华东理工大学学报(自然科学版)
Journal of East China University of Science and Technology(Natural Science Edition)
ISSN 1006-3080,CN 31-1691/TQ

《华东理工大学学报(自然科学版)》网络首发论文

题目：基于深度学习的驾驶场景关键目标检测与提取
作者：张雪芹，魏一凡
DOI：10.14135/j.cnki.1006-3080.20181023002
收稿日期：2018-10-23
网络首发日期：2019-01-04
引用格式：张雪芹，魏一凡．基于深度学习的驾驶场景关键目标检测与提取[J/OL]．华东理工大学学报(自然科学版).
https://doi.org/10.14135/j.cnki.1006-3080.20181023002

华 东 理 工 大 学 学 报 （自 然 科 学 版）
Journal of East China University of Science and Technology (Natural Science Edition)

DOI：10.14135/j.cnki.1006-3080.20181023002

基于深度学习的驾驶场景关键目标检测与提取

张雪芹，魏一凡
（华东理工大学信息科学与工程学院，上海 200237）

摘要：包含目标识别与边界框选定的目标检测是无人驾驶视觉感知中的关键技术之一。采用基于深度计算机视觉组网络（VGGNet）的新型单次多框检测算法（SSD）进行驾驶环境中的关键目标检测、语义标注和目标框选；同时，针对具体驾驶场景，提出了改进的 SSD_ARS 算法。通过优化梯度更新算法、学习率下降策略和先验框生成策略，在提高平均检测精度的同时使得小目标类别的检测精度得到明显提升。在实际驾驶场景中 9 类关键目标的检测实验上验证了本文算法的有效性，实验结果表明，检测速度满足实时检测需求。

关键词：目标检测；VGG 网络；SSD；驾驶场景
中图分类号：TP391.4　　**文献标志码**：A

Deep Learning Based Key Object Detection and Extraction for Driving Scene

ZHANG Xueqin，*WEI Yifan*
（*School of Information Science and Engineering*，*East China University of Science and Technology*，*Shanghai* 200237, *China*）

Abstract: Object detection with object recognition and bounding box generation is important in visual perception of autonomous driving. With the rapid development of deep learning, a series of object detection algorithms based on deep convolution networks have been proposed. These kinds of vision-based algorithms can be widely used when dealing with driving scenes or other complicated situations. In this paper, the novel detection algorithm-single shot multi-box detector（SSD）based on deep convolution network-visual geometry group network（VGG）is been applied to target recognition, semantic annotation and bounding box selection in driving scene. At the same time, an improved algorithm is proposed for object detection in auto-driving called single shot multi-box detector with aspect ratio selection strategy（SSD_ARS）. Momentum optimization has been used in the gradient descent algorithm. And the learning rate reduction strategy get optimized. Further, the aspect ratio selection strategy is improved when generating the priori box. Results on different selection strategies of aspect ratio when generating the priori box shows that the accuracy of object detection is improved. Especially the detection accuracy of small objects is improved. The feasibility and effectiveness of the algorithm is verified by the detection experiments of 9 significant object classes in the real driving scene. Object detection result at different distances in the driving scene shows that the algorithm performs well at near and medium distances. Experiment on driving video demonstrates that the detection speed of the algorithm meets the requirements of real-time detection.

Key words: object detection；VGG network；SSD；driving scene

近年来，辅助驾驶技术和无人驾驶技术受到广泛关注，辅助驾驶技术与无人驾驶的感知模块前端主要依赖于激光雷达和机器视觉。应用于监控视频或者其他静态检测场景的传统的目标检测技术无法应用于运动性较强的场景，近几年，在处理驾驶视野场景时，基于深度学习的驾驶场景目标检测成为研究热点。

Sermanet 等[1] 将多尺度卷积神经网络应用于交通标志分类任务中，取得了较高的准确率。Chen 等[2]提出了一种单目标视觉下的 3D 目标对象检测

收稿日期：2018-10-23
基金项目：国家自然科学基金资助项目（31671006）
作者简介：张雪芹（1972−），女，副教授，博士，主要从事模式识别研究。E-mail:zxq@ecust.edu.cn

方法，该方法生成一组候选特定类的对象区域提议框，借助能量最小化方法，基于 CNN 管道实现对象检测。Ayúegül 等[3]设计了一个 10 层的 CNN 结构和一个 9 层的类似 AlexNet 的结构用于自动驾驶下的目标识别。Chen 等[4]提出了一种新的多任务学习（MTL）方法，用基于笛卡尔积的多任务组合策略对目标检测和距离预测联合建模，实现了自动驾驶中的危险目标检测。许明文[5]研究了基于无人驾驶平台，设计了基于 CNN 特征和 HOG 特征的 SVM 分类器用于交通灯及数字检测与识别。Tian[6]通过添加语义信息，提出了任务辅助卷积神经网络（Task-Assistant Convolutional Neural Network，TA-CNN）结构，提高行人的检测准确率与效率。葛园园等[7]提出用浅层 VGG16 网络作为物体检测框架 R-FCN 的主体网络，并改进了 VGG16 网络，实现自动驾驶场景下的交通标志检测。

上述研究大都针对行人、交通标志或障碍物等单目标的检测，缺乏通用性的关键目标识别框架，而且在目标识别的同时，也无法实现精确的目标自动框选。本文基于 Liu 等[8]提出的 SSD（Single Shot Multi-Box Detector）检测算法，针对驾驶场景下的关键目标检测任务，提出了一种改进的 SSD_ARC 算法，能够实现快速多目标识别、语义标注和定位框选。

1 SSD检测算法

1.1 SSD网络结构

SSD 本质上是利用密集采样的思想，基于 CNN 网络和新增的多尺度特征图进行目标检测。它借鉴 Faster R-CNN 中锚（Anchor）的概念[9-11]，通过尺度不同的先验框来预测目标边界框（Bounding Boxes）。边界框预测包含目标类别预测和框选区域预测。

SSD 采用 VGG16 网络[12]作为基础深度学习网络。VGGNet 有 6 种不同的网络结构，但是每种结构都有含有 5 组卷积，每组卷积都使用 3×3 的卷积核，每组卷积后链接一个 2×2 池化层，接下来是 3 个全连接层（FC），全连接层的配置在所有网络中一致。最后一层是 Soft-max 层[13]用于分类。所有的隐含层都采用非线性修正单元（ReLU）。另外网络基本都包含了局部响应归一化（LRN）。这个归一化主要用于增强在数据集上的泛化能力[12]。VGGNet 在训练高级别的网络时，可以先训练低级别的网络，用前者获得的权重初始化高级别的网络，可以加速网络的收敛。VGG16 的网络结构如图 1 所示。

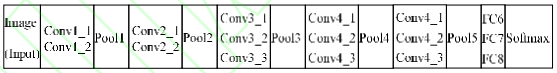

图 1　VGG16 网络结构

Fig.1 Network structure of VGG16

SSD 网络结构如图 2 所示。输入图像大小为 300×300。SSD 采用的 VGGNet 网络为 VGG16，该 VGG16 网络在 ILSVRC CLS-LOC 数据集进行预训练得到。SSD 改进了 VGG16，移除了丢包层（Dropout）和全连接层 FC8，将 Conv4_3 层作为用于检测的第一个特征图，并在其后面增加了一个 L2 正则化层。为了缩减特征图大小，其池化层 pool5 由原来的滑动步长 stride=2 的 2×2 卷积核变成 stride=1 的 3×3 卷积核。将 VGG16 的全连接层 FC6 和 FC7 转换成卷积层 Conv6 和卷积层 Conv7。SSD 网络新增卷积层 Conv8_2，Conv9_2，Conv10_2，Conv11_2，都用于提取检测所用的特征图。Conv6 采用空洞卷积（Dilation Convolution），在不增加参数与模型复杂度的条件下指数级扩大卷积的视野。Conv4_3、Conv7、Conv8_2、Conv9_2、Conv10_2、Conv11_2 卷积层共提取了 6 个特征图。得到特征图后，对这些特征图分别进行 3×3 卷积，生成一系列不同尺度的先验框。6 种尺度的特征图

共产生 6 个不同的分支，这些分支中的所有特征图上生成的先验框就是整个 SSD 网络一次检测的边框数量。

在 SSD 中，生成的先验框与带有类标和框标注的真实目标（Ground Truth，GT）进行匹配，一方面形成正负样本，一方面通过框解码，完成边框回归（预测）。检测目标的类别置信率由 Softmax 分类器得到，并通过非极大值抑制[14]筛选得到检测目标的预测框（最终框选区域）。SSD 网络的损失函数为置信度误差与位置误差的加权和。

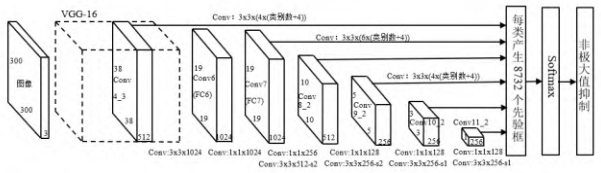

图 2 SSD 检测器网络结构

Fig2. Network structure of SSD

1.2 目标预测框的生成

1.2.1 框回归

框回归即生成检测目标的预测框。6 个特征图经过 3×3 卷积后得到的先验框往往和 GT 标注中的目标区域有一定的误差，还需要通过边框回归对每个先验框的四维坐标进行调整。该过程是通过边框回归使得预测得到的边界框的位置与真实目标的位置更逼近，通过损失函数的位置误差项来评估模型预测的好坏，以完成训练。

设 C 为待测目标的类别数。对于一个大小为 $M\times N$ 的特征图，共有 MN 个单元。设边界框的四维输出为（cx,cy,w,h），分别表示边界框的中心坐标以及宽、高。设每个单元的先验框数目为 K，那么每个单元共产生 $(C+4)K$ 个预测值，所有的单元共产生 $(C+4)KMN$ 个预测值。

若先验框的位置用 $d=(d^{cx},d^{cy},d^{w},d^{h})$ 表示，其对应的 GT 边界框用 $b=(b^{cx},b^{cy},b^{w},b^{h})$ 表示，则边界框的预测值 l 是 b 相对于 d 的转换值：

$$l^{cx}=(b^{cx}-d^{cx})/d^{w} \tag{1}$$

$$l^{cy}=(b^{cy}-d^{cy})/d^{h} \tag{2}$$

$$l^{w}=\log(b^{w}/d^{w}) \tag{3}$$

$$l^{h}=\log(b^{h}/d^{h}) \tag{4}$$

上述这个转换过程称为边界框的编码。

框解码时反向这个过程，通过平移变换和尺度变换，寻找一种关系使得输入的窗口经过映射得到一个与真实窗口更加接近的回归窗口，即从预测值 l 中得到边界框的真实预测位置 B。为了更逼近目标的真实位置，解码时设置超参数 variance 来进行调整，此时边界框解码如下：

$$B^{cx}=d^{w}(\mathrm{variance}[0]*l^{cx})+d^{cx} \tag{5}$$

$$B^{cy}=d^{y}(\mathrm{variance}[1]*l^{cy})+d^{cy} \tag{6}$$

$$B^{w}=d^{w}\exp(\mathrm{variance}[2]*l^{w}) \tag{7}$$

$$B^{h}=d^{h}\exp(\mathrm{variance}[3]*l^{h}) \tag{8}$$

将 6 个不同尺度的特征图分别输入独立的 3×3 大小的卷积层中，得到的输出维度为 $M\times N\times(4\times6)$，表示 6 个特征图上的每个位置单元回归后，以每 4 个为 1 组（四维向量）产生 24 个偏移量。

1.2.2 框选择

SSD 分别使用两个不同的阈值 S_1 和 S_2 来筛选可能包含目标物体的边框，并且去除包含同一个目标的不同边框。具体过程如下：

（1）SSD 从所有回归产生的边框中，选择包含目标物体（非背景类别）且类别置信率 T_C 大于阈值 S_1 的边框。

（2）如果不同的边框包含同一个目标物体，采用非极大值抑制算法，去除掉包含同一目标物体且重叠区域较大的边框。

非极大值抑制是一个递归的过程[14]，算法如下：首先对所有的边框，依据包含目标物体的类别置信率（由 Softmax 输出）从大到小进行排序。其

次，从概率第二大的边框开始，与概率最大的边框进行比较，如果它们的交并比大于阈值 S_2，且包含的目标种类相同，则将该边框从候选边框列表中移除。依此类推，不断地更新候选边框列表，直到所有的边框都比较完成，则候选边框列表中的边框，就是最终得到的目标物体位置的边框。

1.3 损失函数

SSD 损失函数定义为位置误差（Localization Loss，Loc）与类别置信度误差（Confidence Loss, Conf）的加权和[8]：

$$L(x,c,l,g)=\frac{1}{N}(L_{conf}(x,c)+\alpha L_{loc}(x,l,g)) \quad (9)$$

其中：N 为先验框的正样本数量，即与 GT 能够成功匹配的先验框；$x_{ij}^{p}\in\{1,0\}$ 为一个指标参数，当 $x_{ij}^{p}=1$ 时表示第 i 个先验框与第 j 个真实目标匹配，并且真实目标的类别为 p；c 为类别置信度预测值；α 为权重系数；l 为先验框所对应的位置预测值；g 为 GT 的位置参数。

对于位置误差，采用 Smooth L1 loss[10]，定义为

$$L_{loc}(x,l,g)=\sum_{i\in Pos}^{N}\sum_{m\in\{cx,cy,w,h\}}x_{ij}^{k}smooth_{L1}(l_i^m-g_j^m) \quad (10)$$

$$g_j^{cx}=(g_j^{cx}-d_i^{cx})/d_i^{w}/\mathrm{var}iance[0] \quad (11)$$

$$g_j^{cy}=(g_j^{cy}-d_i^{cy})/d_i^{h}/\mathrm{var}iance[1] \quad (12)$$

$$g_j^{w}=\log(g_j^{w}/d_i^{w})/\mathrm{var}iance[2] \quad (13)$$

$$g_j^{h}=\log(g_j^{h}/d_i^{h})/\mathrm{var}iance[3] \quad (14)$$

$$smooth_{L1}(x)=\begin{cases}0.5x^2,if\ |x|<1\\ |x|-0.5,otherwise\end{cases} \quad (15)$$

由于 $x_{i,j}^{p}$ 的存在，所以位置仅针对正样本进行计算。由于预测值 l 是编码值，g 也为对 ground truth 的 g 编码得到。

对于置信误差，采用 Softmax 损失[13]，定义为

$$L_{conf}(x,c)=-\sum_{i\in Pos}^{N}x_{ij}^{p}\log(\hat{c}_i^{p})-\sum_{i\in Neg}\log(\hat{c}_i^{0}) \quad (16)$$

其中

$$\hat{c}_i^{p}=\frac{\exp(c_i^{p})}{\sum_p\exp(c_i^{p})} \quad (17)$$

2 改进的SSD算法

由于原始的训练条件设置并不适用于交通驾驶场景，因此直接采用 SSD 进行驾驶场景下目标识别和定位时，检测准确率并不理想。因此，针对交通驾驶应用场景，本文对模型训练进行优化，提出了改进的 SSD_ARS 算法。

2.1 基于动量优化的梯度更新算法

SSD 算法中采用 VGG16 训练时，使用的优化方法为随机梯度下降法（SGD）[15]。SGD 为每一次迭代计算最小批次（Mini-batch）的梯度，然后对参数进行更新，即

$$g_t=\nabla_{\theta_{t-1}}f(\theta_{t-1}) \quad (18)$$

$$\Delta\theta=-\eta*g_t \quad (19)$$

其中：η 为学习率；g_t 为梯度；θ 为初始参数；t 为训练时的迭代步数。SGD 训练速度快，对于大数据集，也能够以较快的速度收敛[16]，其缺点是容易收敛到局部最优。在实际训练过程中，预测框与 GT 的误差使得每一次迭代的梯度受数据增广（训练过程的预处理）和难例挖掘中的抽样的影响比较大[8]，梯度含有比较大的噪声，不能很好地反映真实梯度。

本文采用基于动量（Momentum）优化的 SGD 梯度优化策略，在面对大噪声对梯度的影响时，可以很好地加速学习。动量梯度下降通过计算梯度的指数加权平均数，并利用该值来更新参数值。在某次迭代时，引入动量优化的梯度更新公式如下：

$$v'_{dW}=\mu v_{dW}+(1-\mu)dW \quad (20)$$

$$v'_{db}=\mu v_{db}+(1-\mu)db \quad (21)$$

$$W^{'} = W - \eta v_{dW} \tag{22}$$

$$b^{'} = b - \eta v_{db} \tag{23}$$

其中：W 和 b 分别为当前的权重项和常数项；$W^{'}$ 和 $b^{'}$ 分别为更新后的权重项和常数项；dW 和 db 为加速项；v_{dW} 和 v_{db} 分别为当前权重项和常数项的动量项；$v^{'}_{dW}$ 和 $v^{'}_{db}$ 分别为更新后的权重项和常数项的动量项。动量项可以看作速度项。μ 是动量因子，通常取值 0.9，动量因子的存在能限制速度过大。当前的速度是渐变的，是动量的过程。一般的梯度下降由于存在上下波动，减缓了梯度下降的速度，因此只能使用一个较小的学习率进行迭代。而使用动量梯度下降时，通过累加过去的梯度值来减少抵达最小值路径上的波动，因此在加速方向下降得更快，能够加速收敛到局部极值。前后梯度方向一致时，动量梯度下降能够加速学习；而前后梯度方向不一致时，动量梯度下降能够抑制震荡。动量梯度下降的计算复杂度为 $O(n)$，n 为样本特征数。

2.2 学习率下降策略优化

SSD 中，学习率下降采用含有下降因子的指数衰减，在此处训练过程中存在不完全收敛和收敛速度慢的问题。为此，本文采用三段阶梯式式学习率下降策略，如图 3 所示，各阶段训练过程的学习率为固定值，可用式（24）表示：

$$l_r = \begin{cases} d_1 l_s, 0 \le t < t_1 \\ d_2 l_s, t_1 \le t < t_2 \\ d_3 l_s, t \ge t_2 \end{cases} \tag{24}$$

式中：d_1，d_2，d_3 分别为三阶段的学习率下降因子；l_s 为训练的初始学习率；t 为训练的迭代步数；t_1，t_2 分别为三阶段的学习率下降边界。训练过程学习率 l_r 的选取，与学习率下降因子和初始学习率有关。

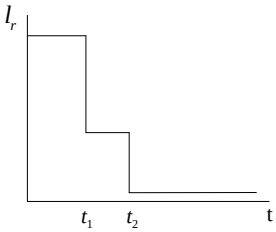

图 3 阶梯式学习率下降示意图

Fig.3 Curve of stepped learning rate reduction

2.3 先验框生成优化

SSD 模型对小物体不敏感，但是驾驶场景下需要对部分距离较远但对驾驶策略起到关键作用的目标进行检测，例如交通标志、红绿灯，因此需要优化先验框生成策略，改善模型对小目标的检测效果。

先验框通过一系列新生成的特征层得到。生成过程位于模型的前端，与尺度比例和长宽比两个参数有关，通过大量的先验框对原图做覆盖。先验框的尺度比例可以通过线性插值得到[8]，如式（25）所示：

$$s_k = \begin{cases} \dfrac{s_{\min}}{2}, k = 1 \\ s_{\min} + \dfrac{s_{\max} - s_{\min}}{4}(k-2), k \in [2,6] \end{cases} \tag{25}$$

式中：k 为特征图序号；s_k 表示第 k 个特征图上的先验框大小相对于原图的尺度比例；$s_{\min}$ 和 $s_{\max}$ 表示尺度比例的最小值与最大值。根据 s_k，通过原图大小换算得到 6 种先验框实际尺度值 $b_k(k=1,...,6))$，相当于先验框的面积。

SSD 为每个特征图设置 6 种长宽比：$\{a_1, a_2, a_3, \dfrac{1}{a_2}, \dfrac{1}{a_3}, a_1^{'}\}$，默认 $a_1 = a_1^{'} = 1$。该长宽比作用于模型前端先验框的生成，企图大面积覆盖原图目标区域。得到尺度值 b_k 后，对于某个特定的长宽比，在面积确定的情况下，按式（26）计算先验框的宽度与高度：

$$w_k^a = b_k\sqrt{a}, h_k^a = b_k / \sqrt{a} \tag{26}$$

在式中的默认情况下，每个特征图都有 1 个$a_1 = 1$且尺度为b_k的先验框，1 个和$a_1^{'} = 1$且尺度为$b_k^{'} = \sqrt{b_k b_{k+1}}$的先验框（即每个特征图都设置有两个长宽比为 1 但大小不同的正方形先验框）。 最后一个特征图需要参考一个人为设定的虚拟值b_7来计算 $b_6^{'}$。

对于交通场景中的多检测目标，其长宽比相差较大，当检测类别较多、检测目标较小时，原长宽比无法完全满足检测精度要求。通过分析发现，Conv7，Conv8_2，Conv9_2 层后接3×3卷积，具有更小的检测“视野”，对于小目标的检测更加敏感。因此本文提出在这 3 个卷积层上进行先验框生成长宽比策略优化。优化策略如下：

（1）针对大检测目标，Conv4_3，Conv10_2 和 Conv11_2 层生成的特征图仅使用 4 个先验框，不使用长宽比为$a_3, \frac{1}{a_3}$的先验框，加快检测速度。

（2）对于小检测目标，Conv7，Conv8_2，Conv9_2 层生成的特征图使用如下策略：

$$ARS_r = \left(a_1, a_2, ..., a_r, \frac{1}{a_2}, ..., \frac{1}{a_r}, a_1^{'}\right) \tag{27}$$

根据检测目标的情况，通过增添或者改变长宽比a_r与$\frac{1}{a_r}$实现策略组合，提高检测精度。

3 基于SSD_ARS模型的驾驶场景关键目标识别与框选

3.1 目标识别

目标识别过程如图 4 所示。在训练过程中，生成的先验框与训练图片的 GT 进行匹配。匹配原则主要有两个。首先，对于训练图片中的每个 GT，与其交并比（IOU）最大且同时大于阈值 S_3 的先验框设为该类的正样本。这里，交并比即产生的先验框与 GT 重叠部分与并集区域的面积比值。对于剩余的先验框，采取第 2 个原则：若先验框与某个 GT 的 IOU 大于阈值 S_3（一般是 0.5），那么认为该先验框与这个 GT 初步匹配。因为先验框可与多个 GT 匹配，取 IOU 最大的那个 GT 作为该先验框的匹配框，该 GT 的类别为该先验框的类别，先验框设为该类的正样本。

由于通常负样本数相对正样本数多，为了平衡正负样本，采用难例挖掘方法对负样本进行抽样。即按照先验框的类别置信度进行降序排列（先验框可与多个 GT 匹配），选取置信度较低的 k 个作为训练的负样本。抽样时，保证正负样本比例接近 1:3。生成的正负样本先送入网络训练后，再经过 Softmax 进行分类。

3.2 目标框选

目标预测框生成过程如图 5 所示。每个先验框生成一个预测框。测试过程中，对于每个预测框，首先将类别置信度最大的类别确定为其类别。之后过滤掉属于背景的预测框和置信度值较低的预测框。对留下的预测框进行解码，得到其真实的位置参数。解码后一般还需要缩减，防止预测框位置超出图片。再根据类别置信度进行降序排列，仅保留 top-k 个预测框。最后，采用非极大值抑制（NMS）算法[14]，过滤掉那些重叠度较大的预测框。最后剩余的预测框就是检测结果。

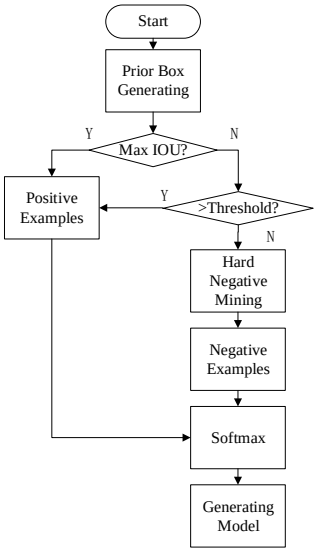

图 4 模型训练
Fig.4 Model training

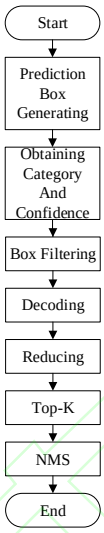

图 5 测试过程
Fig.5 Model test

4 实验数据及分析

4.1 实验数据与环境描述

实验一共收集了 813 张国内驾驶视野场景样本图像，其中包括 665 张训练图像，148 张测试图像。选取对驾驶中最为关键的目标进行标注，分别为轿车、大巴车、其余大型车型、非机动车、人、交通信号灯、提示类交通标志、指路类交通标志、警示类交通标志共 9 大类，分别用序号 O1~O9 表示。交通信号与灯交通标志一般情况下属于小目标范畴。每类目标的图像数量如表 1 所示。

实验采用的 CPU 是 Intel(R) Core(TM) i7-7700HQ，内存 16G， GPU 为 Nvidia GTX 1060，操作系统为 Windows 10，开发环境为 Spyder(Python 3.5)。

表 1 训练集与测试集描述
Table1 Description of training set and testing set

	O1	O2	O3	O4	O5	O6	O7	O8	O9
Object（training）	3169	401	270	372	460	670	454	648	243
Object（training）	579	213	195	206	174	276	236	322	165
Object（test）	745	90	39	92	138	125	110	149	30
Object（test）	140	51	25	49	45	62	53	73	20

4.2 评价指标

本实验是多目标检测任务，涉及目标的分类与框选定位，检测采用的评估指标为平均精度均值（Mean Average Precision，mAP）。mAP 为各类别目标的平均精度（Average Precision，AP）的均值。某类目标的 AP 即为该类别检测结果的 PR 曲线（Precision-Recall）下的面积。

$$AP=\frac{1}{11}\sum_{r\in\{0,0.1,\ldots,1\}}\rho_{interp}(r) \tag{28}$$

$$\rho_{interp}(r)=\max_{\tilde{r}:\tilde{r}\geq r}\rho(\tilde{r}) \tag{29}$$

AP 采用插值平均精度表示。通过取一组 11 个等间距的召回率值 [0, 0.1, 0.2, … ,1]所对应的精度值来代表 PR 曲线的形状。其中，$\rho(\tilde{r})$ 是召回时的测量精度。即通过在每个插值 r 处获取其召回率值大于 r 的最大精度 $\max_{\tilde{r}:\tilde{r}\geq r}\rho(\tilde{r})$，作为插值 r 处的插值精度 $\rho_{interp}(r)$，以获得 AP 值。

4.3 实验结果及分析

4.3.1 实验一

本实验主要验证 SSD 目标检测算法应用于驾驶场景的可行性,以及采用基于动量优化的梯度更新后的检测器性能改善。实验采用 3.1 中所述的数据集，动量因子 μ 取经典值 0.9，实验结果如表 2 所示。

表 2 算法检测精度
Table2 Detection accuracy of the algorithm

Algorithm	AP O1	O2	O3	O4	O5	O6	O7	O8	O9	mAP
SSD	0.835	0.801	0.758	0.743	0.729	0.670	0.741	0.762	0.733	0.752
SSD_M	0.890	0.865	0.834	0.815	0.803	0.734	0.833	0.837	0.802	0.824

从表 2 中可知，采用原始 SSD 算法，mAP 为 0.752。其中，除轿车与大巴车外，其余类别的 AP 均低于 0.8。SSD 算法采用基于动量优化的梯度更新（SSD_M）后，mAP 指标为 0.824，提高 7.2%。除交通信号灯外，其他目标的 AP 均在 0.8 以上。

4.3.2 实验二

本实验主要为了寻找 SSD_ARC 算法的相关最优参数，动量因子 μ 固定为 0.9。

（1）三阶段学习参数实验

SSD_ARC 算法中三阶段学习率下降因子分别设置为 1，0.1，0.01，终止学习率为 0.000001，学习率下降边界分别为 80000 步与 100000 步。实验结果如表 3 所示。

表 3 不同学习率下的检测精度与训练时间
Table3 Detection accuracy and training time with different learning rates

lr	AP O1	O2	O3	O4	O5	O6	O7	O8	O9	mAP	训练时间/h
0.0006	0.883	0.868	0.745	0.806	0.781	0.704	0.826	0.807	0.790	0.801	21
0.0008	0.891	0.859	0.817	0.813	0.801	0.726	0.824	0.812	0.775	0.813	33
0.0010	***0.890***	***0.865***	***0.834***	***0.815***	***0.803***	***0.734***	***0.833***	***0.837***	***0.802***	***0.824***	***40***
0.0012	0.887	0.853	0.822	0.799	0.794	0.733	0.815	0.827	0.779	0.812	44

从表 3 可以看出，当初始学习率为 0.001 时，mAP 值最高，为 0.824，训练耗时 40h。适当降低学习率，可缩短训练时间。

（2）先验框长宽比数量优化实验

固定 μ = 0.9，*lr* = 0.001，先验框长宽比生成策略选择包括 $\{1,2,4,\frac{1}{2},\frac{1}{4},1\}$，$\{1,2,3,4,\frac{1}{2},\frac{1}{3},\frac{1}{4},1\}$，$\{1,2,3,5,\frac{1}{2},\frac{1}{3},\frac{1}{5},1\}$，$\{1,2,3,4,5,\frac{1}{2},\frac{1}{3},\frac{1}{4},\frac{1}{5},1\}$。分别命名为 SSD_ARS1，SSD_ARS2，SSD_ARS3，SSD_ARS4，分别表示为 S1~S4。实验结果如表 4 所示。

表 4 不同先验框长宽比生成策略下的检测精度
Table 4 Detection accuracy with different generation strategies of default box at different aspect ratio

	AP O1	O2	O3	O4	O5	O6	O7	O8	O9	mAP
S1	0.884	0.862	0.832	0.803	0.796	0.731	0.834	0.833	0.804	0.819
S2	0.910	0.884	0.858	0.834	0.823	0.792	0.851	0.864	0.836	0.850
S3	0.902	0.876	0.847	0.837	0.811	0.793	0.848	0.866	0.831	0.846
S4	***0.913***	***0.897***	***0.853***	***0.841***	***0.830***	***0.801***	***0.850***	***0.871***	***0.837***	***0.855***

从表 4 可以看出，当采用 SSD_ARC1，即 $a_3 = 4$ 时，mAP 指标为 0.819，与原始策略的测试结果相比精度更低。当采用 SSD_ARC2，即保留原有比例，增加 $a_4 = 4$ 后， mAP 提高 2.6%，交通信号灯的 AP 提高 5.8%。当采用 SSD_ARC3，即增加 $a_5 = 5$ 时，mAP 与原始策略相比提高 2.2%，交通信号灯的 AP 提高 5.9%。当采用 SSD_ARC4 生成策略，即增加 $a_4 = 4$ 与 $a_5 = 5$ 时，mAP 为 0.855，为 4 组策略中最高值，交通信号灯的 AP 进一步提高。交通信号灯与交通标志通常为检测难度较大的远景小目标。

可见，改进的 SSD_ARC4 算法性能最好，与原始 SSD 算法相比，平均精度提高 10.3%。轿车的 AP 提高 7.8%。大巴车的 AP 提高 9.6%。其余大型车型的 AP 提高 9.5%。非机动车的 AP 提高 9.8%。人的 AP 提高 10.1%。交通信号灯的 AP 提高 13.1%。提示类交通标志的 AP 提高 10.9%。指路类交通标志的 AP 提高 10.9%。警示类交通标志的 AP 提高 10.4%。交通信号灯与 3 类交通标志属于小目标范畴，其 AP 提升相对较大，均超过 10%。综上，小目标的检测精度提高更加明显。

4.3.3 实验三

考虑到驾驶场景中，近景对当前驾驶策略的影响最为关键，而远景对当前驾驶策略影响较弱，本实验进行目标距离测试实验。该实验从 web 中收集了 180 张驾驶场景图片用于测试。数据分为 3 组。第 1 组 60 张做近景目标标注，第 2 组 60 张做中景目标标注，第 3 组 60 张做远景目标标注。采用

SSD_ARC4 策略的模型进行测试，实验结果如表 5 所示。

表 5 不同距离目标的检测精度
Table 5 Detection accuracy at different distance

	AP									mAP
	O1	O2	O3	O4	O5	O6	O7	O8	O9	
C	***0.974***	***0.953***	***0.928***	***0.885***	***0.804***	***0.844***	***0.910***	***0.897***	***0.902***	***0.900***
G	0.883	0.875	0.826	0.816	0.655	0.807	0.896	0.893	0.872	0.836
F	0.784	0.755	0.721	0.733	0.642	0.603	0.637	0.741	0.788	0.712

可以看出，对于远景目标（F）的 mAP 为 0.712，中景目标（G）的 mAP 为 0.836，而对驾驶视野最关键的近景目标（C）的 mAP 达到了 0.9。进一步验证了该算法在驾驶场景中的有效性。同时，中景和远景目标仍具有较好的识别率，可以作为驾驶策略规划的辅助。

4.3.4 实验四

本实验选取实际行车记录仪记录的视频对模型进行测试。实验采用 SSD_ARC4 模型，测试视频 1（V1）为直行路段驾驶视频，帧数为 200 帧，检测重点为车辆、非机动车、人。视频 2（V2）为十字路口弯道转向直道驾驶视频，帧数为 250 帧，检测目标包含交通信号灯与交通标识。视频 3（V3）为直行路段驾驶视频，帧数为 150 帧，检测目标中三大类机动车辆数量较密集。实验结果如表 6 所示。

从表 6 中可以看出，在实际视频帧场景测试中，最为关键的车辆和交通信号灯检测效果均比较理想，AP 均达到 0.8 以上。视频 1、2 中人和非机动车的检测精度较差，这是由于非机动车包含部分人的躯干，这两类存在一定的误检情况。同时视频 1 中，人的目标较小并存在重叠的情况。

表 6 实际驾驶场景检测精度
Table6 Detection accuracy of video frames in real driving scenes

	AP									mAP	Frame Rate （FPS）
	O1	O2	O3	O4	O5	O6	O7	O8	O9		
V1	0.881	/	/	0.714	0.673	/	/	0.848	/	0.779	20.03
V2	0.873	/	/	0.712	/	0.851	0.857	0.862	/	0.831	20.05
V3	0.890	0.852	0.834	/	0.801	/	/	0.857	0.725	0.827	20.07

考虑远景目标对于驾驶的重要性相对较低，该算法适用于实际驾驶场景，检测帧率在 20 FPS 左右，满足实时检测要求。实际驾驶场景视频帧检测效果如图 6 所示。

图 6 实际检测效果图
Fig.6 Detection in real driving scene

5 结束语

目标识别与框选是实现无人驾驶的关键技术。本文提出采用基于深度网络 VGGNet 的新型检测算法 SSD 进行驾驶场景中的关键目标检测，可同时实现驾驶场景中的 9 类关键目标的识别、语义标注和目标框选。根据实际应用场景，提出了改进的 SSD_ARC 算法，通过优化梯度更新算法、学习率下降策略，先验框生成策略，提高了检测精度，特别是小目标的检测精度，检测速度满足实时检测需求。

参考文献

[1] Sermanet P, Lecun Y. Traffic sign recognition with multi-scale convolutional networks[C]// International Joint Conference on Neural Networks. USA: IEEE, 2011:2809-2813.

[2] Chen X, Kundu K, Zhang Z, *et al*. Monocular 3D object detection for autonomous driving[C]// IEEE Conference on Computer Vision and Pattern Recognition. USA: IEEE , 2016: 2147-2156.

[3] Uçar A, Demir Y, Güzeliş C. Moving towards in object recognition with deep learning for autonomous driving applications[C]// International Symposium on Innovations in Intelligent Systems and Applications. USA: IEEE, 2016:1-5.

[4] Chen Y, Zhao D, Le L, et al. Multi-task learning for

dangerous object detection in autonomous driving[J]. Information Sciences, 2018, 432: 559-571.

[5] 许明文. 基于无人驾驶平台的交通灯及数字检测与识别系统[D].南京：南京理工大学, 2017.

[6] TIAN Y L, LUO P, WANG X G, *et al*. Pedestrian detection aided by deep learning semantic tasks[C]//Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition. Boston, USA: IEEE, 2015: 5079-5087.

[7] 葛园园, 许有疆, 赵帅,等. 自动驾驶场景下小且密集的交通标志检测[J]. 智能系统学报, 2018, 13(3):366-372.

[8] Liu W, Anguelov D, Erhan D, *et al*. SSD: Single Shot MultiBox Detector[C]// European Conference on Computer Vision. Cham： Springer, 2016:21-37.

[9] Ren S, He K, Girshick R, *et al*. Faster R-CNN: Towards real-time object detection with region proposal networks[C]// International Conference on Neural Information Processing Systems. USA: MIT Press, 2015:91-99.

[10] Girshick R. Fast R-CNN[C]// IEEE International Conference on Computer Vision. USA: IEEE, 2015:1440-1448.

[11] Girshick R, Donahue J, Darrell T, *et al*. Rich feature hierarchies for accurate object detection and semantic segmentation[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. USA: IEEE, 2014: 580-587.

[12] Simonyan K, Zisserman A. Very Deep Convolutional Networks for Large-Scale Image Recognition[C]//3rd International Conference on Learning Representations (ICLR). Hilton San Diego: Computer Science, 2015: 1150-1210.

[13] Boureau Y L, Ponce J, Lecun Y. A theoretical analysis of feature pooling in visual recognition[C]// International Conference on Machine Learning. Israel: DBLP, 2010:111-118.

[14] Neubeck A, Gool L V. Efficient non-maximum suppression[C]// International Conference on Pattern Recognition. USA: IEEE, 2006: 850-855.

[15] Bottou L. Large-scale machine learning with stochastic gradient descent[C]// Proceedings of COMPSTAT'2010. Hamburg: Springer, 2010:177-186.

[16] Li M, Zhang T, Chen Y, *et al*. Efficient mini-batch training for stochastic optimization[C]// ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. USA: ACM, 2014:661-670.