中,EVO-MTL在Mean,Median指标中也达到了最低值。即分别为15.92和12.31,且在Within 11.25°指标中达到最大值(即46.47)。这表明EVO-MTL成功从两个不同的任务中学习到了有用信息。

[0127] 对于目前最新的多任务自动学习算法(MTL-NAS)比较,很明显,MTL-NAS搜索的网络架构存在任务不平衡的问题。就表面估计任务而言,这三个指标要优于其他手工设计的模型,但是在语义分割任务中,这两个指标均在分割方面比手动设计的NDDR-CNN更差,这表明MTL-NAS搜索的体系结构偏向于语义分割任务。相反,我们所提出的算法(EVO-MTL)搜索的模型架构在平衡不同任务方面具有更好的性能,并且所有指标均优于手动设计的模型。

[0128] 本发明是参照根据本发明实施例的方法、设备(系统)、和计算机程序产品的流程图和/或方框图来描述的。应理解可由计算机程序指令实现流程图和/或方框图中的每一流

程和/或方框、以及流程图和/或方框图中的流程和/或方框的结合。可提供这些计算机程序指令到通用计算机、专用计算机、嵌入式处理机或其他可编程数据处理设备的处理器以产生一个机器,使得通过计算机或其他可编程数据处理设备的处理器执行的指令产生用于实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能的装置。

[0130] 这些计算机程序指令也可存储在能引导计算机或其他可编程数据处理设备以特定方式工作的计算机可读存储器中,使得存储在该计算机可读存储器中的指令产生包括指令装置的制造品,该指令装置实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能。

[0131] 这些计算机程序指令也可装载到计算机或其他可编程数据处理设备上,使得在计算机或其他可编程设备上执行一系列操作步骤以产生计算机实现的处理,从而在计算机或其他可编程设备上执行的指令提供用于实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能的步骤。

[0132] 本发明中应用了具体实施例对本发明的原理及实施方式进行了阐述,以上实施例的说明只是用于帮助理解本发明的方法及其核心思想;同时,对于本领域的一般技术人员,依据本发明的思想,在具体实施方式及应用范围上均会有改变之处,综上所述,本说明书内容不应理解为对本发明的限制。

[0133] 本领域的普通技术人员将会意识到,这里所述的实施例是为了帮助读者理解本发明的原理,应被理解为本发明的保护范围并不局限于这样的特别陈述和实施例。本领域的普通技术人员可以根据本发明公开的这些技术启示做出各种不脱离本发明实质的其它各种具体变形和组合,这些变形和组合仍然在本发明的保护范围内。

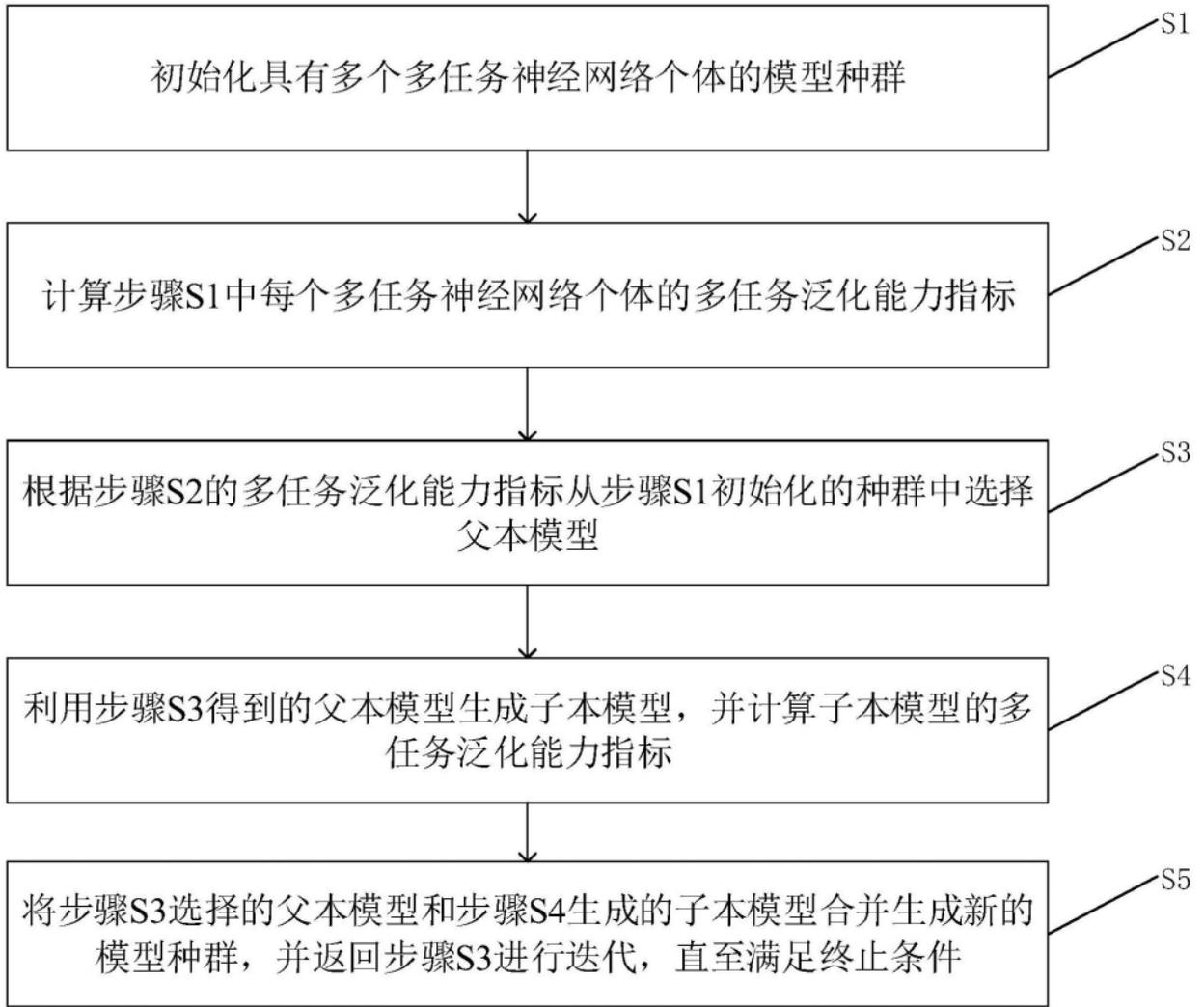


图1

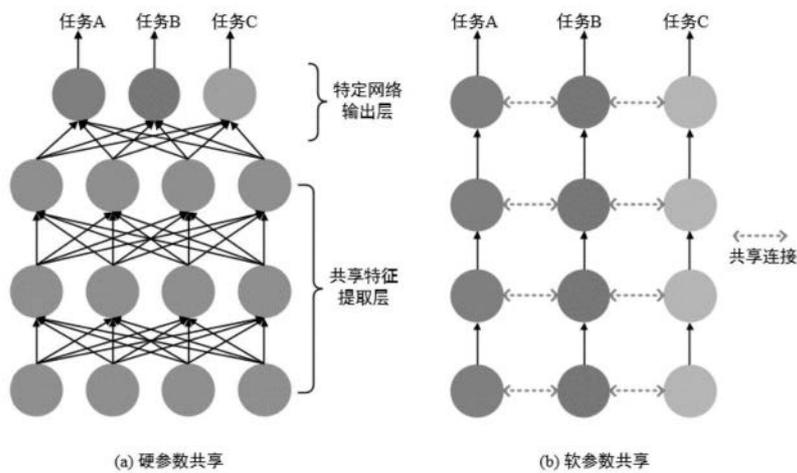


图2

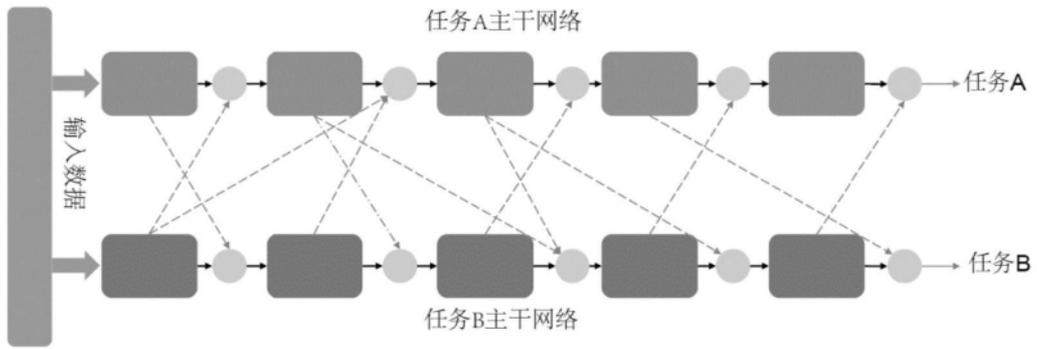


图3

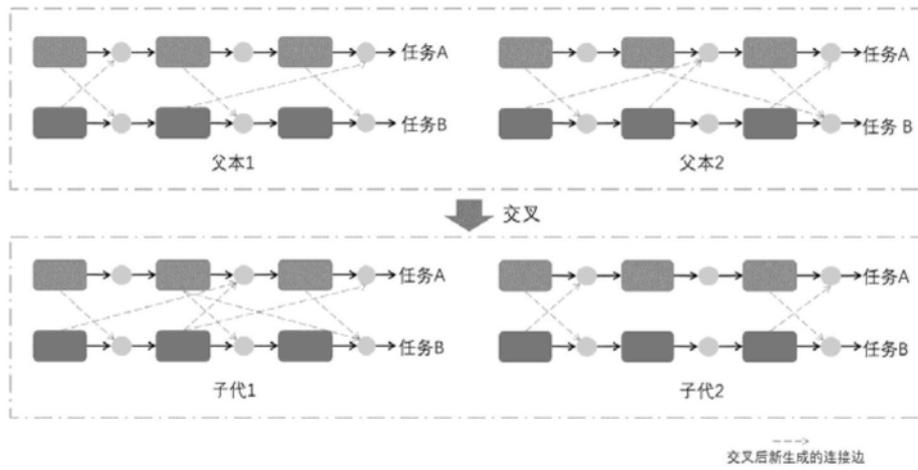


图4