

文章编号: 2095-2163(2023)08-0205-05

中图分类号: TP391

文献标志码: A

基于 UNet 模型的赛道识别算法研究

徐威, 于海滨, 余胤翔, 巩荣芬

(辽宁科技大学 电子与信息工程学院, 辽宁 鞍山 114051)

摘要: 随着赛道识别问题不断升级, 赛道识别的精确度需求大大提升, 传统的赛道识别方法受环境变化的影响大。近几年来, 深度学习方法在赛道识别方面也取得了很好的效果。本文就赛道识别问题提出了一种基于 Keras 的 UNet 模型的赛道识别方法, 该方法是对赛道图像根据 UNet 模型训练结果进行分割, 并用细化算法来识别中线, 由此判断小车接下来的行径。最后, 经实际对比实验测试, 新方法识别时效性一般, 识别准确率高。

关键词: UNet 模型; 赛道识别; 细化算法; 深度学习

Research on track recognition algorithm based on UNet model

XU Wei, YU Haibin, YU Yinxiang, GONG Rongfen

(School of Electronics and Information Engineering, Liaoning University of Science and Technology, Anshan Liaoning 114051, China)

【Abstract】 With the continuous upgrading of track recognition issues, it is required to greatly improve the accuracy of track recognition, and traditional track recognition methods are significantly affected by environmental changes. In recent years, deep learning methods have also achieved good results in track recognition. This article proposes a race track recognition method based on Keras's UNet model for race track recognition. This method segments the race track image based on the training results of the UNet model, and uses a thinning algorithm to identify the centerline for determining the next behavior of the car. Finally, through actual comparative experimental testing, the new method has average recognition timeliness and high recognition accuracy.

【Key words】 UNet model; track recognition; refinement algorithm; deep learning

0 引言

近些年, 智能汽车获得了迅猛发展, 陆续涵盖了工程控制、信息与通信、模式识别、传感技术、电气工程、计算机等多个学科及领域。研究可知, 智能汽车技术在交通运输、智能驾驶等方面有着广阔的应用前景与发展空间^[1]。智能汽车的核心部分是赛道元素识别、方向和速度控制, 精准的赛道元素识别算法是方向和速度控制的基础和前提, 尤其是面对复杂路况时赛道的识别是系统设计的难点^[2]。智能小车相比于真实汽车有着专属的特点和用途, 智能小车具有机械结构简单轻便、驾驶模式低速安全、整车易于改造实现等特点, 可以用于智能汽车核心控制系统的研发设计^[3]。最近几年, 深度学习方法在赛道识别方面也取得了很好的效果。这些方法可以实现自动学习特征, 从而更加精确地识别出赛道。本文

就智能小车的赛道识别问题提出了一种基于 Keras 的 UNet 模型的识别方法, 可对逐帧图像进行识别处理, 划出中线集, 来判断道路状况, 以此来实现对赛道元素的精准识别。

1 传统赛道识别现状

赛道识别是从图像或视频数据中准确识别出赛道, 这是自动驾驶、机器视觉和计算机视觉等领域的基础。目前, 赛道识别研究已经较为成熟, 其中一些传统的赛道识别方法包括颜色分割、特征提取、边缘检测算法。最近几年, 深度学习的赛道识别方法也取得了很好的效果。这些方法可以实现自动学习特征, 以此更加准确地识别出赛道。为了更准确、更快速地对赛道进行识别, 需要结合深度学习和计算机视觉等技术进行改进。

基金项目: 大学生创新创业训练计划项目(S202210146037)。

作者简介: 徐威(2002-), 男, 本科生, 主要研究方向: 机器视觉; 于海滨(2002-), 男, 本科生, 主要研究方向: 嵌入式; 余胤翔(2002-), 男, 本科生, 主要研究方向: 嵌入式软件和机器视觉; 巩荣芬(1979-), 女, 博士, 副教授, 主要研究方向: 模式识别、机器学习。

通讯作者: 巩荣芬 Email: 1161555381@qq.com

收稿日期: 2023-01-11

2 赛道识别实现思路

2.1 基于 UNet 模型的赛道识别算法

基于 UNet 模型的赛道识别算法主要可以分成 2 个部分。第一部分是对于输入进的图像根据网络训练出的结果进行预测,将图像分割成背景和赛道两部分;第二部分是对赛道部分进行细化算法求取中点集。最后,根据中点集就可以预测智能车未来的行径。

2.2 UNet 模型实现

UNet 是一种用于图像分割任务的卷积神经网络模型。由 Ronneberger 等学者在 2015 年提出,并广泛应用在医学图像分割领域。该模型的特点是具有对称的 U 形结构^[4],其中包含一个下采样路径和

一个上采样路径,使得在保持空间分辨率的同时,能够对图像进行有效的语义分割。UNet 的训练过程通常使用交叉熵损失函数,并且可以使用数据增强技术来增加数据的多样性。该模型在医学图像分割方面的应用,不仅能够对肿瘤和器官进行有效的分割,而且在其他领域、如自然图像分割和道路分割等方面也取得了可观成果。

2.2.1 主干特征提取

采用的主干特征提取网络为 VGG16。VGG16 总共有 16 层,包括 13 个卷积层和 3 个全连接层。第 1 次经过 64 个卷积核的 2 次卷积后,采用 1 次 pooling,第 2 次再经过 2 次 128 个卷积核卷积后,再采用 pooling,再重复 2 次 3 个 512 个卷积核卷积后,再采用 pooling,最后经过 3 次全连接^[5]。

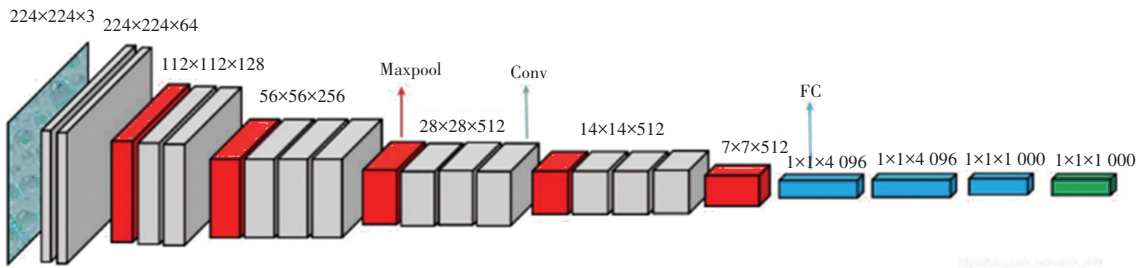


图1 VGG16 结构图

Fig. 1 Structure diagram of VGG16

利用 VGG16 提取后的卷积层和最大池化层,经过图 2 的卷积及池化操作后,获得 5 个初步的有效特征层。

input(512, 512, 3)	
↓	
Conv2d filters=64(512, 512, 64)	
Conv2d filters=64(512, 512, 64)	→ f1
Maxpooling s=2(256, 256, 64)	
↓	
Conv2d filters=128(256, 256, 128)	
Conv2d filters=128(256, 256, 128)	→ f2
Maxpooling s=2(128, 128, 128)	
↓	
Conv2d filters=256(128, 128, 256)	
Conv2d filters=256(128, 128, 256)	
Conv2d filters=256(128, 128, 256)	→ f3
Maxpooling s=2(64, 64, 256)	
↓	
Conv2d filters=512(64, 64, 512)	
Conv2d filters=512(64, 64, 512)	
Conv2d filters=512(64, 64, 512)	→ f4
Maxpooling s=2(32, 32, 512)	
↓	
Conv2d filters=512(32, 32, 512)	
Conv2d filters=512(32, 32, 512)	
Conv2d filters=512(32, 32, 512)	→ f5

图2 有效特征层示意图

Fig. 2 Schematic diagram of the effective feature layer

2.2.2 主干特征提取

经过主干特征提取网络,可以获得 5 个初步的有效特征层,在加强特征提取网络过程中,会利用这 5 个初步的有效特征层进行特征融合^[6],特征融合的方式就是对特征层进行上采样后、再进行堆叠。为了方便网络的构建与呈现出更好的通用性,在上采样时直接进行 2 倍上采样再进行特征融合,最终获得的特征层和输入图片的高宽相同。

2.2.3 Loss 解析

Loss 解析是指对一个神经网络模型的训练过程中,计算出的误差损失值进行分析和解释。可以帮助了解模型的训练效果和优化方向,从而更好地调整模型参数,提升模型性能。

本文使用的 Loss 由 Cross Entropy Loss 和 Dice Loss 两部分组成。Cross Entropy Loss 就是交叉熵损失,在对像素点进行分类时使用^[7]。Dice Loss 将语义分割的评价指标作为 Loss, Dice 系数是一种用于度量 2 个样本相似度的指标,常用于自然语言处理中的文本匹配任务。是用来测量 2 个样本中共同出现的元素或特征的比例,取值范围在 $[0, 1]$ 。其值可用如下公式进行计算:

$$s = \frac{2|x \cap y|}{|x| + |y|} \tag{1}$$

2.2.4 预测结果

获得特征层后, 就可以利用输入进来的图片特征预测结果, 使用 1×1 的卷积进行通道调整。

2.3 细化算法实现

细化算法是一种图像处理算法, 现已广泛应用于数字图像处理领域。算法可以将图像中的线条或轮廓进行细化, 以达到消除噪声、减少数据冗余等目的。细化算法在很多应用领域都有着重要的作用, 例如医学图像处理、指纹识别、人脸识别等。

细化算法是一个迭代算法, 整个迭代过程分为 2 部分, 对图像的像素进行处理。数学方法公式分别见如下:

$$\begin{cases} 2 \leq N(P1) \leq 6 \\ S(P1) = 1 \\ P2 * P4 * P6 = 0 \\ P4 * P6 * P8 = 0 \end{cases} \tag{2}$$

$$\begin{cases} 2 \leq N(P1) \leq 6 \\ S(P1) = 1 \\ P2 * P4 * P8 = 0 \\ P2 * P6 * P8 = 0 \end{cases} \tag{3}$$

其中, $N(P1)$ 表示与 $P1$ 相邻的 8 个像素点中, 为实景像素点的个数; $S(P1)$ 表示像素中出现 0 ~ 1 的累计次数^[7], 0 为背景, 1 为实景。

符合上述全部算法时该格子的算法为 1, 并且

根据这 2 个部分进行迭代, 直到结果不再变化为止。最后的结果就是细化算法后的骨架。

3 数据集处理

3.1 数据集构建

UNet 的工作实际上就是对图片的每个像素点进行预测, 以此来对各像素点位的每个类别概率进行预测。UNet 模型训练用的数据集, 采用 VOC 的格式, 分为 2 部分, 第一部分是原图, 为 RGB 图像 [高, 宽, 3]; 第二部分是标签, 为灰度图图像 [高, 宽]。原图和标签的数据集图片如图 3 所示, 是 239 个原图与标签图。

3.2 数据增强

为了扩大样本的数据量, 需提高模型在复杂环境下的准确性和泛化能力。本文对数据集下的图像进行了改变亮度、锐化处理, 模糊处理等操作。改变亮度、锐化处理、模糊处理后的图像数据, 可以模拟出赛道在不同光照强度下的场景。对原图的亮度处理是在 $(-40, 40)$ 的范围内随机加减, 通过对每个像素点加减值, 来达到改变图像亮度的效果; 对原图的锐化处理是增强图像的边缘对比度, 需要先对图像进行高通滤波处理^[8], 来突出其特征边缘, 再对特征边缘在 $(1.1, 1.3)$ 范围内做随机倍数加强; 对原图的模糊处理上是随机添加均值为 0、方差为 1 的高斯噪声^[9]。原图和标签的数据集图片如图 4 所示。

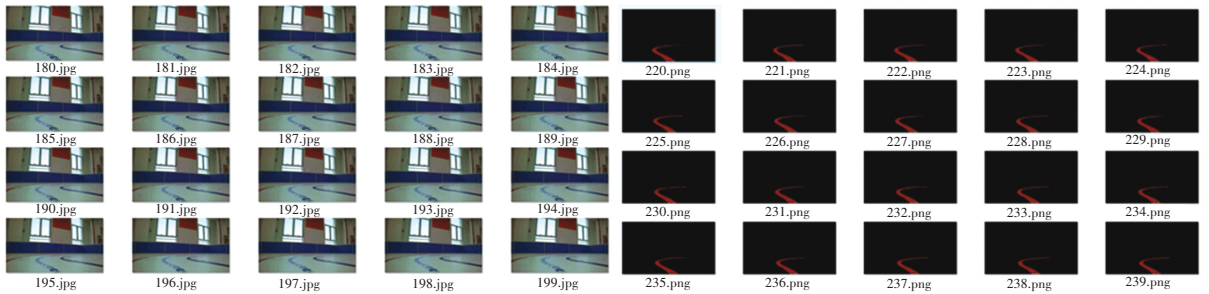


图 3 原图和标签的数据集图片

Fig. 3 Dataset image of the original image and labels

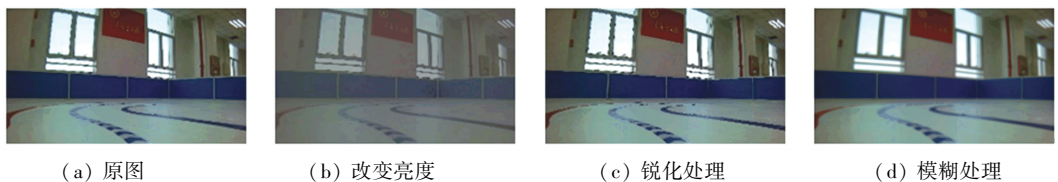


图 4 数据集增强

Fig. 4 Dataset enhancement

4 实验与结果分析

4.1 试验平台

在 Windows10 系统下, CUDA v10.0, cuDNN 7.5.0, 运用 Python 3.6.4 语言进行编译, 基于 Tensorflow 2.6.2 下的 Keras 2.6.0 框架下搭建的, 在

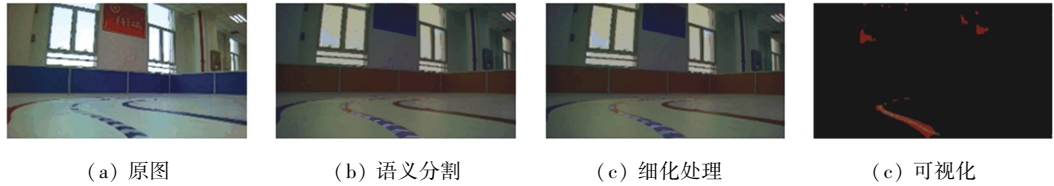


图5 识别效果图

Fig. 5 Recognition rendering

4.3 时效性分析

本文对算法进行了测试, 选取了 30 帧的视频作为检测对象, 记录每一帧下的频率, 画出时效性分析图来体现基于 UNet 模型的赛道识别算法的时效性。时效分析结果如图 6 所示。

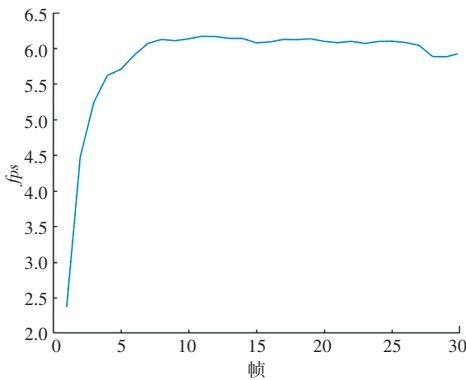


图6 时效分析图

Fig. 6 Aging analysis

4.4 准确度分析

本文为了验证基于UNet模型的赛道识别的准

VS Code 2021 下运行。

4.2 识别效果

使用基于 UNet 模型的赛道识别算法, 对输入视频的每一帧进行识别, 选取其中一帧, 识别效果如图 5 所示。

确性, 与传统赛道识别算法进行结果比对。对于传统赛道识别算法, 研究中采用的是颜色分割和边缘检测的方法。

实验时对相同的赛道视频进行识别, 并对其输出的中点集与正确中点集进行比较。为了加强准确度分析的严密性, 研究采取分别对其在不同环境情况下(低光照、正常光照、高光照)的 8 张赛道图片进行识别, 对输出的中点集使用编辑距离算法, 计算与实际中点集的误差, 公式具体如下:

$$f(i, j) = \begin{cases} \max(i, j) & \text{if } \min(i, j) = 0 \\ \min \begin{cases} f(i-1, j) + 1 \\ f(i, j-1) + 1 \\ f(i-1, j-1) + 1_{(a_i \neq b_j)} \end{cases} & \text{if } \min(i, j) \neq 0 \end{cases} \quad (4)$$

其中, a, b 是 2 个数组, $f(i, j)$ 是 a 中的前 i 个字符和 b 中前 j 个字符的编辑距离。

研究得到的识别偏差结果对比见表 1。

表 1 识别偏差结果对比表

Tab. 1 Comparison table of identification bias results

序号	低光照		正常光照		高光照	
	传统法误差/ 像素	新法误差/ 像素	传统法误差/ 像素	新法误差/ 像素	传统法误差/ 像素	新法误差/ 像素
1	47	25	5	5	41	21
2	47	24	3	5	45	21
3	16	15	3	3	15	14
4	47	22	11	8	45	19
5	45	23	9	9	35	19
6	29	19	6	6	14	10
7	16	7	12	6	21	7
8	41	24	3	7	31	20
平均偏差	36	17.7	5.8	5.4	30.9	14.6