



蘇州科技大學

SUZHOU UNIVERSITY OF SCIENCE AND TECHNOLOGY

电子与信息工程学院

# 毕业论文撰写指南

## V2.0

计算机工程系 沈忠伟

2025-4

## 毕设类型与论文思路

毕设要求，1. 需要一定的编程开发工作量 2. 具有一定的探索创新。

因此定题有 2 种情况：

1. **面向算法**的任务。需要学习算法匹配应用，如“基于字典学习的人脸识别系统设计”。此类任务大部分算法代码都可以找到现成的代码，主要的编程工作是算法在具体任务上的落实（即匹配算法设计完成一个软件外壳），系统完成后需要基于基本科研思路做后期分析和实验。主要难点是在论文中准确表达算法核心思路，附核心代码，进行性能分析。
2. **面向开发**的任务。需要学习技术，如“基于 Django 框架的分布式云盘设计”。此类任务存在一定的开发复杂性，但需求明确，解决的问题也很清晰（如此例中主要需求是云端存储和分布式系统）。完成开发后将相关分析和实现过程讲述清楚即可完成论文。

★论文主要是向评阅老师**展示自己解决问题的思路**，具体实现写代码是相对次要的，实现效果在结果展示中体现。所以一定**不可以**出现增删改查等基本操作，而应当面向**需求、设计、性能** 3 个方面展开。

论文内容主要跟随软件工程的思路，两种任务都需要从需求出发，预期达成什么功能（不仅仅是算法的核心功能，界面 UI 需要包含的功能也非常重要），功能分为哪些模块，分别怎么规划，实际怎么做，达到了怎样的效果。

面向算法的任务要凝练需求，识别、去噪等任务在物理意义上要做什么，我们现实中的数据有什么特点（一般就是数量少，类别不均匀等等），理论上算法能做什么要和需要什么匹配上。在结果分析部分就算法各个步骤展开测试，需要和同类算法进行一些对比体现被选用的算法的优越性，**说明选择这个算法是自己研究的结论**，且具有一定的性能。

## 总纲要求

1. 论文一共有 8 个章节，分别是 1.绪论 2.相关基础知识 3.需求分析 4.系统设计（概要设计和详细设计）5.系统实现 6.结果展示与分析 7\*.系统测试 8.总结。
2. 面向算法的任务，需要分析其他方法的不足和自己所用方法的优势。通常要学习一些数学公式的，都是传统方法，可以针对小规模数据的应用场景展开分析，谈实际场合的应用结合深度学习的收敛性要求，由于传统机器学习方法收敛快、资源需求低，具有实际应用上的优势（为什么这个算法），然后根据算法原理展开谈。而深度学习算法就谈性能优势，根据算法原理展开。对应的论文和相关博客上都有算法原理的优势分析，可以参照。
3. 一票否决型红线要求：
  - 1) 不可以用网图，必须自己画自己截，不会用 visio 这些比较专业的软件的建议用 PPT 画；
  - 2) 不可以复制粘贴任何网上的描述，也不可以用 GPT，都会造成查重问题。全部用自己的话说，且不可故意水；

- 3) 图中字体除了确实需要强调的标题，一般都不可大于正文，字体大小应统一，至少看起来差不多，图应比较紧凑，不可以故意把图画得稀疏来占据篇幅，会给评阅老师带来很差的印象；
  - 4) **注意字体等格式要求**，一般来说，图里面的字体要差不多大，不可拉伸变形，不可大于正文字体（但强调的重要模块字体是可以大的）；
  - 5) 不应在除了章节末尾的地方存在大段空白。
4. 描述性语言层面的要求：
- 不允许笼统地说“具有更好的性能”，而应该具体说明原因。如下范例：
- 算法自适应地将特征表示集中在人脸的外观表征上，排除了无关信息的干扰，因而提升了识别性能；
- 用户在访问外网之前，内网的验证服务器在请求中附加了签名。此节点对于外网用户而言是不可访问的，因此在系统注册步骤即构造了网段限制，提高了安全性。

## 内容安排

### 中英文摘要

摘要需要简短精炼，概括说明论文的内容。这部分比较不好写，可以在论文完成最后写，包含如下内容，每条一句话：

1. 论文的课题有什么意义；
2. 当前的研究或者业界的一般解决方案有什么问题；
3. 针对此问题的有什么办法；
4. 怎么做；
5. 进一步完善的点在哪。

有一些课题，如出票系统研究，因为业界解决方案高度太高而且闭源，我们是很难说出他有什么问题的。第二条就应该谈此系统面临的挑战在哪里，如站点多车次繁杂、票务仅涉及部分区间导致乘客的票务操作会相互影响、用户操作量极大、系统稳定性要求高等等。

相当一部分同学都会犯的错误就是将摘要和绪论一开始的部分重复。

英文摘要可以让 GPT 帮助翻译，这是唯一可以采用 AI 的地方，其他地方凡运用 AI 皆属于严重的学术不端。

### 绪论

#### 1.1 研究背景与意义

开场通常是“随着 XXX 的发展，XXX 逐渐进入了日常生活 XXX 方面，并在 XXX 以及 XXX 中起到了重要用途”的句式，注意自行调整避免查重问题。愿意的同学可以去百度一下国务院文件、白皮书、政府工作安排、十四五规划等涉及的相关内容，拔高主题然后引用一下。

第一段写 2-3 行，然后画个图，将论文研究内容在现实生活中的应用贴一下，适当分析差不多就完成了一页。

1.2 国内外研究现状，不要分开说国内如何国外如何，就把相关技术详细讲解一下。

- 面向开发的工作，简单谈谈国内头部企业是怎么做的，不妨说说国内外因为产业基础的差异，解决方案有什么不同，可以简单聊聊国内从业者如何运用聪明才智填补产业技术的差距，以及说一些正能量。
- 面向算法的工作，需要调研一下相关的研究基础，以人脸识别为例，从 **eigenface** 开始到各种稀疏低秩方法以及深度学习方法，进行一些简单的解析，尽管如此这也会耗费相当的时间，可以放一些图，同样不能截图而应该自己重绘。

面向算法的论文建议篇幅是 7-9 页，面向开发的论文则建议 5-6 页。绝大部分文献引用都发生在这个部分。

1.3 本文研究内容

1.4 本文章节结构。这个部分注意绪论也要包含在内。

## 相关基础知识

2.1 框架技术介绍；

2.2 -2.x 逐一具体介绍系统所用的相关算法。

本章节内容比较简单，注意查重问题，不再赘述。

## 需求分析

该部分面向开发的任务要做到至少 4 页，而面向算法的任务应做到至少 5 页。

主要分为如下几个小节：

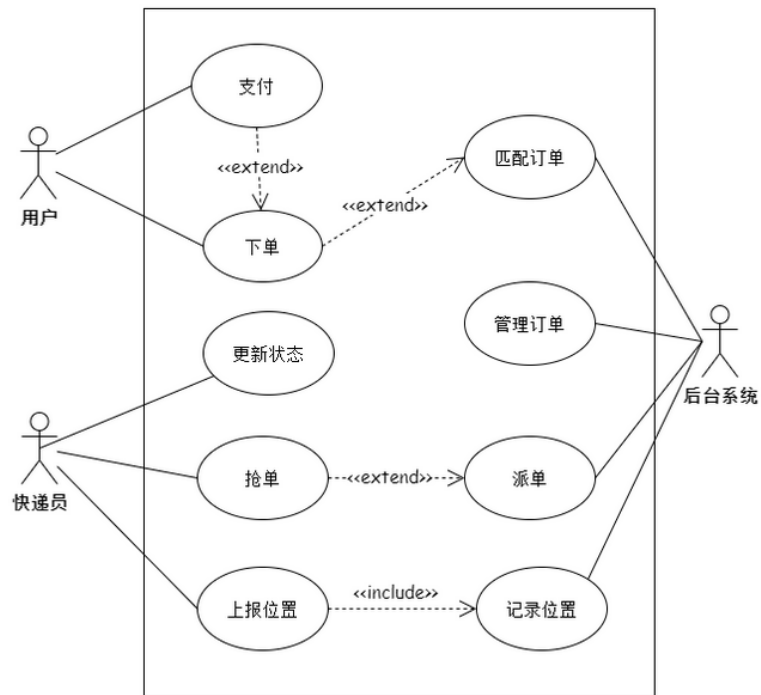
3.1 需求概览

3.2 可行性分析

3.3 业务功能分解（列举功能）

3.4 功能性需求分析（以用例图和用例分析辅助说明问题）

用例图示例：



用例图中包含以下三种关系：

- 包含关系使用符号《include》，想要查看订单列表，前提是需要先登录。
- 扩展关系使用符号《extend》，基于查询订单列表的功能，可以增加一个导出数据的功能
- 泛化关系，子用例继承父用例所有结构、行为和关系。

注意每一个插图都需要对应文字解析,功能有哪些相互关系是什么等等。注意用例需要和 3.3 罗列的功能一一对应。

在 3.4 中需要详细说明每个功能模块的作用。对于核心功能，可以做用例分析表（不重要的功能只需要简单叙述即可），以下是一段较为完整的核心功能模块用例分析示例：

### 3.4.1 商品信息展示功能模块

商品信息展示模块是电商平台的首页。旨在满足用户的需求。该模块包含四个子功能模块：商品分类展示、商品轮播展示、商品推荐展示、商品热卖展示。商品分类展示平台中的商品按照分类进行组织,使用户能够方便快捷地浏览不同类别的商品，增强了用户的购物体验。商品轮播展示平台中的热门或推荐商品，为用户提供了一种动态的、吸引人眼球的浏览商品的方式，提高了用户对商品的关注度和购物欲望。商品热卖展示为用户提供了浏览热销商品的便捷方式，帮助用户发现热门商品并提高购买意愿。商品推荐展示为了满足用户的个性化需求，平台在商品信息展示页面上提供了当前平台内推荐的商品，使用户能够体验到个性化定制的便捷性。用户可以通过点击相应推荐商品，快速定位到该商品的详情页。

该模块以商品分类展示为例进行用例分析，商品分类展示用例的分析已在表 3-1 中详细描述。

表 3-1 商品分类展示用例分析表

用例编号	UC-001
用例名称	商品分类展示
参与者	用户（已登录或游客）
优先级	高
用例描述	用户在电商系统上选择浏览商品分类
触发事件	用户进入商品分类展示页面
前置条件	用户已登录电商系统或作为游客访问电商系统
基本事件流	1) 进入系统首页 2) 系统显示可用的商品分类列表 3) 用户选择感兴趣的商品分类 4) 系统根据用户的选择显示对应商品分类下的商品列表
异常事件流	无
后置条件	系统显示选定商品分类下的商品列表

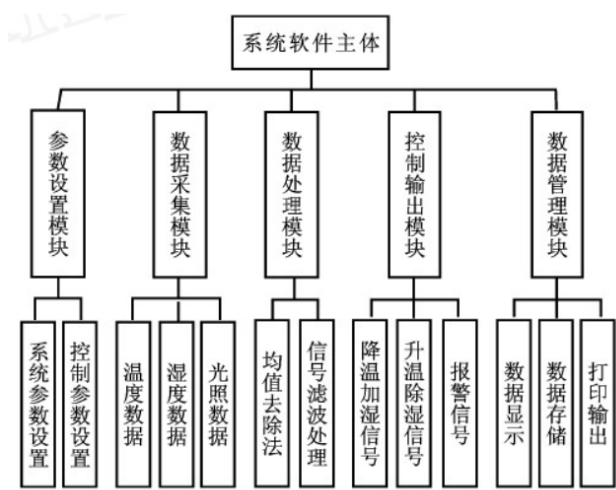
系统设计

概要设计和详细设计合成系统设计章节，概要设计主要围绕**软件模块结构体系**展开分析，而详细设计则基于概要设计的模块划分针对模块内功能进行分析。所以本章节大体上的结构是：

- 4.1 系统模块体系
- 4.2-4.x 各个模块内部的设计

概要设计

软件模块结构体系图示例：



对应文字解析模块和功能的关系，模块划分的依据，等等。  
擅长 OOP 的同学也可以在这个部分画接口图、类图、组件图和部署图并展开分析。这些内容相对比较灵活，只需要围绕核心模块展开分析即可。

其中接口的概念指的是实现该接口的类包含的方法声明，只涉及有什么功能(函数)的声明，而不关注具体实现。接口设计的意义是说明体系结构之间的功能关联，是可以和数据流图结合起来分析的。

以下是类图的示例：

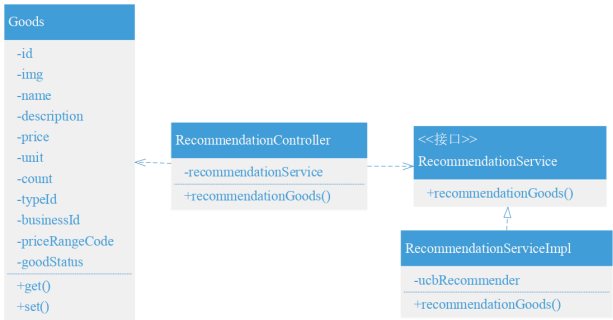


图 5-4 商品推荐功能类图

不熟悉软件工程流程的同学可以采取匹配界面 UI 进行反推的策略，至少每个网页的页面/软件的界面对话框都是一个模块，有时候一个页面比较复杂，就可以对应更多的模块。我们可以先写系统实现中 UI 设计图部分（下一节），然后反过来抽象体系结构和接口设计。

详细设计

概要设计中划分了多少模块，详细设计中就应当有多少个小节与之对应。

主要通过以下图辅助相关分析（根据需求选用），解构功能模块：

- 1. 活动图（流程图）
- 2. 状态图
- 3. 时序图

UML 图绘制的方式可以参照这个链接，也可以按需选择其他的 UML 图辅助说明软件的设计思路，不再赘述：<https://zhuanlan.zhihu.com/p/520475069>

本章节“系统设计”中，不需要将概要设计和详细设计拆分得特别明确。第一小节阐述软件的整体架构之后，主要模块与其他模块之间的联系（比如流程逻辑关系、数据流关系等）都可以在模块的详细设计中阐述。且不局限于上述的 3 种 UML 图，比较善于 OOP 的同学也可以用对象图等说明对象模块之间的关系。

无论是哪一种任务，软件设计都是论文的关键组成部分。即使是非常简单的软件框架，依然在设计中需要对功能模块进行划分，考虑不同功能模块之间的关系，所以即便在实际应用中 UML 图意义较小，依然是展示设计思路的重要途径。在论文行文过程中，要注意详略得当，涉及到每一个模块，但是只有关键的功能模块需要引入 UML 图辅助说明。

特别地，在面向算法的任务中，核心算法是重要模块，需要针对核心算法模块进行分析，匹



配需求。下面以“潜在低秩表示”给出一个语言叙述的示例：

先谈干净数据的分布特征：

因为人脸数据具备天然稀疏性[此处需要文献引用论证]，当字典可以合理表示数据特征分布的时候，字典各行之间构成高度冗余，从而可用低秩约束字典的线性相关性。放公式、推导论证，注意加引用。

这个部分是潜在低秩项的分析：

而训练样本分布往往不能与实际样本分布完全一致，从而加入自适应的不可见样本分布表示可以进一步提升特征的拟合。放公式、推导论证，注意加引用。也可以放一个流程图。

然后谈噪声的均匀分布与干净数据稀疏性之间的冲突，所以低秩模型可以用于滤除噪声：相比于传统稀疏表示[此处引用两三个，然后逐一简单说明]，如 XX 采用 XXXXXX 策略，在 XXXX 场景下具有 XXXX 问题[此处需要文献引用论证]；XX 采用 XXXXXX 策略，在 XXXX 场景下具有 XXXX 问题[此处需要文献引用论证]。我们关注 XXX 噪声，因为 XXXXXXXX，从而在传统稀疏表示中往往具有 XXXX 不足，而潜在低秩表示因为考虑到 XXXX，更适合此类任务。后续实验中我们也将就此进行验证研究。

**数据库设计（面向开发的任务）**

如果在系统中用到数据库，相对独立的数据库设计可以放在详细设计中进行阐述。这个部分可以借助 E-R 图分析，还可以用关系代数描述数据表的存储关系。注意，数据表中应有**外键连接**，否则大概率设计中存在问题。这部分的要求是文字描述+插图，表结构展示只需要最关键的两三个，不应将表结构大规模复制到论文中凑篇幅。大概篇幅是 2-3 页，其中文字描述应当有一半。

E-R 图示例如下，注意实体关系：

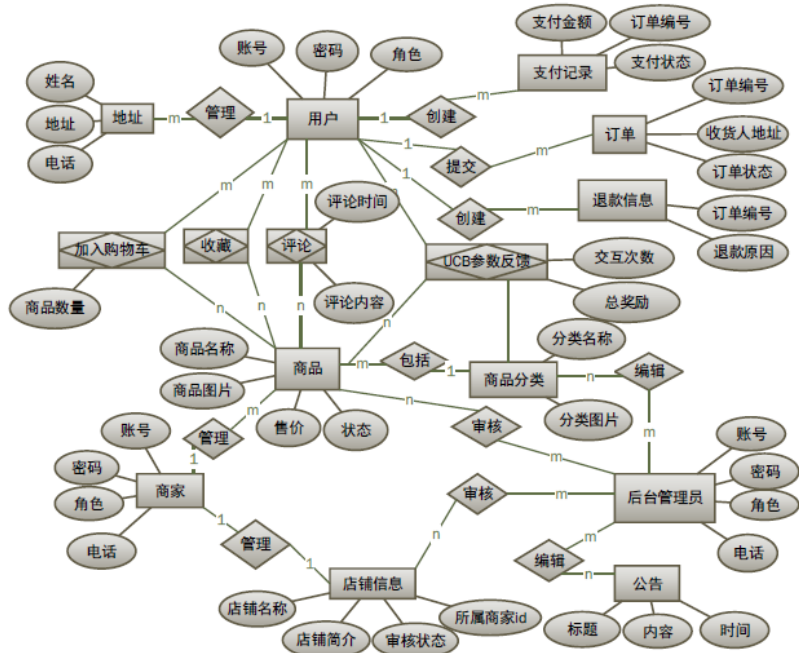


图 4-6 电商系统整体 E-R 图



系统设计较为重要，篇幅应不低于 10 页。

## 系统实现

系统实现的主体内容是对应每个模块的程序流程、UI 设计图及其对应核心代码。其中如果一个界面对应一个功能模块，那么基于界面设计图展开阐述；如果一个界面包含多个功能模块，则需要结合体系结构进行说明，方法比较多样，比如可以用程序流程图（应当有至少一个）也可以直接文字描述。

针对 UI 的说明，建议采用这样的形式：

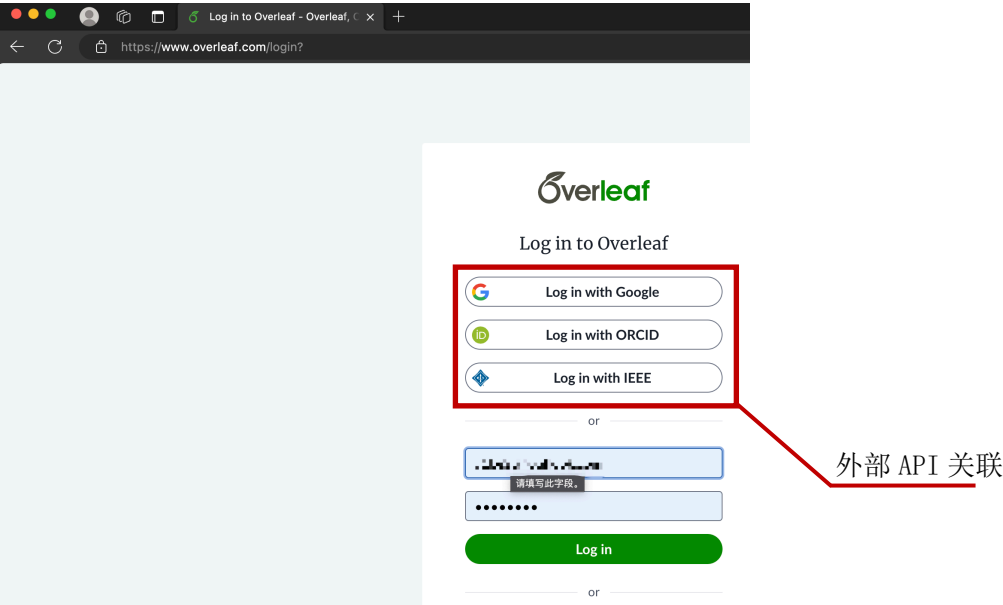


图 1 UI 标注示例

如图 1 所示，外部 API 关联了系统的登陆，使用户可以跳过注册与繁杂的账户管理快捷地完成系统登录，提升用户体验，降低信息泄露风险。同时，复用外部信息管理 API 使站点可以以低成本维护用户数据安全，提升系统稳定性。其核心代码如下：

核心代码一般控制在 5-6 行，插入一个 1x1 的表格，然后用 vscode 打开代码后把它保留原格式贴进去。如果是面向算法的任务，就需要结合代码再讲一下这段代码具体的功能，尤其是超参数的设定之类的。如下面的示例，有注释更好：

```
# 用随机数据测试模型的 FLOPs 和参数量
input = torch.randn(8, 3, 224, 224)
flops, params = profile(model, inputs=(input, ))
flops, params = clever_format([flops, params], "%.3f")
```

如果采用的是神经网络，核心代码可能比较长，建议进行一下拆分，比如先画网络结构图，然后插入网络结构定义的代码（PyTorch 位于构造函数中），适当讲解，如有较为复杂的逻辑，也可适当插入 forward 函数代码。都需要有对应分析。

本章节由于需要插入核心代码，篇幅大约在 5-10 页。

结果展示与分析

面向算法的任务：

这个部分非常重要。主要包含 2 方面内容：1.结果展示 2.算法对照。

建议找 2 个，至少要有 1 个同类对照算法进行定性和定量的分析：

- 1. 定性分析指的是直观视觉效果是否符合算法原理，是否能够一目了然地看出算法有效。定性分析需要截图对比，然后分析图像表征上的优劣势。
- 2. 定量分析指的是指标点的计算和对照。比如人脸识别，那么识别率（accuracy），后验概率（MAP），F1 Score 等都是。通常我们画一个表格进行直观对照，进一步也可以画 ROC 曲线计算 AUC。对于图像质量的评价，通常我们选择结构相似性（SSIM）和峰值信噪比（PSNR）。在画表的时候，注意要标注↑和↓表示这个值是越大越好还是越小越好。

算法性能分析分为 2 步骤，首先写消融实验，然后写同类算法对比，注意次序。

消融实验指的是控制变量法。这个算法分为多少个模块（可以参照算法的原论文），每个模块加上对比不加，都可以做定性和定量分析。如下示例：

为了保持实验的合理性，限制 tol 误差容忍度的值为默认值 1e-8。以 barbara.png 为例，启动 LRR 去噪模型生成去噪后的图像 Image\_lamb.png 之后尝试移除正则化项的约束，我们将 lamb 的值设置为 0，再次通过 LRR 去噪模型生成处理后的图像 Image\_nolamb.png 此时引入图像质量衡量指标 PSNR 以及 SSIM 通过横向对比前后的差异，尝试从中了解正则化项在控制低秩矩阵和稀疏矩阵之间平衡的作用。移除正则化项的影响之后，尝试通过构造一个正则化项的递增序列，拟设置为 0.001、0.01、0.1，基于此序列观察正则化项的影响。实验结果如表 6-2 所示：

表 6-2 正则化项消融实验结果表

图像	lamb=0		lamb=0.001		lamb=0.01		lamb=0.1	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
barbara	22.07	0.48	29.88	0.82	30.25	0.83	28.96	0.81

从上表中的数值观察可得当我们移除了正则化项之后，算法可能无法有效地找到数据的低秩近似，从而影响数据的去噪效果，从图像质量衡量指标也可看出不管是 PSNR 还是 SSIM 值相较于引入正则化项的约束与原始图像的 PSNR 还是平均结构相似性都是很低的，当我尝试引入正则化项之后，即将正则化项设置为 0.001 之后 SSIM 平均结构相似性有了显著的提升，几乎是翻倍式的提升，随着将正则化项从 0.001 改成 0.01，模型的去噪性能迎来了进一步的提升，这在 PSNR 和 SSIM 数值变化上都有体现，此时的去噪效果通过肉眼观察已经相对良好，此时进一步修改正则化参数，图像质量不增反而略微下降了一些，说明此时正则化项对于模型的影响已经趋于稳定。



图 6-1 LRR 正则化项消融实验效果图

示例对于超参数的选择进行了定性（图 6-1 的可视化）以及定量（表 6-2 的数据）分析。文字分析也需要一定的篇幅。一般可以这样组织语言：

这个部分我们分析了 XXX 模块的作用。如表 XX 所示，对照未加入 XXX 模块的情形，XXX 模块提高了 X% 的识别精度/mAP，在 XX，XX，XX 指标上都有较为稳定的提升，但在 XX 指标上的表现不明显，这可能是由于 XXX 原因所致。

我们分析了 XXX 参数对算法精度的影响。如图 XX/表 XX 所示，XXX 参数取值为 X 的时候，算法的识别精度达到峰值 X，其变化趋势较为平稳，对一般情况下的超参数取值有明确的指导意义。

对于一些特定算法，主要是人脸识别和图像去噪，都需要做重构。这些实验在原论文中都有。本质上这些算法就是集中提取关键性特征去除无关信息，所以人脸识别的重构可以去除遮挡，比如墨镜、围巾等等，而去噪的重构则是很直观的提升图像的质量。

面向算法的任务这个部分的篇幅不要少于 6 页。

## 软件测试

面向开发的任务可以做一些软件的单元测试。只做黑盒测试的话这个部分就和上一章节合并，篇幅 2 页左右，简单提一下某某模块做一些黑盒测试来验证正确性；有白盒测试的话可以单独开一章，说一说白盒测试的设计思路，怎样做的代码，然后配合一些黑盒测试结果，加起来 3-5 页为好，形式上与系统实现章节比较相似。