

# 初级图像图形算法工程师测试题 - 答案与评分标准

本文档仅供内部评分使用

---

## 一、传统图像处理(25分)

### 1. 选择题(每题2分,共10分)

1.1 答案: B - 高斯滤波用于去除高斯噪声 (正态分布噪声)

1.2 答案: D - Canny使用高斯滤波先降噪, 对噪声最鲁棒

1.3 答案: B - 腐蚀使物体变小, 边界向内收缩

1.4 答案: D - 傅里叶变换是可逆的, 不会损失信息

1.5 答案: B - 直方图均衡化用于增强图像对比度

---

### 2. 简答题(15分)

#### 2.1 滤波器对比 (5分)

噪声类型:

- 高斯滤波: 适合高斯噪声, 加权平均平滑
- 中值滤波: 适合椒盐噪声, 用中值替换

边缘保持:

- 中值滤波更好 (非线性, 不模糊边缘)
- 高斯滤波较差 (线性平均, 会模糊)

计算复杂度:

- 高斯:  $O(k^2)$ , 可分离优化到 $O(k)$
- 中值:  $O(k^2 \cdot \log k)$ , 需要排序

评分: 噪声类型2分, 边缘保持2分, 复杂度1分

---

#### 2.2 形态学操作应用 (5分)

## 去除噪点方法：开运算

### 步骤：

1. 腐蚀：小噪点被完全去除，主要物体缩小
2. 膨胀：恢复主要物体大小，噪点不再恢复

### 伪代码：

```
python  
kernel = cv2.getStructuringElement(cv2.MORPH_RECT, (3,3))  
result = cv2.morphologyEx(image, cv2.MORPH_OPEN, kernel)
```

### 开运算 vs 闭运算：

- 开运算(腐蚀+膨胀)：去除白色噪点，分离粘连物体
- 闭运算(膨胀+腐蚀)：填充黑色空洞，连接断裂轮廓

**评分：** 方法2分，步骤2分，对比1分

---

## 2.3 图像增强 (5分)

### 使用场景：

- 低对比度图像（灰度集中）
- 曝光不足/过度
- 需要增强细节（医学影像、卫星图像）

### 副作用：

- 噪声被放大
- 可能过度增强，不自然
- 全局处理，不同区域需求不同

### CLAHE改进：

- 分块处理：图像分小块( $8 \times 8$ )，每块独立均衡化
- 限制对比度：设置直方图最大高度阈值，防止过度增强
- 块间插值：双线性插值平滑过渡

**评分：** 场景1.5分，副作用2分，CLAHE1.5分

## 二、深度学习图像算法(30分)

### 3. 选择题(每题2分,共8分)

3.1 答案: B -  $1 \times 1$  卷积降低计算复杂度和调整通道数

3.2 答案: B - ResNet残差连接解决梯度消失问题

3.3 答案: B - NMS用于去除重复检测框

3.4 答案: D - mAP是目标检测指标, 不是分割指标

### 4. 卷积计算题(7分)

输入:  $64 \times 64 \times 32$ , 卷积:  $3 \times 3$ , stride=1, padding=1, 输出64通道

卷积后尺寸 (2分):

$$H = (64 - 3 + 2 \times 1) / 1 + 1 = 64$$

$$W = (64 - 3 + 2 \times 1) / 1 + 1 = 64$$

$$C = 64$$

答案:  $64 \times 64 \times 64$

池化后尺寸 (1分):

$2 \times 2$  max pooling, stride=2

答案:  $32 \times 32 \times 64$

3×3卷积参数量 (2分):

权重:  $3 \times 3 \times 32 \times 64 = 18,432$

偏置: 64

总计: 18,496

1×1卷积参数量 (2分):

权重:  $1 \times 1 \times 32 \times 64 = 2,048$

偏置: 64

总计: 2,112 (减少88.6%)

## 5. 目标检测与分割(8分)

### 5.1 目标检测基础 (4分)

#### IoU计算 (1.5分):

$$\text{IoU} = \frac{\text{交集面积}}{\text{并集面积}}$$

$$\begin{aligned}\text{交集} &= \max(0, \min(x_{2\_1}, x_{2\_2}) - \max(x_{1\_1}, x_{1\_2})) \times \\ &\quad \max(0, \min(y_{2\_1}, y_{2\_2}) - \max(y_{1\_1}, y_{1\_2}))\end{aligned}$$

$$\text{并集} = \text{area1} + \text{area2} - \text{交集}$$

取值：0到1，越大越重叠

#### 两阶段 vs 单阶段 (2分):

维度	两阶段(Faster R-CNN)	单阶段(YOLO)
流程	RPN生成候选框 → 分类回归	直接预测
精度	高，小目标好	略低
速度	慢	快，可实时

#### Anchor作用 (0.5分):

- 预定义多尺度多比例的候选框
- 预测相对anchor的偏移量
- 简化学习，覆盖各种目标

评分： IoU 1.5分，对比2分， anchor 0.5分

## 5.2 图像分割 (4分)

#### 语义 vs 实例分割 (1.5分):

- 语义分割：像素分类，同类物体不区分（3只猫都标为“猫”）
- 实例分割：像素分类+实例区分（分别标为猫1、猫2、猫3）

#### FCN改进 (1.5分):

- 全卷积：用 $1 \times 1$ 卷积替代全连接层，接受任意尺寸输入
- 上采样：反卷积恢复分辨率
- 跳跃连接：结合浅层细节和深层语义

#### U-Net跳跃连接 (1分)：

- 恢复细节：传递高分辨率特征，恢复精细边界
- 梯度流动：提供更短梯度路径
- 多尺度融合：浅层细节+深层语义

评分：分割对比1.5分，FCN 1.5分，U-Net 1分

---

## 6. 实践题(7分)

#### 选择迁移学习 (2分)：

- 10,000张数据有限，从头训练易过拟合
- 预训练模型已学习通用特征
- 收敛更快，精度更高

#### Fine-tune方法 (2分)：

```
python

# 方法1：冻结backbone，只训练分类头
model = resnet50(pretrained=True)
for param in model.parameters():
    param.requires_grad = False
model.fc = nn.Linear(2048, 2)

# 方法2：全部微调（不同学习率）
optimizer = optim.Adam([
    {'params': model.layer1.parameters(), 'lr': 1e-5},
    {'params': model.fc.parameters(), 'lr': 1e-3}
])
```

#### 数据增强 (2分，至少3种)：

- 随机水平翻转、旋转(-15°~15°)
- 随机缩放和裁剪
- 亮度、对比度、色调调整

- Random Erasing / Cutout

### 评估方法(1分):

- 指标: Accuracy, Precision, Recall, F1, 混淆矩阵
- 划分: 训练70%, 验证15%, 测试15%
- 监控: loss和accuracy曲线, 观察过拟合

评分: 迁移学习2分, fine-tune 2分, 增强2分, 评估1分

---

## 三、计算机视觉基础(17分)

### 7. 选择题(每题2分,共6分)

7.1 答案: D - SIFT基于灰度图, 不考虑颜色

7.2 答案: D - Canny是边缘检测算子, 不是特征描述子

7.3 答案: C - RANSAC用于去除误匹配

---

### 8. 特征匹配与图像拼接(11分)

#### 8.1 特征点检测与匹配 (6分)

##### 检测步骤 (2分):

1. 检测关键点: 多尺度DoG寻找极值点
2. 关键点筛选: 去除低对比度和边缘响应点
3. 方向分配: 基于梯度方向直方图
4. 生成描述子: SIFT 128维, ORB 256位二进制

##### RANSAC (2分):

思想: 随机采样找最优模型, 区分内点和外点

步骤:

1. 随机选最小样本集(4对点), 计算变换H
2. 计算所有点的重投影误差, 统计内点数
3. 重复多次(如1000次), 选内点最多的模型
4. 用所有内点重新估计最优H

### **ORB优势 (1分):**

- 速度快100倍 (基于FAST)
- 免费开源 (SIFT有专利)
- 二进制描述子，匹配用汉明距离更快

**应用场景 (1分):** 图像拼接、物体识别、3D重建、图像检索、SLAM

**评分:** 步骤2分，RANSAC 2分，ORB 1分，应用1分

---

### **8.2 图像配准与拼接 (5分)**

#### **拼接步骤 (2分):**

1. 特征提取：检测特征点和描述子
2. 特征匹配：最近邻匹配
3. 变换估计：RANSAC估计单应性矩阵H
4. 图像变换：用H变换图像
5. 图像融合：处理重叠区域

#### **融合方法 (2分):**

- 渐入渐出： $\text{result} = \alpha \times \text{img1} + (1-\alpha) \times \text{img2}$ ， $\alpha$ 从1到0
- 多频段融合：低频宽过渡，高频窄过渡
- 泊松融合：保持梯度一致性

**失败情况 (1分):** 重叠区域太少、光照差异大、透视变形严重、运动物体、重复纹理

**评分:** 步骤2分，融合2分，失败1分

---

## **四、计算几何(15分)**

### **9. 选择题(每题2分,共4分)**

**9.1 答案:** B - Voronoi单元内的点到该单元种子点距离最近

**9.2 答案:** B - Delaunay三角剖分最大化最小角，避免狭长三角形

---

## 10. 简答题(11分)

### 10.1 Voronoi图与Delaunay三角剖分 (5分)

Voronoi图定义 (2分):

给定n个种子点，Voronoi图将平面划分为n个区域

$$V(p_i) = \{ x \mid d(x, p_i) \leq d(x, p_j), \forall j \neq i \}$$

即：到 $p_i$ 距离最近的所有点的集合

对偶关系 (1.5分):

- Voronoi边  $\leftrightarrow$  Delaunay边 (相邻单元的种子点连边)
- Voronoi顶点  $\leftrightarrow$  Delaunay三角形 (围绕顶点的3个种子点)

应用 (1.5分):

- Voronoi: 最近邻查询、区域划分、路径规划、图像分割
- Delaunay: 网格生成、散点插值、地形建模、曲面重建

评分： 定义2分，关系1.5分，应用1.5分

---

### 10.2 凸包 (3分)

定义 (1分): 包含点集S的最小凸多边形 (橡皮筋套住所有点)

Graham扫描 (1.5分):

1. 找最低点 $p_0$ 作基准
2. 其他点按相对 $p_0$ 的极角排序
3. 用栈维护凸包，依次加点
4. 形成右转则弹出栈顶 (非凸)，左转保留

复杂度 (0.5分): Graham扫描:  $O(n \log n)$

评分： 定义1分，算法1.5分，复杂度0.5分

---

### 10.3 点与多边形 (3分)

射线法 (1.5分): 从点P发射水平向右射线，计算与多边形边的交点数

- 奇数个：P在内部
- 偶数个：P在外部

伪代码 (1分)：

```
python

def point_in_polygon(P, polygon):
    count = 0
    for i in range(len(polygon)):
        v1, v2 = polygon[i], polygon[(i+1)%len(polygon)]
        if (v1.y <= P.y < v2.y) or (v2.y <= P.y < v1.y):
            x = v1.x + (P.y-v1.y)*(v2.x-v1.x)/(v2.y-v1.y)
            if x > P.x:
                count += 1
    return count % 2 == 1
```

特殊情况 (0.5分)：

- 点在顶点/边上：单独判断
- 射线过顶点：只计算下顶点或使用严格不等号

评分：原理1.5分，代码1分，特殊情况0.5分

## 五、综合应用题(13分)

### 11. 停车场车位检测系统(13分)

(a) 方案设计 (4分)

推荐方案：深度学习分类

流程：

```
输入图像 → 车位ROI提取 → CNN分类(空/占用) → 时间平滑 → 输出状态
```

技术选择：

- 预先标注或自动检测车位位置
- 轻量级CNN(MobileNet/EfficientNet-Lite)二分类
- 多帧投票避免抖动

**评分：** 方案合理2分，步骤清晰2分

---

### (b) 技术细节 (5分)

#### 传统方法 (2分):

- 背景差分(GMM)检测前景
- 边缘检测：车辆边缘密度高
- 纹理分析：LBP/HOG特征
- 颜色直方图：车辆与地面差异
- 特征组合输入SVM分类

#### 深度学习 (2分):

- 模型：MobileNetV2二分类（空/占用）
- 输入：裁剪的车位图像
- 迁移学习：预训练+微调
- 数据增强：光照、旋转、遮挡

#### 光照/阴影/遮挡 (1分):

- 光照：数据增强模拟多时段，CLAHE归一化
- 阴影：HSV空间分析，亮度归一化
- 遮挡：训练时加入遮挡样本，时间序列跟踪

**评分：** 传统2分，深度学习2分，鲁棒性1分

---

### (c) 工程考虑 (4分)

#### 提升准确率 (2分):

- 数据：多样化采集（时段/车型/角度），高质量标注
- 模型：迁移学习，超参数调优，模型集成
- 后处理：时间平滑(5帧投票)，空间一致性，规则约束

#### 挑战与方案 (2分):

挑战	解决方案
实时性	降低检测频率(1-5fps), 轻量级模型, GPU加速
光照变化	多时段训练, 图像归一化, 红外摄像头
遮挡	多角度摄像头, 时间序列跟踪
边界情况	明确规则(>50%算占用), 时间辅助

**评分：** 准确率方法2分，挑战识别2分