

教 案

2025-2026 学年第一学期

课程名称 人工智能通识与应用
专业班级 机电一体化技术 251 (三二分段)
总学时数 48 学时
任课教师 郭奕裕

课程基本信息

课程名称	人工智能通识与应用			
课程性质	专业基础课	学分	3	
学时	总学时 48 学时。其中：课堂讲授 16 学时；实训/实验 32 学时；			
开课部门	机电工程系	任课教师	郭奕裕	
授课专业、班级	机电一体化技术 251（三二分段）	开课学期	2025-2026 第一学期	
成绩评定	平时成绩占 30%；期末成绩占 70%	考核方式	考查	
选用教材	书名	主编	出版社	出版日期
	人工智能技术应用	何琼	高等教育出版社	2020.8
本课程在本专业人才培养方案中的地位和作用	本课程是机电一体化技术专业的核心数字化能力支撑课程。在制造业智能化转型背景下，课程聚焦为智能装备的感知、决策与控制环节提供 AI 技术赋能，培养学生运用计算机视觉、语音交互、机器学习等技术解决设备智能运维、产线柔性控制、产品质量检测等典型工业场景问题的能力。其核心价值体现在打通传统机电设备与人工智能技术的融合路径，对接智能制造领域新兴岗位需求。			
本课程教学目标	1、掌握人工智能领域的核心概念与基本原理，包括机器学习、深度学习等关键技术，理解其在工业自动化中的实际应用价值和发展趋势； 2、系统理解计算机视觉技术和智能决策系统在工业场景中的技术实现原理，能够分析不同算法在生产线检测、质量管控等具体场景的适用性； 3、熟练掌握 Python 编程语言及其应用，理解 OpenCV、PyTorch 等主流工具库在设备控制中的开发逻辑和实现方法； 4、了解机器学习工作流程和神经网络基础理论，具备调用主流 AI 开放平台 API 接口的能力能够针对特定工业需求选择合适的模型架构。			
素质(思政)内容与要求	1、科技报国精神：通过国产技术突破案例及中美技术专利对比分析，引导学生认同国产化替代战略，要求撰写技术对比分析报告。 2、技术向善导向：围绕动态定价的公平性辩论，分组讨论 AI 应用伦理检查清单，明确技术禁用场景。 3、创新实践与工匠精神的融合培养：在教育教学中，注重通过多样化的实践活动，引导和激励学生深入理解并切实践行“工匠精神”的核心内涵。 4、数据伦理责任：以严谨务实的作风，在技术创新与隐私保护之间精准探寻恰当的平衡点，积极构建可持续发展、健康有序的数字伦理生态。			

学生用主要参考资料	何琼，《人工智能技术应用》，高等教育出版社
-----------	-----------------------

第 1 章 走近人工智能

一、教学目标

1. 理解人工智能 (AI) 的定义、发展历程及核心技术分类 (弱 AI vs 强 AI)。
2. 掌握 AI 在产业中的应用场景 (互联网、医疗、安防、制造、农业)。
3. 熟悉 AI 系统的核心要素 (数据、算法、算力、场景) 及关键技术 (机器学习、知识图谱)。

二、教学方法与手段

1. 方法

- a. **案例教学法**: 以 AlphaGo、医疗影像诊断等案例贯穿课堂。
- b. **问题导向法**: 通过“机器能否思考?” “AI 如何改变生活?” 等问题驱动探究。
- c. **分组讨论法**: 围绕 AI 应用的伦理与社会影响展开辩论。

2. 手段

- a. **多媒体课件**: 结合 PPT 图文、视频片段 (如《模仿游戏》)。
- b. **互动演示**: 实时展示腾讯 AI 开放平台的功能 (OCR、图像识别)。

三、素质 (思政) 内容与要求

1. **科技报国精神**: 强调中国 AI 技术突破 (如百度自动驾驶、阿里鹿班)。
2. **伦理责任意识**: 讨论 AI 隐私风险 (如安防监控) 与“技术向善”原则。通过案例引导学生思考技术发展的社会边界 (如《终结者》引发的通用 AI 恐惧)。
3. **创新与实践结合**: 鼓励学生参与高职院校“垃圾分拣装置”类实践项目。

四、教学重点与难点

类别	内容
----	----

重点	1. AI 的核心技术分类 (弱 AI vs 强 AI) ; 2. AI 在产业的应用逻辑 (推荐系统、计算机视觉) ; 3. 机器学习与深度学习的核心作用。
难点	1. 通用人工智能 (AGI) 的技术瓶颈与伦理争议; 2. 算法原理抽象性 (如策略网络与价值网络) 。

五、教学内容 (总时长: 180 分钟)

1. 导入: AI 如何改变生活? (10 分钟)

- **案例切入:**

- 展示阿里巴巴鹿班系统案例: 2017 年双 11 生成 4 亿张广告图, 等同 200 名设计师 200 年工作量。
- **提问:** “如果由你设计该系统, 需哪些技术支持?” (引导学生联想图像识别、大数据) 。

2. 新课讲授 (160 分钟)

(1) 人工智能的定义与发展史

- **核心概念:**

- **定义:** AI = 模拟人类智能的理论与技术 (约翰·麦卡锡, 1956) 。
- **图灵测试** (1950) : 通过问答判断机器是否具有智能 (播放《模仿游戏》片段) 。

- **发展脉络:**

- **阶段梳理:**

时期	标志事件
开天辟地 (1950s)	图灵测试提出 (1950) ; 达特茅斯会议命名 “AI” (1956)。
百花齐放 (1980s)	专家系统 XCON 成功 (1980) ; 日本五代机计划 (1981) 。
峰回路转 (2010s)	深度学习爆发 (2012) ; AlphaGo 击败李世石 (2016) 。

(2) AI 的技术分类：弱 AI vs 强 AI

- **弱 AI (专用人工智能) :**

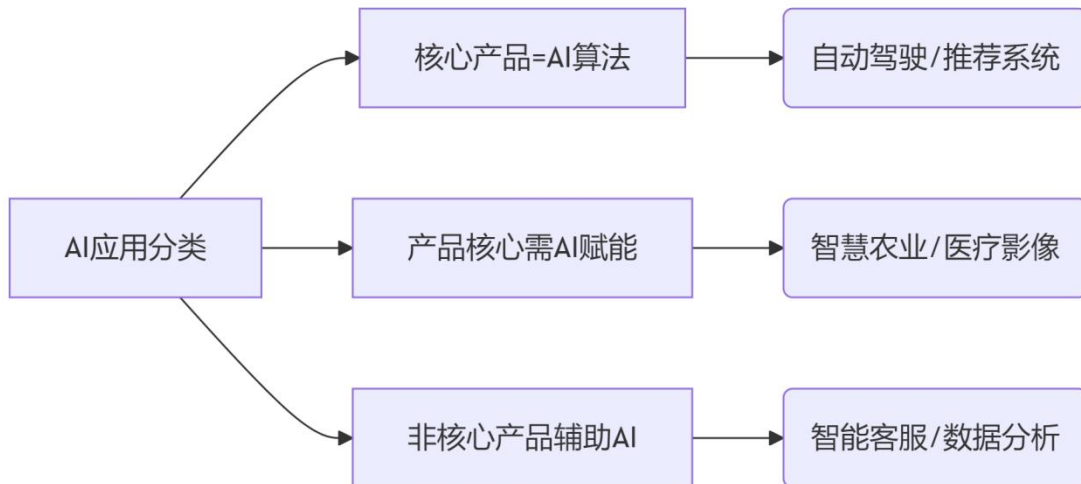
- **特点:** 单一任务专家 (如推荐算法、人脸识别) 。
- **案例:** 淘宝 “猜你喜欢” 、抖音推荐系统 (学生现场操作演示) 。

- **强 AI (通用人工智能) :**

- **特点:** 拟人化全能智能 (尚未实现) 。
- **争议:** 播放《终结者》片段, 讨论 “机器统治人类” 的可能性与伦理边界。

(3) AI 的产业应用

- **应用层级:**



- **重点场景解析:**

- **医疗:** AI 辅助肺腺癌 vs 肺鳞癌病理切片识别 (对比误诊率下降 30%) 。
- **安防:** 城市摄像头+深度学习算法实现嫌疑人追踪 (效率提升 100 倍) 。
- **制造:** 自适应机械臂学习复杂工序 (播放汽车工厂智能装配视频) 。

(4) AI 系统的核心要素与技术

- **关键技术:**

- **机器学习:** 监督学习 (图像分类)、无监督学习 (用户聚类)、强化学习 (AlphaGo) 。
- **知识图谱:** 实体关系网络 (举例: 医疗知识库辅助诊断) 。

(5) 如何开始 AI 学习 (5 分钟)

- **学习路径:**

- **基础工具:** Python 语言 (演示 `print("Hello AI")`) 。
- **开源框架:** TensorFlow (谷歌)、PyTorch (Facebook)、PaddlePaddle (百度) 。
- **实践平台:** 腾讯 AI 开放平台 (现场演示验证码识别 API 调用) 。

3. 课堂总结

- **知识闭环:**

- AI = 数据 + 算法 + 算力 + 场景;
- 弱 AI 已深入生活, 强 AI 需技术突破与伦理约束。

- **升华思考:** 技术不应替代人类, 而是解决 ‘干不了、干不好、不想干’ 的问题。

第 2 章 初识 Python

一、教学目标

1. 掌握 Python 变量的定义与赋值方法。
2. 理解常用数据类型 (字符串、整数、浮点数) 及转换方法。
3. 熟练使用 `input()`和 `print()`函数实现数据输入输出。
4. 能调试数据类型转换引发的错误 (如字符串转数值) 。

二、教学方法与手段

1. **方法**

- a. **案例教学法:** 以 “超市记账本” 为项目载体贯穿课堂。
- b. **任务驱动法:** 分步实现商品录入→计算→输出功能。
- c. **纠错演练法:** 故意制造类型错误引导学生调试。

2. **手段**

- a. **多媒体演示:** PPT 展示代码逻辑与运行效果。
- b. **实时编码:** 教师演示 VSCode 环境搭建与代码编写。
- c. **互动问答:** 针对 `input()`返回值类型等关键点提问。

三、素质（思政）内容与要求

1. **技术惠民**：结合阿里鹿班系统案例，说明编程可解决重复劳动（如自动生成广告图）。
2. **理性消费**：通过家庭支出管理程序倡导合理规划生活。
3. **工匠精神**：强调代码规范性（如标识符命名规则）。

四、教学重点与难点

类别	内容
重点	1. 变量赋值与数据类型转换； 2. input()和 print()的灵活应用； 3. 算术运算符的使用。
难点	1. 理解 input()统一返回字符串的特性； 2. 类型转换失败的处理（如非数字输入）。

五、教学内容（360 分钟）

1. 导入：AI 如何解决重复劳动？（10 分钟）

● 情景切入：

- 展示阿里巴巴鹿班系统案例：2017 年双 11 生成 4 亿张广告图，节省设计师工作量。
- **提问**：“如果让你设计超市记账程序，需要记录哪些信息？”（引导思考商品名称、价格等）。

● 任务明确：

“今天我们将化身‘家庭财务官’，用 Python 编写超市记账本程序！”

2. 新课讲授（340 分钟）

(1) Python 开发环境搭建

- **VSCode 配置** (图示操作) :

步骤 1: 安装 Python 3.7.5 (勾选 "Add to PATH")

步骤 2: 安装 VSCode → 配置中文插件 → 安装 Python 扩展

步骤 3: 创建 `market_notebook.py` 文件 → 输入 `print("Hello Python")` 测试环境

- **易错点强调:**

- 必须使用英文输入法写代码, 中文仅用于字符串内容。

(2) 变量与数据类型

- **核心概念:**

变量赋值 (贴标签比喻)

```
product = "牛奶" # 字符串(str)
```

```
price = 3.5 # 浮点数(float)
```

```
quantity = 2 # 整数(int)
```

- **数据类型转换** (对比演示) :

```
total = price * quantity # 正确: 3.5 * 2 = 7.0
```

```
total = price * int(quantity) # 显式转换更安全
```

```
total = float("3.5") * int("2") # 字符串转数值计算
```

```
total = str(7.0) + "元" # 数值转字符串拼接
```

- **调试实战:**

```
# 错误案例: 字符串直接相乘
```

```
print("3.5" * "2") # 引发 TypeError!
```

(3) 输入输出函数

- **input()与 print()详解:**

```
# 输入商品信息 (返回均为字符串)
```

```
name = input("请输入商品名称: ")
```

```
price = float(input("请输入单价: ")) # 直接转换避免后续错误
```

```
# 输出格式化 (多种方式)
```

```
print("商品: ", name, "单价: ", price) # 逗号分隔自动加空格
```

```
print(f"总价: {price * quantity:.2f}元") # f-string 精确到分
```

- **实战任务:**

在 VSCode 中编写代码，实现从键盘读取一件商品信息并打印小票。

(4) 超市记账本完整实现

- **代码逐行解析:**

```
# 输入数据 → 类型转换 → 计算 → 格式化输出
```

```
name = input("请输入商品名称: ")
```

```
price = float(input("请输入单价 (元) : "))
```

```
quantity = int(input("请输入数量: "))
```

```
total = price * quantity
```

```
# 打印带表格线的收据
```

```
print("-" * 30)
```

```
print(f"商品: {name}\n 单价: {price}元\n 数量: {quantity}\n 总价: {total}\n 元")
```

```
print("-" * 30)
```

(5) 课堂实践任务

- **基础任务:**

扩展程序功能：连续录入 3 件商品，并计算订单总金额。

```
total_cost = 0
```

```
for i in range(3):
```

```
    name = input(f"第{i+1}件商品名称: ")
```

```
    price = float(input("单价: "))
```

```
    quantity = int(input("数量: "))
```

```
    total_cost += price * quantity
```

```
print(f"订单总额: {total_cost}元")
```

- **挑战任务:**

若输入单价时用户误输字母，如何让程序友好提示重新输入？

(提示：用 try-except 捕获 ValueError)

3. 总结与升华 (10 分钟)

- 调研超市 POS 机系统，思考如何增加“折扣计算”功能。
- 尝试用列表存储多件商品信息（为下节课铺垫）。

第三章 让机器认识你

一、教学目标

1. 掌握人脸识别四大核心流程：人脸检测（定位图像中人脸位置）、人脸配准（标记五官关键点）、人脸特征提取（将图像转化为数值串）、人脸识别（比对身份库确认身份）。
2. 理解 OpenCV 库的 Haar 级联分类器原理，Pillow 库的图像处理功能。列举刷脸支付、安防追踪、课堂情绪分析等应用场景。

二、教学方法与手段

- **案例教学法**：以公安系统追逃案例说明人脸识别的精准性（千万级数据库毫秒比对），以刷脸支付失败案例引出活体检测必要性。
- **任务驱动法**：设计“教室门禁系统”项目，要求学生描述从摄像头捕获到身份识别的全流程。
- **对比分析法**：传统 PCA 算法（对光线敏感）与现代深度学习模型（支持多姿态）的优劣对比。
- **技术手段**：用 Python 代码片段示意图说明 OpenCV 人脸检测函数调用逻辑（非完整代码）。

三、课程思政内容与要求

1. **科技伦理**：某小区强制刷脸进门被工信部叫停事件，引导学生思考“技术便利”与“个人信息权”的平衡。
2. **民族科技自信**：旷视科技跨境追逃系统协助警方 30 小时抓获嫌疑人，体现中国 AI 技术硬实力。2023 年全球人脸识别专利 TOP10 中中国企业占 7 席。

3. **技术向善导向**：Deepfake 换脸技术伪造明星视频的诈骗案例，强调技术滥用危害。

四、教学重点与难点

1. **重点**：**四步流程闭环逻辑**：检测（是否有人脸）→配准（眼睛鼻子在哪）→特征提取（脸变成数字密码）→识别（匹配身份库）。**Haar 分类器工作机理**：通过 16×16 像素窗口扫描图像，用黑白矩形特征组合快速排除非人脸区域。

2. **难点**：**特征抽象化过程**：LBPH 算法将人脸分割为 8×8 小区域，计算每个区域的纹理直方图并拼接成特征向量。**活体检测防欺骗**：红外摄像头识别皮肤血氧信号，眨眼动作验证破解高清照片攻击。

五、教学内容（180 分钟）

1. 导入：刷脸时代的真实体验（10 分钟）

- **情景创设**：

“当走进便利店挑选商品，无需手机扫码，抬头瞬间已完成支付——这是如何实现的？”引发学生对无感支付的思考。

- **数据冲击**：

2024 年中国刷脸支付用户突破 8 亿，渗透率超 60%，说明技术已深度融入生活。

2. 核心内容精讲（160 分钟）

(1) 技术框架解剖

- **人脸检测**：

类比“找不同游戏”：计算机以滑动窗口扫描图像，通过 Haar 特征（如眼眶比脸颊更黑）逐区域判断是否含人脸。典型工具 OpenCV 的 `detectMultiScale()` 函数。

- **人脸配准**：

定位 68 个关键点意义：左眼内角点 (x_1, y_1) 、鼻尖 (x_2, y_2) 、右嘴角 (x_3, y_3) 等坐标构成面部“数字地图”。

- **特征提取：**

LBPH 核心三步：①灰度化人脸图像 ②划分 64 个区块 ③计算每个区块的纹理直方图并连接为 1024 维特征向量。

- **人脸识别：**

相似度计算原理：提取的特征与数据库向量求欧氏距离，距离 < 阈值判定为同一人（如支付系统阈值 0.003）。

(2) 关键技术突破

- **活体检测：**

动态防御三机制：①光学特性（红外检测真人皮肤反光） ②微动作（指令式眨眼/点头） ③生物信号（摄像头测心率波动）。

- **深度学习革新：**

卷积神经网络 (CNN) 替代传统算法：通过多层卷积提取瞳孔纹路、法令纹等深层特征，误识率降至 0.0001%。

(3) 应用与挑战

- **应用矩阵：**

金融领域：银行 VIP 系统刷脸确认身份，误识率要求 < 0.001%。

教育领域：分析学生课堂表情（嘴角下垂/皱眉频率）评估专注度。

公共安全：机场动态布控系统，每秒比对 2000 万张人脸。

- **现存瓶颈：**

极端姿态识别：侧脸 > 60°时关键点丢失率达 70%。

伦理争议：某中学用表情分析系统监控学生是否走神，引发侵犯心智自由批评。

3. 课堂总结 (10 分钟)

- **技术闭环复现：**

梳理“摄像头捕获→人脸定位→特征数字化→模型比对→身份输出”全链条，强调配准环节对后续精度的影响。

- **未来展望：**

多模态融合趋势：人脸+声纹+步态三重验证解决双胞胎识别难题。

第 4 章 机器怎么学习

一、教学目标

1. 掌握机器学习核心概念：监督学习（通过已知输入输出训练模型）与无监督学习（无标签数据自主聚类）的本质区别，理解神经网络模仿生物神经元树突的工作机理。
2. 系统认知字符识别（OCR）全流程：从图像二值化（区分目标与背景）、降噪滤波（消除干扰像素）、倾斜校正（仿射变换调整角度）、字符分割（定位独立文字区域）到特征匹配（对比特征向量）。
3. 深入理解卷积神经网络（CNN）的三层结构。
4. 能独立描述神经网络训练四步骤：数据预处理（归一化/增强）→ 模型构建（层结构设计）→ 损失函数选择（分类用交叉熵）→ 反向传播优化权重。
5. 掌握图像处理技术应用：使用 OpenCV 完成灰度转换、中值滤波、轮廓检测等操作。

二、教学方法与手段

1. **项目驱动**：以 MNIST 手写数字识别为载体（文档 1），贯穿数据加载、CNN 搭建、训练验证全流程。基于 ResNet18 迁移学习实现垃圾分类（文档 2），对比预训练模型与随机初始化的效率差异。
2. **概念可视化解析**：用“字体识别经验”类比特征提取：学生回忆辨认不同手写体“1”时抓住“竖线”共性特征的过程，对应卷积层提取数字图像边缘。以“习题集训练与考试验证”比喻数据集划分：训练集如日常练习题，测试集如期末考试卷。
3. **分层实践任务**：基础任务：调用 PyTorch 内置函数完成 MNIST 数据加载，观察 batch_size 参数对训练速度的影响。进阶挑战：修改 ResNet18 全连接层输出维度（4 分类），分析学习率（lr=0.001→0.0001）对收敛速度的影响。

三、素质（思政）内容与要求

1. **科技向善责任**：在垃圾分类项目导入环节，剖析技术如何解决环保痛点。要求学生列举 AI 改善民生的其他案例（如医疗影像诊断）。
2. **数据隐私红线**：讲解 MNIST 数据集采集原理时强调，60000 张手写数字均来自匿名志愿者。

3. **精益求精态度**：模型训练中设置“超越基准线”挑战，在模型调优过程中体会从99%到99.2%精度的工匠精神。

四、教学重点与难点

1. 重点

- 神经网络权重核心作用：通过“儿童学写字”场景说明——人类反复练习形成书写习惯，类似模型通过损失函数反向传播调整权重。
- 图像预处理四步法技术细节：
 - 二值化：动态阈值计算（Otsu 算法）应对光照不均；
 - 降噪：中值滤波消除椒盐噪声；
 - 校正：霍夫变换检测倾斜角度；
 - 分割：投影直方图定位字符边界。
- 迁移学习实践要点：加载 ImageNet 预训练权重后，通过 `requires_grad=False` 冻结卷积层参数。

2. 难点

- **卷积抽象机制具象化**：用“像素扫雷”比喻说明 3×3 过滤器在图像上滑动计算（局部像素加权和=特征响应值）。
- **维度变换逻辑**：以 28×28 图片为例，经卷积（输出 24×24 ）→池化（压缩为 12×12 ）→Flatten 操作展开为 1×576 向量。
- **损失函数选择依据**：分类任务用交叉熵（预测概率分布与真实标签差异），回归任务用均方误差（预测值与真实值距离）。

五、教学内容（360 分钟）

1. 导入：人类学习与机器学习的共鸣（10 分钟）

● 场景还原

教师引导学生回忆幼年识字经历：“当我们第一次看到老师书写歪斜的‘1’时，如何确认它仍是数字1？”学生总结关键特征（竖直走向）。类比指出：MNIST 数据集的 55000 张训练图片，正是机器观察“数字多种形态”的样本库。特别强调人类通过反复练习强化认知，对应机器学习的 epoch 训练机制。

- **认知冲突**

提出问题：“若将数字旋转 45°，人类仍可识别，但未经数据增强的模型会失效。”引出图像校正技术的必要性，为后续实践埋下伏笔。

2. 核心理论精讲 (340 分钟)

- **模块 1：机器学习基石**

- 监督学习本质：以垃圾邮件识别为例，系统学习 “spam” 与 “ham” 邮件的关键词特征（如 “免费” “发票” ），建立输入（邮件文本）到输出（分类标签）的映射。对比无监督学习典型场景：对未知商品自动聚类（如电商用户行为分组）。
- 神经网络生物原型：详细解析树突接收信号（输入层）、细胞体整合刺激（加权求和）、轴突阈值激发（ReLU 激活函数）、突触传递信息（权重参数）四阶段。重点说明 2012 年 Geoffrey Hinton 团队突破——5 层 CNN 在 ImageNet 竞赛中将错误率从 26%降至 15%。

- **模块 2：OCR 技术全流程拆解**

- **二值化痛点攻坚**：分析黑板粉笔字照片中的干扰因素（反光/阴影），引入自适应阈值算法：对图像分块计算局部最优阈值，确保深浅区域均有效分割。
- **校正原理实操推演**：通过两组坐标点（原图倾斜角点/目标垂直角点）求解仿射变换矩阵，演示如何用 OpenCV 的 warpAffine 函数拉正图像。
- **特征匹配本质**：对比人类识别 “田” 字（寻找封闭方框与内部十字），说明计算机将处理后的字符图片转换为特征向量（如笔划端点坐标、曲率值），与模板库进行欧氏距离计算。

- **模块 3：CNN 突破性结构**

- 卷积层核心价值：以识别猫脸为例，第一层过滤器响应边缘（胡须线条），第二层组合局部特征（眼部轮廓），第三层抽象整体结构（面部器官布局）。
- 池化层抗干扰能力：说明 2×2 最大池化如何应对特征偏移——即使猫眼位置移动 10 像素，高层特征响应仍保持稳定。
- 全连接层决策过程：展示数字识别任务中，2000 维特征向量如何通过两层线性变换（Linear(2000,500)→Linear(500,10)）压缩为 10 分类概率。

- **任务 1：手写数字识别实战**

- **数据加载关键点**：解释 DataLoader 参数意义：batch_size=512（每批处理图像数）、shuffle=True（每个 epoch 重排数据防止模式固化）。
- **网络结构定制**：逐层分析 LeNet-5 改造版：第一卷积层: Conv2d(1,10,5) 接收 1 通道灰度图，输出 10 通道特征图（5×5 卷积核）
 - 池化层：2×2 窗口取最大值，尺寸减半
 - 第二卷积层：Conv2d(10,20,3) 深化特征提取
 - 展平操作：将 20×10×10 特征矩阵转换为 2000 维向量
 - 全连接层：Linear(2000,500)→ReLU→Linear(500,10)
- **训练技巧实证**：通过对比实验说明 Adam 优化器优势（自适应学习率），演示查看 GPU 利用率命令（nvidia-smi）确认计算加速生效。
- **任务 2：垃圾分类迁移学习**
 - **数据差异化处理**：重点解析垃圾图片的特殊处理：统一分辨率至 120×120（解决原始图片尺寸差异）
 - 三通道转灰度但保留 3 通道（兼容 ResNet 的 ImageNet 预训练权重）
 - 随机旋转±5°（增强模型泛化能力）
 - **迁移学习精髓操作**：加载 ResNet18 预训练模型：model = models.resnet18(pretrained=True)
 - 冻结特征提取层：for param in model.parameters():
param.requires_grad = False
 - 替换末层全连接：model.fc = nn.Linear(512, 4) # 原输出 1000 类改为 4 类垃圾
 - **部署难点突破**：详述机械臂坐标转换原理：标定阶段：采集图像三点坐标（A/B/C）与机械臂实际坐标（P1/P2/P3）
 - 构建转换矩阵：通过 np.linalg.solve 求解 AX=B 方程组
 - 实时计算：输入目标图像中心坐标(x,y)，输出机械臂目标坐标(b1,b2)

3. 总结升华（10 分钟）

- **技术闭环**

梳理从基础理论（神经元模型）到技术实现（CNN），再到应用创新（OCR/垃圾分类）的完整链条。强调特征提取的核心地位：数字识别抓笔划走向，垃圾分类辨材质纹理。

● 伦理反思

抛出问题：“当垃圾分类准确率达 99% 时，余下 1% 的错误可能来自拾荒老人的模糊照片，技术应如何保障公平？” 引导学生撰写《负责任 AI 开发宣言》要点。

第 5 章 让机器听你指挥

一、教学目标

1. 深入掌握语音识别（ASR）的核心流程：声学处理→特征提取→模型解码
2. 系统理解语音合成（TTS）的文本韵律处理与波形生成原理
3. 明确区分自然语言处理（NLP）中语义理解（NLU）与生成（NLG）的技术边界

二、教学方法与手段

1. 核心方法：

分层拆解法：将语音交互拆解为“听清→听懂→执行”三层架构

故障复现法：人为制造语义歧义案例（如“打开苹果”），引导学生调试

2. 辅助手段：

声学可视化工具：实时展示声波分帧后的频谱包络变化

HTTP 请求模拟器：用 Postman 演示云 API 的请求/响应全流程

三、思政内容与要求

1. **科技自立**：对比中美语音技术专利数量（中国近 5 年增长率超 200%）
2. **技术向善**：讨论欧盟 GDPR 对语音数据存储的合规性要求

四、教学重点与难点

重点 1：MFCC 特征提取的物理意义

● 突破策略：

类比人耳耳蜗滤波机制：

原始声波（空气振动）→ 鼓膜传递 → 耳蜗频率分组（类似三角滤波）→ 神经信号（13

维特征向量） **重点 2：云平台双服务协同机制**

- **关键矛盾**：
- 百度 ASR 输出文本 → 讯飞 NLU 解析意图 → 机械臂执行动作 **数据流验证**：
在代码中插入 `print(f"ASR 结果: {text} → NLU 指令: {command}")` 实时跟踪 **难点 1**：

语义理解的上下文关联

- **典型问题**：
- 用户说"向上移动"后紧接"再五厘米"，需记忆前一指令方向 **解决方案**：

引入对话状态跟踪 (DST) 概念：

```
history = {"last_direction": "up"} # 上下文记忆容器
if "再" in text:
    direction = history["last_direction"] # 复用历史方向
```

难点 2: 机械臂动作的毫米级精度控制

- **误差分析**：
- 演示 10 厘米指令实际移动 9.8 厘米的机械误差 **补偿措施**：

增加校准函数：

```
def calibrate(distance):
    return distance * 1.02 # 根据硬件测试得出的补偿系数
```

五、教学内容设计 (180 分钟)

(一) 课堂导入 (10 分钟)

1. 现实矛盾切入

- 播放两段对比视频：
 - 成功案例**：用户说"室温调至 25 度"，空调精准响应
 - 失败案例**：用户说"太冷了"，音箱回复"已为您播放《冷雨夜》"
- 追问**：为何后者无法理解真实需求？引出语义理解的复杂性

2. 技术演进脉络

- 时间轴展示：
- 1952 年贝尔实验室数字识别 → 2011 年苹果 Siri → 2023 年 ChatGPT 语音交互

3. 国产化突破：重点标注科大讯飞 2016 年医学语音识别夺冠事件

(二) 核心内容 (160 分钟)

模块 1: 语音技术深度解析

1. 语音识别全流程拆解

a. 声学前端处理

- i. **端点检测** : 模拟打电话时"喂"声触发系统唤醒
- ii. **分帧加窗** : 用 50ms 汉明窗演示语音信号的平滑截取
- iii. **噪声抑制** : 对比空调噪声环境下降噪前后的频谱图变化

b. MFCC 特征工程

- i. **物理本质** : Mel 刻度模拟人耳对 1kHz 以下声音的线性感知
- ii. **数学转换** : FFT 获取频域能量 → 2. 40 个 Mel 三角滤波器组降维 → 3. 对数压缩动态范围 → 4. DCT 得到 13 维倒谱系数
- iii. **实战意义** : 播放同一人"开灯"的 10 段录音, 展示 MFCC 向量相似性

c. 深度学习解码

- i. **CTC 损失函数** : 图解"k ai _ d eng"到"开灯"的标签对齐
- ii. **注意力机制** : 动态显示解码器对"设置闹钟"中"闹钟"的聚焦权重

2. 语音合成技术

- a. **韵律建模** : 对比"中国银行"两种歧义读法的基频轨迹:
- b. **端到端合成** : 播放 Tacotron2 生成的带情感语音 (惊喜/平静语气差异)

模块 2: 云平台开发实战

1. 双平台协同架构

a. 密钥安全方案 :

演示.env 文件 + python-dotenv 库的密钥加载:

```
from dotenv import load_dotenv
BAIDU_API_KEY = os.getenv("BAIDU_KEY") # 从环境变量读取
```

b. 错误熔断机制:

```
try:
    response = requests.post(API_URL, timeout=3.0)
except ConnectionError:
    tts_play("网络异常, 请检查连接") # 语音提示故障
```

2. 机械臂精准控制

a. 空间映射算法 :

```
# 坐标系说明: x(前后), y(左右), z(上下)
def move_arm(direction, cm):
    mm = cm * 10 # 厘米转毫米
    current_pos = dType.GetPose(api)
```

```
if direction == "前":
    new_pos = (current_pos[0] + mm, current_pos[1], current_pos[2])
# 限位保护: 防止撞击桌面
if new_pos[2] < 50:
    new_pos = (new_pos[0], new_pos[1], 50) # z轴最低保护
```

b. **多指令流水线** :

演示"向上 3 厘米→右移 5 厘米→抓取"的连续动作队列

模块 3: 技术伦理研讨

1. **隐私保护红线**

a. **数据生命周期** :

b. *录音→传输加密 (HTTPS) →服务器内存处理 (不落盘) →30天自动删除* **合规对比** : 播放 Google 语音侵权诉讼案新闻片段

2. **人机责任边界**

a. **致命场景推演** :

b. *如果语音控制核电站阀门, 该设置几重验证?* **学生公约签署** :

《负责任 AI 开发承诺书》第一条: 不开发危及生命的语音控制系统

(三) 课堂总结 (10 分钟)

1. **技术闭环强化**

a. **板书核心公式链**:

Voice → MFCC → 拼音序列 → 语义树 → 坐标增量 → 机械臂动作

2. **前沿技术展望**

a. **低资源方言** : 粤语语音识别准确率突破 92% (讯飞 2023)

b. **脑机接口延伸** : 展示 Neuralink 猴子意念控制机械臂视频

第 6 章 走进人工智能

一、教学目标

1. 掌握无人零售系统的三大核心技术: 计算机视觉 (商品识别)、传感器融合 (行为追踪)、智能决策 (动态定价)。
2. 理解 "即拿即走" (Just Walk Out) 模式从进店到离店的全流程技术闭环。
3. 区分 RFID 标签方案与机器视觉方案的核心差异 (成本、精度、可扩展性)。

二、教学方法与手段

● **对比分析法**：**成功案例**：Amazon Go 的 98.7%识别精度 (500+摄像头+深度学习算法)。**失败教训**：缤果盒子因 RFID 标签成本占商品价 15%退出市场。

● **情景任务驱动**：

- 分组设计"抗疫无接触超市"方案, 聚焦三大矛盾: 零接触需求 vs 身份认证速度
- 行为追踪精度 vs 硬件成本
- 动态定价收益 vs 公平性质疑

三、思政内容与要求

1. **数据伦理铁律**：人脸数据实时 RSA 加密传输; 购物记录 48 小时后自动清除
2. **国产化攻坚**：对比中美技术卡脖子点: Amazon Rekognition 闭源 → 华为自研 NPU 芯片突破算力封锁

四、教学重点与难点

1. **重点: 视觉定位**：YOLOv5 检测商品位移 (误差 $\pm 2\text{cm}$)；**多源验证**：货架压力传感器+红外测距交叉纠错
2. **难点: 动态定价的伦理争议: 正向价值**："Happy 购"情绪营销提升客单价 30%**法律红线**：对同一商品向老人定价高于年轻人属违法

五、教学内容设计 (180 分钟)

(一) 问题导入：生死局中的无人零售 (10 分钟)

1. **技术高光时刻** 播放 Amazon Go 顾客拿取咖啡离店视频："25 秒无感支付背后：152 个手势识别模型+73 类商品 3D 建模"
2. **残酷退场警示**
拆解缤果盒子败因：**技术硬伤**：RFID 标签遇金属包装失效 (漏检率 23%)；**体验灾难**：40°C 高温致巧克力融化触发误报警

(二) 核心技术深度解析 (40 分钟)

1. **身份认证: 活体检测双保险**
 - a. **3D 结构光**：投射 3 万个红外点阵防照片伪造 (华为 P40 方案)
 - b. **声纹定位**：6 麦克风阵列判定声源方位 (误差 $\leq 10^\circ$)
2. **行为追踪: 多传感器博弈**

a. **决策逻辑树**：

```
IF 压力传感器检测重量减少
  AND 摄像头捕捉商品位移轨迹
  AND 红外测距确认手部离开货架
THEN 标记为购买
ELSE IF 压力未变但视觉检测位移 → 启动人工复核
```

b. **纠错案例**：顾客衣袖碰倒商品不触发结算

3. **机械臂分拣：柔性抓取策略**

a. **易碎品**：真空吸盘+0-5N 力反馈 (鸡蛋抓取破损率<0.1%)

b. **不规则品**：3D 视觉生成最优抓取点 (如葡萄梗部避让)

(三) 项目实战：抗疫超市设计 (35 分钟)

1. **需求冲突破解 (10 分钟)**

a. **矛盾**：无接触要求 vs 老人手机操作困难

b. **方案**：主通道：手机扫码+人脸识别 (20-50 岁人群)

i. 应急通道：声控呼叫人工协助 (≥60 岁或特殊人群)

2. **硬件部署规范 (15 分钟)**

a. **传感网络拓扑**：顶层：广角摄像头 (全局监控)

i. 中层：货架压力传感器 (商品变动感知)

ii. 底层：UWB 定位标签 (顾客移动轨迹)

b. **安全三原则**：机械臂工作区设红外栅栏 (侵入时断电)

i. 急停按钮串联所有动力电路

ii. 关节力矩传感器防过载 (阈值 10N·m)

3. **伦理大辩论 (10 分钟)**

a. **辩题**：“*基于情绪的动态定价是否合理？*”

b. **正方**：笑容折扣提升购物愉悦感 (某超市复购率+40%)

c. **反方**：匆忙状态被加价 15%属价格歧视 (引用价格法第 14 条)

(四) 总结升华 (10 分钟)

1. **技术本质** 核心公式：

2. **安全 = 身份认证 × (行为追踪 + 支付可信)** **国产突围** 华为智能售货柜 2.0：

鸿蒙系统直连健康码 (疫区顾客自动消毒)

i. 机械臂分拣效率 600 件/小时 (误差率 0.03%)

实验内容

实验 1: 文本类 AIGC 应用实践

目的: 学习使用 AI 生成文本内容, 如文章、摘要、PPT 等。

工具: 通义千问 (网页版)

内容:

- (1) 设计提示词生成文章
- (2) 制作学术论文摘要
- (3) 生成小组汇报 PPT
- (4) 使用阅读助手分析论文

实验 2: 图片类 AIGC 应用实践

目的: 掌握 AI 生成和编辑图片的能力。

工具: 文心一格、即梦 AI

内容:

- (1) 图片扩展 (补全画面)
- (2) 涂抹消除 (去掉不需要的部分)
- (3) 图片叠加 (融合两张图)
- (4) 智能抠图+换背景

实验 3: 语音类 AIGC 应用实践

目的: 使用 AI 进行语音转写、合成和复刻。

工具: 腾讯云语音服务

内容:

- (1) 语音实时转文字
- (2) 文本转语音 (TTS)
- (3) 语音复刻 (用你的声音合成语音)

实验 4: 视频类 AIGC 应用实践

目的: 学习用 AI 生成和编辑视频。

工具: 剪映专业版、智普清言、Vidu

内容:

- (1) 文字生成视频
- (2) 图片生成视频
- (3) 多主体参考生成视频

实验 5: AI 搜索实践

目的: 使用 AI 进行数据分析、教育问答和行程规划。

工具: 智普清言、通义千问

内容:

- (1) 生成和分析模拟数据
- (2) 解答数学题、出考题
- (3) 制定旅游计划、锻炼计划等

实验 6: 初识 Python 实验

实验目标

1. 掌握 Python 环境搭建方法 (Python 3.7.5 + VSCode)
2. 理解变量、数据类型及标识符命名规则
3. 学会使用 print()和 input()实现基础输入输出

实验准备

- 软件: Python 安装包、VSCode、pip 国内镜像源 (清华/阿里云)
- 前置知识: 无

实验步骤

1. 环境配置

- a. 演示 Python 安装流程 (自定义路径+环境变量配置)
- b. 在 VSCode 中安装 Python 插件并配置中文语言包
- c. 测试环境: 在终端打印 print("Hello, 人工智能!")

2. 基础语法练习

- a. 变量定义与赋值: python

```
name = "小明"
age = 18
print(name, "年龄: ", age)
数据类型转换: python
price = float(input("商品单价: "))
total = price * 3
print("总价: ", total)
```

3. 调试与拓展

- a. 常见错误：中文标点、缩进缺失、未转换 input()类型
- b. 拓展任务：设计程序计算圆的面积（输入半径，输出面积）

实验 7：超市记事本实验

实验目标

1. 综合运用变量、输入输出和数据类型
2. 实现商品信息记录（名称、单价、数量）
3. 计算并打印消费总金额

实验步骤

1. 核心代码实现

```
# 输入购买的商品名称，单价和数量，计算总价，显示一条购买记录
# 用 input()函数获取数据，默认 str 类型
name = input("请输入商品的名称:")
price = input("请输入商品的单价:")
numbers = input("请输入商品的数量:")
# 数据类型转换
price = float(price)
numbers = int(numbers)
# 计算总价
totalPrice = float(price) * int(numbers)
print("购买商品:",name,",单价:",price,"元,数量:",numbers,"件,总价:",totalPrice,"元
\r\n")
```

2. 异常处理

- a. 演示非数字输入时的报错（如输入字母代替价格）
- b. 引入 try-except 基础保护：python

```
try:
    price = float(input("单价: "))
except
    ValueError: print("请输入数字! ")
```

3. 拓展任务

- a. 增加多商品录入功能（使用列表暂存数据）
- b. 添加时间戳记录购买日期（import time）

实验 8: 猜数字游戏实验

实验目标

1. 掌握 random 模块生成随机数
2. 使用 while 循环和条件分支实现游戏逻辑
3. 理解循环控制 (break 终止)

实验步骤

游戏逻辑实现

```
import random

# 随机数范围[0,20]
num = random.randint(0,20)
print('***** 猜数字游戏 *****')

while True:
    print('请输入 0-20 之间的整数')
    guess = int(input('猜猜她心里想的是哪个数字:'))
    # 判断输入的值是否等于随机数
    if guess == num:
        print('恭喜你,猜对了^_^')
        print('游戏结束,不玩啦')
        break
    else:
        if guess > num:
            print('猜大啦')
        elif guess < num:
            print('猜小啦')
```

1. 难度升级:

- a. 增加尝试次数限制 (如 5 次后失败)
- b. 添加游戏重启功能 (外层循环询问"再玩一局? ")

2. 调试重点

- a. 处理非数字输入导致的崩溃
- b. 随机数范围验证 (避免超出提示区间)

实验 9：图书管理系统实验

实验目标

1. 使用列表存储多本图书信息
2. 通过字典管理学生借阅记录
3. 实现增删查改功能

实验步骤

1. **数据结构设计**
2. **功能实现**
 - a. 借书功能
 - b. 查书功能:

调试与拓展

- a. 键不存在时报错处理 (if stu_id in students)
- b. 拓展：持久化存储 (数据保存到 books.txt)

实验 10 使用 Pillow 库打开、显示和保存图像

实验目标

1. 掌握图像的基本操作，包括打开、显示和修改图像属性。
2. 通过 Pillow 库实现图像格式转换与简单滤镜效果应用。
3. 理解图像处理在实际项目中的应用场景。

实验步骤

```
# 加载 Pillow 库
from PIL import Image
# 打开一幅图像
PIL_Image = Image.open("./cat.png")
# 打印图像信息
print(PIL_Image.format, PIL_Image.size, PIL_Image.mode)
# 显示图像
PIL_Image.show()
# 保存图像
PIL_Image.save('./cat_2.png')
```

实验 11 OpenCV 使用和常见库实验

一、核心库功能说明

- OpenCV : 核心图像处理库, 用于图像读取、预处理 (二值化/降噪/通道转换/缩放)、特征提取及目标检测¹²。
- Matplotlib : 数据可视化工具, 用于显示图像处理结果及模型输出⁴。
- PyTorch : 深度学习框架, 支持构建卷积神经网络 (CNN) 等复杂模型¹。
- Torchvision : 计算机视觉工具包, 提供预训练模型和数据集加载功能 (如 MNIST)¹。
- NumPy : 科学计算基础库, 支撑 OpenCV 的矩阵运算和图像数据操作⁴。

二、OpenCV 关键操作详解

1、图像读取与通道转换

```
import cv2

img = cv2.imread('Lena.png') # BGR 格式读取
b, g, r = cv2.split(img)      # 通道分离
rgb_img = cv2.merge([r, g, b]) # 转 RGB 格式
```

2、图像处理技术

```
二值化    : cv2.threshold()
降噪      : cv2.medianBlur()
几何校正  : cv2.getPerspectiveTransform()
```

3、可视化增强

```
enlarge_img = cv2.resize(img, (int(width*5), int(height*5))) # 放大 5 倍显示
cv2.imshow('Enlarged Image', enlarge_img)
cv2.waitKey(0) # 等待按键关闭窗口
```

三、多库协作流程

(1) 数据准备 (Torchvision)

```
from torchvision import datasets, transforms
```

```
transform = transforms.Compose([
```

```
    transforms.ToTensor(),
```

```
    transforms.Normalize((0.5,), (0.5,))
```

```
])
```

```
train_data=datasets.MNIST(root='./data',train=True,transform=transform, download=True)
```

(2) 模型训练 (PyTorch)

```
import torch.nn as nn
```

```
class ConvNet(nn.Module):
```

```
    def __init__(self):
```

```
        super().__init__()
```

```
        self.conv1 = nn.Conv2d(1, 10, kernel_size=5) # 卷积层
```

(3) 结果可视化 (Matplotlib)

```
import matplotlib.pyplot as plt
```

```
# 显示预测结果
```

```
plt.imshow(np.transpose(img.numpy(), (1, 2, 0)))
```

```
plt.title('Prediction Result')
```

```
plt.show()
```