



## 水下机器人目标识别系统

课程名称:	机器视觉
专 业:	计算机科学与技术
班 级:	21 计科人工智能 1
学 号:	2122550079
姓 名:	颜孙炜

2024 年 6 月

# 目 录

摘要 .....	1
1 选题背景 .....	1
2 开发环境 .....	2
3 数据处理 .....	3
4 模型设计 .....	5
5 模型训练 .....	8
6 模型评测 .....	10
7 机器人的结构设计和运动控制 .....	11
7 课程感想 .....	13
8 参考文献 .....	14
附录 .....	15

## 摘要

中国拥有丰富的水下渔业资源，对粮食安全和生态环境具有重要意义。利用高科技水下勘测设备可加强对这些资源的科学开发。然而，海底复杂环境、生物自身特性和模型复杂性使得水下生物识别面临挑战。基于第八届浙江省水下机器人竞赛，本课程作业通过自主式水下潜器（AUV）搭载相机拍摄水下图像，采用图像增强和轻量模型深度学习算法，实现对海洋生物资源的检测和识别。使用机器视觉功能进行基础的图像切割和画面的优化，比较了 yolov5 和 MobileNetV2 两种模型在相关任务中的性能。

**关键词：**AUV，图像处理，图像增强，深度学习，轻量模型。

## 1 选题背景

中国拥有丰富的水下渔业资源，包括海胆、贝类和海参等多种水生动物。这些资源对于保障国家粮食安全、维护海洋生态环境都具有重要意义<sup>[1]</sup>。通过高科技水下勘测设备可加强对水下渔业资源的勘测以便于科学开发和利用。

基于国内外研究现状，海洋生物的水下识别存在以下难点<sup>[2]</sup>：

- (1). 海底复杂环境影响水下图像成像质量，阻碍网络特征的提取。
- (2). 海洋生物由于尺寸、颜色等自身因素的影响，提高了水下目标识别的难度。
- (3). 受限于网络模型的复杂，增加了水下机器识别的实时性检测负担。

本次作业基于第八届浙江省水下机器人竞赛，通过自主式水下潜器 (Autonomous Underwater Vehicle, 简称 AUV)，搭载水下相机对水下图像进行拍摄，然后运用图像增强算法和基于轻量模型深度学习的目标检测算法对增强图像中的渔业资源进行检测和识别。

## 2 开发环境

本项目设计的水下机器人水下识别系统是一套完备的机器人视觉系统,如图,该机器人包括机器人系统和视觉系统,其中机器人部分采用 STM32 作为主控芯片,搭载多种传感器,使用 KEIL5 进行编程。机器视觉部分基于 K210 在 maixpy 上进行开发。深度学习部分基于 pytorch 使用 python 语言进行开发。

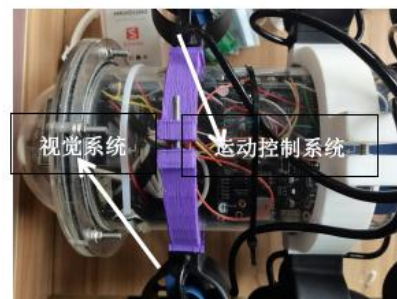


表 1 语言及库对应表

平台	STM32	K210	Mobilenet	Yolov5
语言	C#	Micropython	Python	Python
库	HAL	MACHINR, CV.....	Pytorch.....	Pytorch

### 3 数据处理

**数据介绍：**

第八届水下机器人竞赛视觉识别要求中要求选手对三部分进行识别，分别是铝条轨迹(可以近似为黑线), 传统方形色块(红蓝绿)，海洋生物图片块(鲨鱼，海龟，八爪鱼)。



图 1 水下机器人识别对象（实拍）

**数据处理：**

**(1) 补光：**

由于海底普遍由于光线分布不均匀，水波冲击相对较大等问题，需要一种能提升水下采样亮度和清晰度的方案。在本次项目中，采用了软件和硬件两种方式实现水下摄影采样的清晰化。

- ①在硬件上采用了补光的方式，在摄像头附近放置了如图一块大功率 led 灯片，类似拍摄时的闪光灯，可以照料目标地区。



图 2 • 大功率 led 灯片

- ②在软件上实现了软补光的方式，打开的摄像头的白平衡模型，使用软件算法提高亮度。

表 2 对照补光实验结果

（夜间）未补光	夜间）硬补光	夜间）软硬补光
---------	--------	---------



可以看出通过软硬件同时补光，图形的清晰度和亮度有了显著的提升。

**(2) 采样:**

由于海底普遍由于光线分布不均匀,即使在采样时通过软硬件进行优化，但采样与水面上实拍任存在较大区别，因此，在竞赛中，为了使模型能更准确的进行判断，我们进行水下采样，将水下采样数据用于训练。组委会提供每种海洋生物各 8 张，通过旋转, 镜像, 随机模糊等数据增强方式共采样海洋图片 144 张。章鱼，鲨鱼，海龟各 48 张。

**(3) 标注:**

本项目对照了目标检测和分类，故采用了两种标注方式，一是分类，主要将不同海洋生物的样本分类到不同文件夹下，二是打标签，如下图通过 `labelimg` 进行标注。标注的格式为 `yolo` 格式。数据结构如下：



```

score
|..└-images
|..|..└-test
|..|..└-train
|..|..└-val
|..└-labels
|.....└-test
|.....└-train
|.....└-val

```

## 4 模型设计

本任务中巡线(铝条), 色块识别使用了较为成熟的 `opencv` 进行处理, 而海洋生物使用则采样了深度学习方法进行分析, 以下是对各方法的说明:

### (1) 机器视觉方法:

#### ①巡铝条:

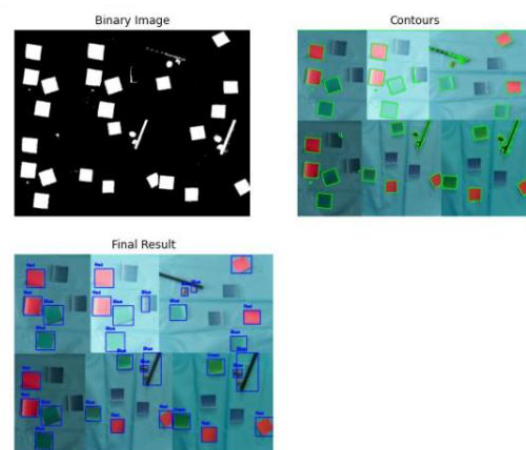
如右图, 铝条在水下可以近似为黑色直线, 因此, 巡铝条任务可以近似为寻黑色线任务, 本任务中通过调用 `image` 模块中现有查找色块方法, 得到偏移量和关键点(如十字路口, 起停线等)的数据, 返回数据给 `STM32f103` 单片机进行巡线, 其核心是通过 `find_blobs` 函数可以找到色块, 其中的 `[green_threshold]` 寻找目标是颜色的阈值, 阈值可以通过 `Maixpy IDE` 工具栏中的工具 -> 机器视觉 -> 阈值编辑器获得。其中 `find_blobs` 返回参数如下表:



外框 y 坐标 (int)	b[1]
外框的宽度 w (int)	b[2]
外框的高度 h (int)	b[3]
像素数量 (int)	b[4]
中心 x 坐标 (int)	b[5]
中心 y 坐标 (int)	b[6]

#### ②色块识别:

如右图, 水下识别的任务需求为在一组水下拍摄的视频中逐帧识别, 即一套算法在应对光照变化、颜色失真等问题下依然可以正常识别, 为模拟水下的实践识别需求, 本题目数据由多组色块在泳池中的实拍合成, 不同组别色块环境(光照角度)不同, 为解决此问题, 这





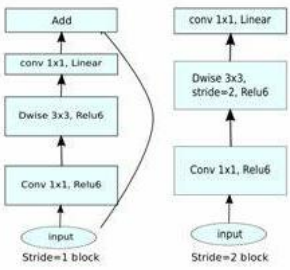
里先去除浅蓝色背景，阈值为 `lower_bg=np.array[90,50,50]`; 和 `upper_bg=np.array[160,255,255]`.之后采样矩形识别，设置大小阈值等排除小干扰，最后通过阈值比较每个色块最接近什么颜色。

(2) 深度学习方法:

① MobileNetV2:

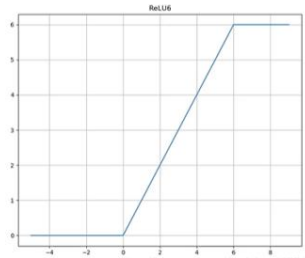
在竞赛中，由于水下自然环境的影响，不仅对图像检测的准确性有着极高的要求，在检测速度方面，也希望能够做到实时监测，因此针对轻量模型的研究也是不可或缺的，故在本次竞赛中选择实验 MobileNetV2 这一轻量模型作为模型训练的主干网络。以下是对 MobileNet v2 的介绍:

MobileNet v2 网络是由 google 团队在 2018 年提出的，相比 MobileNet V1 网络，准确率更高，模型更小。网络中的亮点：Inverted Residuals（倒残差结构）Linear Bottlenecks（结构的最后一层采用线性层）



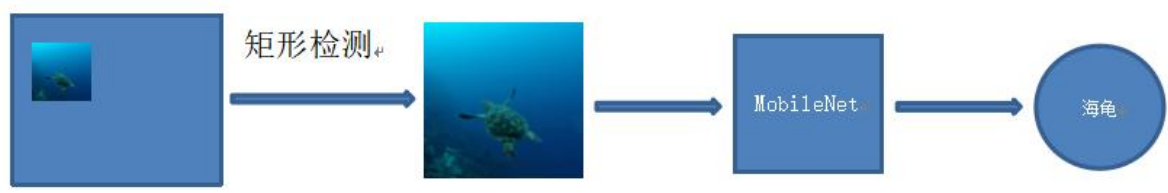
(d) Mobilenet V2

$$y = \text{ReLU6}(x) = \min(\max(x, 0), 6)$$



其中，在 MobileNetV2 结构中，采用了新的激活函数：ReLU6。（在 MobileNetV1 和许多其他神经网络架构中，使用的激活函数是标准的 ReLU, ReLU6 对输出的上限进行了约束，将其最大值限制为 6。）

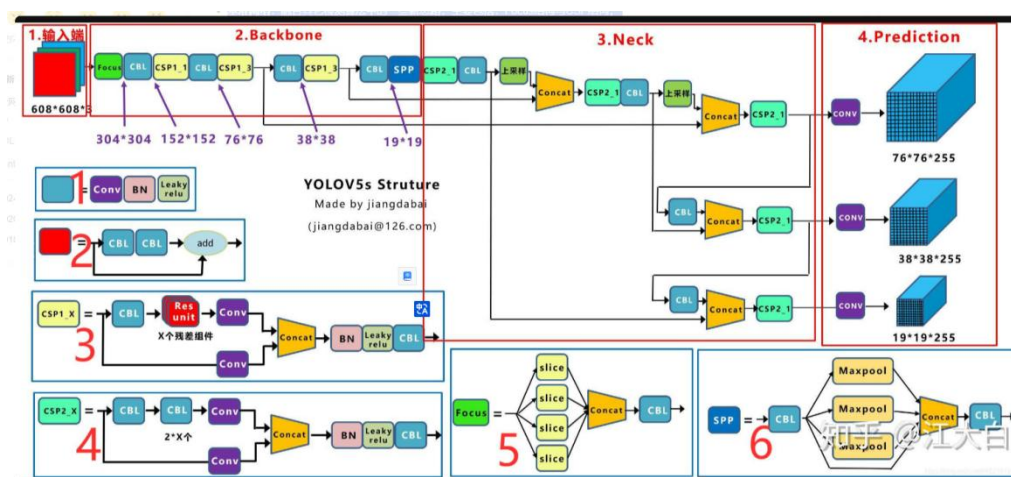
由于在竞赛中无明显干扰，细小的干扰可以通过调节阈值避免，为提升性能，并保证模型的轻量化，使用模型进行分类任务，使用了如下流程，先使用 opencv 进行矩形检测，在将矩形部分送入神经网络获取推演结果。



## ②Yolov5:

当面对真实海洋生物时，海洋生物的形状均为不规则形状，并且在真实海洋环境下，存在许多干扰因素，因此本实验中，另外选取了 yolov5 作为目标检测的模型网络，希望在小量数据下实现目标的检测定位。

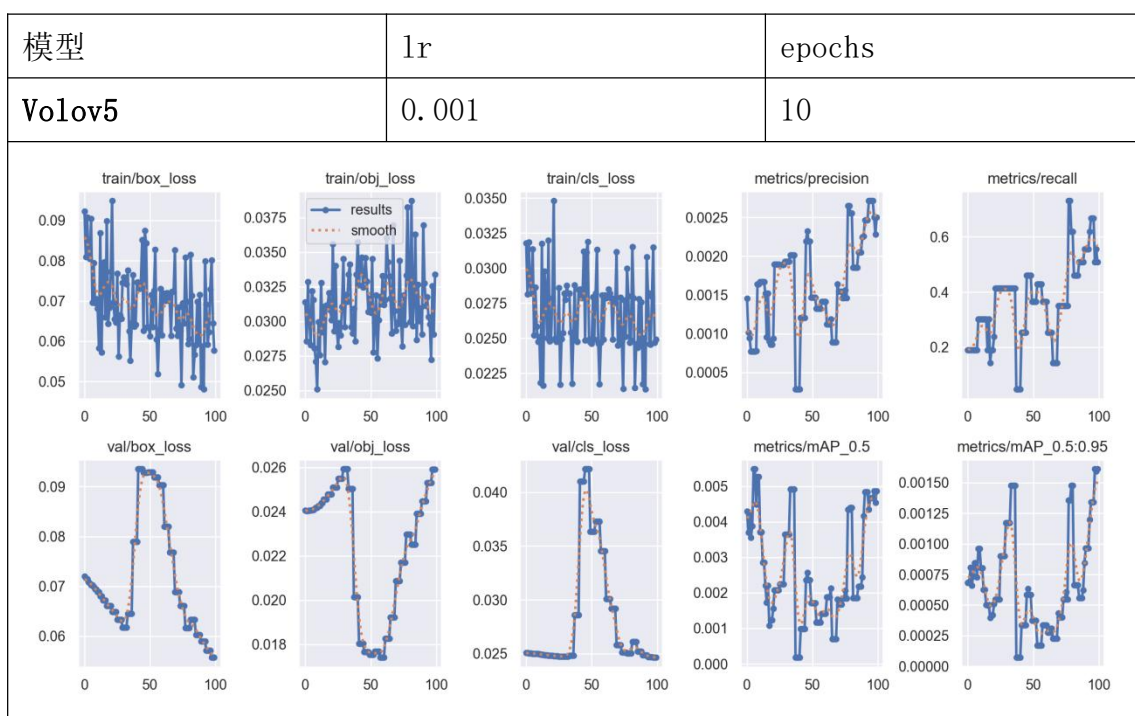
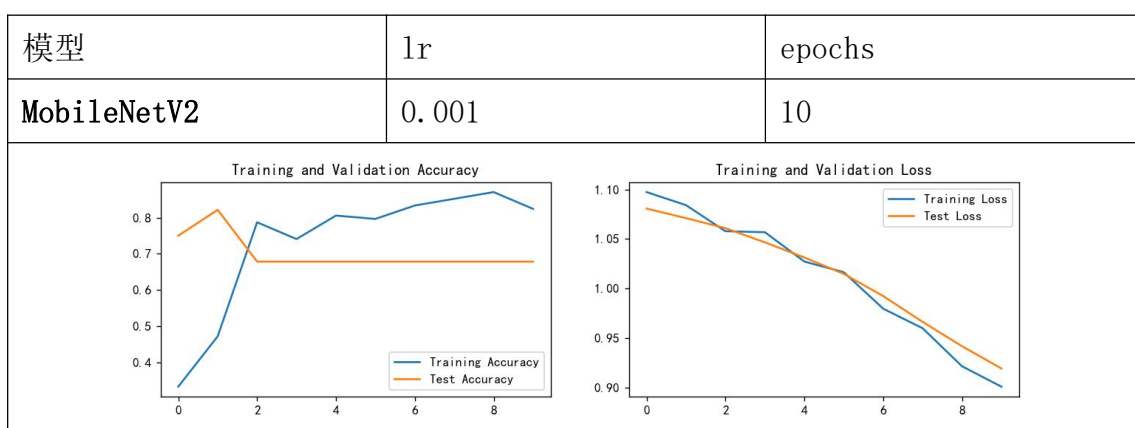
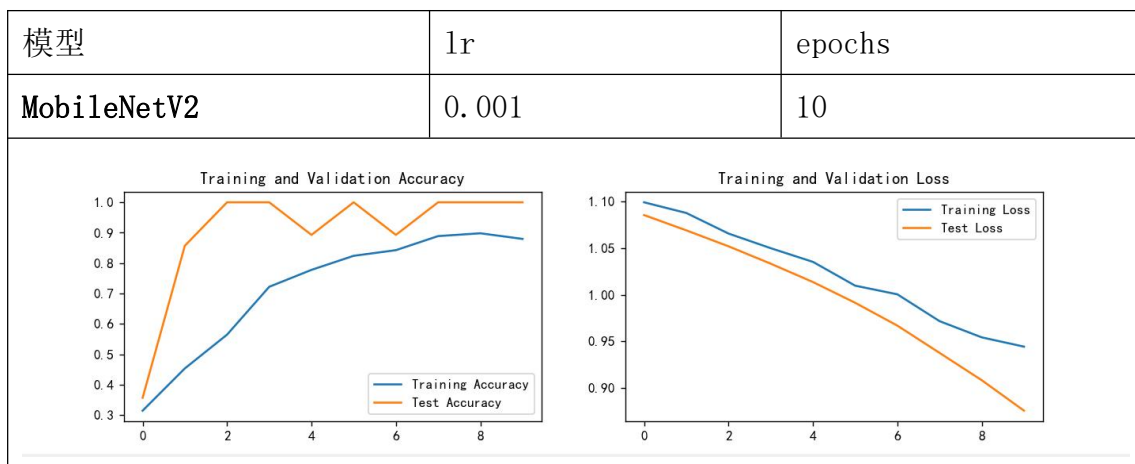
YOLOv5 是一种单阶段目标检测算法，该算法在 YOLOv4 的基础上添加了一些新的改进思路，使其速度与精度都得到了极大的性能提升。主要的改进思路如下所示：



1. 输入端：包括 Mosaic 数据增强、自适应锚框计算、自适应图片缩放。
2. 基准网络：包括：Focus 结构与 CSP 结构。
3. Neck 网络：Yolov5 中添加了 FPN+PAN 结构。
4. Head 输出层：改进损失函数 GIoU\_Loss，以及预测框筛选的 DIOU\_nms。

使用 yolov5 算法，在一定量的数据训练下，模型能够高效，快速的去实现对特定几类目标的检测，包括确定目标位置和分类结果。

## 5 模型训练



在本次的模型训练中，我们选择了两种不同的模型，分别是 yolov5 和 MobileNetV2，同时还进行了不同的学习率(lr)值的对比实验。

实验中采用了数据增强技术和迁移学习技术：

数据增强技术：随机水平翻转

```
train_transforms = transforms.Compose([
    transforms.Resize([224, 224]), # 将输入图片resize成统一尺寸
    transforms.RandomHorizontalFlip(), # 随机水平翻转
    transforms.ToTensor(), # 将PIL Image或numpy.ndarray转换为tensor，并归一化到[0,1]之间
    transforms.Normalize(
        mean=[0.485, 0.456, 0.406], # 标准化处理-->转换为标准正太分布（高斯分布），使模型更容易收敛
        std=[0.229, 0.224, 0.225]) # 其中 mean=[0.485,0.456,0.406]与std=[0.229,0.224,0.225] 从数据集
])
```

数据增强技术

迁移学习：使用大型标准数据集对模型进行训练，之后将模型学习到的通用特征的知识迁移到目标任务中，此时再利用目标任务的样本对迁移的知识进行微调并解决具体的问题。

```
model = models.mobilenet_v2(pretrained=True).to(device)

for param in model.features.parameters():
    param.requires_grad = False

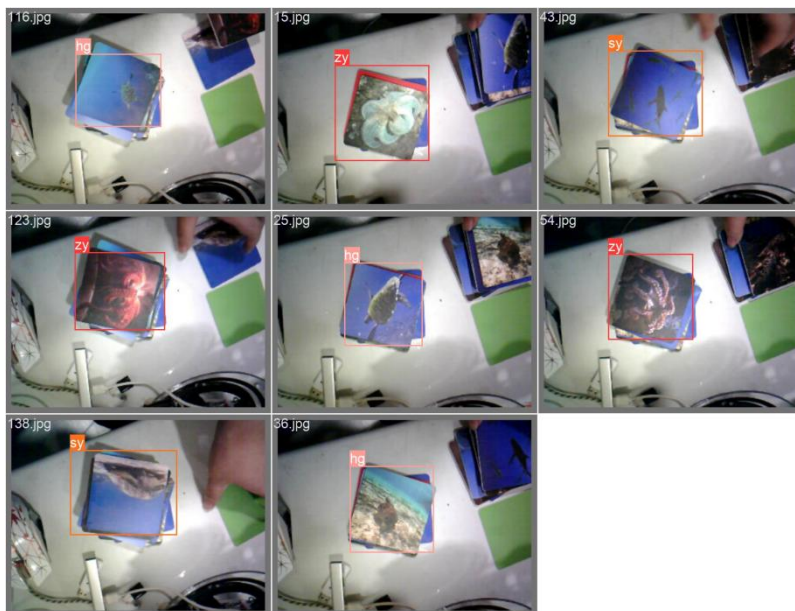
num_classes = len(classNames)
model.classifier[1] = nn.Sequential(
    nn.Linear(model.classifier[1].in_features, 512),
    nn.ReLU(),
    nn.Dropout(0.5),
    nn.Linear(512, 256),
    nn.ReLU(),
    nn.Dropout(0.5),
    nn.Linear(256, num_classes),
    nn.Softmax(dim=1)
)
```

迁移学习技术

## 6 模型评测

yolov5 的目标检测功能和 MobileNetV2 的分类功能，都在一定程度上实现了竞赛的基本要求。

在准确率上，两个模型都实现了较为精准的预测，



其中 MobileNetV2 在数据集较少情况下无法获得较高正确率，而 yolov5 模型在少量进行标注的数据集下也体现出较好的预测结果。

MobileNetV2 中的算子全部兼容 k210 提供的 SDK，可在虚拟环境中通过 onnx 转为 kmodel，通过 k210 进行推演使用，而 yolov5 需要重新替换主干网络 darknet 后才能转换，且对硬件要求较高，一般部署在树莓派等 soc 开发板上，且需要注意散热。

模型	MobileNetV2	yolov5
准确率	约等于 93%	约等于 70%
数据量	多（远大于提供数量）	少（组委会提供数量）
复杂程度	简单	负责
配置要求	要求低	要求高
稳定程度	良好	好

综上所述：MobileNetV2 作为轻量级模型适合部署在 k210 等 AI 计算板上，而 yolov5 模型更为稳定，且在小数据下更为合适部署在高性能的 soc 开发板上。



## 7 机器人的结构设计和运动控制

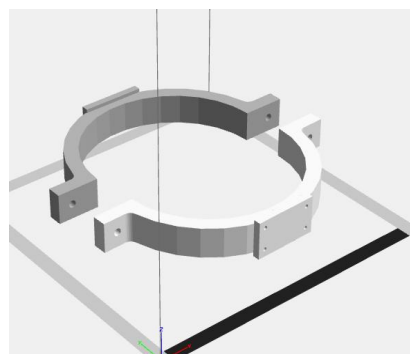
以上介绍了该机器人系统的视觉系统，本节将详细描述水下机器人的结构设计和运动控制部分。

水下机器人，特别是自主水下机器人（AUV），常见的设计有飞碟式和潜艇式两种。飞碟式机器人的设计具有稳定的重心，其结构类似飞碟，重心位于中心，对机械控制的要求较小。这种设计的优点在于其稳定性，但由于摄像头通常安装在机器人下方，缺乏前瞻性。这种摄像头布局限制了机器人的视野，使其在识别前方障碍物或路径变化时反应不够及时，因而对巡线算法的控制要求较高，需要更加复杂的算法来保持精确的路径跟踪。



相比之下，潜艇式机器人设计尽管在重心上不如飞碟式稳定，但它具有显著的前瞻性优势。潜艇式机器人通常将摄像头布置在前端，这使得机器人能够提前识别前方的路径和障碍物，提供更好的环境感知能力。尽管这种设计重心不稳定，但通过精密的控制系统和算法，潜艇式机器人可以在偏离路径时迅速调整姿态回到正确轨迹上，从而提高路径识别的成功率。鉴于这些优点，本系统选择采用潜艇式设计。

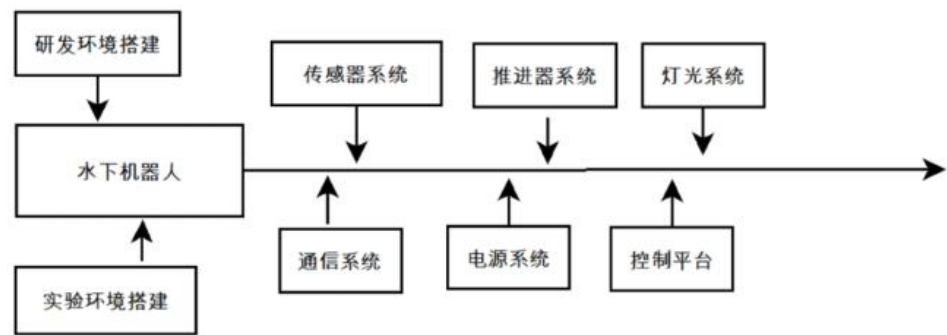
为了进一步优化潜艇式机器人的结构设计，本系统在设计中加大了机器人与水面的接触面积，以达到重心左右平衡。具体而言，本系统设计了三组支架，这些支架并非一体成型，而是采用两两拼接的形式。这种设计方法确保了支架的灵活性和稳定性，使螺旋桨始终保持在水平线上，从而提升了机器人的运动稳定性。



在制作工艺上，使用圆角，倒角等技术，加强了该支架的强度和稳定性。

在运动控制部分，系统选择了 STM32F 系列芯片，该芯片性能强大且稳定，适用于复杂的控制任务。为了实现实时控制，系统搭载了 FreeRTOS 实时操作系统。FreeRTOS 以其高效的任务管理和响应能力，确保机器人在不同操作环境下都能快速响应。此外，机器人配备了陀螺仪和深度计两个传感器。陀螺仪用于实时检测机器人的姿态变化，而深度计则用于监测机器人在水中的深度，提供重要的状态反馈。

通信系统是机器人系统中至关重要的一部分，用于实现上位机和下位机之间的数据传输和控制信号的交换。如图，在本系统中，上位机使用先进的计算机视觉算法对摄像头采集到的图像进行处理和分析，识别出水下的路径和障碍物信息。识别结果通过串口实时传输给下位机。下位机接收到来自上位机的路径信息后，根据这些信息调整机器人的运动轨迹。



控制算法方面，系统采用了如图多级 PID（比例-积分-微分）控制算法。PID 算法是一种经典的控制算法，通过调节比例、积分和微分三个参数，实现对系统的精确控制。多级 PID 算法进一步优化了控制过程，分多个阶段对机器人进行精确的运动调节。在机器人平衡控制方面，PID 算法通过调整机器人的姿态传感器数据，使其始终保持稳定状态；在高度控制方面，PID 算法根据深度计的数据，精确调节机器人的垂直位置；在前进姿态控制方面，PID 算法结合多传感器数据，确保机器人在前进过程中保持正确的姿态和方向。



## 7 课程感想

通过本课程的学习，我们掌握了关于人工智能理论的一些重要知识，并将这些理论知识转化为实践，通过训练自己的模型来实现具体的任务。刚开始接触人工智能的理论时，我们可能会感到晦涩难懂，但当我们将这些理论应用到实际问题中时，理论知识就会变得清晰明了。

本课程的难度可能较大，但内容却十分有趣。我们通过实际操作和实验，深入了解了人工智能的原理和应用。这种实践性的学习方式不仅使我们能够更好地理解理论，还为我们提供了解决实际问题的技能和经验。

通过本课程的学习，我们希望能够获得更多深入学习人工智能领域知识的机会。

人工智能是一个快速发展的领域，不断涌现出新的理论和应用。我们希望能够继续深入学习，跟上人工智能领域的最新发展，掌握更多高级的技术和方法。总之，通过这门课程，我们克服了理论的困难，将其转化为实际应用，并对人工智能的前景充满了兴趣和热情。我们期待未来有更多机会深入学习人工智能，并在这个领域中取得进一步的成就。



## 8 参考文献

[1] Raphael A, Dubinsky Z, Iluz D, et al. Neural Network Recognition of Marine Benthos and Corals[J].Diversity, 2020, 12(1): 29.

[2] [1] 李莎. 基于深度学习的海洋生物识别技术研究[D]. 青岛科技大学, 2024. DOI:10.27264/d.cnki.gqdhc.2023.001265.

## 附录

requirements.txt

```
gitpython>=3.1.30
matplotlib>=3.3
numpy>=1.23.5
opencv-python>=4.1.1
pillow>=10.3.0
psutil # system resources
PyYAML>=5.3.1
requests>=2.32.0
scipy>=1.4.1
thop>=0.1.1 # FLOPs computation
torch>=1.8.0 # see https://pytorch.org/get-started/locally (recommended)
torchvision>=0.9.0
tqdm>=4.64.0
ultralytics>=8.0.232
```