

doi:10.19306/j.cnki.2095-8110.2021.02.013

# 一种基于语义分割的机场跑道检测算法

王旒军<sup>1</sup>, 蒋海涛<sup>2</sup>, 刘崇亮<sup>1</sup>, 裴新凯<sup>1</sup>, 邱宏波<sup>1</sup>

(1. 北京自动化控制设备研究所, 北京 100074;

2. 海装驻北京地区第三军事代表室, 北京 100074)

**摘要:**针对复杂电磁作战环境下无人机自主着陆应用场景,提出了一种基于图像语义分割的机场跑道检测算法,并设计构建了轻量高效的端到端跑道检测神经网络 RunwayNet。在特征提取部分,使用空洞卷积对 ShuffleNet V2 进行改造,得到了输出特征图分辨率可调的主干网络,并利用自注意力机制设计了自注意力网络模块,使网络具备全局跑道特征提取能力。设计了解码器模块将网络浅层丰富的细节、空间位置信息与顶层粗略、抽象的语义分割信息相融合,从而获得精细的跑道检测输出结果。实验结果表明,RunwayNet 网络在无人机着陆全过程都可以对跑道区域进行精准的分割识别,并且在嵌入式计算平台上能达到接近实时的处理速度,具有很强的实用价值。

**关键词:**语义分割;机场跑道检测;自注意力模块;主干网络

中图分类号:TP242.6

文献标志码:A

文章编号:2095-8110(2021)02-0097-10

## An Airport Runway Detection Algorithm Based on Semantic Segmentation

WANG Liu-jun<sup>1</sup>, JIANG Hai-tao<sup>2</sup>, LIU Chong-liang<sup>1</sup>, PEI Xin-kai<sup>1</sup>, QIU Hong-bo<sup>1</sup>

(1. Beijing Institute of Automatic Control Equipment, Beijing 100074, China;

2. The Third Representative Office of the Naval Equipment Department in Beijing Area, Beijing 100074, China)

**Abstract:** For the application scenarios of autonomous landing of UAVs in complex electromagnetic combat environments, an airport runway detection algorithm based on image semantic segmentation is proposed, and a lightweight and efficient end-to-end runway detection neural network called RunwayNet is designed and constructed. In the feature extraction part, ShuffleNet V2 is modified by using atrous convolution to obtain a backbone network with adjustable output feature map resolution. Self-attention module is designed using the self-attention mechanism to make the network capable of global runway feature extraction. And the decoder module is designed to fuse the rich details, the spatial location information of the low-level layers, and the rough, abstract semantic segmentation information of the high-level layers to obtain a fine runway detection output. The experimental results show that RunwayNet can accurately segment the runway area during the entire landing of the UAVs, and can achieve near real-time processing speed on the embedded computing platform, which has strong practical value.

**Key words:** Semantic segmentation; Airport runway detection; Self-attention module; Backbone network

收稿日期:2019-08-30;修订日期:2019-12-02

基金项目:国家自然科学基金重大科研仪器研制项目(41527803)

作者简介:王旒军(1992-),男,硕士,工程师,主要从事视觉导航方面的研究。E-mail: wangliujun\_bit@163.com

## 0 引言

在军事领域,无人机(Unmanned Aerial Vehicles, UAVs)可以替代有人作战飞机执行战场管理、空中和海上监视、战场巡逻与制图、战时安全通信、目标跟踪和作战打击等多种任务。随着各项技术的成熟,各型军用无人机,如美国 RQ-4 全球鹰和中国彩虹无人机等越来越多地参与到实际作战中,变得越来越流行。

固定翼无人机在实际使用中面临的一大挑战是如何确保无人机在执行完任务后能安全着陆。目前常用的无人机着陆导航系统为卫星/惯性等多源信息融合的组合导航系统。但是,面对越来越复杂的电磁作战环境,严重依赖外界定位信息的着陆系统的安全性难以保证。相对于依赖外界定位信息的着陆方案,使用视觉信息辅助的全自主着陆系统在自主性、安全性和价格方面展现出优势<sup>[1]</sup>。

机场跑道检测(Airport Runway Detection)是视觉着陆导航系统的重要一环。Vezinet 等<sup>[1]</sup>基于跑道的参考图像执行图像配准,以检测点特征进行着陆引导。Gibert 等<sup>[2]</sup>使用机场跑道边界和中线作为特征进行跑道检测。Khaled 等<sup>[3]</sup>结合区域竞争分割和最小化能量函数的方法,构建了实时跑道检测和跟踪系统。在之前的团队工作中,刘畅等<sup>[4]</sup>提出了一种视觉/惯性组合导航算法,采用 Kalman 滤波分别完成位置和姿态匹配,实现了视觉测量与惯导信息的融合。刘崇亮等<sup>[5]</sup>提出了一种着陆视觉导航 P3P 问题唯一解的求解方法,实现了无人机与跑道之间的六自由度位姿解算。文献[6]利用可见光相机、红外相机和雷达高度计等传感器,通过提取跑道直线和轮廓特征,实现了跑道检测和位姿计算。

人工智能(Artificial Intelligence, AI)技术的崛起给机场跑道检测算法研究带来了新的契机。本文研究基于图像语义分割的机场跑道检测算法,在特征提取部分,利用注意力模型(Attention Model)设计构建了自注意力模块,融合了特征图空间维度和通道维度的全局相似性信息,以捕获特征图中大范围全局特征,可以提高跑道检测网络的全局特征提取能力。主干网络选用轻量高效的 ShuffleNet V2<sup>[7]</sup>,并使用空洞卷积对网络进行改造,从而获取更高分辨率的稠密特征图。最后,设计了简洁高效的解码器模块,使用跳跃连接将网络浅层特征图引

入顶层,使浅层丰富的细节、空间位置信息与顶层粗略、抽象的语义分割信息相融合,以获得精细的跑道检测输出结果。

## 1 自注意力模型

注意力模型可以对特征图中全局特征之间的关系进行建模,在图像、视频和音频处理等工作中广泛使用<sup>[8-13]</sup>。两个向量点积的几何意义是计算其相似程度,点积结果越大,表明两个向量越相似。自注意力机制(Self-attention Mechanism)利用这个原理进行特征图中关系权值的学习,不需要外部辅助,通过特征图之间的特征变换获取特征图空间、通道或者时序的全局关系。

本节利用自注意力机制,设计构建了自注意力模块(Self-attention Module)。首先介绍了位置注意力模块(Position Attention Module)和通道注意力模块(Channel Attention Module)的网络结构,然后阐述了融合这两种注意力模块的自注意力模块网络设计。

### 1.1 位置注意力模块

图像语义分割任务对网络的特征抽象能力要求比较高。理论上,网络中深层的卷积核具有很大的感受野,更容易提取抽象出高级语义特征,但是深层网络的实际感受野要远小于理论值<sup>[14]</sup>,导致全局特征缺失。位置注意力模块通过特征图之间的特征变换,计算每个像素与其他像素的全局相似性关系,可以增加网络的感受野,通过建模特征图中局部特征的全局上下文信息,从而增强网络的全局特征表达能力。

位置注意力模块的网络结构如图 1 所示。首先对尺寸为  $[H, W, C]$  的输入特征图使用  $1 \times 1$  卷积压缩特征图的通道维度,对于用于特征变换的第一路分支,通道压缩比为 8,得到尺寸为  $[H, W, C/8]$  的特征图  $f$ ;第二路分支的通道压缩比为 2,保留了较多的原始信息,得到尺寸为  $[H, W, C/2]$  的特征图  $g$ 。然后再分两路对  $f$  特征图进行特征变换:一路进行维度变换和转置操作,合并高度和宽度维度,将通道变换为第一维度,得到  $[C/8, H * W]$  的特征图  $f_1$ ;另一路对  $f$  特征图只进行维度变换操作,合并高度和宽度维度,得到  $[C/8, H * W]$  的特征图  $f_2$ 。接着对  $f_1$  和  $f_2$  特征进行矩阵乘法,消去通道维度  $C/8$ ,再使用 Softmax 函数归一化,获得尺寸为  $[H * W, H * W]$  的注意力图。注意力图描

述了特征图中每个像素的全局相关性信息,特征图中两个位置的特征相似度越高,注意力图中相应的值越大。同时,对第二路分支的特征图  $g$  进行维度变换,得到  $[H * W, C/2]$  的特征图  $g_1$ 。最后,使用

注意力图与  $g_1$  特征图进行矩阵乘法,将获取的全局相关性权值信息重新分布到原始特征图中,并使用维度变换操作恢复特征图的高度和宽度,输出通道压缩后  $[H, W, C/2]$  的特征图。

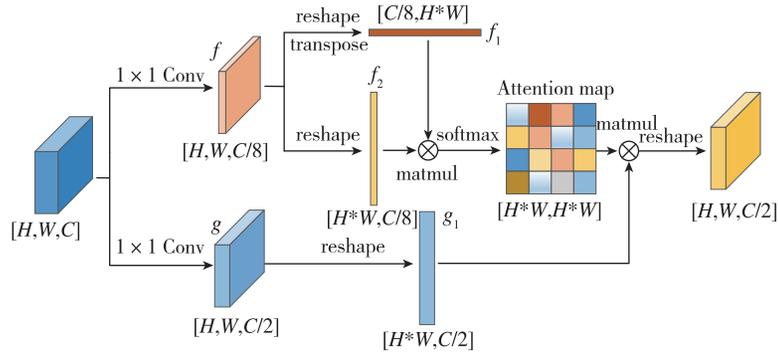


图 1 位置注意力模块

Fig. 1 Position attention module

## 1.2 通道注意力模块

一般情况下,深度卷积神经网络中顶层不同的通道会关注不同的分割类别。位置注意力模块可以提取特征图中不同像素的全局相似性关系,而通道注意力模块利用相似的特征变换操作,在计算特征图矩阵乘法时,将相乘顺序调换了一下,保留通道维度,使网络具有全局通道相关性表达能力<sup>[15]</sup>。

通道注意力模块的网络结构如图 2 所示。首先使用  $1 \times 1$  卷积压缩特征表达得到特征图  $h$ ,  $h \in \mathbf{R}^{H \times W \times (C/2)}$ 。然后分成三路对特征图进行特征变换,

使用维度变换或转置操作,分别得到尺寸为  $[C/2, H * W]$  的特征图  $h_1$ , 尺寸为  $[H * W, C/2]$  的特征图  $h_2$  和  $h_3$ 。接着,对特征图  $h_1$  和  $h_2$  进行矩阵乘法,不同于位置注意力模块,这里消去  $H * W$  维度,保留通道维度。为了防止训练期间的损失值不收敛<sup>[12]</sup>,这里将特征图中每个位置去除最大值,设得到的特征图为  $G$ , 其中一个像素点  $G_{i,j}$  为

$$G_{i,j} = \max(G) - G_{i,j} \quad (1)$$

其中,  $\max(G)$  表示特征图中像素的最大值;  $i, j$  分别表示像素点坐标,  $i, j \in [1, C/2]$ 。

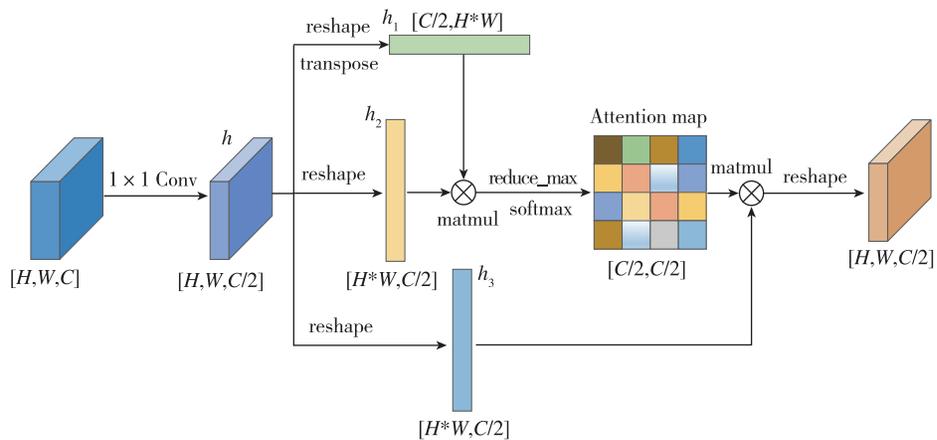


图 2 通道注意力模块

Fig. 2 Channel attention module

随后对特征图  $G$  使用 Softmax 函数归一化,获得尺寸为  $[C/2, C/2]$  的注意力图,注意力图描述了

特征图中所有通道之间的相关性信息。特征图中两个通道的特征相似度越高,注意力图中相应的权

值越大。

最后,将注意力图与特征图  $h_3$  进行矩阵乘法,消去  $H \times W$  维度,并恢复特征图高度和宽度维度。使用注意力图更新通道关系权值,有选择性地强调所有通道中相关联的通道,并将全局通道相关性信息分布到原始特征图中,获得尺寸为  $[H \times W, C/2]$  的输出特征图。

### 1.3 自注意力模块

自注意力模块整合了位置注意力模块和通道注意力模块,可以融合特征图空间维度和通道维度的全局相似性信息,从而获得更好的语义分割效果。

自注意力模块如图3所示,首先使用位置注意力模块和通道注意力模块对输入特征图进行操作,分别提取特征图全局像素相似性和通道相关性信息。然后,使用  $3 \times 3$  卷积进一步提取两个模块输出的特征图特征,获得两路尺寸为  $[H, W, C/2]$  的特征图。接着融合位置注意力和通道注意力输出特征图。不同于文献[12]使用两个可训练参数分别对两路特征图进行加权后再和原始特征图逐像素求和的方法,本文设计的自注意力模块将两个注意力模块输出的特征图直接与原始特征图在通道维度进行拼接,再使用  $1 \times 1$  卷积对拼接的特征图进行特征整合,在达到相同特征融合目的的情况下可以使特征的融合更加自由。最后,添加失活概率为0.1的随机失活层,随机失活一些特征,以避免网络对某些特定特征的过度依赖,从而增强模块的鲁棒性。自注意力模块中每层卷积操作后,使用批量归一化和 ReLU 激活函数来减少梯度消失等现象,以加快收敛速度。

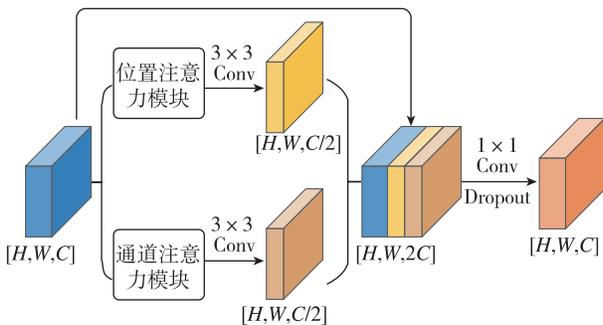


图3 自注意力模块

Fig.3 Self-attention module

自注意力模块原理简单、结构清晰,融合了位置注意力模块的空间相似性特征提取能力和通道

注意力模块的全局通道关系提取能力,在不需要外界信息辅助的情况下,通过特征图之间的特征变换就可以捕获特征图中大范围全局相似性信息,在没有增加很多参数的情况下增加了网络感受野,提高了网络全局特征提取能力。

## 2 跑道检测网络

在语义分割领域,多数学者的研究工作致力于提升图像分割的精度,相对忽略了网络推理的实时性。本文着眼于构建实用跑道检测网络,结合学术界最新的研究成果和理念,在网络设计时兼顾精度和实时性。以下小节分别阐述了主干网络构建、解码器设计和整个跑道检测网络的结构。

### 2.1 主干网络构建

分类网络作为语义分割网络的主干,承担了特征提取器的重要角色,在很大程度上决定了整个语义分割网络的性能。

ShuffleNet V2 轻量级分类网络遵循高效设计理念:网络模块的输入和输出特征图通道数应当一致,以减小内存访问耗时(Memory Access Cost, MAC);避免使用过多的分组卷积(Group Convolution);减少网络分支结构,以提高并行计算效率;逐像素操作如 ReLU 和  $1 \times 1$  卷积等,虽然具有较小的计算量(FLoat point OPerations, FLOPs),但是内存访问消耗较高。

深度可分离卷积(Depthwise Separable Convolution)首先使用  $3 \times 3$  卷积核在输入特征图上逐通道分别进行卷积运算,然后使用  $1 \times 1$  卷积进行通道关系映射。通常,逐通道计算的  $3 \times 3$  卷积被称为“Depthwise Convolution”, $1 \times 1$  卷积被称为“Pointwise Convolution”。深度可分离卷积的使用可以大幅减少网络的计算量。

设深度可分离卷积模块输入特征图尺寸(高 $\times$ 宽 $\times$ 通道)为  $h \times w \times c_i$ , 输出特征图尺寸为  $h \times w \times c_o$ 。则常规卷积卷积核尺寸为  $k \times k \times c_i \times c_o$ ,  $c_o$  为卷积核个数,深度可分离卷积中逐通道卷积核尺寸为  $k \times k \times c_i$ ,  $1 \times 1$  卷积核尺寸为  $1 \times 1 \times c_i \times c_o$ 。深度可分离卷积与常规卷积的计算量之比为

$$\frac{\underbrace{k \times k \times c_i \times h \times w}_{\text{depthwise convolution}} + \underbrace{1 \times 1 \times c_i \times c_o \times h \times w}_{\text{pointwise convolution}}}{k \times k \times c_i \times c_o \times h \times w} = \frac{(k^2 + c_o)c_i \times h \times w}{k \times k \times c_i \times c_o \times h \times w}$$

$$= \frac{k^2 + c_o}{k^2 \times c_o} = \frac{1}{c_o} + \frac{1}{k^2} \quad (2)$$

由于通常卷积核尺寸远小于输出通道数,所以深度可分离卷积与常规卷积的计算量之比约为  $1/k^2$ ,当 Depthwise Convolution 卷积核设置为  $3 \times 3$  时,深度可分离卷积比常规卷积可使计算量仅为原来的  $1/8 \sim 1/9$ 。

网络模块的输入和输出特征图通道数相同可以减少内存访问。简便起见,设卷积核的大小  $k = 1$ ,则卷积计算量  $FLOPs = F = h\omega c_i c_o$ ,内存访问次数  $MAC = h\omega(c_i + c_o) + c_i c_o$ ,由均值不等式可得

$$\begin{aligned} MAC &= h\omega(c_i + c_o) + c_i c_o \\ &\geq \sqrt{(h\omega)^2 4c_i c_o} + \frac{F}{h\omega} \\ &= 2\sqrt{h\omega F} + \frac{F}{h\omega} \end{aligned} \quad (3)$$

由式(3)可知,当 FLOPs 确定时, $c_i = c_o$  时模型 MAC 最小。

ShuffleNet V2 舍弃了 ShuffleNet V1<sup>[16]</sup> 中  $1 \times 1$  分组卷积,引入了通道分离(Channel Split)操作,将输入  $c$  通道数的特征图分为  $c - c'$  和  $c'$  (实现时  $c' = c/2$ ) 两部分,一路分支使用恒等连接,另一路分支使用输入和输出通道数相等的  $1 \times 1$  Conv +  $3 \times 3$  DWConv +  $1 \times 1$  Conv 三个卷积层(DWConv: Depthwise Convolution),最后将两路结果在通道维度进行拼接(Concatenate)。为了使两路分支(组)的特征相互交流,对拼接后的特征进行通道洗牌(Channel Shuffle)操作。接着进入到下一个网络模块,这样拼接、通道洗牌和下一个模块的通道分离操作就合并成了一个逐像素运算的模块,从而减少了内存访问。

实验表明,ShuffleNet V2 与 ShuffleNet V1、MobileNet<sup>[17-18]</sup> 和 modified Xception<sup>[19]</sup> 等网络相比,在推理速度大为提升的同时具有最佳分类精度<sup>[7]</sup>。因此,本文选用 ShuffleNet V2 “ $1 \times$ ”版本构建主干网络。

首先,将 ShuffleNet V2 网络 Stage4 网络块后的池化层和全连接层等去除,构建为全卷积网络(Fully Convolutional Networks, FCN)。这样,网络的 Output Stride(网络输入图像与输出特征图的尺寸比值)为 32。而图像分割任务希望网络输出特征图相对稠密,以获得精细的语义分割结果,并且为了提取抽象的语义信息,在输出较大分辨率特征图的情况下,顶层网络的感受野不能太小。空洞卷

积(Atrous Convolution or Dilated Convolutions)可以解决这个问题。

如图 4 所示,空洞卷积引入膨胀比率(Dilation Rate) $r$ ,在原有卷积核的中间插入空洞构造成为空洞卷积。常规卷积(图 4(a))可以认为是空洞卷积在  $rate=1$  时的特例。设常规卷积核尺寸为  $k \times k$ ,则空洞卷积的卷积核大小等效于

$$\hat{k} = k + (k - 1)(r - 1) \quad (4)$$

由式(4)可知,空洞卷积在不增加卷积核参数数量和卷积计算量的情况下,可以增加卷积核的感受野,进而提升网络的特征提取能力。

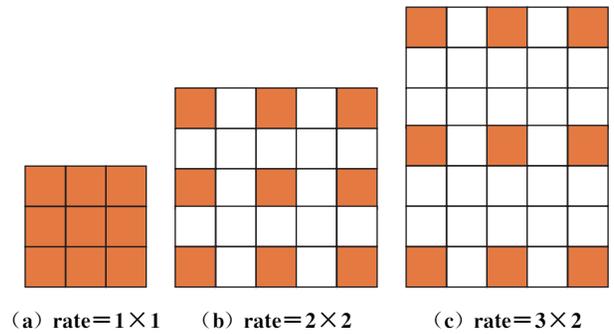


图 4 空洞卷积

Fig. 4 Atrous convolution

设卷积层的输入图像或者特征图的尺寸为  $h \times \omega$ ,每次滑动的步长(Stride)为  $s$ ,在边缘填充(Padding)零像素的圈数为  $p$ ,输出特征图的尺寸为  $m \times n$ ,则使用常规卷积核时输出特征图的尺寸为

$$\begin{cases} m = \lfloor (h + 2p - k)/s \rfloor + 1 \\ n = \lfloor (\omega + 2p - k)/s \rfloor + 1 \end{cases} \quad (5)$$

使用空洞卷积核时输出特征图的尺寸为

$$\begin{cases} m = \lfloor h + 2p - k - (k - 1)(r - 1)/s \rfloor + 1 \\ n = \lfloor \omega + 2p - k - (k - 1)(r - 1)/s \rfloor + 1 \end{cases} \quad (6)$$

由式(6)可知,通过设置相应的膨胀比率和步长,空洞卷积使主干网络输出的特征图尺寸可控,可以根据分割任务调整特征图尺寸,对语义分割任务非常有利。

将 Stage4 网络块中第一层步长设为 1,并将随后的三个网络层改造为膨胀比率为 2 的空洞卷积,使得卷积层感受野继续增大的同时,特征图尺寸不再减小。最终选用 Stage4 网络块输出 464 通道 Output Stride 为 16 的特征图作为主干网络的输出。

本文构建的主干网络如图 5 所示。主干网络中

使用了通道分离、通道洗牌、深度可分离卷积和空洞卷积等技术,在有效降低参数量、计算量和内存访问量的同时,具备很强的特征提取能力。

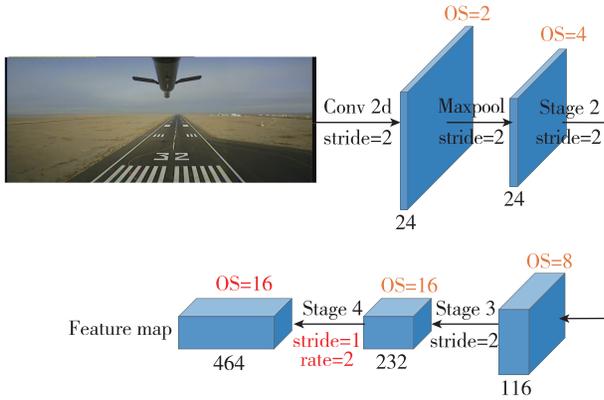


图5 主干网络  
Fig. 5 Backbone network

### 2.2 解码器设计

主干网络和自注意力模块编码了图像中丰富的语义信息,但是输出特征图分辨率低,丢失了部分空间位置信息。解码器使用上采样逐步增大特征图的尺寸,并使用跳跃连接融合浅层丰富的细节和空间位置信息,可以恢复清晰的物体边缘,从而获得更加稠密、精细的语义分割结果。

本文设计的简单高效的解码器模块如图6所示。图6中,DSConv表示深度可分离卷积(Depth-wise Separable Convolution),OS表示Output Stride。对输入OS=16的特征图首先进行双线性

插值2倍上采样,获得OS=8的特征图,然后引入主干网络中相同空间分辨率的Stage2网络输出特征图,并使用 $1 \times 1$ 卷积降维 $1/3$ 得到38通道浅层特征。随后将两路特征图拼接后,使用两层 $3 \times 3$ 的深度可分离卷积进一步整合细化深层和浅层特征,最后输出256通道OS=8的整合了深层网络丰富语义信息和浅层网络丰富细节及空间位置信息的精细化分割特征图。

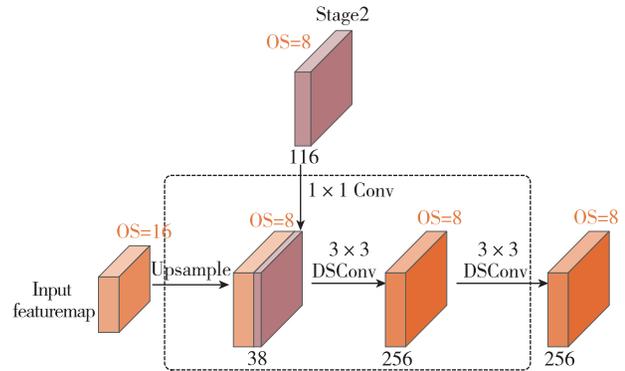


图6 解码器模块  
Fig. 6 Decoder module

### 2.3 跑道检测网络设计

整个跑道检测网络(命名为RunwayNet)可分为编码器和解码器两部分,网络结构如图7所示。编码器部分包括主干网络和自注意力模块:主干网络对输入的三通道图像逐步提取抽象语义特征,最终输出464通道OS=16的特征图;自注意力模块进一步对主干网络输出特征图进行特征变换,捕获

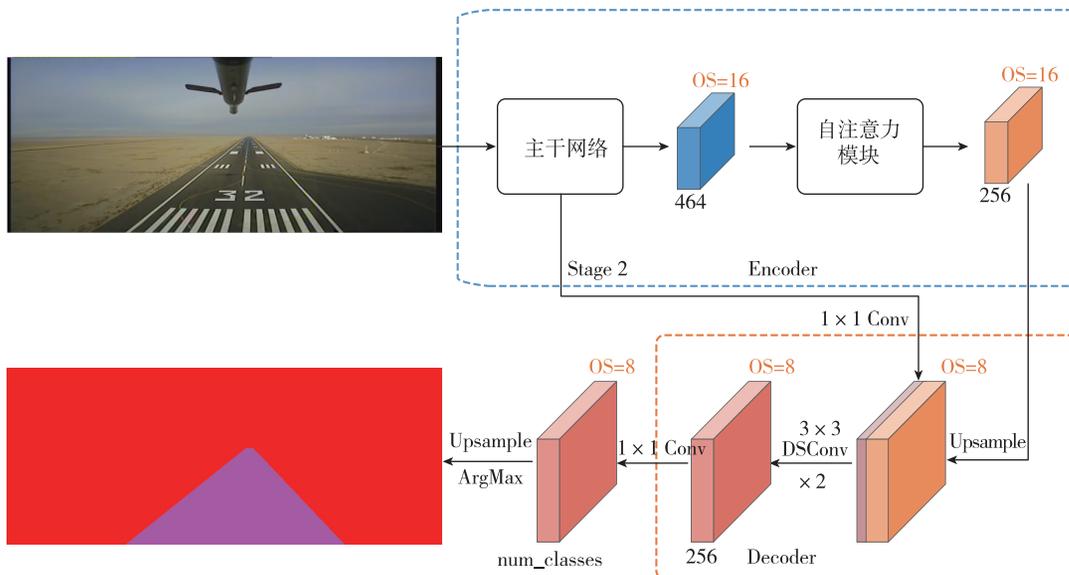


图7 RunwayNet网络结构图  
Fig. 7 RunwayNet network architecture

特征图空间维度和通道维度的全局相似性信息,提高了编码器的全局特征提取能力。编码器输出的 256 通道特征图随机进入解码器,经过跳跃连接和双线性插值上采样融合浅层特征,输出  $OS=8$  更加稠密、精细的特征图。最后,使用  $1 \times 1$  卷积将特征图映射为两通道(分类类别数)的分割图,将分割图上采样 8 倍并在通道维度取最大值(ArgMax)操作,从而获得最终的分割结果。

### 3 实验

#### 3.1 数据集介绍

实验部分使用某型国产固定翼无人机进行跑道图像数据采集,收集了数个跑道不同季节、不同时刻、不同天气下的机场跑道图像序列,使跑道数据具有多样性。采集的跑道图像序列经过降采样和人工筛选得到 1753 张跑道图像。此外,为了丰富实际采集的跑道数据,在互联网上搜集并筛选得到 948 张着陆跑道图像。最后共收集了 2701 张图像。

随后使用 labelme 标注工具对跑道图像进行精细标注工作。设置“跑道”和“背景”两个类别语义标签:跑道区域定义为左右跑道边线、起始斑马线和终止斑马线之间的区域,其他图像区域划分为背景。接着将数据划分为 2124 张训练集和 557 张测试集,命名为 Runway 数据集。Runway 数据集数据统计如表 1 所示,表中以序列为单位统计了图片

表 1 Runway 数据集统计

Tab. 1 Runway dataset statistics

序列	图片数	数据集	季节	时刻	天气	来源
01	506	Train	冬季	黄昏	晴	采集
02	106	Train	夏季	白天	晴	采集
03	96	Train	夏季	白天	晴	互联网
04	221	Train	秋季	白天	阴	互联网
05	140	Train	夏季	白天	晴	互联网
06	83	Train	夏季	黄昏	阴	采集
07	150	Train	夏季	白天	晴	互联网
08	302	Train	春季	白天	阴	采集
09	271	Train	夏季	白天	阴	采集
10	100	Train	夏季	白天	暴雨	互联网
11	149	Train	春季	早晨	晴	采集
12	216	Test	冬季	白天	阴	采集
13	241	Test	冬季	白天	阴	互联网
14	75	Test	夏季	黄昏	阴	采集
15	45	Test	夏季	白天	阴	采集

数、划分为训练集或测试集的数据集、季节、时刻、天气和数据来源等详细信息。Runway 数据集关注真实机场场景下跑道的分割能力,贴近机场跑道检测的实际应用需求,任务难度相对较高。

#### 3.2 训练细节

训练平台为搭载两块 Nvidia GTX 1080Ti GPU 的服务器,CPU 型号为 I7-8700K,拥有 32GB 机身内存,运行 Ubuntu 16.04 操作系统。软件代码基于 TensorFlow<sup>[20]</sup> 开源框架实现。

网络训练时首先使用在 ImageNet 数据集<sup>[21]</sup> 上预训练的 ShuffleNet V2 来初始化主干网络权重,然后将整个网络在 MS COCO 数据集<sup>[22]</sup> 上进行端到端预训练,最后在 Runway 数据集上进行最终的训练。

数据增强方面,首先使用缩放因子为  $[0.5, 2.0]$ ,步长为 0.25 的随机缩放操作。然后对训练输入图像进行  $1242 \times 375$  尺寸的随机裁剪。最后对训练图像使用概率为 0.5 的随机左右翻转操作。

网络使用 Softmax 函数计算每个像素的分类概率,并使用交叉熵函数计算总损失值。权值正则化系数设为  $4 \times 10^{-5}$ ,批量归一化的 Batch Sizes 设置为 16,在反向传播和网络参数更新时使用 Adam 优化器<sup>[23]</sup>,学习速率使用 ploy 策略,初始学习率设置为  $1 \times 10^{-3}$ ,power 设置为 0.9,最大迭代次数设置为 60K。

#### 3.3 实验结果

本节首先进行 RunwayNet 各个模块的消融实验(Ablation Experiments),然后对自注意力模块中间结果进行可视化,最后给出 RunwayNet 在英伟达 Jetson AGX Xavier 嵌入式平台上的跑道检测效果图。

首先在 Runway 数据集上进行 RunwayNet 各个模块的消融实验,以验证各个模块及其组合的性能、参数量和计算量。跑道分割精度用平均交并比<sup>[24]</sup>(Mean Intersection over Union, MIoU)来评价,参数量的单位为 MB(兆字节),网络计算量用 GFLOPs 表示,GFLOPs =  $1 \times 10^9$  FLOPs,GFLOPs 在网络输入为  $1242 \times 375 \times 3$  分辨率下测得。

实验结果如表 2 所示,表中 SAM 和 Decoder 分别表示自注意力模块和解码器模块,Our-Basic 方法表示只使用主干网络进行语义分割,推理速度 fps 在英伟达 Jetson Xavier 平台上测得,MIoU 得分由 Runway 测试集计算得出。

表2 RunwayNet各模块消融实验

Tab. 2 Result of ablation experiment for RunwayNet modules

Method	SAM Decoder	Params	GFLOPs	fps	MIoU/%
Our-Basic	—	1.09	4.41	24.5	85.28
Our-SAM	✓	1.58	6.33	23.1	87.53
Our-Decoder	✓	1.24	6.62	22.3	86.32
RunwayNet	✓	1.72	8.54	21.8	88.53

由表2中数据可知:Our-Basic网络配置在参数量仅1.09MB、计算量仅4.41GFLOPs的情况下,MIoU得分可达到85.28%,在Xavier上推理速度达到24.5fps,可见本文构建的主干网络在极少参数量和计算量的条件下具有很强的特征提取能力;Our-SAM网络配置在主干网络的基础上添加了自注意力模块,MIoU评分提升了2.25%达到87.53%,参数量略微增加0.49MB,计算量增加1.92GFLOPs,推理速度也略微下降到23.1fps,可见自注意力模块在没有增加很多参数量和计算量的情况下,分割精度提升明显;Our-Decoder网络配置在主干网络的基础上添加了解码器模块,MIoU评分相比Our-Basic网络提升了1.04%,参数量略微增加0.15MB,计算量增加2.21GFLOPs,推理速度稍微下降,可见解码器模块融合了浅层细节和空间位置信息后能够获得更好的跑道检测结果。RunwayNet网络整合了自注意力模块和解码器模块的优点,参数量为1.72MB,计算量为8.54GFLOPs,在Xavier上推理速度为21.8fps,最终MIoU评分达到88.53%,相比于Our-Basic网络评分提高了3.25%,验证了本文设计的自注意力模块、解码器模块和RunwayNet跑道检测网络的优异性能。

为了更好地理解注意力模型的原理并验证本文构建的自注意力模块的有效性,将网络中间结果:图像标签、网络预测结果、位置注意力和通道注意力图进行可视化。

由1.1节的分析可知,对于 $H \times W \times C$ 的输入特征图,每一个像素位置都对应一幅 $H \times W$ 分辨率的位置注意力图,该注意力图描述了当前像素与特征图中所有像素的相似性信息。由1.2节可知,通道注意力图的尺寸为 $[C/2, C/2]$ ,为了使通道注意力特征可视化,对融合了原始特征图和通道相关性信息的通道注意力模块输出特征图进行可视化,其尺寸为 $H \times W \times C/2$ 。需要说明的是,由于原图和标签尺寸为 $1242 \times 375$ ,网络预测结果 $OS=8$ ,所以网络预测原始分辨率为 $47 \times 156$ ;而自注意力模块

输入和输出特征图 $OS=16$ ,所以注意力图实际分辨率为 $24 \times 78$ ,为了可视化方便,将各图像缩放至统一尺寸。此外,实际训练时标签图像和预测输出的像素值为其对应的类别,所以其像素取值为 $(0, 1)$ 。可视化时将标签图像和预测结果进行了 $[255/2]$ 因子的加权,方便区分不同的类别标签。

自注意力模块可视化结果如图8所示。图8中,每一列对应一张图的可视化结果,第一行为输入原图,第二行为真值标签,第三行为网络预测结果。PAM表示位置注意力图,其标号对应图中像素位置 $(x, y)$ ;CAM表示通道注意力图,其标号对应通道号。由第四行第一列可见,红色坐标点 $(14, 7)$ 对应的位置注意力图提取了整个图中“背景”类别的信息,分类准确,边界清晰。第四行第二列绿色坐标 $(38, 15)$ 像素点对应跑道区域,其位置注意力图对整个标签真值定义“跑道”区域响应明显,对非标签定义但是人类认知上的跑道区域有所响应且有较强的抑制作用,对“背景”类别没有任何响应。由最后一行可见,通道注意力图同样提取了不同类别清晰的分割区域,如第一列第五列中通道109提取了“跑道”类别,第一列第五列中通道50对“背景”类别响应强烈。

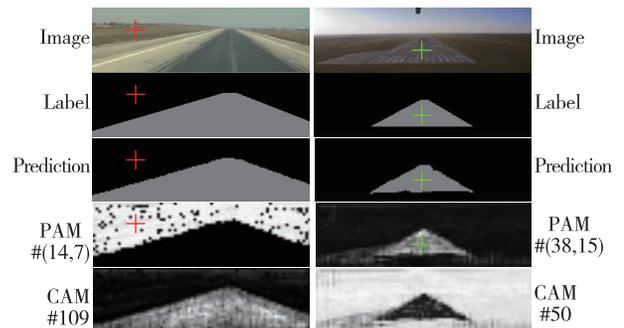


图8 自注意力模块中间层可视化结果

Fig. 8 Visualization of self-attention module intermediate layer

由以上可视化结果分析可知,自注意力模块如预期的一样提取了特征图中各像素位置之间的全局相似性关系和通道之间的全局相关性信息。

为了使评测得分结果更加直观,可视理解解本文设计的RunwayNet的有效性,给出了RunwayNet在Runway测试集上的分割结果,并做出分析。

分割结果如图9所示,跑道区域用紫色标记,前四行图像为机载实验采集所得,后两行图像由互联网搜集所得。由图中第一列结果可见,RunwayNet网络在距离跑道很远(大于3km)时就对跑道区域

正确感知,人眼隐约看见跑道的时候,网络就能大致识别出跑道区域。由最后一列结果图可见,从无人机即将着陆到完全降落到跑道上滑行,网络都准确地分割出了跑道区域。结合图中第一列、第二列和第三行结果可见,网络对由远及近着陆过程中各个阶段的跑道图像都能精准的分割识别,并且对不同时、不同季节、不同背景、不同铺设条件的跑道

都能准确进行分割检测。综合以上实验结果和分析可知,相比于其他方法<sup>[1,6]</sup>,RunwayNet 网络具有媲美于人眼的检测精度和作用距离,通过大量数据的训练具备对陌生机场跑道的检测识别能力,并且对无人机着陆全过程成像尺度变化剧烈的跑道区域进行了精准的分割识别,具有很强的多尺度全局特征提取能力。

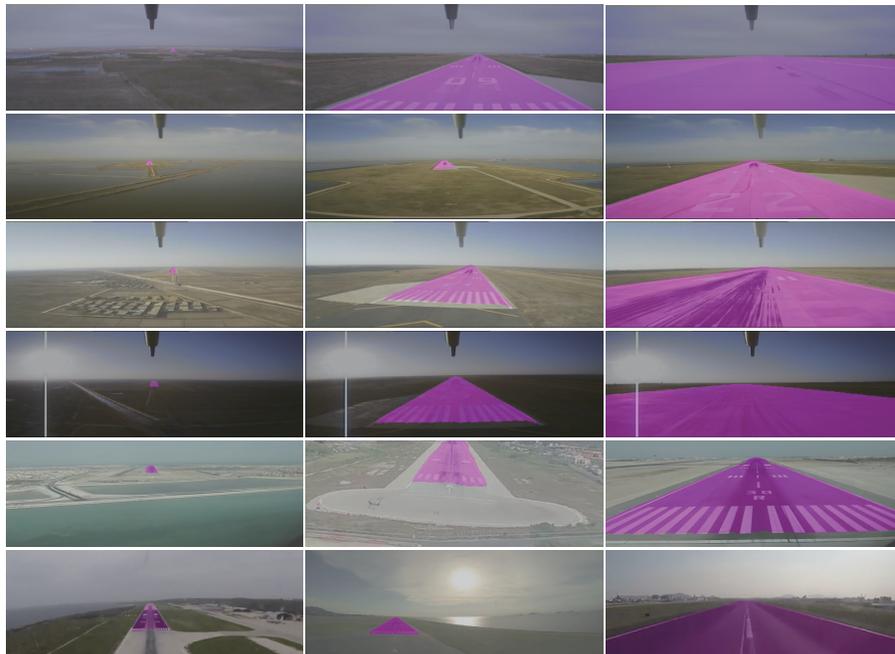


图 9 Runway 测试集分割结果

Fig. 9 Segmentation results of Runway test set

## 4 总结

本文针对无人机自主着陆应用场景,研究了基于图像语义分割的机场跑道检测算法,构建了轻量高效的 RunwayNet 跑道检测网络。在编码器部分,设计构建了自注意力模块,选用 ShuffleNet V2 构建了主干网络。在解码部分设计了简洁高效的解码器以获得更加精细的跑道分割结果。实验部分使用无人机进行跑道图像数据采集,收集了丰富多样的机场跑道图像,形成 Runway 跑道数据集。实验结果显示,本文设计的自注意力模块和 RunwayNet 跑道检测网络具有优异的性能。最终 RunwayNet 网络在 Runway 测试集上取得了 88.53% 的 MIoU 评分,1242×375 图像分辨率下在英伟达 Jetson Xavier 平台上能达到 21.8 fps 的处理速度,且无人机着陆全过程都可以对跑道区域进行精准的分割识别,具有很强的实用价值。

## 参考文献

- [1] Vezinet J, Macabiau C, Escher A C, et al. State of the art of image-aided navigation techniques for aircraft approach and landing[C]// Proceedings of International Technical Meeting of the Institute of Navigation, 2013: 473-485.
  - [2] Victor G, Guilhem P. Landing of an airliner using image based visual servoing[J]. IFAC Proceedings Volumes, 2013, 46(23): 74-79.
  - [3] Abu-Jbara K, Alheadary W, Sundaramorthi G, et al. A robust vision-based runway detection and tracking algorithm for automatic UAV landing[C]// Proceedings of International Conference on Unmanned Aircraft Systems (ICUAS). IEEE, 2015: 1148-1157.
  - [4] 刘畅, 杨良军, 刘峰, 等. 一种基于惯性/视觉信息融合的无人机自主着陆导航算法[J]. 导航定位与授时, 2016, 3(6): 6-11.
- Liu Chang, Yang Liangjun, Liu Feng, et al. Naviga-

- tion algorithm based on inertial/vision information fusion of UAV autonomous landing[J]. *Navigation Positioning and Timing*, 2016, 3(6): 6-11(in Chinese).
- [5] 刘崇亮, 刘垒, 扈光锋, 等. 一种着陆视觉导航 P3P 问题的解法[J]. *导航定位与授时*, 2018, 5(1): 58-61.
- Liu Chongliang, Liu Lei, Hu Guangfeng, et al. A P3P problem solving algorithm for landing vision navigation[J]. *Navigation Positioning and Timing*, 2018, 5(1): 58-61(in Chinese).
- [6] Hecker P, Bestmann U, Wolkow S, et al. Integrity enhancement of an integrated navigation system with optical sensors[J]// *Proceedings of 26<sup>th</sup> Saint Petersburg International Conference on Integrated Navigation Systems (ICINS)*, 2019: 79-88.
- [7] Ma N, Zhang X, Zheng H T, et al. ShuffleNet V2: practical guidelines for efficient cnn architecture design[C]// *Proceedings of European Conference on Computer Vision (ECCV)*. Springer, Cham, 2018: 116-131.
- [8] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[C]// *Proceedings of Advances in Neural Information Processing Systems*, 2017: 5998-6008.
- [9] Zhang H, Goodfellow I, Metaxas D, et al. Self attention generative adversarial networks [C]// *Proceedings of International Conference on Machine Learning*, 2019: 7354-7363.
- [10] Wang X, Girshick R, Gupta A, et al. Non-local neural networks[C]// *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2018: 7794-7803.
- [11] Hu J, Shen L, Sun G, et al. Squeeze-and-excitation networks[C]// *Proceedings of IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018: 7132-7141.
- [12] Fu J, Liu J, Tian H, et al. Dual attention network for scene segmentation[C]// *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2019.
- [13] Shen T, Zhou T, Long G, et al. DiSAN: directional self-attention network for RNN/CNN-free language understanding[C]// *Proceedings of AAAI Conference on Artificial Intelligence*, 2018: 5446-5455.
- [14] Zhou B, Khosla A, Lapedriza A, et al. Object detectors emerge in deep scene CNNs[C]// *Proceedings of International Conference on Learning Representations*, 2015.
- [15] Woo S, Park J, Lee J Y, et al. CBAM: convolutional block attention module[C]// *Proceedings of European Conference on Computer Vision*, 2018: 3-19.
- [16] Zhang X, Zhou X, Lin M, et al. ShuffleNet: an extremely efficient convolutional neural network for mobile devices[C]// *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2018: 6848-6856.
- [17] Howard A G, Zhu M, Chen B, et al. Mobilenets: efficient convolutional neural networks for mobile vision applications[J]. *arXiv preprint arXiv: 1704. 04861*, 2017.
- [18] Sandler M, Howard A, Zhu M, et al. MobileNetV2: inverted residuals and linear bottlenecks [C]// *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Salt Lake City, UT. IEEE, 2018: 4510-4520.
- [19] Li Z, Peng C, Yu G, et al. Light-head R-CNN: in defense of two-stage object detector[J]. *arXiv preprint arXiv:1711.07264*, 2017.
- [20] Abadi M, Agarwal A, Barham P, et al. TensorFlow: large-scale machine learning on heterogeneous distributed systems[J]. *arXiv preprint arXiv: 1603.04467*, 2016.
- [21] Deng J, Dong W, Socher R, et al. ImageNet: a large-scale hierarchical image database[C]// *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2009: 248-255.
- [22] Lin T Y, Maire M, Belongie S, et al. Microsoft coco: common objects in context[C]// *Proceedings of European Conference on Computer Vision*. Springer, Cham, 2014: 740-755.
- [23] Kingma D P, Ba J L. Adam: a method for stochastic optimization[J]. *arXiv preprint arXiv: 1412.6980*, 2014.
- [24] Everingham M, Eslami S M A, Gool L V, et al. The pascal, visual object classes challenge: a retrospective [J]. *International Journal of Computer Vision*, 2015, 111(1): 98-136.

(编辑:李瑾)