计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2019,28(7):240−245 [doi: 10.15888/j.cnki.csa.006955]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于 Cascade 结构的牛脸姿态估计①

苟先太, 黄　巍, 刘琪芬

(西南交通大学 电气工程学院, 成都 611756)
通讯作者: 黄　巍, E-mail: huangwei_9521@163.com

摘　要: 随着牲畜牛逐年增加, 单角度特征编码这种身份认证方法在数据容量上已无法满足目前的需求. 本文给出一种 cascade 结构, 使用 SSD 模型对牛脸进行检测, 之后使用 MobileNet 对牛脸姿态角度进行估计. 为多角度特征编码打下坚实的特征基础. 实验表明, cascade 结构在牛脸检测、姿态角度估计任务中均能得到一个较高的准确度.

关键词: cascade; 牛脸检测; 姿态角度

引用格式: 苟先太,黄巍,刘琪芬.基于 Cascade 结构的牛脸姿态估计.计算机系统应用,2019,28(7):240–245. http://www.c-s-a.org.cn/1003-3254/6955.html

Gesture Estimation of Cattle Face Based on Cascade Structure

GOU Xian-Tai, HUANG Wei, LIU Qi-Fen

(School of Electrical Engineering, Southwest Jiaotong University, Chengdu 611756, China)

Abstract: The single-angle feature coding identity authentication method cannot meet the current demand in terms of data capacity because of the increasing number of cattle. A cascade structure is used to detect the cattle's face and then estimate the angle of the cattle's face, which build a solid feature base for multi-angle feature coding. The result of experiments shows that the cascade structure can obtain a higher accuracy in both the face detection and attitude angle estimation tasks.

Key words: cascade; cattle face detection; gesture angle

引言

随着我国畜牧业的发展, 作为畜牧业的基本构成成员, 牛的数量也在急剧增加. 急剧增加的数量使牛的溯源、认责、投保理赔等问题变得更为严峻. 牲畜身份认证是解决该一系列问题的主要方法. 通过牲畜身份认证, 相关部门可以精确地追踪市场上牛肉制品的来源, 这对于保证我国牛肉制品的安全性是至关重要的. 但逐年增加的数量也为牲畜牛的身份认证提出了挑战. 如何找到一种成本低、稳定性高、大容量的身份认证方式是目前我国牲畜牛管理所面临的主要困难.

传统的身份认证方法包括耳标、激光打标、芯片植入等, 但因为其稳定性不高、易损毁、成本高, 目前并没有得到广泛的使用. 目前常使用的是单角度特征编码的方法, 该方法具有稳定性高、成本低等优点. 但随着牲畜牛的数量逐渐增加, 单角度的方法因为不能采集到牛脸的全部特征, 所以在容量上无法满足现在的需求. 为了提升容量, 目前常用的解决方案为多角度特征编码, 通过采集牛脸各个角度的特征进行编码. 为了保证牛脸的各个角度的特征均得到采集, 现在迫切需要得到采集图片中牛脸的姿态角度 (即 α, β, γ 三个角度, 分别表征牛脸沿着 x 轴、y 轴以及 z 轴的旋转角度).

目前国内牛脸姿态估计研究尚属空白, 而为了保

① 基金项目: 四川省重大科技专项 (18ZDZX0162); 四川省重点研发项目 (2017GZ0159)
Foundation item: Science and Technology Major Program of Sichuan Province (18ZDZX0162); Major Science and Technology Research and Development Plan of Sichuan Province (2017GZ0159)
收稿时间: 2019-01-21; 修改时间: 2019-02-21; 采用时间: 2019-03-04; csa 在线出版时间: 2019-07-01

证多角度特征编码的顺利实施, 需要得到各个角度的牛脸特征, 所以获得精度高的牛脸姿态角度非常重要. 本文对牛脸姿态估计进行了相关研究, 提出一种 cascade 结构, 首先使用 SSD[1]模型将输入图中的所有牛脸检测出来, 按照预测框截图之后, 使用 MobileNet[2]对截图中的牛脸进行姿态角度估计. 从实验效果来看, 检测精度以及角度估计平均误差, 均在一个较好的水平.

1 cascade 结构组成及原理

cascade 结构流程如图 1 所示.

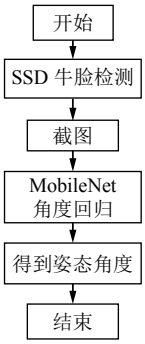

图 1 cascade 结构主流程图

1.1 牲畜脸部检测

传统方法对牲畜脸部检测通常基于 LBP[3]、SIFT[4]、HOG[5,6]等特征提取算法. 在文献[7]中, 采用集成学习方法 Adaboost[8]对牛脸进行检测并取得了较好的效果, 但 Adaboost 最终效果依赖于弱分类器的选择, 同时算法对离群点较为敏感.

本文拟采用深度学习的方式对牛脸进行检测. 在目标检测领域中, 通常分为 two-stage 和 one-stage 两种方案, two-stage 方案主要包括 RCNN[9], Fast-RCNN[10], Faster-RCNN[11]等方法, 其主要特点是先进行区域建议 (regional proposal, RP), 再进行边界框回归 (bounding box regression); one-stage 方案主要包括 SSD(Single Shot MultiBox Detector)、YOLO(You Only Look Once)[12]等, 其特点是没有进行区域建议, 而是直接在特征图上划分区域与边界框回归. 所以速度相比于 two-stage 方案更快, 同时可以保持较高的精度. 相比于 YOLO, SSD 因为采用多尺度特征图预测, 在小物体检测、精度上都略优于 YOLO. 本文采用 SSD 作为牛脸检测模型, 同时对模型进行优化.

1.1.1 SSD 网络结构

SSD 网络结构如图 2 所示, 模型基于 VGG-16, 将最后两层全连接层变为卷积层, 之后添加 FCN 网络进行特征提取. 分别抽取 Conv4_3、Conv7、Conv8_2、Conv9_2、Conv10_2、Conv11_2 层的 feature map 进行多尺度特征提取.

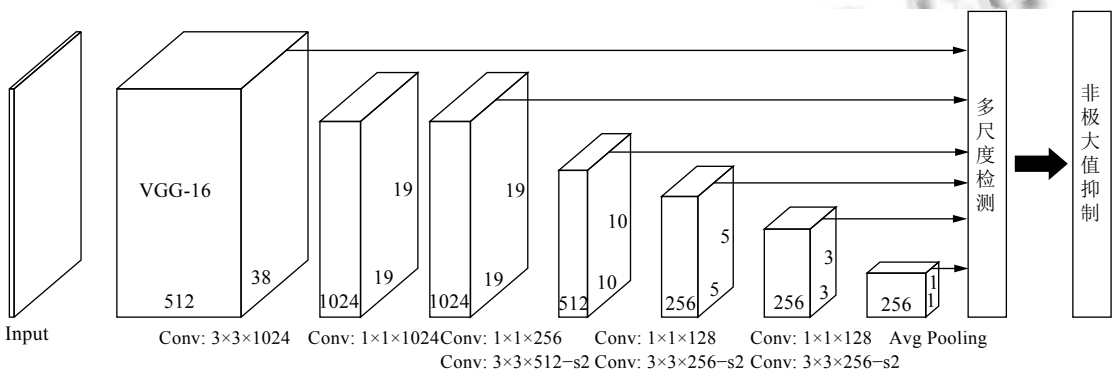

图 2 SSD 网络结构

因为不同大小的 feature map 对应的感受野不同, 对应原图中物体的大小也不同, 所以 SSD 对不同大小的物体都有较高准确率.

1.1.2 损失函数

SSD 的损失函数由两部分组成: 分类置信度 L_{conf} 和坐标误差 L_{loc}.

$$\begin{cases} L(x,c,l,g)=\dfrac{1}{N}\left(L_{conf}(x,c)+aL_{loc}(x,l,g)\right) \\ L(x,c,l,g)=\dfrac{1}{N}\left(L_{conf}(x,c)+\alpha L_{loc}(x,l,g)\right) \\ L(x,c,l,g)=\dfrac{1}{N}\left(L_{conf}(x,c)+aL_{loc}(x,l,g)\right) \end{cases} \quad (1)$$

式中, N 是默认框 (default box) 的个数. 如果 $N=0$, 将

loss 置为 0.

其中分类置信度的损失函数使用的是 softmax loss.

$$L_{conf}(x,c) = -\sum_{i\in Pos}^{N} x_{ij}^{p}\log\left(\hat{c}_i^p\right) - \sum_{i\in Neg}\log\left(\hat{c}_i^0\right) \quad where\ \hat{c}_i^p = \frac{\exp\left(c_i^p\right)}{\sum_p \exp\left(c_i^p\right)} \tag{2}$$

坐标误差损失函数使用的是 Smooth L1 loss:

$$\begin{cases} L_{loc}(x,l,g) = \sum_{i\in Pos}^{N}\sum_{m\in\{cx,cy,w,h\}} x_{ij}^{k}\,\text{smooth}_{L1}\left(l_i^m - \hat{g}_j^m\right) \\ \hat{g}_j^{cx} = \left(g_j^{cx} - d_i^{cx}\right)/d_i^w \quad \hat{g}_j^{cy} = \left(g_j^{cy} - d_i^{cy}\right)/d_i^h \\ \hat{g}_j^w = \log\left(\frac{g_j^w}{d_i^w}\right) \quad \hat{g}_j^h = \log\left(\frac{g_j^h}{d_i^h}\right) \\ where\ \text{smooth}_{L1}(x) = \begin{cases} 0.5x^2 & if\ |x|<1 \\ |x|-0.5 & otherwise \end{cases} \end{cases} \tag{3}$$

式中, g代表 ground truth, l代表预测框 (predicted box), cx、cy、w、h 分别对应默认框的中心、宽和高. 由上式可以看出, SSD 不是直接回归的边框坐标, 而是从默认框到 ground truth 的偏移以及变换.

1.1.3 困难负样本挖掘

因为 SSD 得到负样本的数量远远多于正样本的数量, 如果随机抽取样本进行训练的话, 网络会过于重视负样本, 这样会使 loss 不稳定. 所以需要将正样本和负样本的数量进行平衡. 同时, 过于简单的负样本对于整个模型的训练几乎没有帮助, 所以需要选取一些易错的、分类较为困难的负样本进行训练. 常用的方法就是困难负样本挖掘, 将正负样本的比例控制在 1:3 左右, 这样模型会取得更好的泛化效果.

1.2 牛脸姿态估计

为了保证牲畜脸部各个角度的特征都可以得到采集, 对于每一个检测出的牛脸, 我们都需要截图并对截图进行过滤, 保证特征采集的图片质量. 同时, 较好的图片质量也能有效降低脸部姿态回归的误差. 通常脸部姿态是使用空间直角坐标系的三个参数进行描述 (即 α, β, γ 三个角度, 分别表征牛脸沿着 x 轴、y 轴以及 z 轴的旋转角度).

考虑到 cascade 模型速度问题, 本文使用轻量级结构 MobileNet 作为角度预测模型, MobileNet 使用深度可分离卷积的方法大量降低了模型参数与计算量, 同时依然可以保持一个较高的精度.

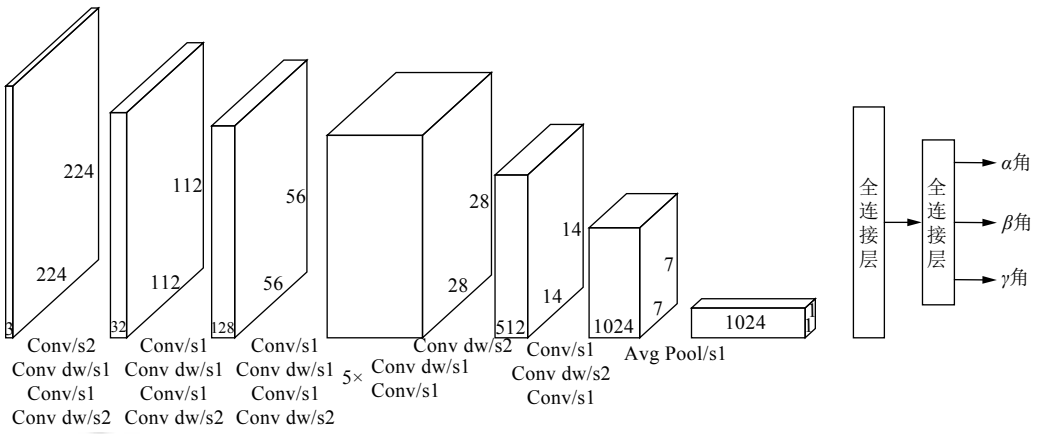

图 3 MobileNet 网络模型

MobileNet 的网络结构如图 3 所示, 使用了大量的 1×1 卷积与深度可分离卷积, 减少了大量参数.

1.2.1 深度可分离卷积

深度可分离卷积和传统的卷积区别在于传统的卷积核会对每一个通道进行卷积, 而深度可分离卷积仅仅是针对于某一个通道, 之后使用 1×1 的卷积进行特征融合. 如图 4 所示.

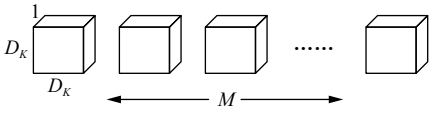

图 4 深度可分离卷积

其中, D_K是深度可分离卷积核的尺寸, M为输入特

征图的通道数, N为输出特征图的通道数. 设输入特征图的大小为$D_F \times D_F$, 可分离卷积的计算量为:

$$D_k \times D_k \times M \times D_F \times D_F + N \times M \times D_F \times D_F \quad (4)$$

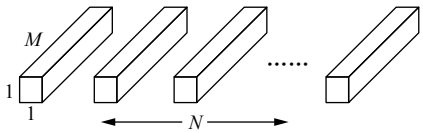

图 5 1×1 卷积

相对的, 传统卷积的计算量为:

$$D_k \times D_k \times N \times M \times D_F \times D_F \quad (5)$$

所以深度可分离卷积相比于传统卷积的计算量为:

$$\frac{D_k \times D_k \times M \times D_F \times D_F + N \times M \times D_F \times D_F}{D_k \times D_k \times M \times N \times D_F \times D_F} = \frac{1}{N} + \frac{1}{D_k^2} \quad (6)$$

由上式可以看出, 深度可分离卷积可以大大减少模型参数与计算量.

在一些对运行速度或者计算机内存有极端要求的场合, 还可以通过调整模型的宽度因子与分辨率因子达到减少模型参数、降低计算量的目的.

宽度因子 α 属于 (0, 1], 附加于网络的通道数, 意义是新网络中每一个模块使用的卷积核数相对于标准 MobileNet 的比例; 分辨率因子同样属于 (0, 1], 附加于每一个模块的输入, 意义是新模型的输入大小相对于标准的 MobileNet 的比例. 结合宽度因子 α 和分辨率因子 β, MobileNet 的计算量为:

$$D_k \times D_k \times \alpha M \times \beta D_F \times \beta D_F + \alpha N \times \alpha M \times \beta D_F \times \beta D_F \quad (7)$$

由式 (7) 可知, 通过调整宽度因子和分辨率因子可以进一步减少 MobileNet 的参数量与计算量.

1.2.2 损失函数

因为模型做的是回归任务, 所以使用回归任务常用的 Huber loss, 公式如下.

$$L_\delta(y, f(x)) = \begin{cases} \frac{1}{2}(y - f(x))^2 if |y - f(x)| \le \delta \\ \delta \cdot \left(|y - f(x)| - \frac{1}{2}\delta\right), \text{otherwise} \end{cases}$$

Huber loss 相较于传统的 L2 loss 有更强的鲁棒性, 当残差 (residual) 很小的时候, loss 函数为 L2 范数, 残差大的时候, 为 L1 范数的线性函数, 所以 Huber loss 对于离群点不敏感, 不易发生梯度爆炸的问题. 同时, 超参δ可以对 Huber loss 的函数曲线进行调整, 使之更适合模型的训练.

2 实验结果与分析

2.1 训练集与测试集

实验以牛脸为实验对象, 使用 ImageNet[13]中标签为 cattle 的图片、PASCAL VOC 2012 数据集上标签为 cow 的图片以及 google 中收集的牛脸图片, 选择大约 5000 张图片使用二维标注软件 labelImg 进行标注. 训练集和测试集按照 9:1 的比例进行划分, 该训练集和测试集用于 SSD 模型的训练与测试. 5000 张标注完成的图片, 按照标注得到的 ground truth 进行截图, 得到的截图使用三维标注软件 blender 获得牛脸的三个姿态角度 (α, β, γ). 得到的图片随机打乱之后也按照 9:1 的比例进行划分, 分别用于 MobileNet 的训练和测试.

2.2 实验平台

本文实验的开发平台为 Ubuntu 18, 基于 tensorflow[14]框架对模型进行构建、训练与测试. tensorflow 是目前使用做多、最广的深度学习架构. 实验具体平台配置如表 1 所示.

表 1 实验平台配置

名称	具体配置
CPU	Intel i7 6700k, 4.0 GHz
内存	16 GB
显卡 GPU	NVIDIA GeForce 1080Ti, 11 GB
GPU 加速	CUDA 9.2, cuDNN v7.3.1
深度学习框架	tensorflow 1.11.0

2.3 训练

在具体参数设置上, SSD 模型的优化器采用 RMSProp[15], 和其它优化器相比, RMSProp 可以对学习率进行自适应衰减, 故只需要对模型的初始学习率进行设置即可, 设置为 0.001. 在数据增广方面, SSD 采用了随机裁剪、随机 padding 以及色彩扭曲等方式. 考虑到模型的最终性能, 训练过程中尝试使用 Focal loss 与更大的输入尺寸对模型进行训练. 最终模型的输入尺寸采用的是 300×300, Focal loss 对模型效果提升不大, 所以没有采用 Focal loss.

和 SSD 相似, MobileNet 也采用 RMSprop 作为优化器, 考虑到 cascade 模型的速度, 适当降低 MobileNet 的宽度因子 α 以及分辨率因子 β, 测试最终的模型效果. 最终模型采用经典 MobileNet(即 α=1 且

β=1). 模型输入大小设置为 224×224, 因为 MobileNet 做的是回归问题, 所以模型最后一层由 softmax 改为全连接层, 最终输出为三个角度, 分别对应预测的 α、β 以及 γ 值. 损失函数也换成做回归任务的 Huber loss.

2.4 SSD 模型试验结果与分析

本文使用 ImageNet 中标签为“cattle”的图片按照 9:1 的比例划分为训练集与测试集, 对 SSD 模型进行训练与测试. 为更好地说明模型效果, 本文的评价指标为精确率 (Precision, P), 召回率 (Recall, R) 以及 F 值 (F-score, F) 进行, 计算公式如下所示.

$$P=\frac{\text{预测正确的牛脸框数}}{\text{预测的牛脸框数}} \tag{8}$$

$$R=\frac{\text{预测正确牛脸框数}}{\text{真实牛脸框数}} \tag{9}$$

$$F=\frac{2\cdot P\cdot R}{P+R}$$

为了加强实验对比性, 本文通过设置是否使用 Focal loss, 分类阈值等参数, 训练多个模型, 使用同一个测试集进行测试, 实验结果和检测效果分别如表 2、图 6 所示.

表 2 不同 SSD 模型检测结果

300×300	Focal loss	阈值		准确率 (%)	召回率 (%)	F 值 (%)
		0.5	0.8			
√			√	92.7	85.8	89.1
√	√		√	91.0	86.6	88.7
√		√		89.3	87.3	88.3
√	√	√		88.9	88.4	88.6

图 6 SSD 检测效果图

从实验结果来看, Focal loss 对模型整体效果没有帮助, 甚至略低于没有使用 Focal loss 的模型, 原因可能是实验的任务仅仅是牛脸识别, 相对于其它的多分类任务更加简单, Focal loss 不能起到太大的作用. 调整阈值可以对准确率以及召回率进行调整.

2.5 MobileNet 角度预测实验结果与分析

MobileNet 的训练数据为 SSD 训练数据按照 groundtruth 进行切分, 切下来的牛脸使用 blender 软件进行标注, 获得牛脸的 x, y, z 三个角度. 为增加实验的对比度, 本文使用不同的宽度因子 α 以及分辨率因子 β. 可以评价指标为预测得到的 x, y, z 三个角度与其对应 ground truth 的平均误差. 实验结果如下:

表 3 不同 MobileNet 模型回归误差

宽度因子	分辨率因子	平均误差/单位: 角度		
		α 角	β 角	γ 角
1.0	1.0	9.73	9.24	10.72
0.8	1.0	15.32	14.73	17.09
1.0	0.8	16.01	15.28	20.43
0.5	0.5	33.65	37.74	40.21

由实验结果可知, 宽度因子 α 为 1.0 与分辨率因子为 1.0 时 (即为标准的 MobileNet), 误差最小. 减小宽度因子和分辨率因子会不同程度上影响模型的效果, 角度平均误差变大. 当宽度因子 α 以及分辨率因子 β 为 0.5 时, 误差最大. 分析原因虽然宽度因子和分辨率因子减少的模型参数, 降低了计算量. 但参数的减少会影响模型的特征表征能力, 会使模型精度降低. 目前模型的主要误差来自于标注误差, 因为训练样本是人为标注的, 会存在 7 度左右的误差. 考虑到模型精度和速度的变化关系, 本文选用标准的 MobileNet 作为角度回归模型.

3 结论与展望

本文基于 SSD 与 MobileNet 提出一种 cascade 结构, 并在原来的标准网络上进行微调, 通过不同微调模型间的对比, 选择最优的模型构建 cascade 结构, 完成了牛脸检测与牛脸姿态估计两个任务, 效果显著. 其中 SSD 检测模型的准确率和召回率均可以控制在一个较好的水平. 使用 MobileNet 进行角度回归, 角度平均误差可以达到 9 度左右, 完全达到使用要求.

目前, cascade 模型可以很好地完成脸部检测与角度回归, 同时考虑到 cascade 模型速度, 检测模型与回归模型均选择的是较快的 one-stage 模型 (SSD) 以及轻量级模型 (MobileNet), 但因为 cascade 结构的限制, 在模型速度上还有进一步提升的空间. 接下来将考虑对模型进行改进, 使用一个模型而不是 cascade 结构完

成牛脸检测以及姿态角度回归.

参考文献

1 Liu W, Anguelov D, Erhan D, *et al*. SSD: Single shot multibox detector. Proceedings of the 14th European Conference on Computer Vision. Amsterdam, The Netherlands. 2016. 21–37.

2 Howard AG, Zhu ML, Chen B, *et al*. MobileNets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv: 1704.04861, 2017.

3 Wang XY, Han TX, Yan SC. An HOG-LBP human detector with partial occlusion handling. 2009 IEEE 12th International Conference on Computer Vision. Kyoto, Japan. 2009. 32-39.

4 Bicego M, Lagorio A, Grosso E, *et al*. On the use of SIFT features for face authentication. Conference on Computer Vision and Pattern Recognition Workshop. New York, NY, USA. 2006. 35-35.

5 Zhu Q, Yeh MC, Cheng KT, *et al*. Fast human detection using a cascade of histograms of oriented gradients. 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. New York, NY, USA. 2006. 1491–1498.

6 Cao XB, Wu CX, Yan PK, *et al*. Linear SVM classification using boosting HOG features for vehicle detection in low-altitude airborne videos. 201118th IEEE International Conference on Image Processing. Brussels, Belgium. 2011. 2421–2424.

7 蔡骋, 宋肖肖, 何进荣. 基于计算机视觉的牛脸轮廓提取算法及实现. 农业工程学报, 2017, 33(11): 171–177. [doi: 10.11975/j.issn.1002-6819.2017.11.022]

8 Zhu J, Rosset S, Zou H, *et al*. Multi-class AdaBoost. Statistics and Its Interface, 2006, 2(3): 349–360.

9 Girshick R, Donahue J, Darrell T, *et al*. Rich feature hierarchies for accurate object detection and semantic segmentation. arXiv preprint arXiv: 1311.2524, 2013.

10 Girshick R. Fast R-CNN. arXiv preprint arXiv: 1504.08083, 2015.

11 Ren SQ, He KM, Girshick R, *et al*. Faster R-CNN: Towards real-time object detection with region proposal networks. Proceedings of 28th International Conference on Neural Information Processing Systems. Montreal, Canada. 2015. 91-99.

12 Redmon J, Divvala S, Girshick R, *et al*. You only look once: Unified, real-time object detection. 2016 IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, NV, USA. 2016. 779-788.

13 Deng J, Dong W, Socher R, *et al*. Imagenet: A large-scale hierarchical image database. IEEE Conference on Computer Vision and Pattern Recognition. Miami, FL, USA. 2009. 248-255.

14 Abadi M, Barham P, Chen JM, *et al*. Tensorflow: A system for large-scale machine learning. arXiv preprint arXiv: 1605.08695, 2016.

15 Tieleman T, Hinton G. Lecture 6. 5-rmsprop: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural Networks for Machine Learning, 2012, 4(2): 26–30.