龙芯中科
LOONGSON TECHNOLOGY

智能应用开发实例

提纲

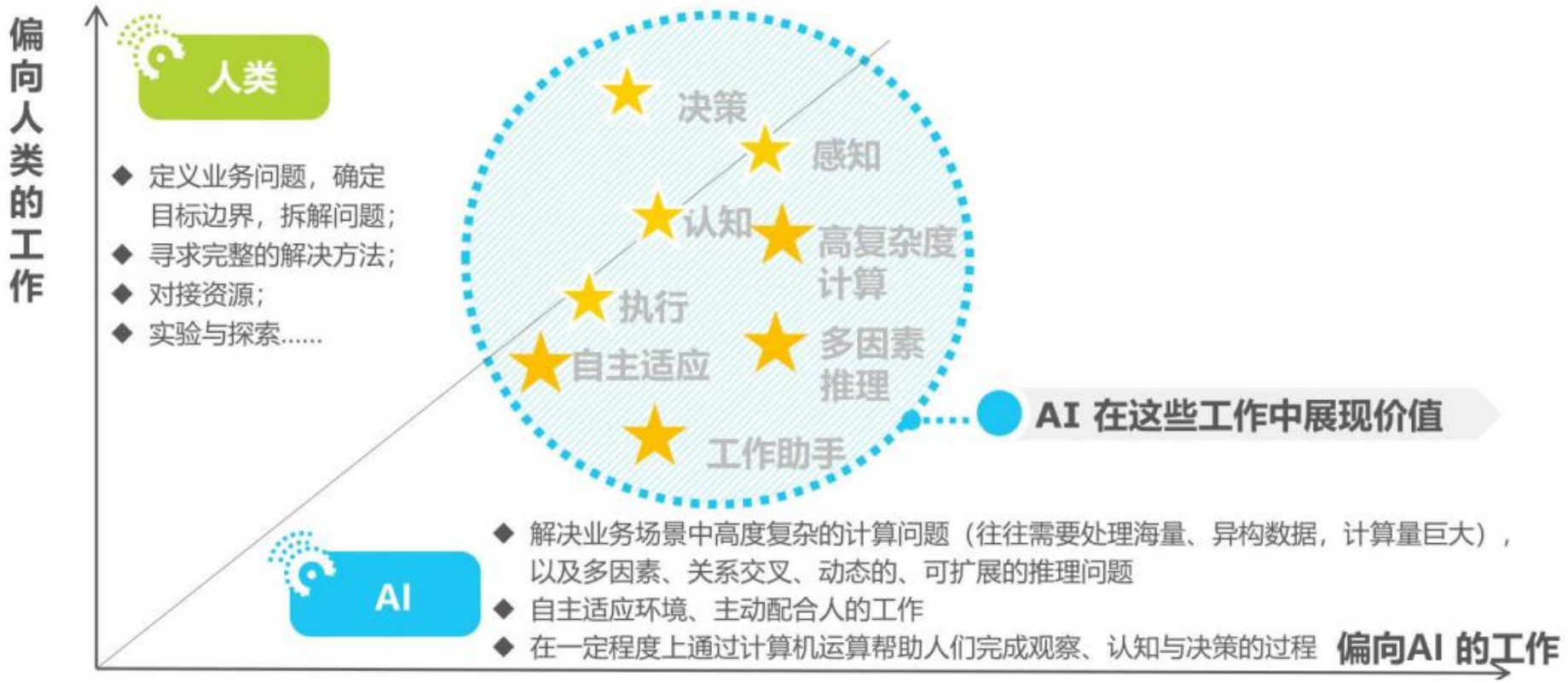
CONTENTS

一、嵌入式人工智能概述

龙芯中科 LOONGSON TECHNOLOGY

人工智能

人工智能（Artificial Intelligence）,简称 AI，即利用机器模拟人类感知、学习、认知、推理、决策、交互等过程的一门技术。



一、嵌入式人工智能概述

龙芯中科 LOONGSON TECHNOLOGY

智能化升级改造

随着人工智能技术的普及应用，传统行业都进入到了智能化升级改造的进程中，如：智能制造、智慧农业、智慧物流、智慧商务、智慧金融、智慧交通、智慧医疗、智慧养老、智慧环保等。同时人工智能技术的发展也促进了新兴产业的发展，如：智能软硬件、智能机器人、智能运载工具、智能终端、VR/AR 等。智能化的升级改造必将是未来发展的主要趋势，使用人工智能技术可以在社会的各行各业中大放异彩，由此可以更加体现出人工智能技术结合行业落地应用的重要性。

一、嵌入式人工智能应用场景

龙芯中科 LOONGSON TECHNOLOGY

人工智能结构

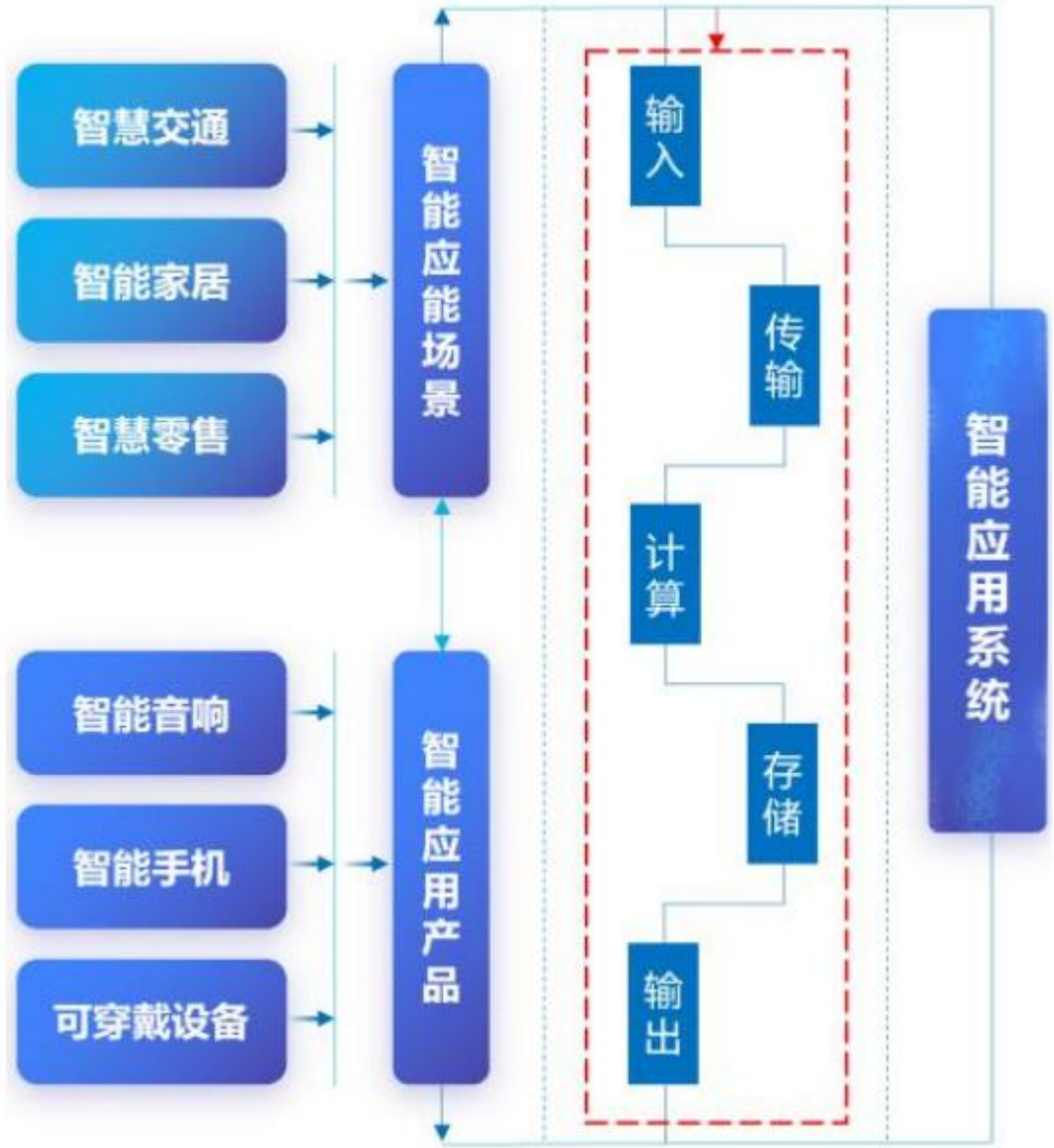
人工智能不仅仅是一项技术，而是一种认识和思考世界的方式。人工智能技术要发挥价值就必须跟行业智能化需求相匹配，针对特定的产品或应用场景做智能化升级改造，通过人工智能基础知识的积累和系统化思维的构建输出产品方案或解决方案，从而实现人工智能技术的落地应用。

一、嵌入式人工智能应用场景

龙芯中科 LOONGSON TECHNOLOGY

智能应用系统的组成

行业智能化的发展离不开人工智能技术的支撑，从系统化的思维角度出发可拆分出，构建一个完整的智能应用系统需要包含：输入、传输、计算、存储、输出五大部分。例如：智慧交通、智能家居、智慧零售、智能音箱、智能手机、可穿戴设备等都需要由这五大部分构成最终的智能系统。

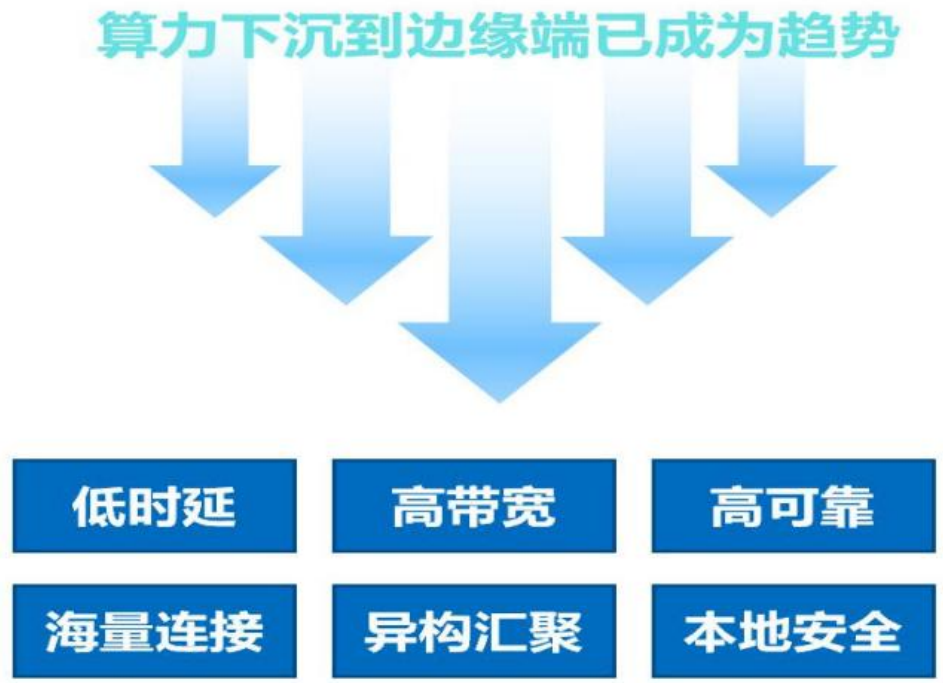


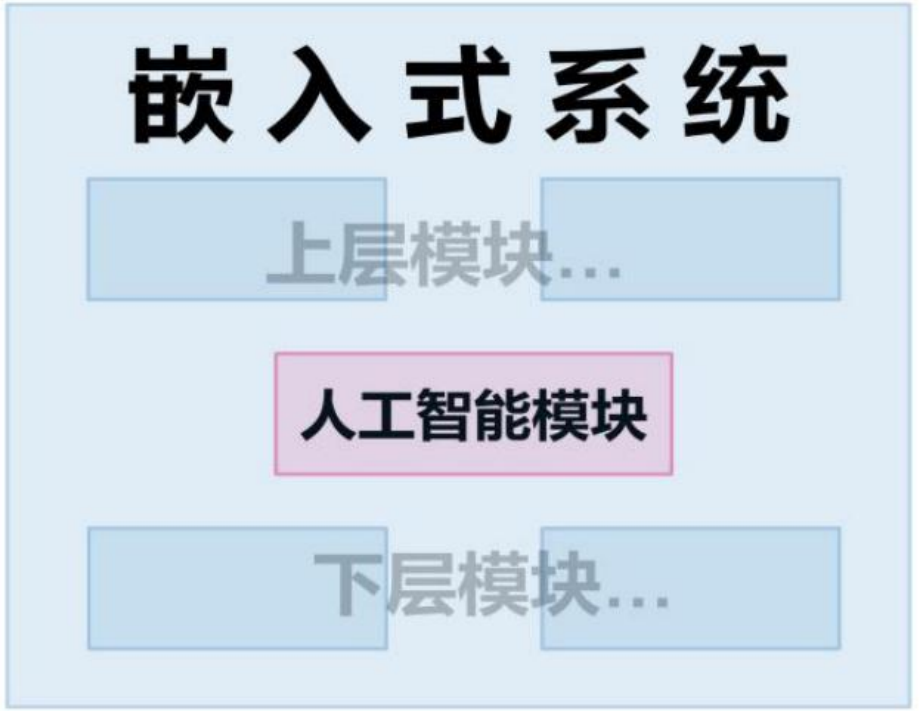
一、嵌入式人工智能应用场景

龙芯中科 LOONGSON TECHNOLOGY

人工智能

人工智能已经从技术走向应用，如何将 AI 技术真正落地，解决每个应用场景中人们的实际需求，才是关键。而这个过程中嵌入式技术将成为 AI 落地的重要承载平台，促使人工智能在各个垂直行业落地生根。嵌入式 AI：让人工智能算法可以在嵌入式智能终端设备上运行的一种技术概念。嵌入式系统是“主体”，人工智能是“装备”，强大的“属性加成”以模块来呈现，提供新功能改进现有功能，让智能更上一层楼。





二、嵌入式人工智能开发步骤

龙芯中科 LOONGSON TECHNOLOGY

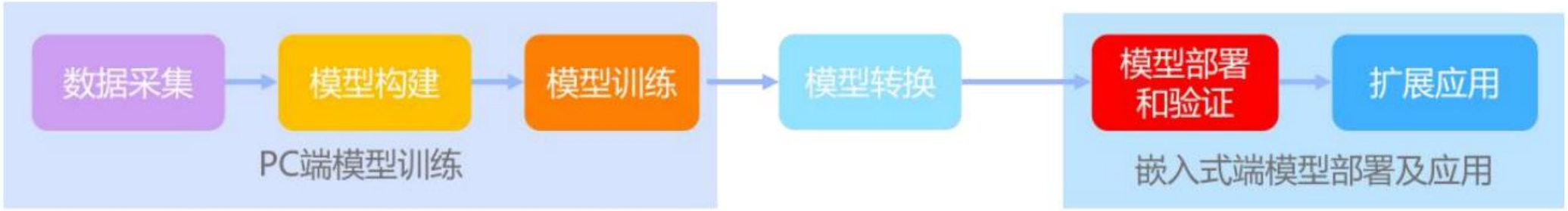
二、嵌入式人工智能开发步骤

龙芯中科 LOONGSON TECHNOLOGY

嵌入式人工智能开发流程

嵌入式人工智能开发主要包括两大部分，即人工智能模型训练和嵌入式终端模型部署。人工智能模型训练主要在 PC 端进行，包括数据处理、模型训练、模型优化等。嵌入式端主要是结合实际场景部署和应用人工智能模型。

嵌入式人工智能开发的三个环节：模型训练、模型转换、模型部署。模型训练需要在 PC 端完成，使用深度学习框架构建模型，并完成模型训练，常用的深度学习框架有：PyTorch、TensorFlow、PaddlePaddle 等；模型转换主要是进行模型优化，对模型进行剪枝、量化等操作，在损失较小模型识别精度的同时，极大的降低模型大小，便于在嵌入式端运行；模型部署根据转换后的模型编写模型推理代码，实现嵌入式端的模型部署及应用。

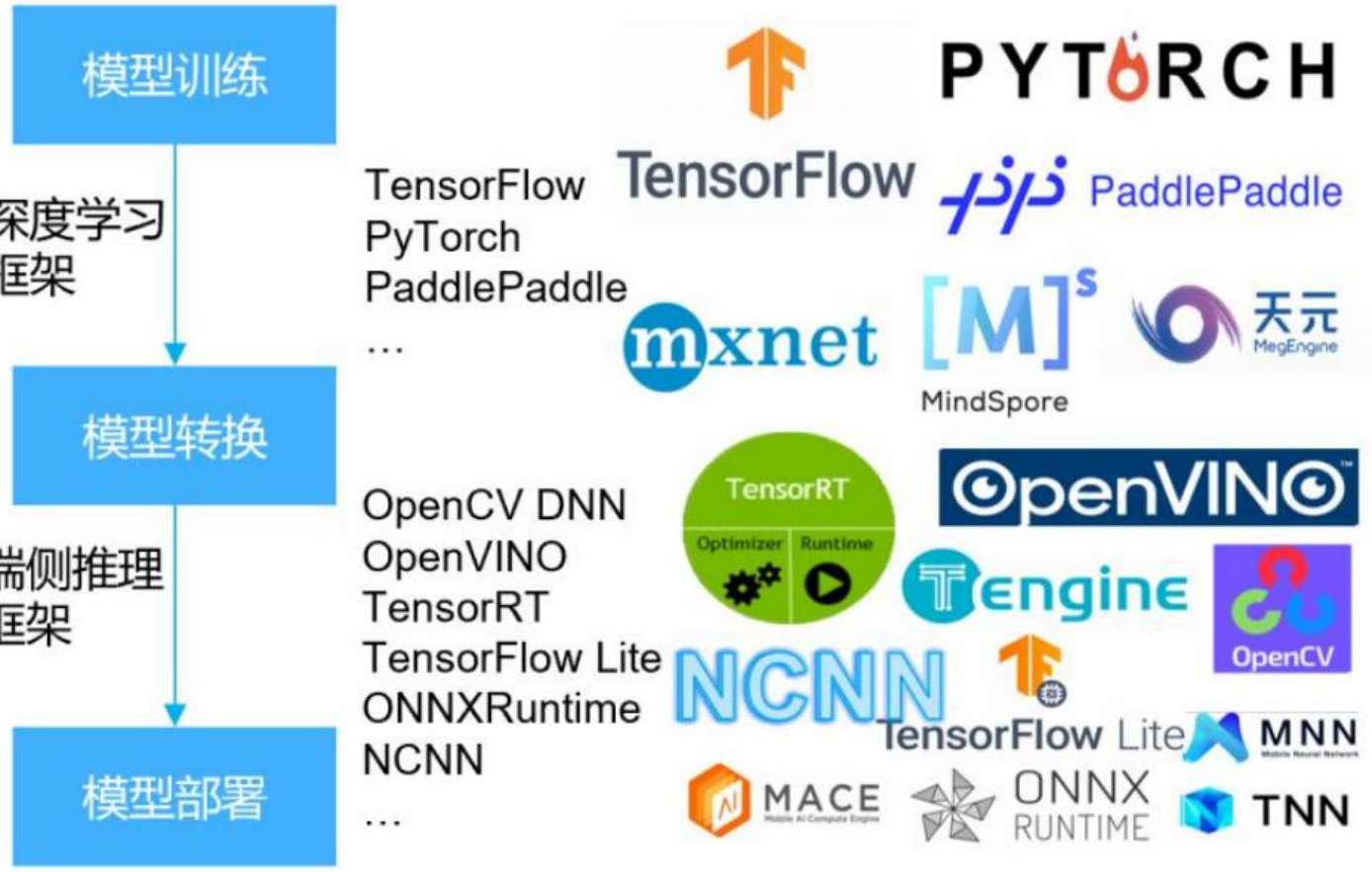


二、嵌入式人工智能开发步骤

龙芯中科 LOONGSON TECHNOLOGY

嵌入式 AI 的三个主要环节

嵌入式人工智能开发的三个环节：模型训练、模型转换、模型部署。模型训练需要在 PC 端完成，使用深度学习框架构建模型，并完成模型训练，常用的深度学习框架有：PyTorch、TensorFlow、PaddlePaddle 等；模型转换主要是进行模型优化，对模型进行剪枝、量化等操作，在损失较小模型识别精度的同时，极大的降低模型大小，便于在嵌入式端运行；模型部署根据转换后的模型编写模型推理代码，实现嵌入式端的模型部署及应用。



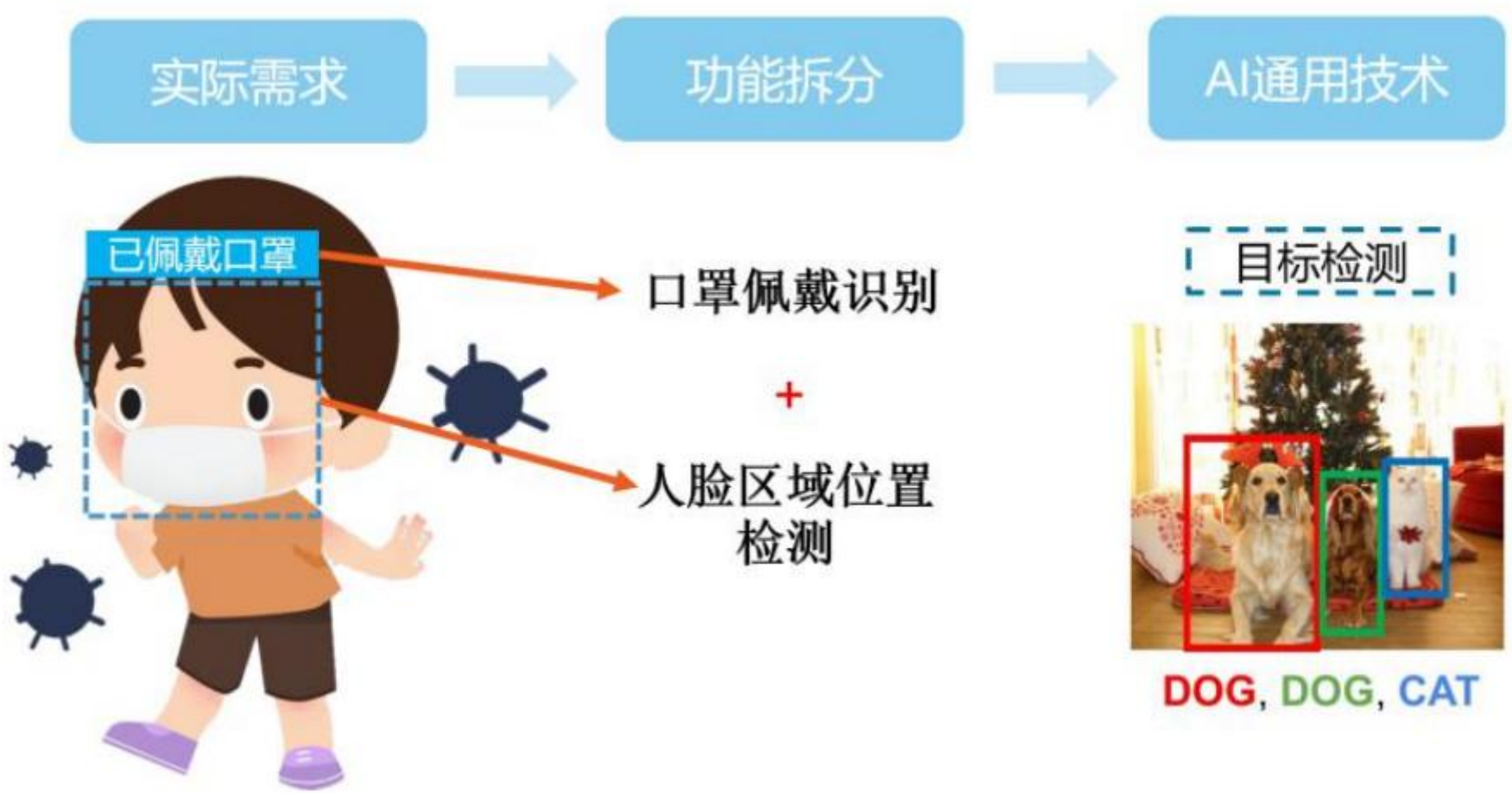
三、人脸口罩佩戴检测

龙芯中科 LOONGSON TECHNOLOGY

人脸口罩佩戴检测

佩戴口罩是疫情防控的主要措施，口罩佩戴检测在各种人流密集场所应用非常广泛，如客运中心、商超、地铁站、火车站、飞机场等。

根据深度学习的基础知识可知目标检测可以定位到物体的位置并且可识别物体的类别，可满足于人脸口罩佩戴检测的功能需求，既可以定位到图像中人脸的位置，也可是识别出该人脸是否佩戴口罩。

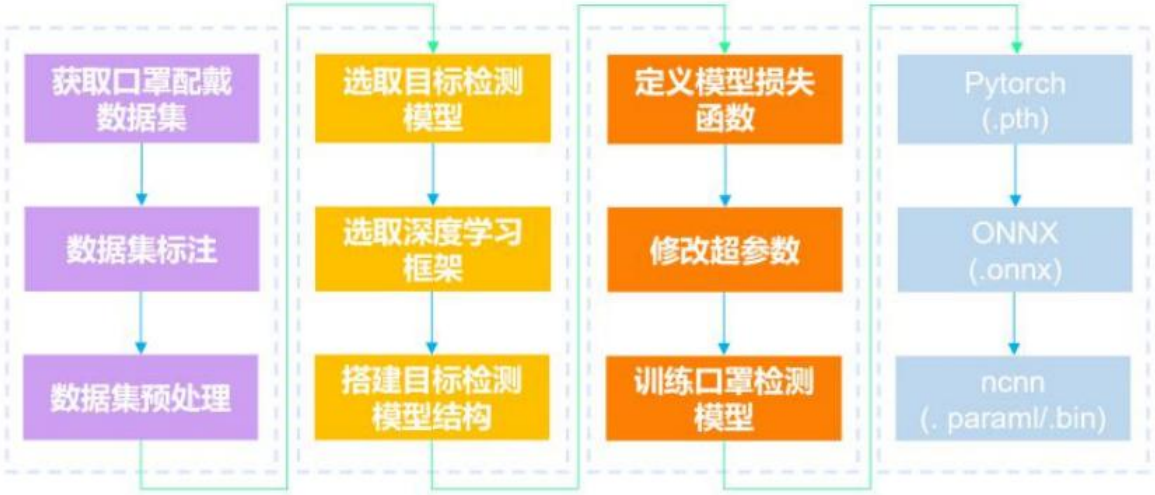


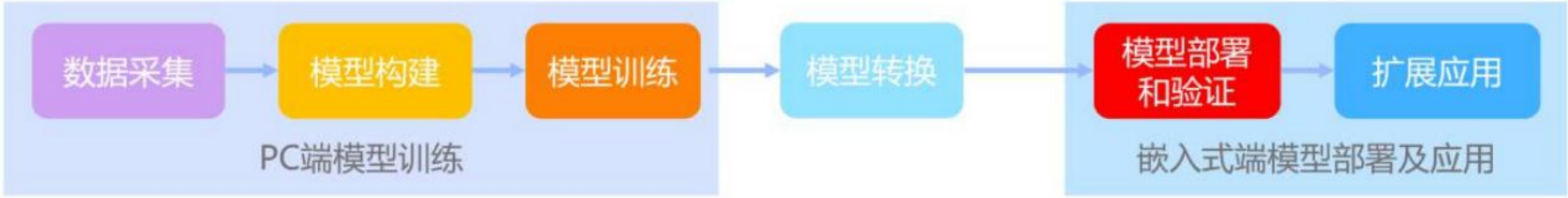
三、人脸口罩佩戴检测

龙芯中科 LOONGSON TECHNOLOGY

人脸口罩佩戴检测

人脸口罩检测嵌入式端模型部署的完整流程如下图所示，主要使用 PyTorch 深度学习框架和 NCNN 模型推理框架实现。主要分为 PC 端模型训练和嵌入式端模型部署及应用，由数据采集、模型构建、模型训练、模型转换、模型部署、扩展应用六大步骤组成。





三、人脸口罩佩戴检测

龙芯中科 LOONGSON TECHNOLOGY

3.1 环境准备

1. 上位机训练 YOLOv5 模型,采用 PyTorch 框架
2. 龙芯 2k 教育派开发板,安装:ncnn

3.2 数据集准备

使用 VOTT 数据集标注软件标注人脸口罩数据集(需要将数据集格式导出为 VOC 的数据集格式)或直接使用已标记完成的口罩数据集

三、人脸口罩佩戴检测

龙芯中科 LOONGSON TECHNOLOGY

3.3 训练模型

YOLOv5 是一种单阶段目标检测算法,该算法在 YOLOv4 的基础上添加了一些新的改进思路,使其速度与精度都得到了极大的性能提升。主要的改进思路如下所示:

输入端:在模型训练阶段,提出了一些改进思路,主要包括 Mosaic 数据增强、自适应锚框计算、自适应图片缩放;

基准网络:融合其它检测算法中的一些新思路,主要包括:Focus 结构与 CSP 结构;

Neck 网络:目标检测网络在 BackBone 与最后的 Head 输出层之间往往会插入一些层,Yolov5 中添加了 FPN+PAN 结构;

Head 输出层: 输出层的锚框机制与 YOLOv4 相同,主要改进的是训练时的损失函数GIOU_Loss,以及预测框筛选的 DIOU_nms。

三、人脸口罩佩戴检测

3.4 实现步骤

YOLOv5 是通用的目标检测模型,只需更换数据集在预训练模型的基础上重新训练即可实现口罩检测功能。注意:模型训练需使用 PC 主机完成。具体实现步骤如下:

(1)环境配置

安装必要的 python package 和配置相关环境

(2)创建数据集的配置文件 dataset.yaml

Dataset.yaml 主要指明了数据集的路径信息,可以基于该 yaml 修改自己数据集的 yaml文件。

(3)数据集格式转换

需要将数据集标注转换为和 darknet format 相同的标注形式,每一个图像生成一个*.txt的标注文件(如果该图像没有标注目标则不用创建*.txt 文件)。创建的*.txt 文件遵循如下规则:

每一行存放一个标注类别

每一行的内容包括 class x_center y_center width height

Bounding box 的坐标信息是归一化之后的(0-1)

class label 转化为 index 时计数是从 0 开始的

3.4 实现步骤

(4)选择模型,修改相应的配置文件

yolov5 官方一共提供了 yolov5s.yaml、yolov5m.yaml、yolov5l.yaml、yolov5x.yaml 四个不同大小的模型,他们的精度与速度也各有不同

(5)模型训练

在 train.py 文件里修改相应的训练参数:batch、epoch、cfg、data 等参数,运行如下命令训练模型:

```
python3 train.py --img-size 640 --batch-size 16 --epochs 300 --data ./data/score.yaml
--cfg ./models/score/yolov5s.yaml --weights weights/yolov5s.pt
```

(6)模型测试

在 detect.py 里自行按需修改,主要是要修改测试图片和视频的路径

3.5 模型部署

在龙芯 2K1000 中部署 YOLOv5 模型,使用 NCNN 框架,所以需要将 YOLOv5 的 PyTorch 模型转换为 NCNN 所需的模型格式。

(1)YOLOv5 PyTorch 模型转 ONNX

在 YOLOv5 的 git 项目里有自带的一个 onnx_export.py 文件,运行该文件,即可将 pt 模型转为 ONNX 模型,需要根据模型配置修改图像输入大小。

转换成功后输出 ONNX 格式模型,但是转 NCNN 模型格式还需要对模型做一个 Simplifier 操作,因为转出的 ONNX 模型还有许多冗余,在 NCNN 里是不支持的,避免转 NCNN 时报错。

可使用如下命令优化模型,其中 input_onnx_model 是你需要简化的 onnx 的路径,out_onnx_model 就是输出简化后模型的路径。

```
python3 -m onnxsim input_onnx_model out_onnx_model
```

三、人脸口罩佩戴检测

龙芯中科 LOONGSON TECHNOLOGY

3.5 模型部署

(2)ONNX 模型转 NCNN 模型

在 PC 主机中编译安装 NCNN,安装成功后使用 onnx2ncnn 命令完成模型转换操作,其中 yolov5s.onnx 即是需要转为 NCNN 的 ONNX 模型,yolov5s.param 和 yolov5s.bin 即为转为NCNN 后输出的两个文件。

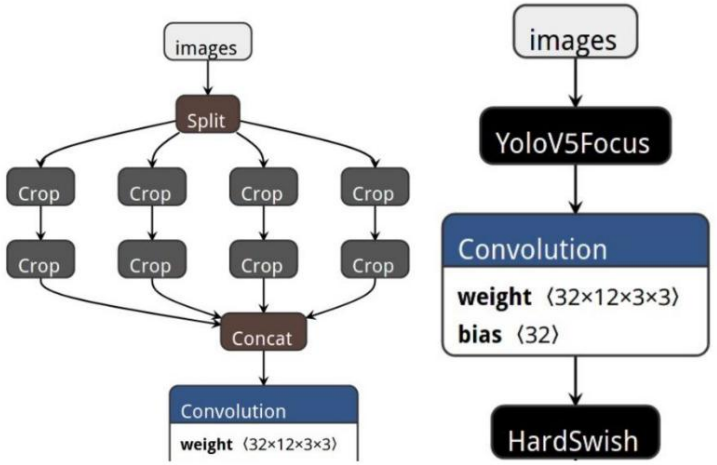
./onnx2ncnn yolov5s.onnx yolov5s.param yolov5s.bin

运行完成之后会出现很多 Unsupported slice step,

这是 focus 模块转换的报错,需要修改NCNN 的模型结构

yolov5s.param 文件,去除 split 模块中的 op,直接使用自

YoloV5Focus 代替。



替换后用 ncnnoptimize 优化一遍模型,转为 fp16 存储减

ncnnoptimize yolov5s.param yolov5s.bin

yolov5s-opt.param yolov5s-opt.bin 65536

```
7767517
192 216
Input            images           0 1 images
YoloV5Focus      focus            1 1 images 207
Convolution      Conv_41          1 1 207 208 0=32 1=3 4=1 5=1 6=3456
HardSwish        Div_49           1 1 208 216 0=1.666667e-01
Convolution      Conv_50          1 1 216 217 0=64 1=3 3=2 4=1 5=1 6=18432
HardSwish        Div_58           1 1 217 225 0=1.666667e-01
Split            splitncnn_0      1 2 225 225_splitncnn_0 225_splitncnn_1
Convolution      Conv_59          1 1 225_splitncnn_1 226 0=32 1=1 5=1 6=2048
HardSwish        Div_67           1 1 226 234 0=1.666667e-01
```

三、人脸口罩佩戴检测

龙芯中科 LOONGSON TECHNOLOGY

3.5 龙芯 2K1000 部署

(1)将转换完成的模型文件拷贝至开发板中将模型文件拷贝至开发板~/ncnn/build/examples 文件夹中

(2)修改 NCNN 中 yolov5 的模型推理代码

cd 到 ncnn\examples 路径中,使用 vim 打开 yolov5.cpp 文件,修改模型名称和路径,

重新编译 yolov5.cpp,生成可执行文件。

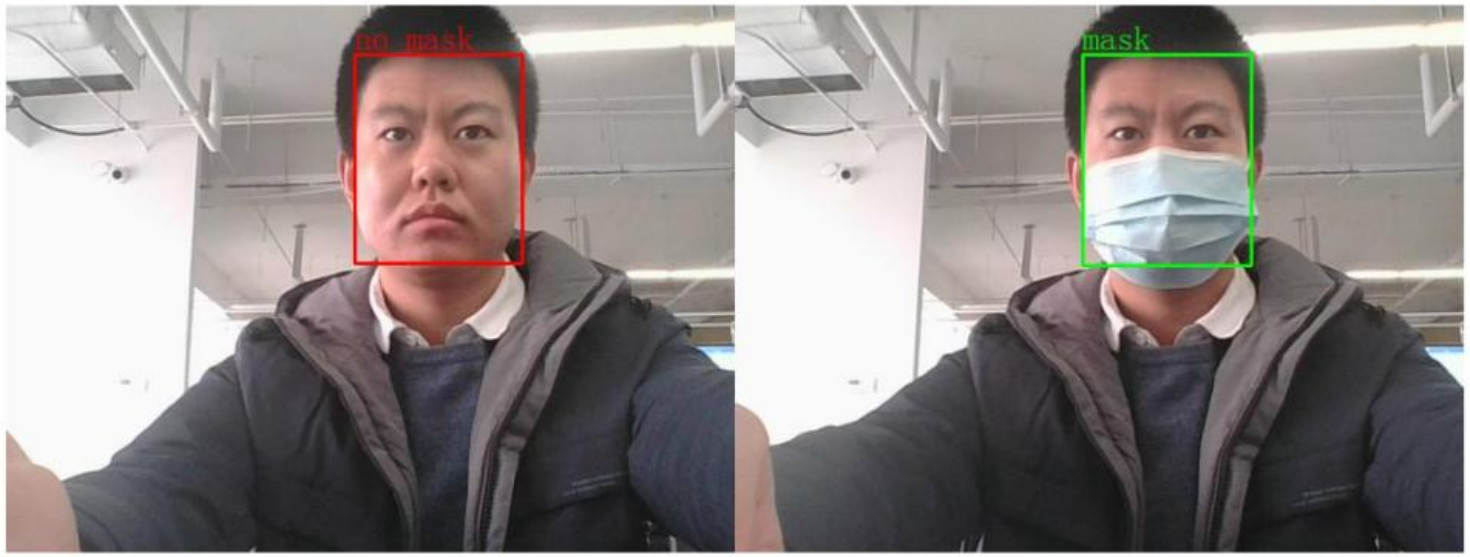
进入的 ncnn 文件夹中的 build 文件夹,重新编译

生成 yolov5 的可执行文件。

```
cd ../build/
make -j2
```

(3)龙芯 2K1000 口罩检测模型运行结果

```
cd /examples/
./yolov5 no_mask.jpg
./yolov5 mask.jpg
```



六、小任务

龙芯中科
LOONGSON TECHNOLOGY

- 1、口罩识别

感谢观看！