

现代计算机技术在遗传学实验教学中的应用 ——移动端轻量级高精度果蝇遗传性状批量识别系统的开发

安钧浩, 赵雪莹, 乔守怡, 卢大儒, 皮妍

复旦大学生命科学学院, 生物科学国家级实验教学示范中心, 上海 200433

摘要: 果蝇是实验教学中最常用的重要生物材料之一。在果蝇实验教学中, 每个学生通常需要针对上百只果蝇进行手工辨认, 并记录每只果蝇身上的数个不同性状, 工作量大且分类标准参差不齐。为了解决这一问题, 本文将现代计算机技术融入到遗传学实验教学中, 使用深度卷积神经网络来自动统计每只果蝇的性状。采用的是目标检测模型+分类模型的两阶段策略模式。在分类模型的训练设计过程中, 创新性利用了关键点辅助分类的方法, 有效地提升了模型的可解释性。此外, 还针对任务特性改善了 RandAugment 方法, 利用渐进式学习与适应性正则化策略, 在有限的计算资源下训练了 MobileNetV3 架构下的多标签分类任务, 并最终在每只果蝇 3 对性状(红/白眼、长/小翅、雌/雄)的分类任务下分别达到了 97.5%、97.5%和 98%的准确率。模型经过优化后, 可以在手机端 10 s 内完成 600 个果蝇性状的分类, 该模型具有轻量化的特点, 大小不到 5 MB, 易于在各类安卓系统手机上安装使用。该系统的开发有利于推进以果蝇为研究对象的遗传规律验证等实验的教学, 也可用于涉及大量果蝇分类统计分析的科研工作。

关键词: 现代计算机技术; 实验教学; 移动端; 果蝇性状识别系统

收稿日期: 2022-12-12; 修回日期: 2023-01-13; 网络发布日期: 2023-03-17

基金项目: 复旦大学本科教学研究与改革实践项目(课程思政, 编号: FD2021E002, FD2021E003, FD2021E004)资助[Supported by Fudan University Undergraduate Teaching Research and Reform Practice Project (Nos. FD2021E002, FD2021E003, FD2021E004)]

作者简介: 安钧浩, 本科在读, 专业方向: 生物信息学。E-mail: anjunhao_16@outlook.com

通讯作者: 皮妍, 博士, 高级讲师, 研究方向: 分子遗传学。E-mail: yanpi@fudan.edu.cn

DOI: 10.16288/j.ycz.22-409

Application of modern computer technology in classical genetics lab course

——Development of a mobile, lightweight and high-precision batch identification system for genetic traits of *Drosophila*

Junhao An, Xueying Zhao, Shouyi Qiao, Daru Lu, Yan Pi

National Demonstration Center for Experimental Biology Education, School of Life Sciences, Fudan University, Shanghai 200433, China

Abstract: *Drosophila* is a crucial biological experimental teaching material extensively utilized in experimental teaching. In this experimental teaching, each student typically needs to manually identify hundreds of fruit flies and record multiple of each fly. This task involves substantial workload, and the classification standards can be inconsistent. To address this issue, we introduce a deep convolutional neural network that classifies the traits of every fruit fly, using a two-stage consisting of an object detector and a trait classifier. We propose a keypoint-assisted classification model with tailored training session for the trait classification task and significantly enhanced the model interpretability. Additionally, we've enhanced the RandAugment method to better fit the features of our task. The model is trained with progressive learning and adaptive regularization under limited computational resources. The final classification model, which utilizes MobileNetV3 as backbone, achieves an accuracy of 97.5%, 97.5% and 98% for the eyes, wings, gender tasks, respectively. After optimization, the model is highly lightweight, classifying 600 fruit fly traits from raw images in 10 seconds and having a size less than 5 MB. It can be easily deployed on any android device. The development of this system is conducive to promoting the experimental teaching, such as verifying genetic laws with *Drosophila* as the research object. It can also be used for scientific research involving a large number of *Drosophila* classifications, statistics and analyses.

Keywords: modern computer technology; experiment teaching; mobile terminal; *Drosophila* character recognition system

果蝇是经典遗传学研究中被广泛使用的一种模式动物^[1,2]。复旦大学生命科学学院开设的遗传学实验必修课程中,都会让学生完成一次果蝇遗传规律验证实验。在该实验中,每名学生需要计数100多只果蝇,并观察统计每一只果蝇的遗传性状。

实验包含两个阶段:第一阶段是由野生型雄蝇和隐性纯合雌蝇(白眼和小翅)杂交,产生第一代果蝇(F₁)。小翅果蝇的翅膀比野生型的翅短些,仅仅长至腹部端,与身体的体长差不多。第二阶段是由F₁代中的雌雄果蝇自交,产生第二代果蝇(F₂)。F₂代果蝇表现出性状分离,学生主要观察的3个性状(红/白眼、长/小翅、雌/雄)都会较为随机地分配在每只果蝇上,产生总共2×2×2=8种果蝇表型。

需要观察统计的3个性状为:果蝇的复眼(eye)颜色、性别(gender)以及翅(wing)的长度,如图1所

示。学生在培养获得F₂代果蝇群体后,需要逐个地在显微镜下分辨果蝇的各个性状并统计计数。

在显微镜下逐个辨别100多只果蝇的不同的几个性状,耗时巨大,任务比较繁重,每个学生需要花费几个小时完成果蝇性状的辨别和计数工作,且由于个体之间的分类标准参差不齐,尤其是针对长/小翅之间的分类标准差异较大,容易给该实验带来较大的误差,在计算验证遗传规律时数据容易产生较大偏差。

在对果蝇遗传性状的鉴别操作中,数据的收集具有如下特点:(1)易于收集数据。可以使用手机直接在解剖镜下拍照,采取数据;(2)易于招募数据标注人员。参与遗传学实验课程的学生均可以作为招募对象,为采集的果蝇数据进行标注;(3)易于获取持续性数据:由于遗传学实验是专业必修课程,果

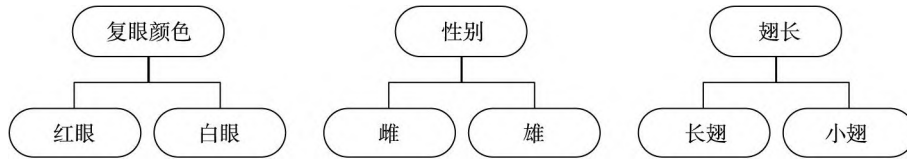


图 1 需要观察统计的果蝇性状

Fig. 1 *Drosophila* traits of interest

蝇实验是经典的实验，每学年都会开设，可以用新的数据不断地迭代模型，并对模型进行优化。

因此，本文希望能引入现代计算机技术，设计出一种移动端轻量级高精度果蝇遗传性状批量识别系统，可以帮助学生直接在手机端上采集果蝇图片后，直接运用模型系统快速统计出所有果蝇的遗传性状。这样不仅可以免去大量的人力工作，节省时间，还可以减少判断误差，提高实验结果的准确性。

1 实验数据的初步收集和模型的初步构建

1.1 数据采集方式

总共招募了 34 名学生采集并标注数据。学生在显微镜下采集原始图片，每张原始图片中包含 40~70 只果蝇。随后在电脑上标注定位框，确定每个果蝇的位置，并依据这些数据训练初期的目标检测模型。最后，通过目标检测模型给出的定位框，裁剪出原始图片中每只果蝇的小图片，并由志愿者学生标注出性状。

1.2 初期数据集的构建

初期数据集的构建，首先需要在 F_1 代果蝇图片上标注出定位框，用于训练初期目标检测模型；并在 F_2 代果蝇上标注出性状，用于训练分类模型。

其中， F_1 代目标检测数据集包括 31 张原始图片，包括约 600 个定位框。 F_2 代数据集基于收集到的 119 份原始图片，通过目标检测算法切割为 3206 张小图片后，随机取约 25% 进行标注，获得分别区分 3 个性状的子集。其中，雌雄子集包括 365 雌+315 雄，共 680 份；红/白眼子集包括 451 红+323 白，共 774 份；长/小翅子集包括 413 长+334 小，共 744 份。

1.3 数据清洗

在初期实验中，训练了基于多种 CNN (convolutional

neural network) 架构的神经网络图像分类模型^[3-6]。在果蝇复眼颜色以及性别这两个性状的分类上，模型能够在验证集上给出约 96% 的准确率，但是在果蝇长/小翅的性状上，只有约 70% 的准确率(图 2)。

通过分析，发现数据集中包含以下两个主要问题：样本不均衡以及错误率高。后期人工将每个样本均补充到 400 份后，运用多种 CNN 架构重新训练分类模型，统计模型分类错误频率最高的数据集样本，并在这些样本中运用人工重新审查归类，并再三核对确认，直到数据集中不再具有错误样本。最终在果蝇性别性状子集中重新归类出了约 50 个错误样本，在长/小翅子集中重新归类出了约 200 个错误样本。这也说明人工归类果蝇性状确实会导致大量错误。

此外，由于初期目标检测模型数据集是在 F_1 代果蝇上标注的， F_1 代果蝇又与 F_2 代果蝇颇为不同。为了提升准确率，本文额外在 F_2 代果蝇上重新标注了新的目标检测数据集。

1.4 最终数据集的建立

最终的目标检测数据集包括 80 张原始图片以及 1733 个定位框。该数据集的一些可视化图像见图 3 和图 4。

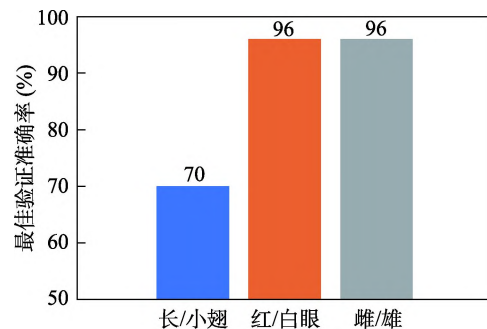


图 2 模型在不同分类任务上的准确率

Fig. 2 Model accuracy on different classification tasks

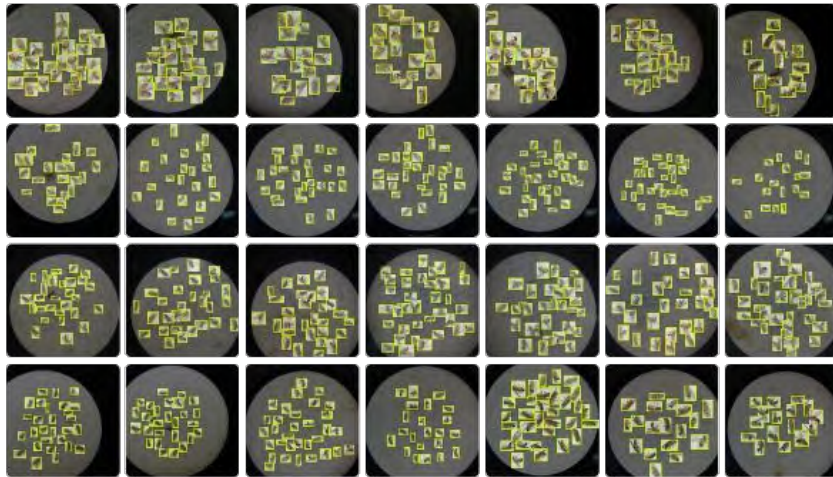


图 3 目标检测数据集中的一些图片样本
Fig. 3 Samples in the object detection dataset

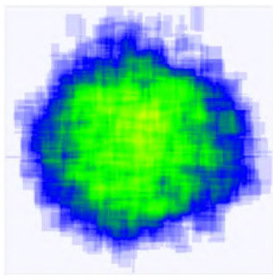


图 4 目标检测数据集的定位框分布热力图
Fig. 4 Bounding box heatmap of the object detection dataset

最终的性状分类数据集分为 3 个子集, 分别对应 3 个果蝇性状。每个子集都有 400+400 个样本,

该数据集的一些可视化图像见图 5 和图 6。

此外, 还额外标注了关键点数据集。被标注的关键点包括: 果蝇的头部顶端、果蝇的尾部尖端以及果蝇的两翅尖端, 即每个果蝇包括 4 个关键点。关于该数据集的一些可视化图像见图 7。

2 模型的优化——关键点辅助的图像分类模型的建立

2.1 设计目标

在初期模型中, 对于果蝇长/小翅的分类准确率



图 5 性状数据集中的一些图片样本
Fig. 5 Samples in the classification dataset

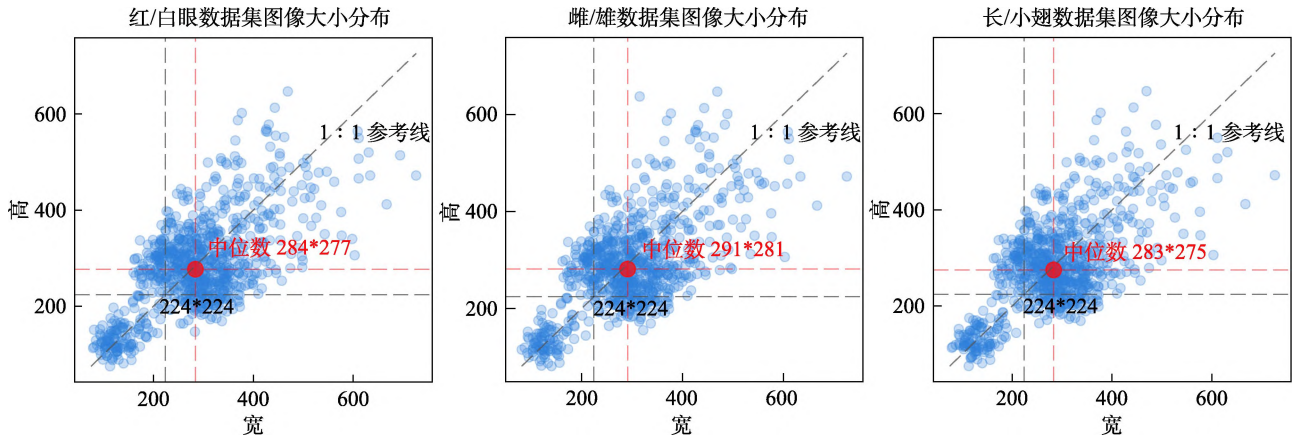


图 6 3 个子集的图片尺寸分布
Fig. 6 Image size map of three subsets

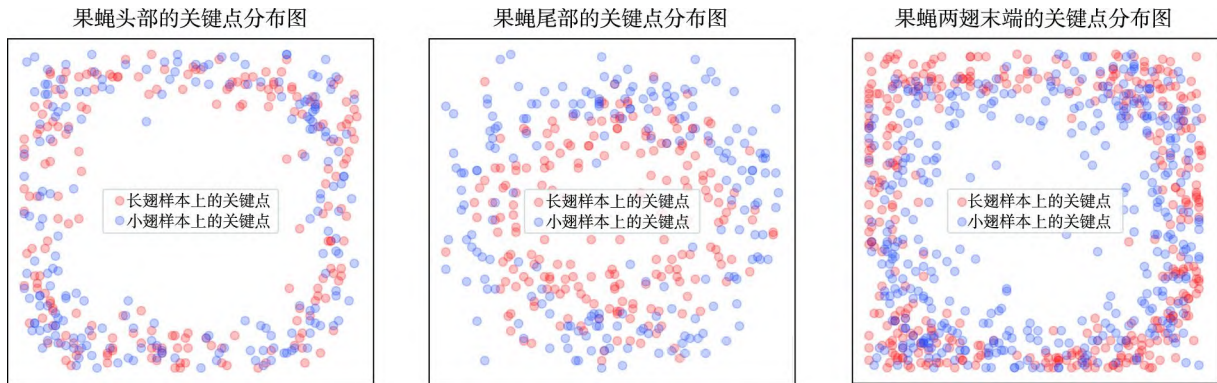


图 7 各个关键点在图片中相对位置的分布
Fig. 7 Distribution of relative locations of the keypoints on images

不高, 不尽如人意。为了提升准确率, 重新调整了果蝇长/小翅的分类标准。由于在区分长/小翅的性状时, 不能单从翅膀的长度考虑, 还需要观察并比较果蝇的身长和翅长。于是本文改进原有识别方式, 将果蝇的身长和翅长作为一种显性模式提供给模型, 补充标记出原来长/小翅数据集中的 800 张样本的身长和翅长的关键点, 希望提高模型分类的识别度, 从而提高准确率。

2.2 关键点编码方式的选择和优化

目前在计算机视觉领域, 主要有两种方式来表示关键点。第一种是目前较为流行的高斯热力图方式, 也就是以关键点为中心, 以高斯分布生成热力图^[7, 8], 这样可以将多个关键点叠加在同一个热力图上。该方法由于需要上采样生成高分辨率的热力图,

往往需要相对较大的模型^[9-11], 难以轻量化。第二种方法是用 XY 坐标来表示关键点^[12], 该方法需要明确关键点的顺序, 但由于只需要输出浮点数坐标, 故不需要大的模型就可以实现。由于本文的模型将来需要在移动端设备部署实现, 因此选择使用第二种方法。

在表示关键点时, 由于果蝇都具有一对翅(两个), 所以每个果蝇都有两个翅尖端的关键点, 这在传统的 XY 坐标表示方法中, 必须要区分左右。而在辨别果蝇长/小翅性状时, 并不需要区分左右。区分左右引入了冗余的信息, 可能会提高模型的区分难度。因此, 本文对该方法进行了改进, 在仍然使用 XY 坐标的前提下, 避免加入区分左右这一额外的维度。

解决方案为, 将原始的坐标 (x_1, y_1) , (x_2, y_2) 替换为 (a, b, c, d, e) , 具体规则如下:

$$\begin{aligned}
 a &= (x_1 + x_2) / 2 \\
 b &= (y_1 + y_2) / 2 \\
 c &= \text{abs}(x_1 - x_2) \\
 d &= \text{abs}(y_1 - y_2) \\
 e &= 1 \text{ if } (x_1 - x_2) * (y_1 - y_2) > 0, \text{ else } 0
 \end{aligned}$$

图 8 展示了用这种规则编码关键点的情况。这种编码方式不仅减少了学习难度，还减轻了数据标注时的工作量。

2.3 模型的建构与模型训练方式的选择

基于上述的关键点编码方式，本文设计的模型架构如图 9 所示。

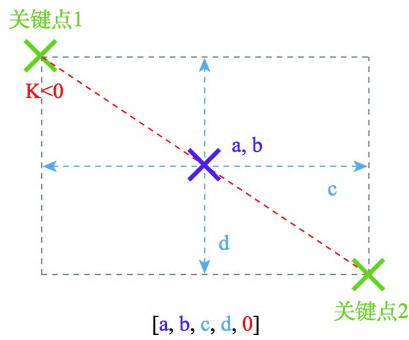


图 8 XY 坐标编码对称关键点举例
 Fig. 8 An example for encoding symmetric keypoints with Cartesian coordinates

图中， $(x_{\text{头}}, y_{\text{头}}, x_{\text{尾}}, y_{\text{尾}})$ 表示果蝇头部顶端以及尾部尖端的两个关键点坐标，这两个坐标不具有对称性，故不适宜使用上述编码方法。 (a, b, c, d, e) 就是利用上述编码方式表示的两个翅尖端坐标。在第一阶段中，着重训练模型的关键点部分。需要注意的是，在计算损失函数时，需要给两个二元分类输出乘以一个小数，来避免对于关键点坐标训练的影响。其原因是二元交叉熵给出的损失函数值往往远大于平均方差给出的数值。第一阶段训练关键点时的训练指标曲线图见图 10 所示。

在关键点损失达到一个不错的指标后，冻结所有层，只训练顶部的长/小翅分类层，当分类指标有所提升后，再解冻所有层，用一个较小的学习率训练整个模型。

运用该方法产生的模型和不用该方法产生的模型，通过运行后比较发现两个模型都具有相似的验证集准确率。但是运用上述方法产生的模型具有更好的可解释性。

运用上述关键点辅助图像分类方法生成的模型和不用上述方法生成的模型，在 Grad-CAM 可视化^[13]方法下的比较结果如图 11 所示。

从图 11 中可以发现，关键点辅助分类产生的模型中，更多与分类相关的区域被激活，例如头部顶

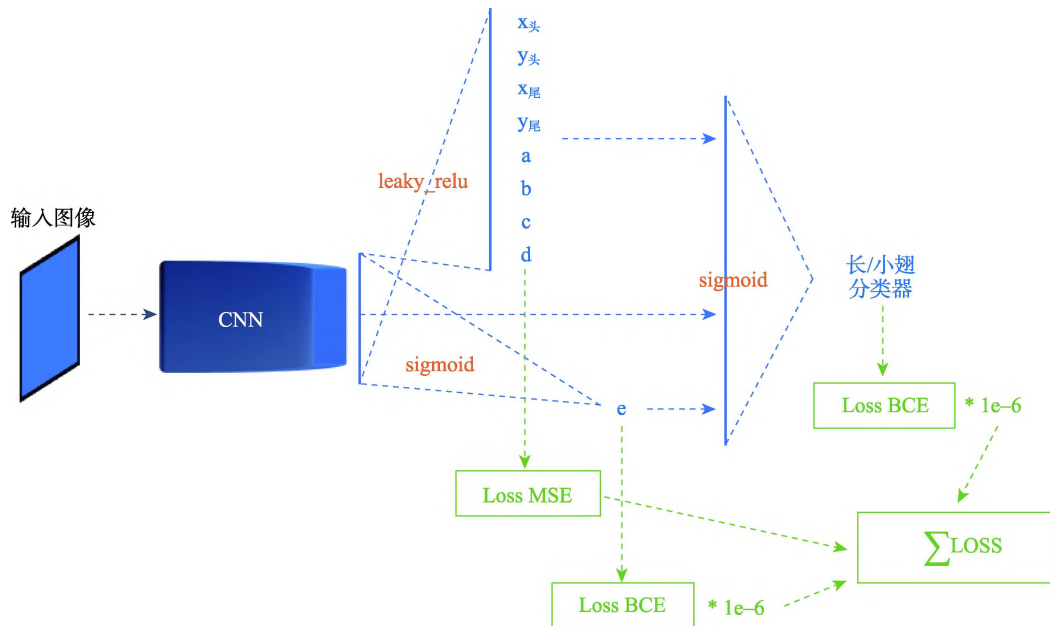


图 9 关键点辅助图像分类模型架构
 Fig. 9 Keypoint-assisted image classification model architecture

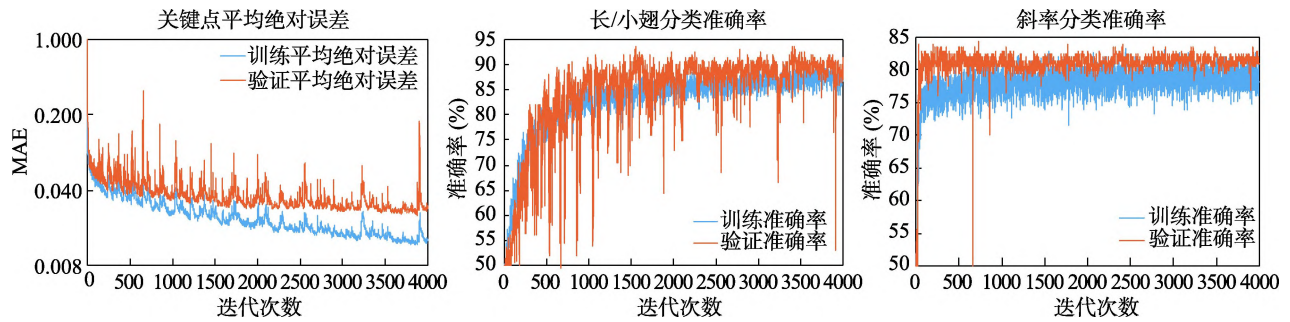


图 10 训练指标曲线图

Fig. 10 Keypoint-assisted image classification model architecture

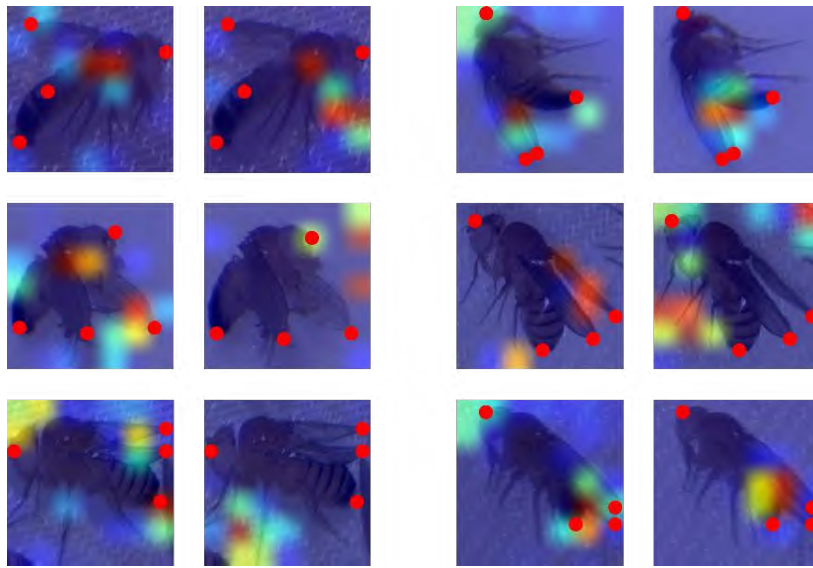


图 11 Grad-CAM 结果

Fig. 11 Result of Grad-CAM

每列的左侧为关键点辅助分类，右侧为普通分类。

端、尾部尖端以及两翅末端等。这与人们分类长/小翅性状的习惯更为相似，因此可以认为该模型提取了更有区分意义的特征。

3 手机移动端模型的构建

3.1 移动端目标检测模型的构建

为了生成占存储空间更小、在移动端 CPU 上运行速度更快的模型，本文采用 EfficientDet 作为目标检测模型^[14]。EfficientDet 是基于 EfficientNet 架构的一种可延展的模型架构。基于此，本文设计了两个不同大小的模型系统——EfficientDetLite0 和 EfficientDetLite4，以应对不同的精度需求以及硬件

条件限制。其中，EfficientDetLite4 模型大小约为 EfficientDetLite0 的 5 倍，推理运算速度相比约慢 7 倍，但具有更好的精度。两个模型在目标检测精度指标上的区别见图 12。

3.2 移动端多标签分类模型的构建

移动端最经典的分类模型是 MobileNet 系列^[15-17]。MobileNetV3 对于超参数的选择十分敏感，为了实现一个模型中同时预测 3 个性状，并且减小模型的大小，本文优化了策略并对原有设计进行了改进。

为了补足训练数据的缺失，首先，本文选用伪标签生成方法^[18]。为了得到高质量的伪标签，则选择使用泛用性较高的 ResNet50V2 架构^[6]，对于每一个性状分类任务分别建模并训练，再充分调优，即

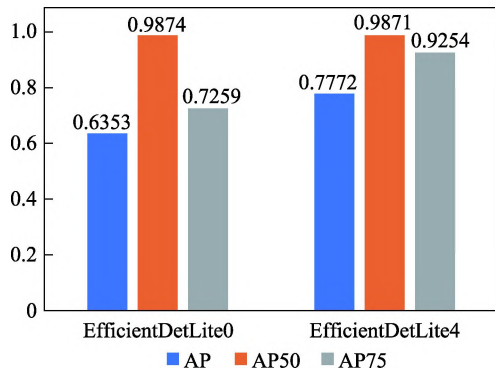


图 12 不同规模目标检测模型的精度对比
Fig. 12 Performance of models with different scale

等模型达到很高的准确率后再生成伪标签。

RandAugment^[19]是一种实用的减少模型训练时图像增强超参数搜索空间的方法。在原始方法中, RnadAugment 会在一个固定的图像增强方法集合中, 随机选取 N 个增强方法, 并以强度 M 对图像增强, 整体表示为 RandAugment(N, M)。在超参数调优方面, 本文对该方法进行了改进。在前期实验中, 我们发现增强方法数量的多样性也有助于模型训练, 于是改为对于每张图片, 都在(N/2, N)的范围内随机取一个数值 k 作为选取的方法数, 并向方法集合中增加了以下项目(<https://github.com/aleju/imgaug>):

(1)卷积操作, 包括有方向的边缘检测及不同方式的模糊化等;

(2)噪声, 包括覆盖式噪声、加乘式噪声以及区域覆盖等;

(3)Jpeg 图像压缩, 用来应对实际情况中微信对图片的压缩等。

此外, 还运用 EfficientNetV2 中提出的渐进式学习 (<https://www.fast.ai/2018/04/30/dawnbench-fastai>) 与适应性正则化策略^[20]来加速训练过程, 并起到一定的正则化效果。最后通过超参数搜索以及前期实验的经验, 最终设置了如下超参数, 可以很好的稳定训练过程:

(1)渐进式学习分 80 阶段, 每 5 次迭代进入下一阶段。在渐进过程中, 线性地进行如下操作: 初始设置为 ImprovedRandAug(N=6, M=60), 最终为 ImprovedRandAug(N=8, M=100); 输入图像大小初始设置为 64×64, 最终为 224×224;

(2)SGD(lr=0.1), 每 5 次迭代, 学习率降为原来的 0.97;

(3)Dropout(ratio=0.5), 在所有分类器前加入^[21]。典型的训练指标趋势图见图 13。

本文选择了两个不同大小的 MobileNetV3 进行训练, 以适应不同的硬件条件与准确率需求。其中, MobileNetV3Large 的模型大小约为 MobileNetV3Small 的 3 倍。不同模型在各个性状分类任务下的准确率比较见图 14。

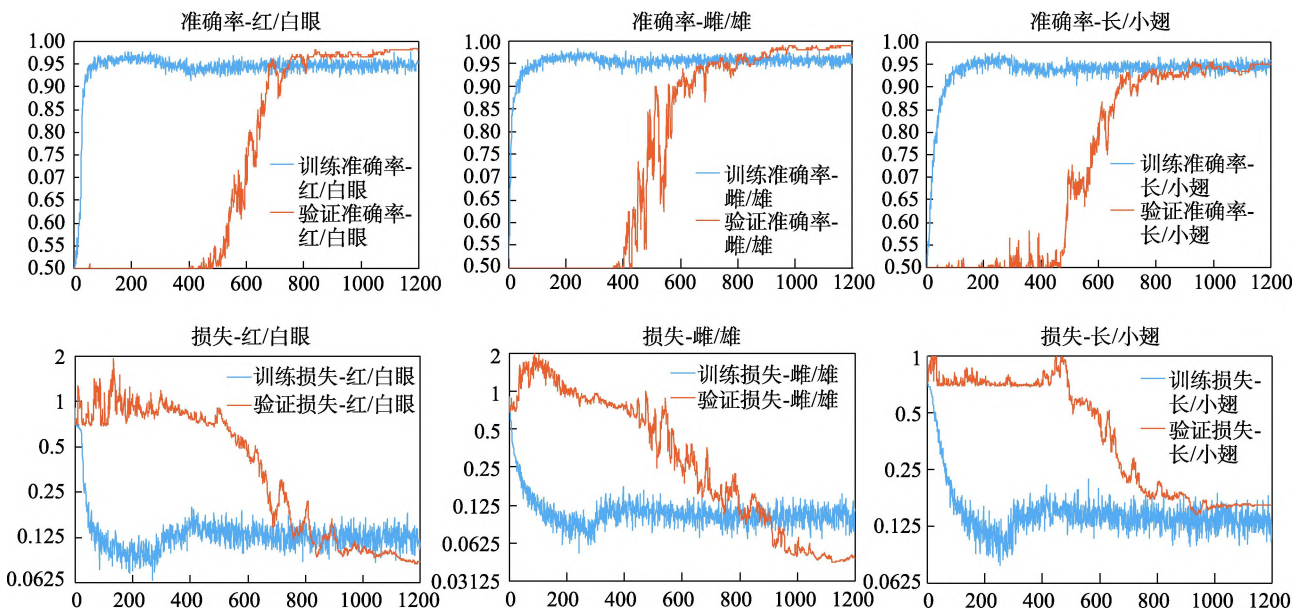


图 13 MobileNetV3 典型训练趋势图
Fig. 13 Typical training session of MoblieNetV3

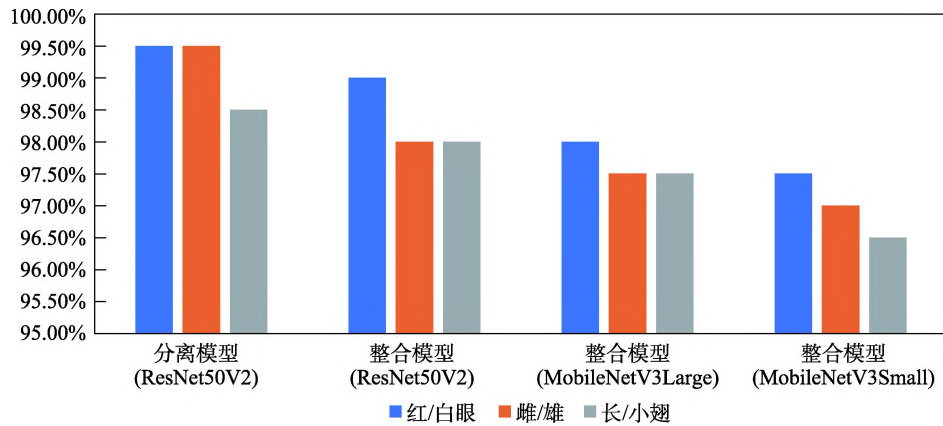


图 14 不同模型的准确率比较

Fig. 14 Accuracy of different models

3.3 训练后模型的量子化

为了让模型能够在 CPU 上运行更快,并且占据存储空间更小,本文比对了多种量子化方式^[22-24]。对于目标检测模型,采用了动态量子化,可以生成最小的模型并最大限度保留模型精度。对于分类模型,采用了浮点优化法,可以得到最小的模型。最终可以用于安卓端的几个模型的大小比较见图 15。

3.4 模型在手机安卓端的测试效果

通过上述条件的不断优化后,最终选择目标检测模型 EfficientDetLite0 和分类模型 MobileNetV3Small 写入 Metadata,并最终生成 Tflite 模型^[25],在安卓端进行测速。其中,目标检测模型 EfficientDetLite0 可以在 37 ms 内处理一张图片,分类模型 MobileNetV3Small 可以在 7 ms 内处理一张图片。在典型实验应用情况下,约需要处理 8 张原

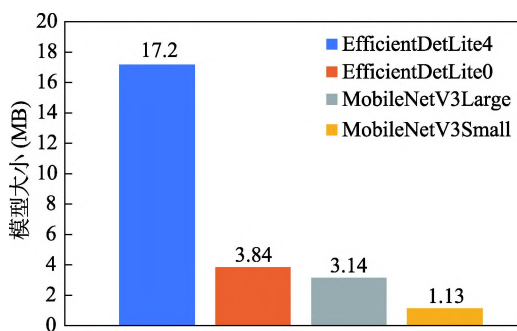


图 15 量子化后模型大小的比较

Fig. 15 Model size after quantization

始图片以及 200 张果蝇图片,则总计模型推理时间约 1.7 s。在实际情况中,还需要考虑硬盘读写速度以及手机内存的限制等,预估耗时共计约 5 s。

为了达到最好的 leave out 测试效果,另外选取一批志愿者学生 8 人采集的果蝇图片进行测试。最终在测试集上,使目标检测模型的召回率与精确率均为 100%(使用 EfficientDetLite4),分类模型在长/小翅上的分类准确率为 93.9%,在红/白眼与性别上的分类准确率均为 98.8%(使用 MobileNetV3Small)。

4 结语与思考

随着技术的发展,现代计算机应用技术为教师的教学和学生的学习都带来了全新的体验。在遗传学实验开展过程中,本文将现代计算机技术融入到传统的经典果蝇杂交实验中,大大提高了该实验关于果蝇性状识别的准确率,解决了繁琐的手工重复劳作实验,提高了学生的兴趣,也开拓了学生的思维。

在完成了数据集的清洗以及完善后,通过关键点辅助分类、伪标签生成以及优化 RandAugment 等策略,建立了分类模型,并通过量子化方法实现了安卓端的模型部署,最终构建了移动端轻量级高精度果蝇遗传性状批量识别系统。该系统安装软件较小,可在任意安卓系统的手机上安装,适于推广使用。

该系统的开发有利于推进以果蝇为研究对象的遗传规律验证等实验的教学,也可用于涉及大量果蝇分类统计分析的科研工作。

参考文献(References):

- [1] Chen DF, Lu DR, Zhang FX, Zhang GF. The development of genetics teaching in China in the last four decades and its future prospect. *Hereditas(Beijing)*, 2018, 40(10): 916–923.
陈德富, 卢大儒, 张飞雄, 张根发. 中国遗传学教学 40 年发展及展望. *遗传*, 2018, 40(10): 916–923.
- [2] 乔守怡, 江绍慧. 遗传学实验—果蝇实验. *遗传*, 1981, 3(2): 40–44.
- [3] Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z. Rethinking the inception architecture for computer vision. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, 2818–2826.
- [4] Chollet F. Xception: deep learning with depthwise separable convolutions. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, 1251–1258.
- [5] He KM, Zhang XY, Ren SQ, Sun J. Deep residual learning for image recognition. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, 770–778.
- [6] He KM, Zhang XY, Ren SQ, Sun J. Identity mappings in deep residual networks. In: *European Conference on Computer Vision*. 2016, 630–645. Springer, Cham.
- [7] Duan KW, Bai S, Xie LX, Qi HG, Huang QM, Tian Q. Centernet: Keypoint triplets for object detection. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019, 6569–6578.
- [8] Law H, Deng J. Cornernet: detecting objects as paired keypoints. In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018, 734–750.
- [9] Newell A, Yang KY, Deng J. Stacked hourglass networks for human pose estimation. In: *European Conference on Computer Vision*. 2016, 483–499. Springer, Cham.
- [10] Xiao B, Wu HP, Wei YC. Simple baselines for human pose estimation and tracking. In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018, 466–481.
- [11] Yu F, Wang DQ, Shelhamer E, Darrell T. Deep layer aggregation. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, 2403–2412.
- [12] Toshev A, Szegedy C. Deeppose: human pose estimation via deep neural networks. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2014, 1653–1660.
- [13] Selvaraju RR, Cogswell M, Das A, Vedantam R, Parikh D, Batra D. Grad-cam: visual explanations from deep networks via gradient-based localization. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2017, 618–626.
- [14] Tan MX, Pang RM, Le QV. Efficientdet: scalable and efficient object detection. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020, 10781–10790.
- [15] Howard AG, Zhu ML, Chen B, Kalenichenko D, Wang WJ, Weyand T, Andreetto M, Adam H. Mobilenets: efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*. 2017.
- [16] Sandler M, Howard A, Zhu ML, Zhmoginov A, Chen LC. Mobilenetv2: inverted residuals and linear bottlenecks. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, 4510–4520.
- [17] Howard A, Sandler M, Chu G, Chen LC, Chen B, Tan MX, Wang WJ, Zhu YK, Pang RM, Vasudevan V, Le QV, Adam H. Searching for mobilenetv3. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019, 1314–1324.
- [18] Lee DH. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In: *Workshop on Challenges in Representation Learning, ICML*. 2013, 3(2): 896–896.
- [19] Cubuk ED, Zoph B, Shlens J, Le QV. Randaugment: practical automated data augmentation with a reduced search space. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*. 2020, 702–703.
- [20] Tan MX, Le QV. Efficientnetv2: smaller models and faster training. In: *International Conference on Machine Learning*. 2021, 10096–10106. PMLR.
- [21] Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. *Commun ACM*, 2017, 60(6): 84–90.
- [22] Guo YH. A survey on methods and theories of quantized neural networks. *arXiv preprint arXiv:1808.04752*. 2018.
- [23] Gholami A, Kim S, Dong Z, Yao ZW, Mahoney MW, Keutzer K. A survey of quantization methods for efficient neural network inference. *arXiv preprint arXiv:2103.13630*. 2021.
- [24] Deng L, Li GQ, Han S, Shi LP, Xie Y. Model compression and hardware acceleration for neural networks: a comprehensive survey. *Proc IEEE*, 2020, 108(4): 485–532.
- [25] Abadi M, Barham P, Chen JM, Chen ZF, Davis A, Dean J, Devin M, Ghemawat S, Irving G, Isard M, Kudlur M. TensorFlow: a system for large-scale machine learning. In: *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*. 2016, 265–283.