

英博超算科技有限公司产学研 合作项目需求

英博超算（南京）科技有限公司

2021-08-03

公司简介

一、公司简介

英博超算（南京）科技有限公司，中兴通讯子公司。成立于 2018 年，前身是中兴通讯的汽车电子产品线，从 2016 年开始中兴通讯累积投入了 300 多人的团队历时 3 年潜心前期研发，前期累积投入研发资金过亿元。

依托中兴通讯 33 年核心技术积累与高素质研发人才队伍起步，英博超算是国内第一家将 IT 和 CT 领域成熟的软件定义理念导入汽车电子领域的新型科技企业，以自主研发自动驾驶智能计算平台、智能安全网关、一机多屏智能车机为核心产品，以开放产业生态打造的模式来全面满足不同档次汽车市场需求，为国内车厂提供全系列、不同计算力和外部接口的汽车电子产品。同时，公司积极布局 V2X、工业互联网等领域。

二、联系方式

联系人：吴豪

联系方式：18603040358

邮箱：wu.hao250@unlimitedai.com.cn

时间：2021.8.3

面向典型单车智能 L3~L4 自动驾驶系统的 功能安全分析

一、任务描述：

1. 综合 ISO 26262 和 ISO 21448 的要求，从整车功能级别分析智能驾驶系统的功能安全要求（FSR）和安全目标（Safety Goal）；
2. 基于 ISO 21448 的要求，结合上述 FSR 和 Safety Goal，分析智能驾驶系统感知部分的潜在失效场景，并分析对应失效场景的风险、危害及安全策略；
3. 基于上述 FSR 和 Safety Goal，根据 ISO 26262 要求，针对智能驾驶域控制器、传感器，分别针对硬件架构、软件架构及功能安全测试大纲，进行功能安全相关的分析，并提供设计建议。
4. 上述所有分析和设计建议，分析如何降维落地至 L2+智能驾驶系统，提供设计建议。

二、评价指标：

系统级功能安全设计文档完整度(%)

三、任务目标：

根据 ISO 26262 和 ISO 21448 的要求，对任务描述中的 4 项内容进行分析和设计，提供对应的设计文档。

算法性能需求

一、单目 3D 目标检测

1. 研究背景：

3D 目标检测在自动驾驶中扮演了非常重要的角色，是车辆感知周围环境的重要手段。但是当前的 3D 目标检测算法往往需要双目相机，激光雷达等传感器来提供准确的 3D 信息。但是激光雷达价格高昂，双目相机标定较为困难。为了得到一个简单易用的，成本低廉的，能真正使用在自动驾驶中的 3D 目标检测算法，我们将关注于单目 3D 目标检测。

所谓“单目 3D 目标检测”，即在算法输入只有一张 RGB 图片的情况下，返回我们感兴趣的物体的 3D 信息（包括物体的类别，物体的中心点 3D 坐标，物体的实际长宽高等）。

近年来，深度学习在单目 3D 目标检测也取得了一定的成果，但能在自动驾驶领域真正落地的算法却不多。主要是由于当前的一些单目 3D 目标检测深度学习算法主要存在以下的一些问题：

- 精度不高。目前单目 3D 目标检测的精度与应用了双目，激光雷达的算法项目仍有一定差距。自动驾驶关系到人的生命安全，算法的精度必须有所保障。
- 速度不高。目前深度学习在单目 3D 目标检测取得了一定成果，但往往需要复杂的模型与海量的参数。由于自动驾驶中，往往使用嵌入式平台来进行运算，过高的算法复杂度使得当前一些精度不错的算法在应用到实际自动驾驶时无法达到实时的效果。
- 模型结构复杂。在实际场景中，嵌入式平台往往使用 AI 芯片来对模型运算进行加速。但 AI 芯片仅仅支持一些最基础的网络层（如 2D 卷积，2D 池化等），使得一些拥有复杂模型结构（如 Self-Attention，可变形卷积）的算法无法部署在 AI 芯片上。

为了能使单目 3D 目标检测真正在自动驾驶领域落地，我们希望您可以开发出一个精度高，速度快，模型结构简单的单目 3D 目标检测深度学习算法。

2. 数据集选型：

“Kitti 3D Object Detection”数据集在 3D 目标检测领域被广泛使用，为了能与前人的算法更好地进行对比，我们希望您可以使用该数据集来进行训练与测试。由于该数据集的测试集没有标注。所以请把该数据集中的“training”文件夹中的数据划分成训练集与验证集，并在验证集上进行测试。

为了与前人的算法公平比较，请遵循论文《3d object proposals for accurate object class detection》中的做法来划分训练集与验证集。

数据集请从 Kitti 3D Object Detection 主页的“Download left color images of object data set (12 GB)”选项中下载，请不要使用额外的数据。

3、算法功能描述

我们希望您的算法可以完成以下功能：在算法的输入只有一张 RGB 图片的情况下，返回我们感兴趣的物体的 3D 信息，包括物体的类别与置信度，物体中心点的 3D 坐标，物体的实际长宽高。

针对“Kitti 3D Object Detection”这个数据集，我们希望您的算法可以返回 车 (Car)，骑行的人(Cyclist)，行人 (Pedestrian) 这三类物体的 3D 信息。

4.验收指标：

我们会根据您的算法在“Kitti 3D Object Detection”验证集上的表现来评判您的算法。希望您的算法可以满足下文将提到的精度，速度，模型结构复杂度三个指标。

5. 精度指标：

我们将会使用 3D BoundingBox 的 AP_{40} 来作为精度指标。 AP_{40} 中的 AP 即 Average Precision, 40 的含义是采用 40 个召回率点来计算 AP。同时遵循“Kitti 3D Object Detection”数据集的做法，在计算车(Car)的 AP_{40} 时，Iou 的阈值使用 0.7，在计算骑行的人(Cyclist)，行人 (Pedestrian) 的 AP_{40} 时，Iou 的阈值使用 0.5

“Kitti 3D Object Detection”数据集曾使用 3D BoundingBox 的 AP_{11} 来作为精度指标，但论文《3d object proposals for accurate object class detection》指出了该计算方法的弊端，并提出了使用 AP_{40} 来作为指标。“Kitti 3D Object Detection”数据集接受了该建议并一直沿用至今。所以为了公平比较，请您务必使用 AP_{40} 来评估您的算法。

AP_{40} 的具体计算方法可参考 Kitti 官方的 Development Kit 与论文《Disentangling Monocular 3D Object Detection》。

我们希望您的算法在“Kitti 3D Object Detection”验证集的车(Car), 骑行的人(Cyclist), 行人(Pedestrian)这三类物体的 3D BoundingBox 的 AP_{40} 可以达到以下水准。

	Moderate	Easy	Hard
Car	$\geq 21.67\%$	$\geq 27.77\%$	$\geq 18.28\%$
Cyclist	$\geq 10\%$	$\geq 13\%$	$\geq 10\%$
Pedestrian	$\geq 12\%$	$\geq 16\%$	$\geq 10\%$

6. 速度指标:

虽然我们希望您的算法可以部署到 AI 芯片上，但各种 AI 芯片的算力，优化方式各不相同，难以统一比较，并且部署到 AI 芯片上也不是一件非常容易的事。为了可以快速地公平比较，我们将会用显卡为 Nvidia-1080ti，CPU 为英伟达 i9-9900k 的电脑上测试您的算法的速度。

我们希望您的算法在显卡为 Nvidia-1080ti，CPU 为英伟达 i9-9900k 的电脑上可以达到 250FPS 的速度。

7. 模型结构复杂度指标

正如上文所述，AI 芯片仅仅支持一些最基础的网络层。我们希望您的网络仅包含以下类型的网络层。

- 2D AveragePooling

- 2D Convolution
- 2D TransposeConvolution
- 2D DepthwiseConvolution
- 2D GlobalAveragePooling
- RoiAlign
- 2D Upsample
- ChannelArgMax
- Dense
- ChannelShuffle
- ChannelSplit
- BatchNormlization
- Relu

需要注意的是,在PC端深度学习框架中一些常见的操作如Permute, View, Reshape, 并不是所有 AI 芯片都支持这样的操作, 所以希望您可以通过 Loss 的计算方式, 后处理等方面来避免使用这些操作。

二、车道线检测

1. 研究背景:

车道线检测技术在自动驾驶中扮演了一个非常重要的角色,在实时定位,路径规划,车道保持中都发挥了重要的作用。传统车道线检测往往通过颜色空间变换,霍夫变换等方法来对车道线进行检测。这种方法往往只适用于工况简单的路面。但是在真实世界中,路面情况纷繁复杂,传统检测算法的弊端渐渐显现出来。近年来,人工智能技术在车道线检测取得了非凡的成就。通过海量的数据与强大的拟合能力,深度学习车道线检测算法在环境适应性与检测精度上都大大超过了传统检测算法。

但是深度学习车道线检测技术并未非常广泛地应用到当前的自动驾驶产品中,主要是有以下的原因:

- 速度不高。目前深度学习车道线检测技术取得了一定成果，但往往需要复杂的模型与海量的参数。由于自动驾驶中，往往使用嵌入式平台来进行运算，过高的算法复杂度使得当前一些精度不错的算法在应用到实际自动驾驶时无法达到实时的效果。
- 模型结构复杂。全局信息在车道线检测中非常重要，为了获得全局信息，当前的深度学习算法往往都使用了一些比较复杂的结构，但是在实际场景中，嵌入式平台往往使用 AI 芯片来对模型运算进行加速，而 AI 芯片仅仅支持一些最基础的网络层（如 2D 卷积，2D 池化等），使得一些拥有复杂模型结构（如 Self-Attention，可变形卷积）的算法无法部署在 AI 芯片上。

为了能使深度学习车道线检测技术在自动驾驶领域得到更广泛的应用，我们希望您可以开发出一个精度高，速度快，模型结构简单的深度学习车道线检测算法。

2.数据集选型：

CULane 数据集在深度学习车道线检测算法开发中被广泛使用。它采集于北京，包含了正常路面，拥挤道路，夜晚，阴影等多种情况，非常贴近于现实的情况。为了让算法能更好的适应现实的情况，同时也为了更好的与前人的算法比较，我们希望您使用 CULane 数据集来进行深度学习模型的训练与测试。具体来说，请使用 CULane 的训练集来进行模型的训练，使用 CULane 的验证集来调整训练策略，使用 CULane 的测试集来测试您的模型。

为了与前人的算法公平比较，请不要使用额外的数据。

3.算法功能描述

我们希望您的算法可以完成以下功能：在算法的输入只有一张 RGB 图片的情况下，检测出图片中所有的车道线（其中车道线请用三次多项式来描述，并返回多项式的系数）。



假设图片中存在两条车道线，则请用 $x_1 = a_1y_1^3 + a_2y_1^2 + a_3y_1 + a_4$, $x_2 = b_1y_2^3 + b_2y_2^2 + b_3y_2 + b_4$ 两个多项式来描述这两条车道线，并返回这两个多项式的系数 (a_1, a_2, a_3, a_4) 与 (b_1, b_2, b_3, b_4)

4.验收指标:

我们会根据您的算法在 CULane 测试集上的表现来评判您的算法。希望您的算法可以满足下文将提到的精度，速度，模型结构复杂度三个指标。

5.精度指标:

参考 CULane 数据集官方的做法，我们也会使用 F1-Score 来作为精度指标。CULane 官方提供了计算 F1Score 的代码，请使用该代码来计算 F1-Score。我们希望您的算法在 CULane 测试集中各种情况的 F1-Score 以及总的 F1-Score 可以达到以下标准。（由于在“Cross”情况中不存在车道线，无法计算 F1-Score，所以“Cross”情况的指标是 False Positive，其余情况的指标都是 F1-Score。）

	Total	Normal	Crowded	Dazzle	Shadow	No Line	Arrow	Curve	Cross	Night
Metric	≥ 75.5	≥ 91.5	≥ 73	≥ 66.0	≥ 71.0	≥ 48.5	≥ 85.5	≥ 63.5	≤ 1150	≥ 69.0

6.速度指标:

虽然我们希望您的算法可以部署到 AI 芯片上，但各种 AI 芯片的算力，优化方式各不相同，难以统一比较，并且部署到 AI 芯片上也不是一件非常容易的事。为了可以快速地公平比较，我们将会 在显卡为 Nvidia-1080ti，CPU 为英伟达 i9-9900k 的电脑上测试您的算法的速度。

我们希望您的算法在显卡为 Nvidia-1080ti，CPU 为英伟达 i9-9900k 的电脑上可以达到 250FPS 的速度。

7.模型结构复杂度指标

正如上文所述，AI 芯片仅仅支持一些最基础的网络层。我们希望您的网络仅包含以下类型的网络层。

- 2D AveragePooling
- 2D Convolution
- 2D TransposeConvolution
- 2D DepthwiseConvolution
- 2D GlobalAveragePooling
- RoiAlign
- 2D Upsample
- ChannelArgMax
- Dense
- ChannelShuffle
- ChannelSplit
- BatchNormlization
- Relu

需要注意的是，在PC端深度学习框架中一些常见的操作如Permute, View, Reshape，并不是所有 AI 芯片都支持这样的操作，所以希望您可以通过 Loss 的计算方式，后处理等方面来避免使用这些操作。

三、人脸检测

1. 研究背景：

人脸检测与识别技术是一种重要的生物特征验证手段,在安全访问控制、视觉监测、智能用户接口、基于内容的图像检索等方向有着重要的应用。随着汽车智能化水平的提高，人脸检测与识别技术在汽车领域也越来越被重视。

人脸识别一般分为四个步骤人脸检测，人脸对齐，人脸表征，人脸匹配。人脸检测作为人脸识别的第一步，其重要意义不言而喻。检测不到人脸的话，识别便无从谈起。

针对汽车领域，人脸检测的对象主要是驾驶员，乘客等车内人员。由于驾驶环境不断变化，车内光照环境也不断变化。光线照射方向和角度不同，光线折射到人脸的部分阴影也不同，此时人脸的特征会发生变化，给检测带来一定难度。同时由于车内人员乘坐位置不同，摄像头安装位置不同，人脸被遮挡的问题也经常出现。

当前，深度学习算法在人脸检测任务上大放异彩，对光照变化，遮挡的问题也有非常不错的解决方案。但是针对车内环境的深度学习人脸检测算法却不多见，我们希望您可以开发出一个精度高，速度快，模型结构简单的，适合部署到车载环境的深度学习人脸检测算法。

2.数据集选型：

由于针对车内环境的人脸检测数据集较少，且不好获取。而公开数据集“WIDER FACE”中的人脸在尺度，姿态，光照、表情、遮挡方面都有很大的变化范围，与车载环境中的人脸有一定的相似之处。而且“WIDER FACE”在人脸检测的算法开发中也被广泛应用。为了更好的与前人的算法相比，且不失一般性，我们希望您使用 WIDER FACE 数据集训练与测试。

为了与前人的算法公平比较，请不要使用额外的数据。

3.算法功能描述

我们希望您的算法可以完成以下功能：在算法的输入只有一张 RGB 图片的情况下，返回图中所有人脸的 BoundingBox。其中 BoundingBox 请使用左上角坐标与右下角坐标共同表示，即（Left_Upper_x, Left_Upper_y, Right_Lower_x, Right_Lower_y）。

4.验收指标:

我们会根据您的算法在“WIDER FACE“的验证集与测试集上的表现来评判您的算法。希望您的算法可以满足下文将提到的精度，速度，模型结构复杂度三个指标。

5.精度指标:

参考“WIDER FACE“数据集官方的做法，我们也会使用 AP(Average Precision) 来作为精度指标。“WIDER FACE“官方提供了计算 AP 的代码，请使用该代码来计算 AP。“WIDER FACE“数据集中根据检测难度的不同，分成了“Easy”，“Medium”，“Hard”三种难度。我们希望您的算法在“WIDER FACE“的验证集与测试集的不同难度上的 AP 可以达到以下标准。

	AP
Val-Easy	≥ 0.972
Val-Medium	≥ 0.965
Val-Hard	≥ 0.92
Test-Easy	≥ 0.965
Test-Medium	≥ 0.958
Test-Hard	≥ 0.914

6.速度指标:

虽然我们希望您的算法可以部署到车载 AI 芯片上，但各种 AI 芯片的算力，优化方式各不相同，难以统一比较，并且部署到 AI 芯片上也不是一件非常容易的事。为了可以快速地公平比较，我们将会 在显卡为 Nvidia-1080ti，CPU 为英伟达 i9-9900k 的电脑上测试您的算法的速度。

我们希望您的算法在显卡为 Nvidia-1080ti，CPU 为英伟达 i9-9900k 的电脑上可以达到 250FPS 的速度。

7.模型结构复杂度指标

由于车载环境中往往使用 AI 芯片来对模型进行运算与加速，而 AI 芯片仅仅支持一些最基础的网络层。所以为了您的模型可以部署到车载环境中，我们希望您的神经网络仅包含以下类型的网络层。

- 2D AveragePooling
- 2D Convolution
- 2D TransposeConvolution
- 2D DepthwiseConvolution
- 2D GlobalAveragePooling
- RoiAlign
- 2D Upsample
- ChannelArgMax
- Dense
- ChannelShuffle
- ChannelSplit
- BatchNormlization
- Relu

需要注意的是，在 PC 端深度学习框架中一些常见的操作如 `Permute`, `View`, `Reshape`，并不是所有 AI 芯片都支持这样的操作，所以希望您可以通过 `Loss` 的计算方式，后处理等方面来避免使用这些操作。

四、人脸关键点检测

1. 研究背景

当前人脸关键点检测技术在各个方面有着广泛的应用。在人脸识别中，人脸关键点是人脸对齐的重要一环，直接影响着人脸的特征提取与编码，对识别率有至关重要的影响。在 DMS 领域中，人脸关键点的位置是驾驶员状态的重要判断依据。准确的检测出人脸的关键点有着重要的意义。

针对于车载环境，人脸关键点检测的对象主要是驾驶员，乘客等车内人员。由于驾驶环境不断变化，车内光照环境也不断变化。光线照射方向和角度不同，光线折射到人脸的部分阴影也不同，此时人脸的特征会发生变化，给检测带来一定难度。同时由于车内人员乘坐位置不同，摄像头安装位置不同，人脸被遮挡的问题也经常出现。而且，由于车内人员头部姿态的不同，关键点也不一定可以完整的出现在相机视野内。

当前，深度学习算法在人脸关键点检测任务上大放异彩，对光照变化，遮挡的问题也有非常不错的解决方案。但是针对车内环境的深度学习人脸关键点检测算法却不多见，我们希望您可以开发出一个精度高，速度快，模型结构简单的，适合部署到车载环境的深度学习人脸检测算法。

2.数据集选型：

由于针对车内环境的人脸关键点检测数据集较少，且不好获取。而公开数据集“300W”中的人脸在尺度，姿态，光照、表情方面都有很大的变化范围，与车载环境中的人脸有一定的相似之处。而且“300W”数据集在人脸关键点检测的算法开发中也被广泛应用。为了更好的与前人的算法相比，且不失一般性，我们希望您使用“300W”数据集来进行训练与测试。

3.算法功能描述

我们希望您的算法可以完成以下功能：输入一张只存在一个人脸的 RGB 图片，返回图中人脸的 68 个关键点的坐标。这 68 个关键点位置的定义请参考 300W 数据集。

4.验收指标：

我们会根据您的算法在“300W”测试集上的表现来评判您的算法。希望您的算法可以满足下文将提到的精度，速度，模型结构复杂度三个指标。

5.精度指标:

参考“300W”数据集官方的做法，我们也会使用 IPN（Inter-pupil Normalization）与 ION（Inter-ocular Normalization）来作为精度指标。

“300W”数据集中测试集中包含了 300w-common, 300w-Challenge, 300W-Full 三个子集，我们希望您的算法在这三个子集中的精度可以达到以下标准。

	Common	Challenge	Full
IPN	≤ 3.01	≤ 6.01	≤ 3.60
ION	≤ 2.62	≤ 4.52	≤ 3.04

6.速度指标:

虽然我们希望您的算法可以部署到车载 AI 芯片上，但是各种 AI 芯片的算力，优化方式各不相同，难以统一比较，并且部署到 AI 芯片上也不是一件非常容易的事。为了可以快速公平比较，我们将会为显卡为 Nvidia-1080ti，CPU 为英伟达 i9-9900k 的电脑上测试您的算法的速度。

我们希望您的算法在显卡为 Nvidia-1080ti，CPU 为英伟达 i9-9900k 的电脑上可以达到 250FPS 的速度。

7.模型结构复杂度指标

由于车载环境中往往使用 AI 芯片来对模型进行运算与加速，而 AI 芯片仅仅支持一些最基础的网络层。所以为了您的模型可以部署到车载环境中，我们希望您的神经网络仅包含以下类型的网络层。

- 2D AveragePooling
- 2D Convolution
- 2D TransposeConvolution
- 2D DepthwiseConvolution
- 2D GlobalAveragePooling
- RoiAlign

- 2D Upsample
- ChannelArgMax
- Dense
- ChannelShuffle
- ChannelSplit
- BatchNormlization
- Relu

需要注意的是,在PC端深度学习框架中一些常见的操作如Permute, View, Reshape, 并不是所有 AI 芯片都支持这样的操作, 所以希望您可以通过 Loss 的计算方式, 后处理等方面来避免使用这些操作。

五、人脸识别

1.研究背景:

随着汽车智能化水平的提高,汽车个性化和安全性的需求越来越强烈,汽车行业对人脸识别技术的关注度也在逐渐上升。人脸识别技术作为一种新型的人车交互方式,有着非常广阔的应用前景。

通过人脸识别技术,可以完成车辆解锁和启动,增加车辆的安全性与便捷性。与此同时,还可以通过识别车主身份,来为车主提供定制化,个性化的功能,进一步提升驾驶体验。

但是人脸识别应用在车载领域也存在一定的困难。由于驾驶环境的不同,车内环境往往面临着侧光、顶光、背光和高光等光照条件,对人脸特征的提取会造成较大影响。由于人脸图像采集的非强制性和非接触性,采集的人脸图像可能出现侧脸、低头、抬头等各种非正脸的姿态和佩戴帽子、黑框眼镜、口罩等饰物现象,加大了人脸识别的难度。

当前,深度学习算法在人脸识别任务上大放异彩,对光照变化,多变姿态等问题也有非常不错的解决方案。但是针对车内环境的深度学习人脸识别算法却不多见,我们希望您可以开发出一个精度高,速度快,模型结构简单的,适合部署到车载环境的深度学习人脸识别算法。

2.数据集

由于针对车内环境的人脸识别数据集较少，且较难获取。而 LFW（Labeled Faces in Wild）数据集，CPLFW(Cross-Pose Labeled Faces in the Wild)数据集 和 CALFW（Cross-Age Labeled Faces in the Wild）数据集中的人脸包含了不同姿态，不同年龄，不同光照等条件，与车载人脸数据有一定的相似之处，且这三个数据集在人脸识别的算法开发中也被广泛应用。为了更好的与前人的算法相比，且不失一般性，我们希望您分别使用 LFW，CPLFW，CALFW 数据集来训练与测试。

为了与前人的算法公平对比，请不要把这三个数据集的训练集加起来一起训练。请对每一个数据集单独训练与测试。

3.算法功能描述

我们希望您的算法可以完成以下功能：在算法的输入只有一张 RGB 图片的情况下，判断该图片中的人脸是否为数据库中已有的人脸，如果是，则同时返回该人脸所属的人的编号。

4.验收指标：

我们会根据您的算法在 LFW，CPLFW，CALFW 测试集上的表现来评判您的算法。希望您的算法可以满足下文将提到的精度，速度，模型结构复杂度三个指标。

5.精度指标：

参考 LFW 数据集官方的做法，我们也会使用 VERIFICATION ACCURACY 来作为精度指标。我们希望您的算法在三个数据集上可以达到以下的指标。

	LFW	CPLFW	CALFW
VERIFICATION ACCURACY (%)	≥ 99.85	≥ 93.50	≥ 96.30

6.速度指标:

虽然我们希望您的算法可以部署到 AI 芯片上,但是各种 AI 芯片的算力,优化方式各不相同,难以统一比较,并且部署到 AI 芯片上也不是一件非常容易的事。为了可以快速地公平比较,我们将会 在显卡为 Nvidia-1080ti, CPU 为英伟达 i9-9900k 的电脑上测试您的算法的速度。

我们希望您的算法在显卡为 Nvidia-1080ti, CPU 为英伟达 i9-9900k 的电脑上可以达到 250FPS 的速度。

7.模型结构复杂度指标

由于车载环境中往往使用 AI 芯片来对模型进行运算与加速,而 AI 芯片仅仅支持一些最基础的网络层。所以为了您的模型可以部署到车载环境中,我们希望您的神经网络仅包含以下类型的网络层。

- 2D AveragePooling
- 2D Convolution
- 2D TransposeConvolution
- 2D DepthwiseConvolution
- 2D GlobalAveragePooling
- RoiAlign
- 2D Upsample
- ChannelArgMax
- Dense
- ChannelShuffle
- ChannelSplit
- BatchNormlization
- Relu

需要注意的是,在 PC 端深度学习框架中一些常见的操作如 Permute, View, Reshape, 并不是所有 AI 芯片都支持这样的操作,所以希望您可以通过 Loss 的计算方式,后处理等方面来避免使用这些操作。

