使用增强学习训练多焦点聚焦模型

刘 畅 [1] 刘勤让 [1]

摘 要 聚焦模型 (Attention model, AM) 将计算资源集中于输入数据特定区域, 相比卷积神经网络, AM 具有参数少、计算量独立输入和高噪声下正确率较高等优点. 相对于输入图像和识别目标, 聚焦区域通常较小; 如果聚焦区域过小, 就会导致过多的迭代次数, 降低了效率, 也难以在同一输入中寻找多个目标. 因此本文提出多焦点聚焦模型, 同时对多处并行聚焦. 使用增强学习 (Reinforce learning, RL) 进行训练, 将所有焦点的行为统一评分训练. 与单焦点聚焦模型相比, 训练速度和识别速度提高了 25 %. 同时本模型具有较高的通用性.

关键词 深度学习, 聚焦模型, 增强学习, 多焦点

引用格式 刘畅, 刘勤让. 使用增强学习训练多焦点聚焦模型. 自动化学报, 2017, **43**(9): 1563−1570

DOI 10.16383/j.aas.2017.c160643

Using Reinforce Learning to Train Multi-attention Model

LIU Chang[1] LIU Qin-Rang[1]

Abstract Attention model (AM) concentrates computing resources on specific areas of the input data. Compared with the convolutional neural network, AM has many advantages: fewer parameters, the amount of computation being independent of the input, higher tolerance for noise input, etc. Generally, the focused area is smaller than the input image and target. However, if the focused area is too small, it will lead to more iterations and a low efficiency; besides, it is difficult to recognize multiple targets in the same input. Therefore, this paper proposes a multi-focus model. However, if on multiple focuses in parallel. This model uses reinforce learning (RL) to train, and scores the behaviors of all focuses uniformly during training. Compared with the single focus model, both the training and recognition speeds are improved by 25 %. At the same time, the model has good generality.

Key words Deep learning, attention model (AM), reinforce learning (RL), multi-attention

Citation Liu Chang, Liu Qin-Rang. Using reinforce learning to train multi-attention model. *Acta Automatica Sinica*, 2017, **43**(9): 1563−1570

深度神经网络掀起了学界和工业界的热潮, 使得人工智能达到了前所未有的高度. 例如物体识别、自然图像识别、语音识别、静态机器翻译、太空游戏和围棋游戏[1−2]. 这些成就往往伴随着大量的训练和运行时间. 尽管采用各种降参手段, 一个大型卷积神经网络 (Convolutional neural networks, CNN) 常常在多 GPU 的机器上训练多天[3]. 在一些研究中, 单 GPU 处理单张图片就需要多秒[4−5]. 这种普遍的情况原因之一, 是大量的研究基于经典的滑动窗口加分类器的模型[6−7]. 如果对整幅图像进行卷积, 计算量随着像素数量线性增长, 所以这种模型是计算昂贵的[8].

聚焦模型 (Attention model, AM) 是一种循环神经网络 (Recurrent neural network, RNN), 是当前新兴的深度学习模型. Attention 是指神经网络在执行任务时, 把焦点 (即计算资源), 集中于输入数据中的特定部分. 因此可以让神经网络每一步从更大的输入中获取信息. AM 源自人类视觉系统[9]. 生物解剖表明: 人类视网膜中央视锥细胞较多较密, 而边缘稀疏. 人类视觉总是集中于视觉中心, 并不断移动[8]. 关于 AM 已经有了广泛的应用研究. 比如视频中的运动识别跟踪. 文献 [10] 可以大致定位目标区域, 而只使用了 7 层神经网络. 文献 [11] 通过对特定特征进行处理, 实现了迭代纠错, 提高了鲁棒性和高噪声的分辨率. 文献 [12−13] 根据不同的描述语言, 在图像中寻找相关的区域. 基于这种视觉模型的相关研究, 国内也有很多应用, 文献 [14] 针对网络中受丢包损伤的视频提出了一种基于视觉注意力变化的全参考客观质量评估方法. 文献 [15] 面向运动目标监测, 构建了一种基于粒子滤波的视觉聚焦模

收稿日期 2016-09-08 录用日期 2017-03-21
Manuscript received September 8, 2016; accepted March 21, 2017
国家高技术研究发展计划 (863 计划) (2014AA01A), 国家自然科学基金 (61572520) 资助
Supported by National High Technology Research and Development Program of China (863 Program) (2014AA01A) and National Natural Science Foundation of China (61572520)
本文责任编委 袁勇
Recommended by Associate Editor YUAN Yong
1. 国家数字交换系统工程技术研究中心 郑州 450002
1. China National Digital Switching System Engineering and Technological Research and Development Center, Zhengzhou 450002

型. 文献 [16] 使用 AM 进行了文本分类问题的探究.

基于 Volodymyr 的 Recurrent models of visual attention (出于翻译习惯缩写为 RAM, recurrent attention model)[8], 本文设计了一种多焦点聚焦模型. RAM 是一种高效灵活的循环神经网络, 使用输入图像中的一小部分作为输入, 使用内部状态选择下一个焦点的位置, 根据所有获得的信息推断目标. RAM 有多种优点, 所有参数数量和计算量都独立于输入图像大小, 可以自动忽视一些复杂的噪声干扰, 具有较高的通用性可以适用于各种视觉任务. RAM 也有一些限制, 每次仅从图像中获取一个局部, 如果目标种类较多, 识别目标较大, 需要循环更多的次数才能完成识别. 在输入图像过大时, 焦点不理想而采样在空白的概率也会增大, 降低效率. 本文在 RAM 基础上, 提出了多焦点聚焦模型, 即每次从图像中采取两个或多个局部. 焦点的采样是逻辑并行的. 整个模型依然使用端到端的增强学习 (Reinforce learning, RL) 训练, 保证了模型的有效性和正确率. 由于采样速率加倍, 所以仅需更少的迭代即可完成识别任务, 提高了识别速度和效率, 也提高了训练时的收敛速度. 同时本模型的实用性得到进一步扩展, 不仅可以应用到视频等动态环境, 也可以作为处理多媒体数据的一种思路, 可以将一部分焦点置于视频的同时将另一部分焦点置于音频, 从而完成需要多种输入处理的复杂任务. 经实验验证, 多焦点聚焦模型相比单焦点聚焦模型, 计算精度略优, 计算量降低, 识别速度和训练速度提高约 25 %. 并保留了 RAM 通用性较高等优点. 本文结构安排: 第 1 节介绍多焦点聚焦模型结构; 第 2 节给出了模型训练算法; 第 3 节通过实验验证模型的有效性并进行性能分析; 第 4 节总结并指出下一步研究方向.

1 多焦点模型结构

AM 受启发于人类视觉系统. 人类观察场景时, 并非一次理解整个场景, 而是动态地观察视觉空间中的多个局部获取信息, 再将获取的信息综合以理解当前的场景[17−20]. 同理, 当 AM 接收到一幅图片时, 提取出输入的局部, 进行分析, 得出下一个兴趣区域. 反复提取兴趣区域分析, 直到基于所有信息完成识别.

单焦点聚焦模型识别过程如图 1 所示. 焦点提取图像范围很小. 使用 MINST 数据集时, 图像大小为 28 像素 × 28 像素, 而焦点大小仅为 4 像素 × 4 像素. 焦点过小, 就会提高焦点轮空的概率, 同时也会增加本模型循环的次数. 同时, 焦点计算量较低, 增加焦点对模型计算量的增加不大. 因此本文提出思路, 训练两个甚至多个焦点, 焦点之间并行提取, 提高信息提取速度, 从而提高识别速度. 同时也降低了焦点轮空的概率, 只要有一个焦点命中目标, 就可以保证后续焦点全部有效.

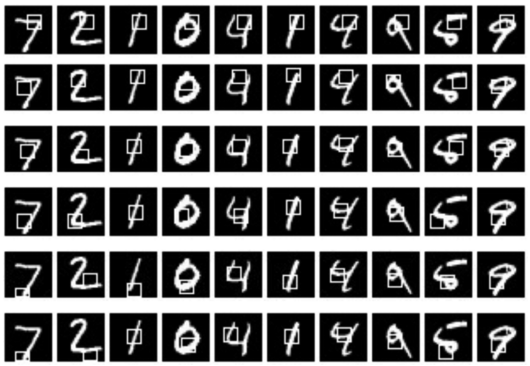


图 1 单焦点聚焦模型识别过程

Fig. 1 Recognition process of RAM

这种思路也可以扩展到多媒体信息的处理, 例如将一个焦点处理视频信息, 另一个焦点处理音频信息. 或者一个焦点在图像中搜索, 一个焦点在文本中搜索, 也能完成在图像中搜索文本相关内容的任务, 例如文献 [13] 中的效果. 所以本模型具有极佳的通用性和扩展性.

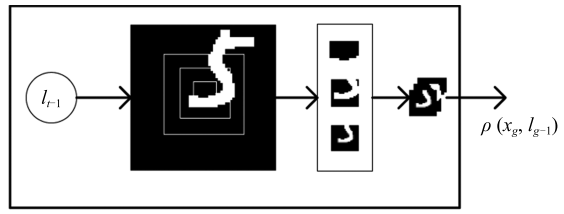
1.1 多焦点模型结构

多焦点模型在整体上是一个 RNN 模型, 如图 2 所示. 图 2 (a) 是图像提取模块. 该模块的输入是完整的图像和坐标, 以坐标为中心提取一个局部. 一个提取模块可以提取一个或多个比例的局部. 稍大的局部可以扩大感知范围, 减少聚焦次数. 提取出的图像后续输入到全联接层, 所以多个比例提取对计算量的影响整体是有利的. 文献 [21] 中, 这种行为称为 Glimpse. 图 2 (b) 是焦点模块. 该模块包含了图像提取模块, 将坐标信息和提取出的图像信息, 各自编码后, 合并在一起进行编码. 图 2 (c) 是多焦点聚焦模型整体结构. 这个模型整体上是一个 RNN. 其中 f 是全联接层, 综合了焦点和坐标信息. 其中 h 是隐藏层, 得到本次观察图象的信息, 结合上一个时刻的状态, 更新当前的状态. 这个状态包含了之前观察到的所有信息. 同时根据内部状态, 行为模块产生新的坐标, 并在下个时刻作为模块的输入. 当模型接收图片后, 每个时间 t, 从图片中提取多个局部信息, 并根据内部状态和所得到的信息选择下一步的行为.

1.1.1 焦点模块

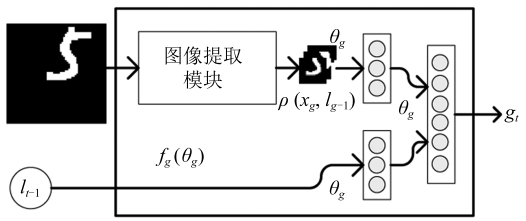
焦点模块输入的是一个完整的图像和坐标, 每一个时刻, 焦点模块以接收到的坐标为中心, 提取图像的部分, 传给后续模块. 初始坐标一般取图像中心为原点 (对于单焦点聚焦模型, 取图像中心为起点可以避免焦点轮空). 该模块也可以坐标为中心, 提取出多幅不同比例的图像. 这样的好处是可以扩大感

知范围. 因为本模块中提取的图像经由全联接层传给后续模块, 所以提取多幅不会增加过多的计算.



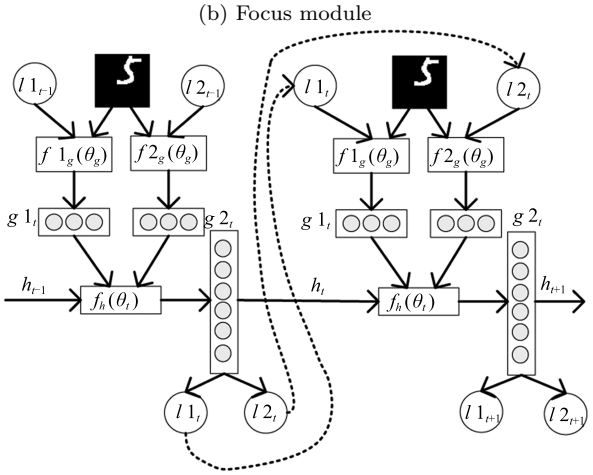
(a) 图像提取模块

(a) Image extraction module



(b) 焦点模块

(b) Focus module



(c) 多焦点聚焦模型整体结构

(c) Multi attention model structure

图 2　模型结构

Fig. 2　Model structure

同时, 当前的坐标也会作为输入, 编码后与处理过的图像局部合并. 多焦点聚焦模型即同时拥有多个焦点模块. 焦点模块之间的工作是独立并行的, 每个焦点模块都仅按照自己收到的坐标采集信息. 所有焦点模块采集的信息会合并在一起进行分析.

1.1.2　内部状态模块

该模块以 RNN 层为核心, 并且将所有焦点模块提取的信息整理到一起. 该模块在每个时间 t, 接收多个焦点模块的信息和本模块上一个时刻的状态. 这个模块内部的状态包含了过去所有焦点观察的信息的总结. 在每个时间 t, 根据过去和当前的信息, 更新内部状态并输出.

1.1.3　行为模块

在每个时刻 t, 行为模块根据内部状态产生下次多个焦点的坐标. 本模型使用 SoftMax 函数对输出进行分类识别. 识别结果与行为之间关系紧密, 可以在识别成功后停止观察, 而不是观察固定的次数. 对复杂的目标可能识别的次数更久, 而对简单的目标可能几次识别就可以退出, 更加灵活. 这样的行为特征也符合人类视觉行为. 对于简单的识别, 人类观察一次即可完成; 当目标复杂时, 人类视觉系统会反复观察, 主动寻找目标的特征. 本模型即模仿了这种行为, 并且利用机器自然的并行性. 本模型中的行为模块, 每次产生多个坐标位置, 而不是分多次产生的, 多个坐标分别送到各个焦点模块. 这是为后面简单使用增强学习进行训练.

1.1.4　评分模块

本模型在每一次采样分析后, 都会得到一个评分信号. 而且对于多焦点聚焦模型, 是对这次所有焦点的采样效果整体评分, 而不是单独给每个焦点评分. 这样的目的是综合考虑多个焦点提取的信息, 避免各个焦点各自行动甚至重复采样的行为. 整个模型的目标就是将这个评分信号的累积和最大化. 累积和信号是非常稀疏并且延时的: $R = \sum_{t=1}^{T} r_t$. 其中 R 是模型完成一次识别后得到的总分, r_t 是一次识别中, 每次循环聚焦行为后得到的分数. 目标识别任务中, 如果经过时间 t 识别成功, 那么 r_t 就是 1, 否则就是 0.

以上步骤在 RL 的领域中是一个典型的局部观察马尔科夫决策过程 (Partially observable Markov decision process, POMDP) 的样例. 这个例子的条件是环境的真正状态不能一次全部观察到. 多焦点模型每次采样得到的只是图像中的局部, 永远不能看到图像整体. 这个条件中, 模型需要学习到一个策略 $\pi\left((l_t, a_t)\,|s_{1:t}; \theta\right)$, 其中 l_t 和 a_t 是时刻 t, agent 决定的下一次的焦点坐标和当前的行为. 在目标识别任务中, 当前的行为就是确定下一次坐标, l_t 和 a_t 是相同的. 其中 $s_{1:t} = x_1, l_1, a_1, \cdots, x_{t-1}, l_{t-1}, a_{t-1}, x_t$, 其中 x_t 是在时刻 t 时的输入, 即图像的一部分; $s_{1:t}$ 即输入和输出的时间序列. 所以策略 π 就是根据当前的输入和之前全部观察的结果, 给出下一次如何行为的策略.

2　模型训练算法

多焦点聚焦模型使用了与 RAM 相同的训练算法. 由于模型每次从图像中观察一部分, 从不将整个图像作为输入, 因此符合 POMDP 的模型. 文献 [8] 对此进行了详细解释.

如何将多个焦点的信息综合在一起处理, 这个问题是训练算法决定的. 多焦点聚焦模型的训练算法是对所有焦点的行为统一打分, 而不是单独打分, 这决定了焦点的整体行为, 保证了所有焦点行为的统一性. 多焦点之间不会重复采样, 而且焦点之间信息共享. 使用 RAM 的训练算法, 在多焦点创新的同时, 保留了原模型端到端的优点, 是一种取巧的解决方案. 训练算法的目标函数, 决定了整个模型的功能目标, 也决定了训练时数据的格式. 多焦点聚焦模型的训练的目标函数为

$$J(\theta)=E_{p(s_{1:T};\theta)}\left[\sum_{t=1}^{T}\sum_{n=1}^{N}r_{nt}\right]=E_{p(s_{1:T};\theta)}[R] \tag{1}$$

$J(\theta)$ 是一次识别任务中循环 T 次, N 个焦点的目标函数. 其中 $p(s_{1:T};\theta)$ 取决于当前的网络参数和策略. r_{nt} 是在一次识别任务中, 焦点 n 第 t 次行为获得的评分. R 是一次识别任务中, 循环 T 次后获取的总分. 为了将多焦点的信息综合在一起, 避免各个焦点各自打分而退化成单焦点模型, 所有焦点的行为被视为同一行为, 所以将 $\sum_{n=1}^{N}r_{nt}$ 写为 r_t. 式 (1) 可以写为

$$J(\theta)=E_{p(s_{1:T};\theta)}\left[\sum_{t=1}^{T}r_t\right]=E_{p(s_{1::T};\theta)}[R] \tag{2}$$

其中

$$r_t=\begin{cases}1, & \text{right}\\ 0, & \text{wrong}\end{cases}$$

之后的推导与文献 [8] 中一致. 多焦点聚焦模型训练算法保留了 RAM 训练算法的优点, 并保证了多个焦点的行为协调一致.

3 实验及结果分析

3.1 数据集

实验基于手写数据集 MNIST 及其变形. MNIST 存储了 60 000 幅 28 像素 × 28 像素的手写数字图像. 在模拟随机位置目标识别的实验中, 将 28 像素 × 28 像素的 MNIST 图像置于 60 像素 × 60 像素图像中的随机位置. 数据标签中只包含识别类型而不包含目标的真实坐标. 在抗噪实验中, 将 MNIST 图像中加入扰动的噪声背景. 在所有实验中, 将数据集分成 20 % 为测试集, 80 % 为训练集, 然后测试多焦点以下方面的性能: 1) 识别中心图像的准确率和速度; 2) 识别随机坐标图像的准确率和速度; 3) 识别复杂的随机坐标图像有效性; 4) 对比单焦点聚焦模型收敛速度; 5) 探索焦点数量的影响

实验环境: 操作系统 MAC OS 10.11.5, 神经网络代码库: Torch. 在 i5 单 CPU 训练 60 像素 × 60 像素的图像约 260 例/s, 单次 epoch 约 50 分钟.

3.2 程序设计实现

根据多焦点聚焦模型设计, 本节给出模型的训练和测试程序的实现流程.

3.2.1 训练程序

输入. 数据集 (图像 – 标签对), 命令行参数 (包括一些可设置的网络参数, 如是否使用 GPU 等).

步骤 1. 准备参数集和数据集.

1) 根据命令行参数, 设置程序参数, 包括模型存储路径, 学习率, 最小学习率, 迭代次数, 是否使用显卡, 批训练规模, RL 尝试深度, 图像提取比例和大小, 数据集等各种参数.

2) 读入数据集并检查数据集规范, 分别存到图像和标签.

步骤 2. 设置网络.

1) 根据参数设置多个焦点模块.

2) 合并多个焦点模块的输出, 作为 RNN 的输入.

3) 定义行为模块, 即坐标生成模块.

4) 封装整个模型为 RL 的 Agent, 并将行为模块的输出在下一步送往整个模型的输入.

5) 定义奖励基准模块等 RL 相关参数.

6) 定义识别模块, 使用 softMax 函数.

步骤 3. 定义训练相关参数.

1) 定义 loss 函数, 其中反向传播和 RL 模块需要分别定义.

2) 根据 Torch 函数设置 Optimizer, Evaluator 和 Tester. 其中 Optimizer 在训练中更新参数, Tester 进行测试.

3) 根据 Torch 设置 Experiment, 这是训练的主要函数, 将整个模型, Optimizer, validator 和 tester 等输入.

4) 判断是否采用 GPU, 如果是则将整个网络设置成相应模式.

步骤 4. 输入数据集, 调用 Torch 的相关函数, 开始训练.

3.2.2 测试和展示程序

该程序可以测试保存好的网络模型, 并且将模型识别的过程展示出来, 效果如图 1.

输入. 数据集和已保存的模型文件.

步骤 1. 准备参数集, 数据集和网络模型.

1) 根据命令行参数, 设置程序参数, 包括模型存储路径, 学习率, 最小学习率, 迭代次数, 是否使用显卡, 批训练规模, RL 尝试深度, 图像提取比例和大小, 数据集等各种参数.

2) 读入数据集并检查数据集规范, 分别存到图像和标签.

3) 读入网络模型.

步骤 2. 定义输入, 整个模型前向传播一次.

步骤 3. 获取并绘制模型信息.

1) 定义绘制焦点函数.

2) 获取网络内部信息, 得到每次网络迭代焦点坐标.

3) 调用绘制函数, 在图像中标出焦点范围并保存. 打印所有信息.

3.3 实验及结果分析

实验 1. 识别中心图像的准确率和速度

为了验证多焦点模型对物体的识别功能的有效性, 在 MNIST 数据集中与单焦点聚焦模型识别效果对比. 实验结果如图 3 和表 1 所示.

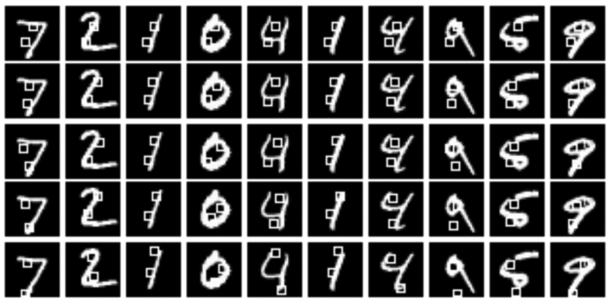


图 3　多焦点模型识别过程

Fig. 3　Recognition process of multi attention model

表 1　多焦点模型错误率

Table 1　Multi-attention model error rate

模型	错误率 (%)
RAM, 2 次	8.11
RAM, 4 次	3.28
RAM, 6 次	2.11
RAM, 8 次	1.55
RAM, 10 次	1.26
多焦点模型, 2 次	4.17
多焦点模型, 4 次	2.59
多焦点模型, 6 次	1.58
多焦点模型, 8 次	1.19
多焦点模型, 10 次	1.19

从图 3 可以看出, 两个焦点沿着不同方向寻找信息内容, 是相互独立而又合作的关系, 是为了同一个目标而努力.

表 1 中 RAM 是单焦点聚焦模型, 由于本文数据未达到文献 [8] 中的最佳, 这里相同实验并未引用文献 [8] 的数据. 焦点模型的焦点范围均设 8 像素 × 8 像素, 采样深度为一层, 即 scale 为 1. 从表 1 可以看出, 在处理简单的图像中央的数据时 (如 MNIST), 观察相同的次数, 多焦点模型比 RAM 正确率更高; 达到同等的正确率, 多焦点聚焦模型需要的次数更少, 约为 RAM 的 75 %. 这足以验证多焦点模型的合理性, 相比 RAM 的识别速度提高约 25 %.

实验 2. 识别随机坐标图像的准确率和速度

为了进一步验证多焦点聚焦模型识别目标位置随机的图像的功能, 使用了 MNIST 的扩展数据集. 将 MNIST 置于 60 像素 × 60 像素图像中的随机位置. 图 4 给出几幅图像效果. 表 2 是 60 像素 × 60 像素随机坐标图像的正确率, 提取窗口 12 像素 × 12 像素.

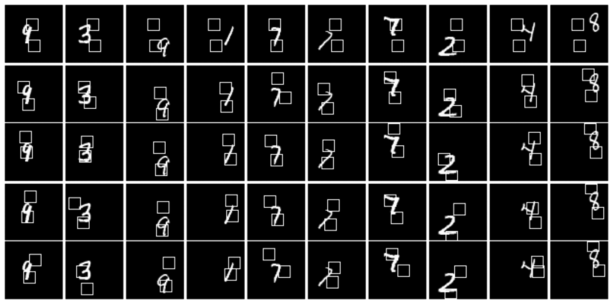


图 4　多焦点模型在 60 像素 × 60 像素图像中识别的效果

Fig. 4　Recognition process of multi attention model in 60 × 60 image dataset

表 2　随机坐标错误率

Table 2　Random position error rate

模型	错误率 (%)
RAM, 2 次	1.51
RAM, 4 次	1.29
RAM, 6 次	1.22
多焦点模型, 2 次	2.81
多焦点模型, 4 次	1.55
多焦点模型, 6 次	1.01

从图 4 可以看出, 两个焦点之间是互相配合的关系, 这是因为模型中两个焦点提取的信息是在一起处理的, 两个焦点的行为也是作为一个整体进行打分的, 因此这是两个焦点重叠在一起的情况.

由于本文数据未达到文献 [8] 中的最佳, 表 2 中 RAM 的相同实验并未引用文献 [8] 的数据. 从表 2 可以看出, 多焦点聚焦模型可以有效判读随机坐标的目标. 相比单焦点聚焦模型, 识别效率更高. 达到模型最高正确率, 多焦点聚焦模型需匹配 4 次, 单焦

点模型需要 6 次, 多焦点聚焦模型的识别速度提高了 20 %～25 %.

实验 3. 识别复杂的随机坐标图像有效性

RAM 具有一定的抗噪能力. 本文对此进行了对照测试. 按照文献 [8] 中的方法生成噪声图像: 在 60 像素 × 60 像素的区域中, 随机放入一幅 MNIST 图像, 再随机从其他 MNIST 图像中截取 8 像素 × 8 像素的局部放入随机的位置. 对于 CNN 等将整幅图像作为输入处理的模型, 必须学会如何忽略局部的噪声. 而理论上, 聚焦模型可以轻易学会避免局部噪声而仅关注信息相关部分. 相比单焦点聚焦模型, 多焦点聚焦模型的焦点 "扫到" 噪声区域的概率更大. 但是通过训练算法可知, 只有当焦点获取的信息有利于识别时, 这样的行为才被认可. 所以理论上, 多焦点聚焦模型应同样不会被局部噪声影响. 测试结果如图 5 和表 3 所示.

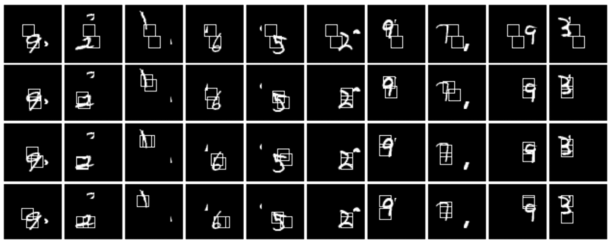


图 5 多焦点聚焦模型相比 RAM 的对照试验

Fig. 5 Control experiment between RAM and multi attention model

表 3 噪声环境对比

Table 3 Noisy dataset error rate between RAM and multi attention model

模型	错误率 (%)
RAM, 2 次	4.96
RAM, 4 次	4.08
RAM, 6 次	4.04
多焦点模型, 2 次	5.25
多焦点模型, 4 次	4.47
多焦点模型, 6 次	3.43

图 5 中第 1 行为模型初始焦点位置, 图 5 为模型 4 次尝试的焦点落点. 从图 5 可以看出, 两个焦点为了识别而共同努力.

由于本实验图像 60 像素 × 60 像素较大, 这里 RAM 数据引用了文献 [8] 的实验数据. 焦点范围为 12 像素 × 12 像素, scale 为 3. 从表 3 可以看出, 多焦点聚焦模型继承了 RAM 的抗噪优点. 这是因为在 RL 算法中, 只对有效信息的提取进行奖励, 局域性的强噪声会被自然地忽略. 实验结果基本符合之前对抗局域噪声性能的分析. 再次证明了用 RL 训练聚焦模型的有效性.

实验 4. 对比单焦点聚焦模型训练收敛速度

本实验采用了之前实验记录的数据. 模型参数: MNIST 数据集, 焦点深度为 1, 焦点范围为 4 像素 × 4 像素, 尝试次数为 7; Noise 数据集, 图像大小 60 像素 × 60 像素, 焦点深度为 3, 焦点范围 12 像素 × 12 像素, 尝试次数为 4. 收集了模型训练期间打印的信息, 得到收敛速度对比如图 6 所示.

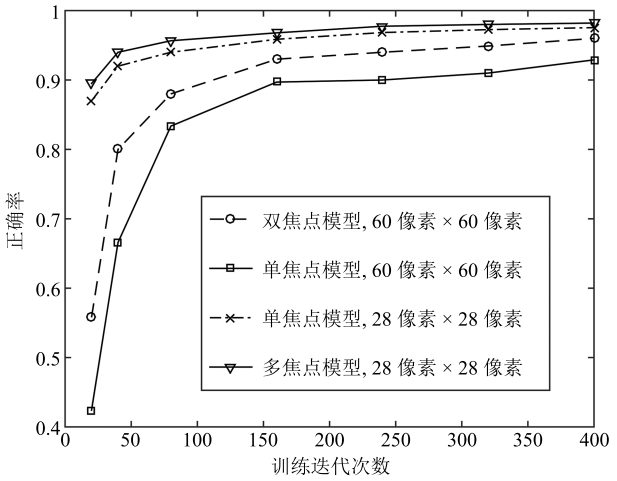


图 6 对比训练时收敛速度

Fig. 6 Convergence speed control experiment

在作者机器中, 训练 MNIST 数据集时, 单焦点模型的训练速度约 800 张/s, 多焦点训练速度约 600 张/s, 训练迭代一次速度相差 20 %. 在实验 1 中, 正确率收敛到 0.92 时, RAM 需迭代 400 次左右, 多焦点聚焦模型需 125 次左右, 综合训练速度多焦点聚焦模型更快, 在单 CPU 上快约 2 倍左右. 这是模型的设计决定的, 如果使用并行度更好的硬件如 GPU 等计算, 应有更好的效果, 因为输入图像一样, 多焦点聚焦模型仅多出一焦点模块的计算, 而且是逻辑上并行的模块. 所以如果硬件支持较好, 多焦点模型的时间复杂度约等于单焦点模型.

实验 5. 不同焦点数量对识别的影响

本实验基于 MNIST 数据集, 对比了不同数量焦点对模型的性能影响. 模型参数方面, 焦点深度均为 1, 焦点范围为 8 像素 × 8 像素. 对比了不同模型间识别的正确率之间的关系. 实验结果如图 7 所示, 其中单焦点数据使用了循环次数为 4 次和 6 次的数据, 多焦点聚焦模型使用了循环 2 次和 3 次的数据. 整体上看相同循环次数下, 焦点越多错误率越低. 但是由于数据集较小, 循环 2～3 次后, 模型正确率已接近极限, 优势较小.

本实验也对比了不同焦点不同循环次数下, 模

型的运算速度. 实验结果如图 8 所示, 其中焦点为 1 的数据循环次数取 4 次和 6 次. 可以看出随着焦点数量的增加, 在 CPU 处理器上计算性能是几乎直线下降的, 而且循环次数增加会加剧性能恶化. 在一次识别中, 循环 2～3 次是达到理想正确率必须的. 考虑到正确率已经饱和, 综合实验结果, 在当前多核 CPU 下, 2 焦点是正确率和运行速度的最佳选择.

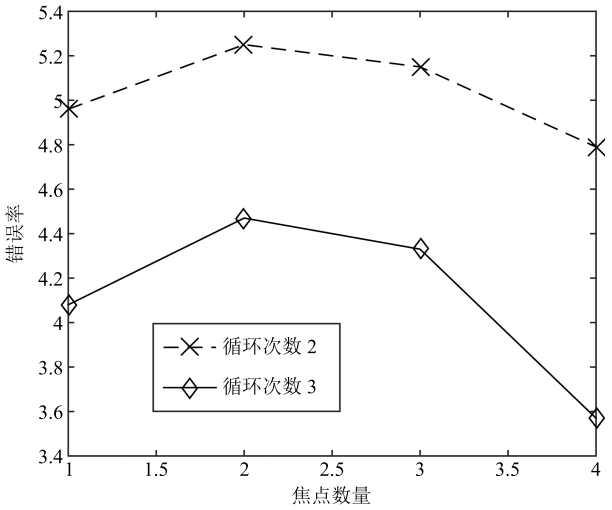


图 7　焦点数量与正确率关系

Fig. 7　Relationship between quantify and accuracy

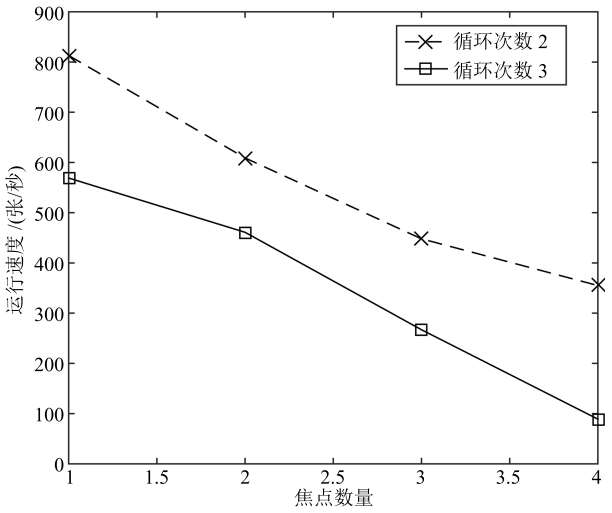


图 8　焦点数量与运行速度关系

Fig. 8　Relationship between quantify and speed

3.4　实验结论和分析

综上实验可以看出, 多焦点聚焦模型的识别效率领先于单焦点聚焦模型, 体现在达到同样的识别率, 观察的次数更少, 或观察相同的次数, 正确率更高. 同时相比于单焦点聚焦模型, 多焦点模型训练的收敛速度更快. 实验对比了焦点数量与正确率和运行速度的关系. 焦点数量增加, 但正确率已饱和, 同时运行时间迅速恶化, 循环次数也会加剧恶化. 相比较下, 2 焦点在正确率和运行速度方面取得了最好的效果. 多焦点聚焦模型的表现与数据集和硬件性能相关度很高.

多焦点聚焦模型的重要特点是通过并行提取多处焦点信息, 提高了提取和处理信息的效率. 多焦点聚焦模型相比单焦点聚焦模型, 在整体计算量上, 多了焦点模块的相关计算. 同时训练期间的收敛速度也会加速. 其中 2 焦点模型单次迭代的时间增加较少, 而整体迭代次数减少, 达到同等的正确率, 2 焦点聚焦模型需要的次数约为 RAM 的 75 %. 整体收敛速度提高 1.33 倍左右.

4　结论

本文基于单焦点聚焦模型, 提出并实现了一个多焦点聚焦模型. 多焦点聚焦模型的主要思想, 是并行提取输入数据中的多处信息. 使用 RL 训练网络模型, 让模型学会寻找最有利的焦点位置. 提取输入数据中最有效的信息, 完成识别任务. 使用 MNIST 数据集及其变种进行各种实验, 验证了多焦点聚焦模型的有效性. 实验结果证明, 相比于单焦点聚焦模型, 正确率略优, 训练速度和识别速度都有提高, 因为多焦点聚焦模型的信息提取效率较高. 同时多焦点聚焦模型保留了单焦点聚焦模型的多种优点, 识别目标位置灵活, 计算量独立于输入图像大小, 一定的抗噪声能力, 端到端的训练算法. 多焦点聚焦模型具有更大的实用潜力, 例如用在多媒体信息处理的任务等.

下一步, 用更多更大的数据集和更大的图像 (如 ImageNet) 进行测试同一图像中多目标的识别. 将最优的焦点形状、面积、数量, 更快的训练速度, 优化 GPU 的表现应用于更多的应用, 例如对多媒体文件的识别.

References

1 Mnih V, Kavukcuoglu K, Silver D, Rusu A A, Veness J, Bellemare M G, Graves A, Riedmiller M, Fidjeland A K, Ostrovski G, Petersen S, Beattie C, Sadik A, Antonoglou I, King H, Kumaran D, Wierstra D, Legg S, Hassabis D. Human-level control through deep reinforcement learning. *Nature*, 2015, **518**(7540): 529−533

2 Mordvintsev A, Olah C, Tyka M. Inceptionism: going deeper into neural networks [Online], available: http://research.googleblog.com/2015/06/inceptionism-going-deeper-into-neural.html, August 22, 2016

3 Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks. In: Proceedings of the 25th International Conference on Neural Information Processing Systems. Lake Tahoe, Nevada, USA: Curran Associates Inc., 2012. 1097−1105

4 Girshick R, Donahue J, Darrell T, Malik J. Rich feature Hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition. Columbus, OH, USA: IEEE, 2014. 580−587

5 Sermanet P, Eigen D, Zhang X, Mathieu M, Fergus R, LeCun Y. OverFeat: integrated recognition, localization and detection using convolutional networks [Online], available: http://arxiv.org/abs/1312.6229, August 22, 2016

6 Felzenszwalb P F, Girshick R B, McAllester D. Cascade object detection with deformable part models. In: Proceedings of the 2010 IEEE Conference on Computer Vision and Pattern Recognition. San Francisco, CA, USA: IEEE, 2010. 2241−2248

7 Viola P, Jones M. Rapid object detection using a boosted cascade of simple features. In: Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Kauai, HI, USA: IEEE, 2001. 511−518

8 Mnih V, Heess N, Graves A, Kavukcuoglu K. Recurrent models of visual attention. In: Proceedings of the 27th International Conference on Neural Information Processing Systems. Cambridge, MA, USA: MIT Press, 2014. 2204−2212

9 Rensink R A. The dynamic representation of scenes. *Visual Cognition*, 2000, **7**(1−3): 17−42

10 Yoo D, Park S, Lee J Y, Paek A S, Kweon I S. AttentionNet: aggregating weak directions for accurate object detection [Online], available: http://arxiv.org/abs/1506.07704, August 22, 2016

11 Stollenga M F, Masci J, Gomez F, Schmidhuber J. Deep networks with internal selective attention through feedback connections. In: Proceedings of the 27th International Conference on Neural Information Processing Systems. Cambridge, MA, USA: MIT Press, 2014. **4**(2): 107−122

12 Legrand J, Collobert R. Jiont RNN-based greedy parsing and word composition [Online], avaliable: https://arxiv.org/abs/1412.7028?context=cs, August 22, 2016

13 Alexe B, Heess N, Teh Y W, Ferrari V. Searching for objects driven by context. In: Proceedings of the 25th International Conference on Neural Information Processing Systems. Lake Tahoe, Nevada, USA: Curran Associates Inc., 2012. 881−889

14 Feng Xin, Yang Dan, Zhang Ling. Saliency variation based quality assessment for packet-loss-impaired videos. *Acta Automatica Sinica*, 2011, **37**(11): 1322−1331
(冯欣, 杨丹, 张凌. 基于视觉注意力变化的网络丢包视频质量评估. 自动化学报, 2011, **37**(11): 1322−1331)

15 Liu Long, Fan Bo-Yang, Liu Jin-Xing, Yang Le-Chao. Particle filtering based visual attention model for moving target detection. *Acta Electronica Sinica*, 2016, **44**(9): 2235−2241
(刘龙, 樊波阳, 刘金星, 杨乐超. 面向运动目标检测的粒子滤波视觉注意力模型. 电子学报, 2016, **44**(9): 2235−2241)

16 Zhang Chong. Text Classification Based on Attention-Based LSTM Model [Master dissertation], Nanjing University, China, 2016.
(张冲. 基于 Attention-Based LSTM 模型的文本分类技术的研究 [硕士学位论文], 南京大学, 中国, 2016.)

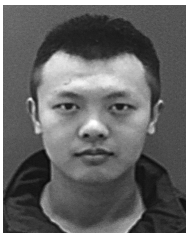
17 Denil M, Bazzani L, Larochelle H, de Freitas N. Learning where to attend with deep architectures for image tracking. *Neural Computation*, 2012, **24**(8): 2151−2184

18 Paletta L, Fritz G, Seifert C. Q-learning of sequential attention for visual object recognition from informative local descriptors. In: Proceedings of the 22nd International Conference on Machine Learning. New York, NY, USA: ACM, 2005. 649−656

19 Ranzato M. On learning where to look [Online], available: http://arxiv.org/abs/1405.5488, August 22, 2016.

20 Stanley K O, Miikkulainen R. Evolving a roving eye for go. In: Proceedings of the 2004 Genetic and Evolutionary Computation Conference. Berlin, Heidelberg, Germany: Springer, 2004. 1226−1238

21 Larochelle H, Hinton G. Learning to combine foveal glimpses with a third-order Boltzmann machine. In: Proceedings of the 23rd International Conference on Neural Information Processing Systems. Vancouver, Canada: Curran Associates Inc., 2010. 1243−1251



刘 畅 国家数字交换系统工程技术研究中心硕士研究生. 主要研究方向为人工智能和芯片技术. 本文通信作者. E-mail: liunux1992@gmail.com
(**LIU Chang** Master student at China National Digital Switching System Engineering and Technological Research and Development Center. His research interest covers artificial intelligence and chip design technology. Corresponding author of this paper.)

刘勤让 国家数字交换系统工程技术研究中心研究员. 主要研究方向为片上网络设计. E-mail: qinrangliu@sina.com
(**LIU Qin-Rang** Researcher at China National Digital Switching System Engineering and Technological Research and Development Center. His main research interest is network-on-chip.)