



本科生毕业论文（设计）

题目： 基于远程直接内存访问的
高性能内存传输优化

姓 名 兰 靖

学 号 18340085

院 系 计算机学院

专 业 计算机科学与技术

指导教师 肖依 (教授)

2022 年 3 月 27 日

**基于远程直接内存访问的
高性能内存传输优化**

**High-Performance Distributed Memory
Transfer Using RDMA**

姓 名	兰 靖
学 号	18340085
院 系	计算机学院
专 业	计算机科学与技术
指导教师	肖依 (教授)

2022 年 3 月 27 日

学术诚信声明

本人郑重声明：所呈交的毕业论文（设计），是本人在导师的指导下，独立进行研究工作所取得的成果。除文中已经注明引用的内容外，本论文（设计）不包含任何其他个人或集体已经发表或撰写过的作品成果。对本论文（设计）的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。本论文（设计）的知识产权归属于培养单位。本人完全意识到本声明的法律结果由本人承担。

作者签名：

日 期： 年 月 日

【摘 要】

随着大数据处理、大规模智能模型训练、强化学习等多样化、高负载的计算需求成为学术界和产业界的焦点,作为计算基础设施的分布式计算框架(例如 Mapreduce, Spark, Ray)正在得到更多的关注和更广泛的应用。通常,计算框架软件负责为用户调度计算任务,同时管理和最大限度地利用计算集群的物理资源。上述场景所产生的快速增长的计算需求,正在使计算集群的性能愈发捉襟见肘:一方面,高性能计算(HPC)集群正在取代廉价的计算机器以加速高强度计算。值得注意的是,前者所拥有的新型高性能网络,明显地区别于传统集群。以智能网卡为硬件基础,高性能计算机能够在无中央处理器(CPU)的介入下,对机器内存直接读写和传输数据。这种被称为远程直接内存访问(RDMA)的传输机制,相比以太网具有低占用、低延迟、高带宽等技术优势。另一方面,作为基础设施的软件框架,也在面临严峻的性能挑战。当前大多数分布式计算框架仍然缺乏对高性能硬件的支持,特别是无法充分利用集群中的高速网络资源。因此,本研究选取了分布式计算框架 Ray 的核心组件:分布式内存对象数据库 Plasma,首先测试和分析了其使用以太网传输数据而导致的性能瓶颈。然后,在高性能集群上,我们提出了一种支持 RDMA 特性的高吞吐数据传输机制。该机制针对大型数据,提出了基于单边读语义的传输协议,实现了用户态的内存零拷贝。并且该机制能够在运行时,根据数据大小动态地选择传输协议,以获得最佳性能。进一步,我们基于并行计算范式 MPI 构造了分布式、可扩展的多节点传输性能测试。在实验中,我们确定了传输协议选择的最优参数,并且展示了在天河高性能集群上,优化后的内存平台能够在常见大小的对象传输中实现至多 8 倍的吞吐率提升。

关键词: 数据传输, 高性能网络, RDMA, 键值对数据库, 分布式缓存

[ABSTRACT]

Keywords: data transfer, high-performance network, RDMA, key-value database, distributed memory

目录

1	绪论	1
1.1	选题背景与意义	1
1.2	国内外研究现状和相关工作	2
1.3	论文主要研究内容	6
1.4	论文结构与章节安排	7
2	本科生毕业论文写作与印制规范	8
2.1	毕业论文的撰写内容与要求	8
2.2	毕业论文的撰写格式要求	11
3	L ^A T _E X 模板配置与使用	17
3.1	使用 Overleaf 编写毕设	17
3.2	编译环境配置	19
3.3	写作环境配置	19
3.4	如何开始写毕业论文（设计）	20
4	简单的使用例子	22
4.1	图像的插入	22
4.2	表格的插入	25
4.3	公式	25
4.4	算法流程图	26
4.5	例子、定理与证明	26
4.6	代码	27
4.7	其他的一些用法	27
5	其他注意事项	29
5.1	关于生僻字	29
6	实验与结果	30

参考文献	31
附录 A 补充更多细节	32
A.1 补充图	32
致谢	33

插图目录

2.1	图片例子	15
3.1	在 Overleaf 上创建并上传压缩包。	17
3.2	在 Overleaf 上调整编译工具	18
3.3	Overleaf 使用例子	18
3.4	vscode 配置好后的样例	20
4.1	镶嵌在文中的图像	22
4.2	单张图像	23
4.3	并排的多张图像	23
4.4	并排的多张图像加各自的注解	23
4.5	复杂的两列对象的插入	24
A.1	一个配有彩色表格的插图	32

表格目录

2.1	字体使用规范	11
2.2	表格例子	14
4.1	典型的实验对比表格	25
4.2	复杂一些的表格	25

1 绪论

1.1 选题背景与意义

分布式计算框架是一种复杂的平台软件。传统的并行计算范式如信息传递接口 (MPI) 和分区全局地址空间 (PGAS), 通常为编程者提供丰富的调用接口和灵活的编程空间。然而, 使用这些编程标准的门槛过高: 编程者通常需要自己管理多台机器的状态, 特别是内存的分配和使用; 编程者还需要使用给定的标准原语, 精确地规定多个进程之间的通信和协作方式, 这通常需要大量时间和精力; 另外使用这些范式将导致程序和功能的强耦合——编程者很可能不得不重新编写代码来更新程序的逻辑。因此, 现代分布式计算框架通常承担了上述硬件管理的角色, 并针对目标任务类型, 为用户设计尽可能少、但简单易用的功能接口, 来降低集群的使用难度。传统的分布式计算框架, 例如 Mapreduce 和 Spark, 都是面向大规模数据分析而设计的。然而, 近些年以强化学习、复杂工作流等为代表的复杂计算需求, 需要更灵活的框架支持。加州大学伯克利分校 RISELab 实验室提出的 Ray 和芝加哥大学 Globus 实验室提出的 Parsl 是其中两个典型。这些框架吸取了云计算领域“函数即服务 (Function as a Service, FaaS)”的设计思想, 能够细粒度地以函数为单位将任务调度到集群上执行, 同时依然对用户隐藏绝大部分实现细节。Ray 通过分布式内存对象存储 Plasma, 实现了框架完全自主的集群内存管理, 让用户可以专注于实现功能逻辑, 无需担心数据的存放和移动。Ray 在保持易用性的前提下, 极大地提升了计算框架的灵活性, 用户只需要修改几行代码就能将单进程程序扩展到整个集群, 进而构建出分布式机器学习等复杂模型。

高性能集群 (或超算集群), 是一种以高端处理器、计算卡、高性能网络、大容量存储为核心硬件的计算集群。随着大数据应用的丰富、超大规模人工智能模型的出现, 高性能集群和超级计算机正在变得愈发重要。首先, 高性能计算机拥有普通机器不能比拟的计算能力, 主要表现为高端的多核处理器和计算卡。而随着大模型时代的到来, 应用对集群网络的需求逐渐提高, 高性能网络所表现出的高带宽、低延迟等特性也逐渐受到了更多的关注。在超算集群中, 网络性能上的优势很大程度上来自于对远程直接内存访问 (RDMA) 机制的支持。然而, 要利用 RDMA, 用户必须在应用中实现基于 RDMA 的通信机制, 而不能依赖于操作系统内核提供的系统调用。因此, 基于套接字 (Socket) 的网络应用大多不能在高性能集群中直接获得显著的性能提升。分布式计算框架 Ray 并不例外, 其分布式内存

存储 Plasma 目前仅有对传统 TCP/IP 协议的支持，因而在超算集群上无法发挥出应有的网络性能，进而影响 Ray 运行在超算上的总体性能。

当前，超算集群普遍使用的是英伟达(Nvidia)公司的 Infiniband 高速网络。对于使用 Socket 通信的网络程序，该架构通过“基于 Infiniband 的互联网协议 (IPoIB)”实现支持。值得注意的是，这是一种依赖操作系统内核的非原生支持，已经有多个工作表明，其网络性能和直接使用 RDMA 技术相比具有明显差距。因此，本研究的目的是：我们是否能为分布式内存存储 Plasma，提出并实现一种支持 RDMA 机制的内存通信协议，从而让 Plasma 乃至整个 Ray 框架在现代超算集群上获得更好的性能？从超算研究的趋势来说，应用软件和先进超算硬件之间的隔阂，正在逐渐成为大家关注的热点。随着超级计算机和云计算两个领域的融合，会有越来越多的软件运行在高性能集群中。然而，它们中的大部分还没有针对高性能硬件提供软件支持——通过提供软件对高性能硬件的支持，我们能够将这些应用的运行性能提升到全新的水平。

1.2 国内外研究现状和相关工作

1.2.1 分布式计算框架

分布式计算框架是一种复杂的平台软件。在底层，它通过实现任务调度、并发执行、内存管理等基本组件，向用户透明地提供分布式计算的功能；在应用层，它通过设计和规范一系列接口，帮助用户高效地运行特定任务：例如大数据分析 (Mapreduce、Spark)，分布式机器学习 (Ray) 等等。Ray 是近年来最受关注的分布式计算框架，它以函数为单位调度任务执行，完全自主地管理内存移动，并且支持异步执行。借助这一平台，目前社区人员已经实现了机器学习 (Ray ML)、强化学习 (Ray RLlib)、模型部署 (Ray Serve)、工作流 (Ray Workflows) 等上层应用库供用户使用。同时，用户也可以直接在框架上编写任意程序：使用常见的 Python 语法构建出完全分布式的计算程序，而且不会有任何表达能力上的限制。

```
import ray
ray.init()

@ray.remote
def f(x):
    return x * x

futures = [f.remote(i) for i in range(4)]
print(ray.get(futures)) # [0, 1, 4, 9]
```

在上述代码片段中，用户通过修饰符“@ray.remote”将函数定义为远程函数，并在下方执行 4 次。值得注意的是，在 Ray 框架的调度下，这四次调用将会逐一调度到不同的进程上并发执行，因此 Ray 能够透明地为任何函数提供并行计算能力。而分布式内存存储 Plasma，则在系统中持续为并发任务调度所需的数据。在示例代码中，Plasma 将会在集群中查找变量 i 的位置，并将其拉取到进程中供任务使用。最后，Ray 原生支持异步执行，任何函数调用都会立刻返回，给予用户一个凭证（future）——用户可以立刻用这一凭证作为参数继续调用其他函数，即使数据的真实值还没有求得。Ray 的执行引擎动态地构建和解决数据依赖，将凭证替换为真实值，然后继续调用依赖它的函数。异步的特性将使 Ray 充分发挥并发任务的性能。

1.2.2 远程直接内存访问技术（RDMA）

RDMA 全称 Remote Direct Memory Access，即远程直接内存访问，是一种于二十一世纪之后逐渐兴起的新型网络通信技术。其核心思想和直接内存访问（DMA）相似——在早期计算机系统当中，内存读写、移动都需要由中央处理器（CPU）直接操作，从而带来相当的性能开销。而现代计算机系统分离出了内存子系统，将内存相关的负载卸载到子系统的专用硬件上。因此，CPU 只需要在访问开始和结束时同子系统协作，而在漫长的访问延迟中间可以执行其他计算任务，大大提高了计算机的总体性能。RDMA 技术在这一方向上更进一步：支持 RDMA 的现代智能网卡，不仅能从网卡端直接寻址并读写本机内存，还能够和远端的另一个网卡协作，直接读写远端机器的内存空间，从而实现通信。这一过程中 CPU 通常只需极少的介入、甚至完全不需要，因此具有相当明显的技术优势：

- 1) 旁路内核（Kernel-bypass）。基于 TCP/IP 协议的网络通信已经逐渐不能适应现代高并发、重负载的网络应用。基于高速 SSD 的存储系统、基于内存的缓存和（键值对）数据库都需要在短时间内应对大量并发的 I/O 操作请求，工作表明这些应用的性能瓶颈位于 CPU，而不是 I/O 部分。臃肿的网络栈、用户态-内核态的切换、内核态的数据处理和拷贝等等，是影响网络系统性能的根本原因。智能网卡代替 CPU 执行内存移动，而且大多数 CPU 和网卡的交互都实现在用户态，因此大大减轻了 CPU 的负担。
- 2) 零拷贝（Zero-copy）。零拷贝是当前操作系统领域的一项热门技术，其思想在于提高内存的共享程度，例如减少进程-进程、用户态-内核态内存拷贝的数量，从而提高 I/O 的整体性能。在支持 RDMA 的应用中，由于内核旁路，一次内存操作往往能省去：用户缓冲到内核缓冲的拷贝、内核缓冲到硬件（驱

- 动) 缓冲的拷贝。数据直接在两端主存之间移动, 因此产生可观的性能提升。
- 3) 低延迟、高并发。简化的网络路径、零拷贝等特性大大降低了机器操作远端内存的延迟。在并发应用中, 更低的延迟意味着相同时间内更强大的并发处理能力。
 - 4) 异步通信。RDMA 是原生异步的通信机制, 进程需要主动访问完成队列 (CQ) 甚至直接检查内存数据, 才能得知通信的发生。这一特性让程序的并发潜力大大提升, 但于此同时, 编程者需要自己设计同步机制来完成通信, 同样增加了编程上的难度。

目前, RDMA 技术是智能网卡技术中较为成熟的一种。硬件支持 RDMA 的网卡通常都具有相当惊人的网络带宽以及其他诱人的硬件特性: 目前, Mellanox NDR Infiniband 网卡能够支持高达 400Gb/s 的网络带宽; 另外, Infiniband 标准实现了链路层的容错机制, 这意味着通常意义上的丢包在 IB 网络上并不存在, 大大降低了用户设计通信机制的难度。加上近十年来新型硬件的价格逐渐走低, 学术界和工业界争相尝试这一新技术, 并已经有了相当多优秀的成果。尽管 RDMA 存在着其他实现方式, 例如基于以太网的 RoCE, 但本研究中的 RDMA 机制在硬件上基于超算集群中广泛使用的 Mellanox Infiniband 架构。

基于 RDMA 机制的网络编程, 目前主流的做法是使用 Verbs 操作原语。在 Infiniband 网络架构中, 两端以队列对 (Queue Pair, QP) 为基本模型进行通信, 一次简单的发送-接收可以描述为如下步骤:

- 1) 两端程序各自创建一个队列对, 并借助套接字等基础通信方式, 交换队列对的基本信息, 建立一对连接。
- 2) Infiniband 通信的基本单位是一个 (发送/接收/完成) 事务: 接收方通过 Verbs 调用将一个构造好的接收事务压入到接收队列 (Receive Queue) 中
- 3) 之后发送方将一个构造好的发送事务压入到发送队列 (Send Queue) 中。
- 4) 此时, 硬件将为两端处理这次事务, 将内存从本机发送事务指定的内存地址发送到远端接收事务指定的内存地址。
- 5) 硬件完成一次事务操作后, 通常将一个完成队列项 (CQE) 压入到完成队列 (Completion Queue, CQ) 中。
- 6) 用户进程可以在任何时刻通过弹出完成队列的头部来确认事务执行完成, 或者得到操作失败的错误代码。

以上流程实现了类似 TCP/UDP 的双边通信语义, 不过, 仍然有着本质的区别: 以上操作均在用户态完成; 且数据从一个进程直接发送到另一个进程, 无需任何系统调用。除此之外, RDMA 机制还支持使用读/写 (Read/Write) 等单边通信语

义。在执行这些单边操作时，远端应用不需要事先发送接收事务，也不会通过完成队列得知通信的发生。使用单边操作语义将会进一步降低远端 CPU 的介入和负担，从而支持更高强度的并发操作。

1.2.3 基于 RDMA 技术的内存系统

近年来，针对 RDMA 机制提供的高带宽、低延迟优势，研究人员在多个方向上尝试将其转变为实际应用上的性能提升。特别是在 RDMA 同时支持双边和单边语义的情况下，如何针对实际应用场景，设计合适的通信和协作机制，一直是这一领域研究的重点。从场景的角度来说，目前的主要研究方向是分布式内存系统和高并发的内存系统。

1.2.3.1 分布式内存系统

Infiniband 网络标准和 RDMA 技术最早应用于高性能计算（HPC）领域。俄亥俄州立大学的^[1]首次将 RDMA 技术应用于优化消息传递接口（MPI）的通信机制，这也是最早的、较为完善的一个 RDMA 协作机制实现。这一设计针对 RDMA 单边操作所产生的协作困难问题，预先建立了固定映射的发送-接收缓冲区对，以及设计了自描述的消息块，深刻影响了后续 RDMA 通信方案的设计。并且在 MPI 多年的演进中，其 RDMA 通信机制也不断有优化方案提出。

然而，从集群内存空间的角度来看，RDMA 技术的逐渐成熟，让研究者看到了颠覆性、普适性优化的可能性：优异的访问带宽和极低的访问延迟，已经极大地拉近了机器内存之间的“距离”，使得集群规模的共享内存逐渐成为可能：^[2]发现 IB 网络的通信延迟和 PCIe 总线上的通信延迟处于一个数量级，这说明网络已经不再是集群通信的性能瓶颈。以 RDMA 为核心，HPC 社区和分布式系统社区均提出了一些分布式共享内存的实现方案，从而以较低的性能代价连接起整个集群的内存空间：前者包括 GasNet，OpenSHMEM 等支持分区全局地址空间（PGAS）计算的中间件；后者以 FaRM，CoRM 等系统为代表。这些研究的共同焦点是分布式的事务机制——除了本地内存的读-写冲突之外，还需要解决远端机器读写和本地访存之间的数据冲突。此外，也有研究避开了细粒度的内存管理，如去中心化、可扩展的分布式页表系统 Infiniswap，就通过粗粒度的内存交换处理集群中存在的物理内存使用不均的问题。

1.2.3.2 高并发内存系统

在另一个方向上, 研究人员希望将 RDMA 低延迟、高并发的特性直接变为应用的性能——内存键值对(key-value)数据库是一个合适的领域。Redis, Memcached 等内存数据库通常被用作一种高速缓存, 其他机器对数据库的访问也主要以并发读取为主, 因此非常适合使用 RDMA 技术进行优化, 特别是使用单边读等通信方式优化单机并发能力。不过, 基于单边操作的读取机制难以应对高强度并发中的冲突问题, 需要借助于更好的哈希函数等其他优化手段才能获得较好的优化效果。因此近期的工作更是提出了双边语义和单边语义混合的通信机制, 从而在 CPU 负载、并发能力和编程难度上都获得较好的结果。总结来说, 在过去的一段时间, 针对应用场景不同, 研究人员已经提出了众多不同的 RDMA 通信范式。

1.3 论文主要研究内容

在本文中, 我们为分布式内存存储 Plasma, 提出了一种原生支持 RDMA 技术的通信机制, 并且在现代超算集群上验证了其在各个数据大小的传输上都获得了更优的性能。作为分布式计算框架的核心组件, Plasma 并没有对 RDMA 通信的支持, 因而在超算集群中存在性能提升的空间。

这一工作存在以下挑战:

- 1) 目前 RDMA 编程仍然是极为“小众”的技术, 如何能在有限的资料和现有研究帮助下实现高性能的网络通信机制。
- 2) 如何在尽可能不破坏项目整体结构的情况下, 为 Plasma 提供原生 RDMA 通信机制。这要求优化后的程序可以无缝运行在以太网和 Infiniband 两种网络架构上。
- 3) 针对 Ray 框架中可能出现的大小不一的数据, 如何实现该机制使得 Plasma 能够在尽可能多的大小范围内都能获得最优的网络性能。

下面总结了本工作的主要贡献:

- 1) 我们通过实验分析了原 Plasma 实现在超算集群上的存储和网络性能, 验证了其无法充分发挥 Infiniband 架构的网络性能, 从而证明了使用 RDMA 技术优化 Plasma 数据传输的可行性。
- 2) 我们分别针对常见的小型数据和分布式训练中常见的大型数据, 分别实现了基于双边和单边通信的传输机制。针对大数据, 单边通信将实现用户态的零拷贝特性, 从而降低了 CPU 数据拷贝而导致的负载和占用时间。
- 3) 我们基于消息传递接口 MPI 实现了分布式、可扩展的数据传输性能测试, 比

较了优化实现和原实现在各个数据大小上的性能。并且通过实验，我们确定了混合传输机制在选择方式时的最佳分界点，从而获得了最优的整体性能。

1.4 论文结构与章节安排

本文共分为五章，这些章节的内容安排如下：

第一章：绪论。简述了本文的研究背景和意义，简述了本研究的核心技术背景，并介绍了国内外相关工作和研究现状。

第二章：Plasma 分布式内存存储的架构和性能分析。这一章将简要分析 Plasma 的分布式架构，并且通过性能测试和常见内存存储 Redis 进行对比，分析了 Plasma 在传统网络结构上的性能瓶颈。

第三章：基于 RDMA 技术的混合通信机制实现。这一章将详细介绍基于 RDMA 技术的优化方案，并针对大小数据提出混合通信机制的实现。

第四章：实验和分析。这一章将在天河高性能集群上验证优化方案的性能优化，

第五章：总结和展望。这一章将总结本文的主要结果，并且进一步分析后续的工作方向。

2 本科生毕业论文写作与印制规范

该部分将学校规定中的毕业论文（设计）写作与印制规范复制了过来，并在其中展示相关例子表明格式的正确性。

2.1 毕业论文的撰写内容与要求

2.1.1 封面

纸质版封面由学校统一印发（电子版请参见文后示例）。封面内容包括论文题目、所在院系专业、学生姓名学号、指导教师（姓名及职称）等信息。论文题目应以简短、明确的词语恰当概括整个论文的核心内容，避免使用不常见的缩略词、缩写字。读者通过题目可大致了解毕业论文的内容、专业特点和学科范畴。论文题目一般不宜超过 25 个字，必要时可增加副标题。

2.1.2 扉页

扉页内容包括论文中英文题目、学生姓名、学号、院系、专业、指导教师（姓名及职称）等信息。格式详见文后示例。

2.1.3 学术诚信声明

内容及格式详见文后示例。

2.1.4 摘要和关键词

1) 中文摘要和关键词

摘要应概括论文的主要信息，应具有独立性和自含性，即不阅读论文的全文，就能获得必要的信息。摘要内容一般应包括研究目的、内容、方法、成果和结论，要突出论文的创造性成果或新见解，不要与绪论相混淆。语言力求精练、准确，以 300-500 字为宜。关键词是供检索用的主题词条，应体现论文特色，具有语义性，在论文中有明确的出处，并应尽量采用《汉语主题词表》或各专业主题词表提供的规范词。关键词与摘要应在同一页，在摘要的下方另起一行注明，一般列 3-5 个，按词条的外延层次排列（外延大的排在前面）。

2) 英文摘要和关键词

英文摘要及关键词内容应与中文摘要及关键词内容相同。中英文摘要及其关键词各置一页内。

2.1.5 目录

目录是论文的提纲，也是论文各章节组成部分的小标题。要求标题层次清晰，目录中的标题要与正文中的标题一致。

2.1.6 正文

正文是毕业论文的主体和核心部分，不同学科专业和不同的选题可以有不同的写作方式。正文一般包括以下几个方面：

1) 绪论

绪论应包括毕业论文选题的背景、目的和意义；对国内外研究现状和相关领域中已有的研究成果的简要评述；介绍本项研究工作研究设想、研究方法或实验设计、理论依据或实验基础；涉及范围和预期结果等。要求言简意赅，注意不要与摘要雷同或成为摘要的注解。

2) 主体

论文主体是毕业论文的主要部分，必须言之成理，论据可靠，严格遵循本学科国际通行的学术规范。在写作上要注意结构合理、层次分明、重点突出，章节标题、公式图表符号必须规范统一。论文主体的内容根据不同学科有不同的特点，一般应包括以下几个方面：

- (a) 毕业论文总体方案或选题的论证；
- (b) 毕业论文各部分的设计实现，包括实验数据的获取、数据可行性及有效性的处理与分析、各部分的设计计算等；
- (c) 对研究内容及成果的客观阐述，包括理论依据、创新见解、创造性成果及其改进与实际应用价值等；
- (d) 论文主体的所有数据必须真实可靠，凡引用他人观点、方案、资料、数据等，无论曾否发表，无论来源于纸质或电子版材料，均应详加注释。自然科学论文应推理正确、结论清晰；人文和社会学科的论文应把握论点正确、论证充分、论据可靠，恰当运用系统分析和比较研究的方法进行模型或方案设计，注重实证研究和案例分析，根据分析结果提出建议和改进措施等。

3) 结论

结论是毕业论文的总结，是整篇论文的归宿，应精炼、准确、完整。结论应着重阐述自己的创造性成果及其在本研究领域中的意义和作用，还可进一步提出需要讨论的问题和建议。

2.1.7 参考文献

参考文献是毕业论文不可缺少的组成部分，它反映毕业论文的取材来源、材料的广博和可靠程度，也是作者对他人知识成果的承认和尊重。凡有引用他人的著作、论文等，均应列于参考文献中。

2.1.8 相关的科研成果目录

本科期间发表的与毕业论文相关的论文或被鉴定的技术成果、发明专利等，应在成果目录中列出。此项不是必需项，空缺时可以省略。

2.1.9 附录

对于一些不宜放在正文中的重要支撑材料，可编入毕业论文的附录中，包括某些重要的原始数据、详细数学推导、程序全文及其说明、复杂的图表、设计图纸等一系列需要补充提供的说明材料。如果毕业论文中引用的实例、数据资料，实验结果等符号较多时，为了节约篇幅，便于读者查阅，可以编写一个符号说明，注明符号代表的意义。附录的篇幅不宜太多，一般不超过正文。此项不是必需项，空缺时可以省略。

2.1.10 致谢

致谢应以简短的文字对课题研究与论文撰写过程中曾直接给予帮助的人员(例如指导教师、答疑教师及其他人员)表达自己的谢意，这不仅是一种礼貌，也是对他人劳动的尊重，是治学者应当遵循的学术规范。内容限一页。

表 2.1 字体使用规范

论文题目	黑体二号居中
中文摘要标题	黑体三号居中
中文摘要内容	宋体小四号
中文关键词	宋体小四号（标题“关键词”加粗）
英文摘要标题	Times New Roman 加粗三号全部大写
英文摘要内容	Times New Roman 小四号
英文关键词	Times New Roman 小四号（标题“Keywords”加粗）
目录标题	黑体三号居中
目录内容	宋体小四号
正文各章标题	黑体三号居中
正文各节一级标题	黑体四号左对齐
正文各节二级及以下标题	宋体小四号加粗左对齐空两格
正文内容	宋体小四号
参考文献标题	黑体三号居中
参考文献内容	宋体五号
致谢、附录标题	黑体三号居中
致谢、附录内容	宋体小四号
页眉与页脚	宋体五号居中
图题、表题	宋体五号
脚注、尾注	宋体小五号

2.2 毕业论文的撰写格式要求

2.2.1 文字和字数

除外国语言文学类专业外，其他专业的毕业论文须采用简化汉语文字撰写。论文正文部分一般不少于 8000 字，各专业可根据需要确定具体的字数要求，并报教务部备案。

2.2.2 字体和字号

标题一般用黑体，内容一般用宋体，数字和英文字母一般用 Times New Roman，具体如表 2.1。

字体样例可见如下（以居中形式展现）：

黑体二号居中
黑体三号居中
黑体四号居中
 宋体四号居中
 宋体小四号居中

宋体五号居中

宋体小五号居中

Times New Roman : Three 三号居中

Times New Roman : Small Four 小四号居中

2.2.3 页面设置

纸张大小：A4。

页边距：上边距 25 mm，下边距 20 mm，左右边距均为 30 mm。

行距：1.5 倍行距，章和节标题段前段后各空 0.5 行。

2.2.4 页码

页面底端居中，从摘要开始至绪论之前以大写罗马数字（I，II，III，…）单独编连续码，绪论开始至论文结尾，以阿拉伯数字（1，2，3…）编连续码。

2.2.5 关键词

摘要正文下方另起一行顶格打印“关键词”款项，后加冒号，多个关键词以逗号分隔。

2.2.6 目录

目录应另起一页，包括论文中的各级标题，按照“一……”、“（一）……”或“1……”、“1.1……”格式编写。

2.2.7 各级标题

正文各部分的标题应简明扼要，不使用标点符号。论文内文各大部分的标题用“一、二……（或 1、2……）”，次级标题为“（一）、（二）……（或 1.1、2.1……）”，三级标题用“1、2……（或 1.1.1、2.1.1……）”，四级标题用“（1）、（2）……（或 1.1.1.1、2.1.1.1……）”，不再使用五级以下标题。两类标题不要混编。

2.2.8 名词术语

- 1) 科学技术名词术语尽量采用全国自然科学名词审定委员会公布的规范词或国家标准、部标准中规定的名称，尚未统一规定或叫法有争议的名词术语，

可采用惯用的名称。

- 2) 特定含义的名词术语或新名词、以及使用外文缩写代替某一名词术语时, 首次出现时应在括号内注明其含义, 如: 经济合作与发展组织 (Organisation for Economic Co-operation and Development, OECD)。
- 3) 外国人名一般采用英文原名, 可不译成中文, 英文人名按姓前名后的原则书写, 如: CRAY P, 不可将外国人姓名中的名部分漏写, 例如: 不能只写 CRAY, 应写成 CRAY P。一般很熟知的外国人名 (如牛顿、爱因斯坦、达尔文、马克思等) 可按通常标准译法写译名。

2.2.9 物理量名称、符号与计量单位

- 1) 论文中某一物理量的名称和符号应统一, 应采用国务院发布的《中华人民共和国法定计量单位》、国际公认或各行业领域惯用的计量单位。单位名称和符号的书写方式, 应采用国际通用符号。
- 2) 在不涉及具体数据表达时允许使用中文计量单位如“千克”。
- 3) 表达时刻应采用中文计量单位, 如“下午 3 点 10 分”, 不能写成“3h10min”, 在表格中可以用“3:10PM”表示。
- 4) 物理量符号、物理量常量、变量符号用斜体, 计量单位符号均用正体。

2.2.10 数字

- 1) 无特别约定情况下, 一般均采用阿拉伯数字表示。
- 2) 年份一律使用 4 位数字表示。
- 3) 统计符号的格式: 一般除 μ 、 α 、 β 、 λ 、 ε 以及 V 等符号外, 其余统计符号一律以斜体字呈现, 如 *ANCOVA*, *ANOVA*, *MANOVA*, *N*, *nl*, *M*, *SD*, *F*, *p*, *r* 等。

2.2.11 公式

- 1) 公式应另起一行写在稿纸中央。一行写不完的长公式, 最好在等号处转行, 如做不到这一点, 可在运算符号 (如“+”、“-”号) 处转行, 等号或运算符号应在转行后的行首。
- 2) 公式的编号用圆括号括起, 放在公式右边行末, 在公式和编号之间不加虚线。公式可按全文统编序号, 也可按章编独立序号, 如 (49)、(4.11)、(4-11) 等。采用哪一种序号应和图序、表序编法一致。不应出现某章里的公式编号

号，有的则不编序号。子公式可不编序号，需要引用时可加编 a、b、c……，重复引用的公式不得另编新序号。公式序号必须连续，不得重复或跳缺。

3) 文中引用某一公式时，可写成“由式（序号）”。

这是一个例子：

$$ax^2 + bx + c = 0 \quad (2.1)$$

如式 2.1 所示，为了求解该一元二次方程，我们可以推导得到该方程的求根公式。因此，由式 2.2 即可求解该方程的两个根。

$$x_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a} \quad (2.2)$$

$$x_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}$$

2.2.12 表格

- 1) 表格必须与论文叙述有直接联系，不得出现与论文叙述脱节的表格。表格中的内容在技术上不得与正文矛盾。
- 2) 每个表格都应有自己的标题和序号。标题应写在表格上方正中，不加标点，序号写在标题左方。
- 3) 全文的表格可以统一编序，也可以逐章单独编序。采用哪一种方式应和插图、公式的编序方式统一。表序必须连续，不得跳缺。
- 4) 表格允许下页接写，接写时标题省略，表头应重复书写，并在右上方写“续表 ××”。多项大表可以分割成块，多页书写，接口处必须注明“接下页”、“接上页”、“接第 × 页”字样。
- 5) 表格应放在离正文首次出现处最近的地方，不应超前和过分拖后。

例子可见表 2.2。

表 2.2 表格例子

这是表格第一行第一列	这是表格第一行第二列
这是表格第二行第一列	这是表格第二行第二列

2.2.13 图

- 1) 插图应与文字内容相符，技术内容正确。所有制图应符合国家标准和专业标准。对无规定符号的图形应采用该行业的常用画法。
- 2) 每幅插图应有标题和序号，全文的插图可以统一编序，也可以逐章单独编序，采取哪一种方式应和表格、公式的编序方式统一。图序必须连续，不重复，不跳缺。
- 3) 由若干分图组成的插图，分图用 a、b、c……标序。分图的图名以及图中各种代号的意义，以图注形式写在图题下方，先写分图名，另起行写代号的意义。
- 4) 图与图标题、图序号为一个整体，不得拆开排版为两页。当页空白不够排版该图整体时，可将其后文字部分提前，将图移至次页最前面。
- 5) 对坐标轴必须进行文字标示，有数字标注的坐标图必须注明坐标单位。

例子可见图 2.1。

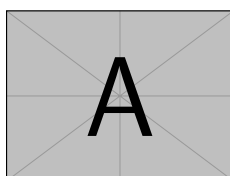


图 2.1 图片例子

2.2.14 注释

毕业论文（设计）中有个别名词或情况需要解释时，可加注说明。注释采用脚注或尾注，应根据注释的先后顺序编排序号。注释序号以“①、②”等数字形式标示在正文中被注释词条的右上角，脚注或尾注内容中的序号应与被注释词条序号保持一致。

脚注例子可见这里^①。

2.2.15 参考文献

参考文献的序号左顶格，并用数字加方括号表示，如“[1]”。每一条参考文献著录均以“.”结束。各类参考文献的具体编排格式请参照国家标准《信息与文献参考文献著录规则》（GB/T 7714-2015）。

参考文献例子可见这里^[1]。

^① 这是一个脚注

2.2.16 附录

论文附录依次用大写字母“附录 A、附录 B、附录 C……”表示，附录内的分级序号可采用“附 A1、附 A1.1、附 A1.1.1”等表示，图、表、公式均依此类推为“图 A1、表 A1、式 A1”等。

3 L^AT_EX 模板配置与使用

本部分内容将让你能够通过本 L^AT_EX 模板生成一份可用的 pdf，并为后面修改源码撰写毕设做准备。

首先，我们会展示最简单的方法：直接使用 overleaf 进行编写。然后，我们整理了不同环境下 L^AT_EX 环境的配置指南与不同写作工具的配置技巧，方便各位同学使用本 L^AT_EX 模板。最后，我们说明了如何开始编写自己的毕业论文（设计）。

3.1 使用 Overleaf 编写毕设

Overleaf^①是一个在线的 Latex 文档协作平台。我们不需要配置任何环境，便能够在上面直接使用本模板进行写作。操作步骤如下：

第一步，下载本项目压缩包（从<https://github.com/SYSU-SCC/sysu-thesis/releases>处下载即可），注意需要下载 zip 格式的压缩包。然后，我们在 Overleaf 上新建项目，并上传该压缩包，可参考图 3.1。

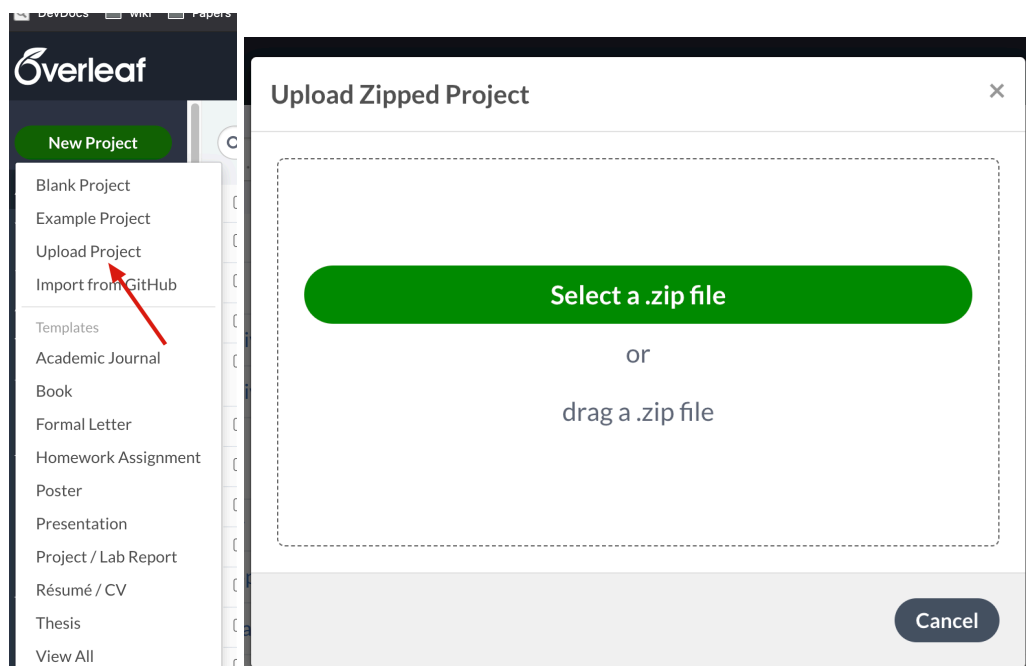


图 3.1 在 Overleaf 上创建并上传压缩包。

第二步，在 Overleaf 的菜单中调整编译工具为 xelatex，可参考图 3.2。

第三步，点击编译，得到本 pdf，可以开始修改 pdf 了！最终可见图 3.3。

^① 网址可见<https://www.overleaf.com/>

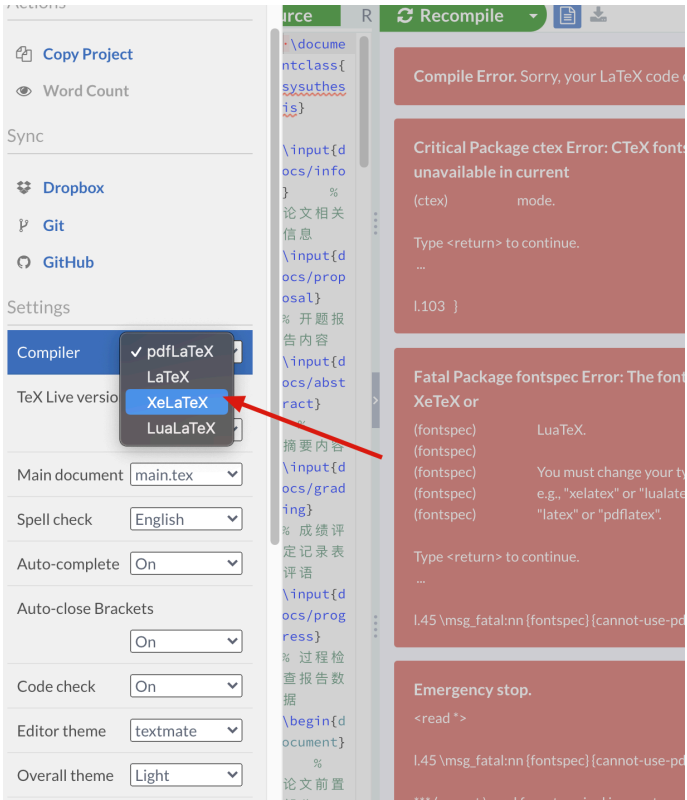


图 3.2 在 Overleaf 上调整编译工具

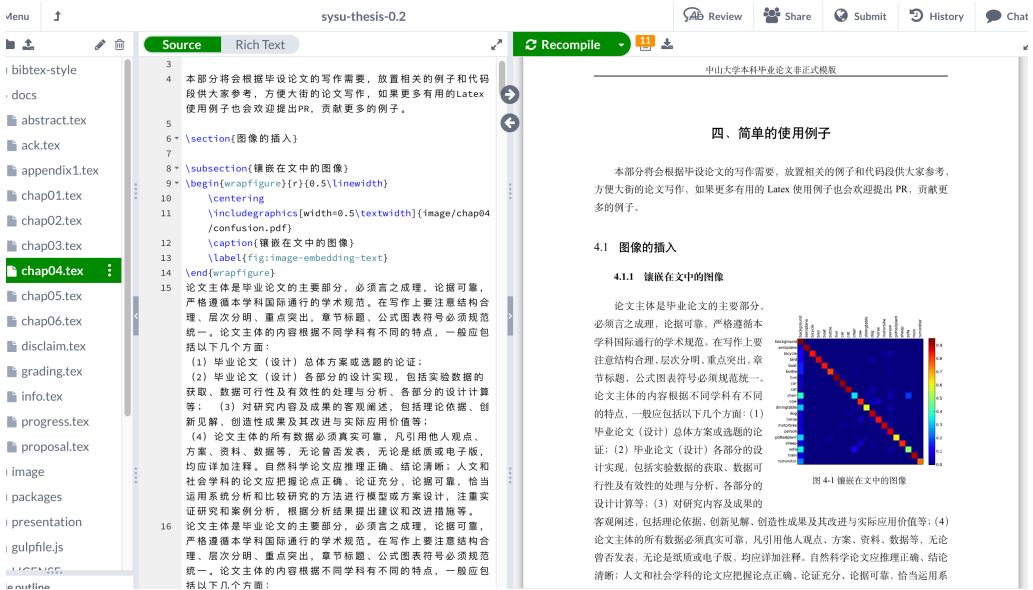


图 3.3 Overleaf 使用例子

3.2 编译环境配置

编译环境配置相对来说比较简单，下载 Tex Live2020 并如同一般的程序一样安装即可。

3.2.1 编译环境配置：Window 篇

在<https://mirrors.tuna.tsinghua.edu.cn/CTAN/systems/texlive/Images> /上下载 Tex Live2020 并参考教程^①安装即可。

3.2.2 编译环境配置：Linux 篇

在<https://mirrors.tuna.tsinghua.edu.cn/CTAN/systems/texlive/Images> /上下载 Tex Live2020 并参考教程^②安装即可。

3.2.3 编译环境配置：MacOS 篇

在 MacOS 上配置 Latex 的环境，这里我们使用的是 MacTex。

- 1) <https://www.tug.org/mactex/>下载 MacTex 安装。
- 2) 安装步骤：不详细展开，按照图形界面点击即可，傻瓜式安装。

TIPS：MacTex 文件比较大，有 2G 多，介意的话可以选择 MacTex_Basic 包，只有 100M 以内，但是如果安装 MacTex_Basic，后期可能会遇到各种缺包的问题。

安装完成之后，可以简单测试一下安装是否成功。如可以查看 Texshop 应用是否安装好，或者在命令行测试一下 `xelatex` 命令是否可用。

3.3 写作环境配置

不同的写作工具对应不同的写作环境。这里我们给出几个工具的配置例子以供参考。

3.3.1 模板编译流程

由于 L^AT_EX 的限制，本模板需要经过四次编译才能生成完整的论文：

- 1) 先使用 `xelatex` 编译一次
- 2) 再使用 `bibtex` 编译一次

^① 可以参考<https://zhuanlan.zhihu.com/p/58811994>

^② 可以参考<https://zhuanlan.zhihu.com/p/55894177>

3) 然后使用 xelatex 编译两次

本编译流程已经写在 Makefile 中，修改模板源码后只需要执行 `make pdf` 即可按照该流程进行编译并生成最终的 pdf。

3.3.2 写作环境配置：Visual Studio Code

Visual Studio Code 是微软公司推出的轻量代码编辑器，我们可以做一些简单的配置，便可以用该编辑器修改我们的 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 模板，并实现一键编译。

- 1) 安装 Visual Studio Code。
- 2) 安装 LaTeX Workshop 插件。

本项目的 `.vscode/setting.json` 下已经包含了与前面所述编译流程相同的配置。正常配置下，每次修改模板源码后按下保存 (`Ctrl+S`)，就能够自动进行编译产生 pdf。效果图如图 3.4 所示。

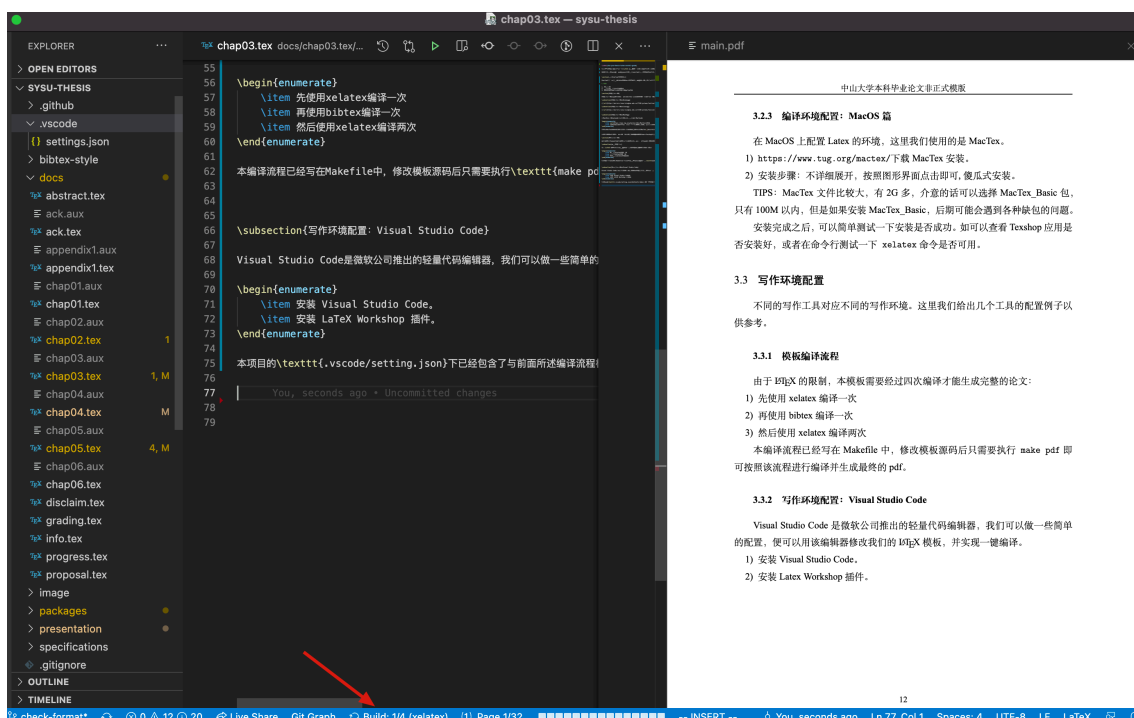


图 3.4 vscode 配置好后的样例

3.4 如何开始写毕业论文（设计）

首先将所有个人信息,包括学号、姓名、专业、论文题目等,在 `./docs/info.tex` 中逐项进行更新。

然后我们再编辑 `./docs/abstract.tex` 补充论文摘要。

到了论文主体部分,我们可以自行编辑`./docs/chap01.tex`,`./docs/chap02.tex`等文件进行编辑。如果章数不够,可以自行修改`main.tex`增加新的章节。

当论文主体编写完成后,我们再编辑`./docs/ack.tex`作为论文致谢。

4 简单的使用例子

本部分将会根据毕设论文的写作需要，放置相关的例子和代码段供大家参考，方便大家的论文写作，如果更多有用的 Latex 使用例子也会欢迎提出 PR，贡献更多的例子。

4.1 图像的插入

4.1.1 镶嵌在文中的图像

论文主体是毕业论文的主要部分，必须言之成理，论据可靠，严格遵循本学科国际通行的学术规范。在写作上要注意结构合理、层次分明、重点突出，章节标题、公式图表符号必须规范统一。论文主体的内容根据不同学科有不同的特点，一般应包括以下几个方面：(1) 毕业论文（设计）总体方案或选题的论证；(2) 毕业论文（设计）各部分的设计实现，包括实验数据的获取、数据可行性及有效性的处理与分析、各部分的设计计算等；(3) 对研究内容及成果的

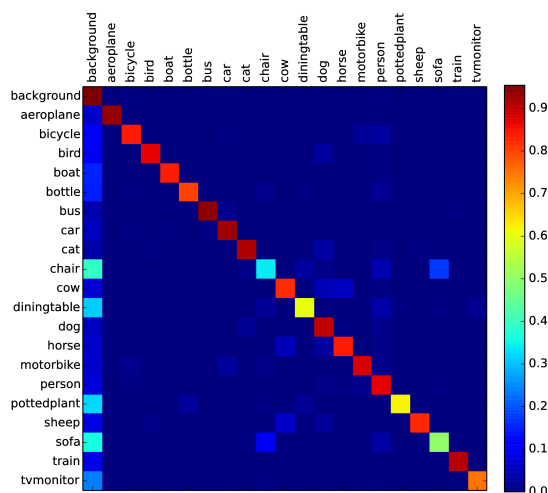


图 4.1 镶嵌在文中的图像

客观阐述，包括理论依据、创新见解、创造性成果及其改进与实际应用价值等；(4) 论文主体的所有数据必须真实可靠，凡引用他人观点、方案、资料、数据等，无论曾否发表，无论是纸质或电子版，均应详加注释。自然科学论文应推理正确、结论清晰；人文和社会学科的论文应把握论点正确、论证充分、论据可靠，恰当运用系统分析和比较研究的方法进行模型或方案设计，注重实证研究和案例分析，根据分析结果提出建议和改进措施等。论文主体是毕业论文的主要部分，必须言之成理，论据可靠，严格遵循本学科国际通行的学术规范。在写作上要注意结构合理、层次分明、重点突出，章节标题、公式图表符号必须规范统一。论文主体的内容根据不同学科有不同的特点，一般应包括以下几个方面：(1) 毕业论文（设计）总体方案或选题的论证；(2) 毕业论文（设计）各部分的设计实现，包括实验数据的获

取、数据可行性及有效性的处理与分析、各部分的设计计算等；(3) 对研究内容及成果的客观阐述，包括理论依据、创新见解、创造性成果及其改进与实际应用价值等；(4) 论文主体的所有数据必须真实可靠，凡引用他人观点、方案、资料、数据等，无论曾否发表，无论是纸质或电子版，均应详加注释。自然科学论文应推理正确、结论清晰；人文和社会学科的论文应把握论点正确、论证充分、论据可靠，恰当运用系统分析和比较研究的方法进行模型或方案设计，注重实证研究和案例分析，根据分析结果提出建议和改进措施等。

4.1.2 单张图像的插入

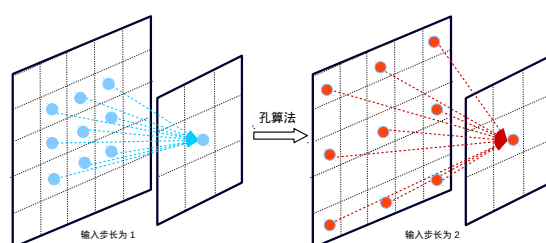


图 4.2 单张图像

4.1.3 多张图像的并排插入

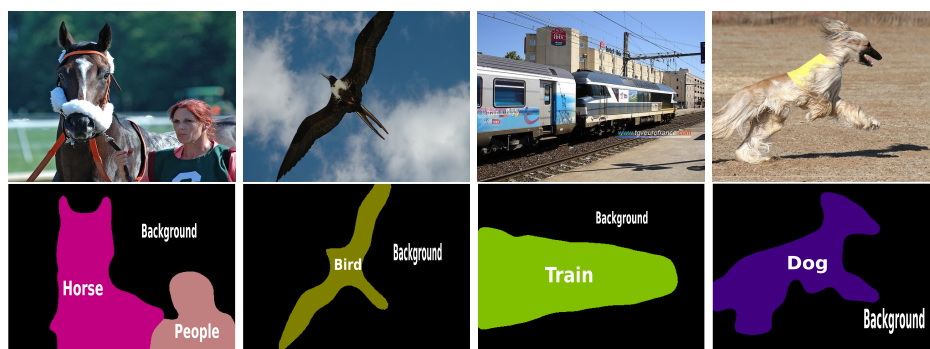


图 4.3 并排的多张图像

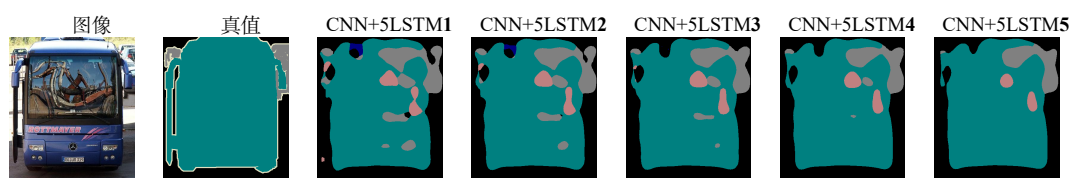


图 4.4 并排的多张图像加各自的注解

4.1.4 两列图像的插入

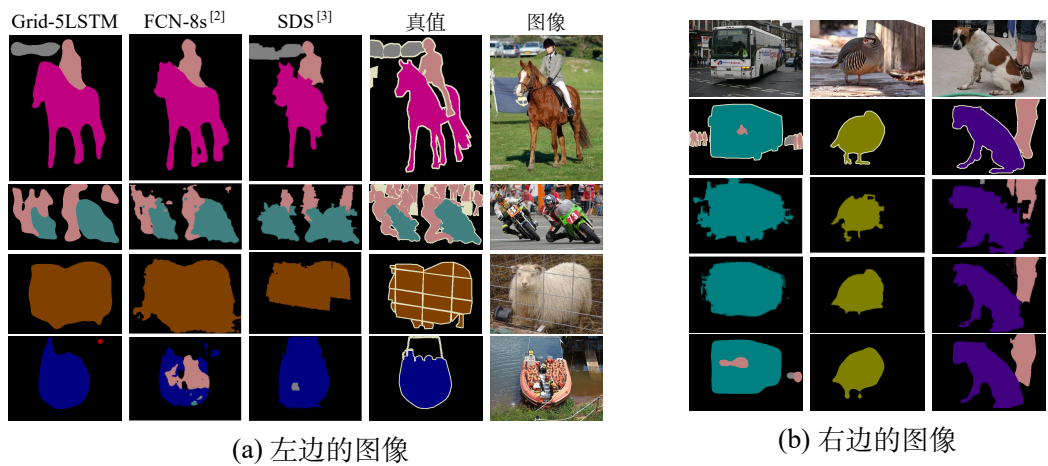


图 4.5 复杂的两列对象的插入

4.2 表格的插入

表 4.1 典型的实验对比表格

Method	Pixel Acc.	Mean Acc.	Mean Iu.
Liu 等人 ^[4]	76.7	-	-
Tighe 等人 ^[5]	78.6	39.2	-
FCN-16s ^[2]	85.2	51.7	39.5
Deeplab-LargeFOV ^[6]	85.6	51.2	39.7
Grid-LSTM5	86.2	51.0	41.2

表 4.2 复杂一些的表格

Method	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	shep	sofa	train	tv	mIoU.
CNN	72.6	29.6	70.2	53.1	65.1	81.0	74.3	79.8	25.0	64.8	47.8	69.5	66.2	65.2	74.2	42.1	69.6	38.8	74.4	58.6	62.5
CNN+1LSTM	71.5	30.6	70.5	53.8	64.9	82.4	77.1	79.5	25.1	65.8	47.8	71.5	64.6	67.0	74.0	43.9	69.6	38.6	74.9	59.4	63.0
CNN+2LSTM	76.1	32.6	72.1	57.0	65.3	83.6	75.4	81.7	24.7	69.3	47.5	72.3	68.9	69.5	74.7	41.5	69.8	38.3	77.8	62.1	64.3
CNN+3LSTM	77.7	32.3	72.6	60.0	68.3	85.5	78.5	82.3	25.3	71.1	49.7	71.5	69.7	70.8	75.9	47.9	71.2	38.9	80.2	61.7	65.8
CNN+4LSTM	79.1	33.7	73.6	62.0	70.4	85.5	80.9	83.7	24.1	70.7	45.7	73.7	69.6	72.1	75.6	47.2	76.0	37.3	80.5	62.2	66.4
CNN+5LSTM	79.9	33.6	73.6	61.7	68.0	88.5	80.9	84.0	23.6	71.3	49.7	73.1	71.3	72.9	76.4	48.9	75.1	38.1	84.5	63.8	67.2
CNN+5LSTM†	84.8	36.4	82.0	69.4	73.0	87.2	81.8	86.1	34.5	82.4	53.1	81.5	77.4	79.0	81.3	54.8	81.1	47.0	84.3	67.3	72.3

4.3 公式

没有编号的公式

$$\mathbf{z}^{(l)} = \mathbf{W}^{(l)} \mathbf{a}^{(l-1)} + \mathbf{b}^{(l)}$$

$$\mathbf{a}^{(l)} = f(\mathbf{z}^{(l)})$$

公式中含有中文

$$\text{像素准确率} = \sum_{i=1}^{n_{cl}} n_{ii} / \sum_{i=1}^{n_{cl}} t_i$$

$$\text{平均像素准确率} = \frac{1}{n_{cl}} \sum_{i=1}^{n_{cl}} (n_{ii} / t_i) \quad (4.1)$$

$$\text{Mean IU} = \frac{1}{n_{cl}} \sum_{i=1}^{n_{cl}} \frac{n_{ii}}{t_i + \sum_j^{n_{cl}} n_{ji} - n_{ii}}$$

公式中含有矩阵

$$\mathbf{H} = \begin{bmatrix} I * \mathbf{x}_i \\ \mathbf{h} \end{bmatrix} \quad (4.2)$$

每行后面都有编号的公式

$$\frac{\partial}{\partial W_{ij}^{(l)}} J(\mathbf{W}, \mathbf{b}; \mathbf{x}, y) = \frac{\partial J(\mathbf{W}, \mathbf{b}; \mathbf{x}, y)}{\partial z_i^{(l+1)}} \cdot \frac{\partial z_i^{(l+1)}}{\partial W_{ij}^{(l)}} = \delta_i^{(l+1)} a_j^{(l)} \quad (4.3)$$

$$\frac{\partial}{\partial b_i^{(l)}} J(\mathbf{W}, \mathbf{b}; \mathbf{x}, y) = \frac{\partial J(\mathbf{W}, \mathbf{b}; \mathbf{x}, y)}{\partial z_i^{(l+1)}} \cdot \frac{\partial z_i^{(l+1)}}{\partial b_i^{(l)}} = \delta_i^{(l+1)} \quad (4.4)$$

4.4 算法流程图

算法 4.1: 梯度下降算法

输入: m 个训练样本

- 1 对于 $l = 1$ 转到 n_l 进行 初始化: $\Delta \mathbf{W}^{(l)} = 0$, $\Delta \mathbf{b}^{(l)} = 0$;
- 2 对于每个 训练样本 进行
- 3 对于 $l = 1$ 转到 $n_l - 1$ 进行 前向传播:
 - $\mathbf{z}^{(l+1)} = \mathbf{W}^l \mathbf{a}^l + \mathbf{b}^l, \mathbf{a}^{(l+1)} = f(\mathbf{z}^{(l+1)})$;
- 4 输出误差计算: $\delta^{(n_l)} = \frac{\partial}{\partial \mathbf{z}^{(n_l)}} J(\mathbf{W}, \mathbf{b}; \mathbf{x}, y)$;
- 5 对于 $l = n_l - 1$ 转到 1 进行 后向传播: $\delta^{(l)} = ((\mathbf{W}^{(l)})^T \delta^{(l+1)}) f'(\mathbf{z}^{(l)})$;
- 6 对于所有 层 l 进行
- 7 计算梯度: $\nabla_{\mathbf{W}^{(l)}} J(\mathbf{W}, \mathbf{b}; \mathbf{x}, y) = \delta^{(l+1)} (\mathbf{a}^{(l)})^T$
- 8 $\nabla_{\mathbf{b}^{(l)}} J(\mathbf{W}, \mathbf{b}; \mathbf{x}, y) = \delta^{(l+1)}$;
- 9 累加梯度: $\Delta \mathbf{W}^{(l)} \leftarrow \Delta \mathbf{W}^{(l)} + \nabla_{\mathbf{W}^{(l)}} J(\mathbf{W}, \mathbf{b}; \mathbf{x}, y)$;
- 10 $\Delta \mathbf{b}^{(l)} \leftarrow \Delta \mathbf{b}^{(l)} + \nabla_{\mathbf{b}^{(l)}} J(\mathbf{W}, \mathbf{b}; \mathbf{x}, y)$;
- 11 对于所有 层 l 进行
- 12 更新权重: $\mathbf{W}^{(l)} \leftarrow \mathbf{W}^{(l)} - \alpha \left[\frac{1}{m} \Delta \mathbf{W}^{(l)} \right]$
- 13 $\mathbf{b}^{(l)} \leftarrow \mathbf{b}^{(l)} - \alpha \left[\frac{1}{m} \Delta \mathbf{b}^{(l)} \right]$

4.5 例子、定理与证明

例 4.1 这是一个例子, 用以验证特殊环境的字体成功更改为楷体。

定理 4.1 (定理例子) 这是一个定理。

推论 4.1 (推论例子) 这是一个推论。

引理 4.1 (引理例子) 这是一个引理。

这里我们先给出定理 4.2

定理 4.2 (中山大学毕业论文模板定理) 中山大学 L^AT_EX 毕业论文模板^[1] 可以用于写各种证明。

下面我们对定理 4.2 进行证明:

证明 下面我们开始证明：

由本定理的证明可见，我可以引用定理 4.1和引理4.1以及推论4.1来证明我这个 L^AT_EX 可以用来写各种证明。

定理 4.2得证。

□

4.6 代码

本模版支持在论文中插入代码片段，或直接从源码文件进行插入。例如，在论文中插入代码片段的效果为：

```
def func():
    print("hello world")
    with open('./output.txt', 'w') as f:
        L = f.readlines()

    if __name__ == "__main__":
        # this is a comment line
    func()
```

也可在行内插入代码片段，例如：Python 中重载加法运算符的函数为`__add__`，类的标识符为`class`。此外，还可直接插入代码文件，例如插入`./code/demo.cpp`的效果为：

```
#include <iostream>
int main()
{
    ::std::cout << "hello world" << ::std::endl;
    return 0;
}
```

4.7 其他的一些用法

4.7.1 子章节编号

4.7.1.1 更小的章节

更小的章节编号也是支持的。

可以如此引用章节：

- 章 4
- 节 4.7

- 小节 4.7.1
- 小小节 4.7.1.1

4.7.2 列表的使用

这是一个无序列表

- 引用文献^[2]
- 引用文献作者Long et al.
- 引用文献年份2015
- 字体变红，粗体，斜体，下划线。

这是一个有序列表

- 1) 索引前面的节 4.3、图像4.5、表格4.1
- 2) 加脚注^①

^① 测试一下脚注和 URL <http://cs231n.github.io/transfer-learning/>

5 其他注意事项

5.1 关于生僻字

测试生僻字

[illegible]

6 实验与结果

参考文献

- [1] CHEN S, HUANG J, YONGFENG W. 中山大学毕业论文 Latex 模板 [EB/OL]. 2020. <https://github.com/SYSU-SCC/sysu-thesis>.
- [2] LONG J, SHELHAMER E, DARRELL T. Fully convolutional networks for semantic segmentation[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. [S.l.: s.n.], 2015: 3431-3440.
- [3] HARIHARAN B, ARBELÁEZ P, GIRSHICK R, et al. Simultaneous detection and segmentation[M]//Computer vision—ECCV 2014. [S.l.]: Springer, 2014: 297-312.
- [4] LIU C, YUEN J, TORRALBA A. Sift flow: Dense correspondence across scenes and its applications[J]. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 2011, 33(5): 978-994.
- [5] TIGHE J, LAZEBNIK S. Finding things: Image parsing with regions and per-exemplar detectors[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. [S.l.: s.n.], 2013: 3001-3008.
- [6] CHEN L C, PAPANDREOU G, KOKKINOS I, et al. Semantic image segmentation with deep convolutional nets and fully connected crfs[C/OL]//ICLR. 2015. <http://arxiv.org/abs/1412.7062>.

附录 A 补充更多细节

A.1 补充图

A.1.1 补充图

这是附录内容，应该用宋体小四号字体。

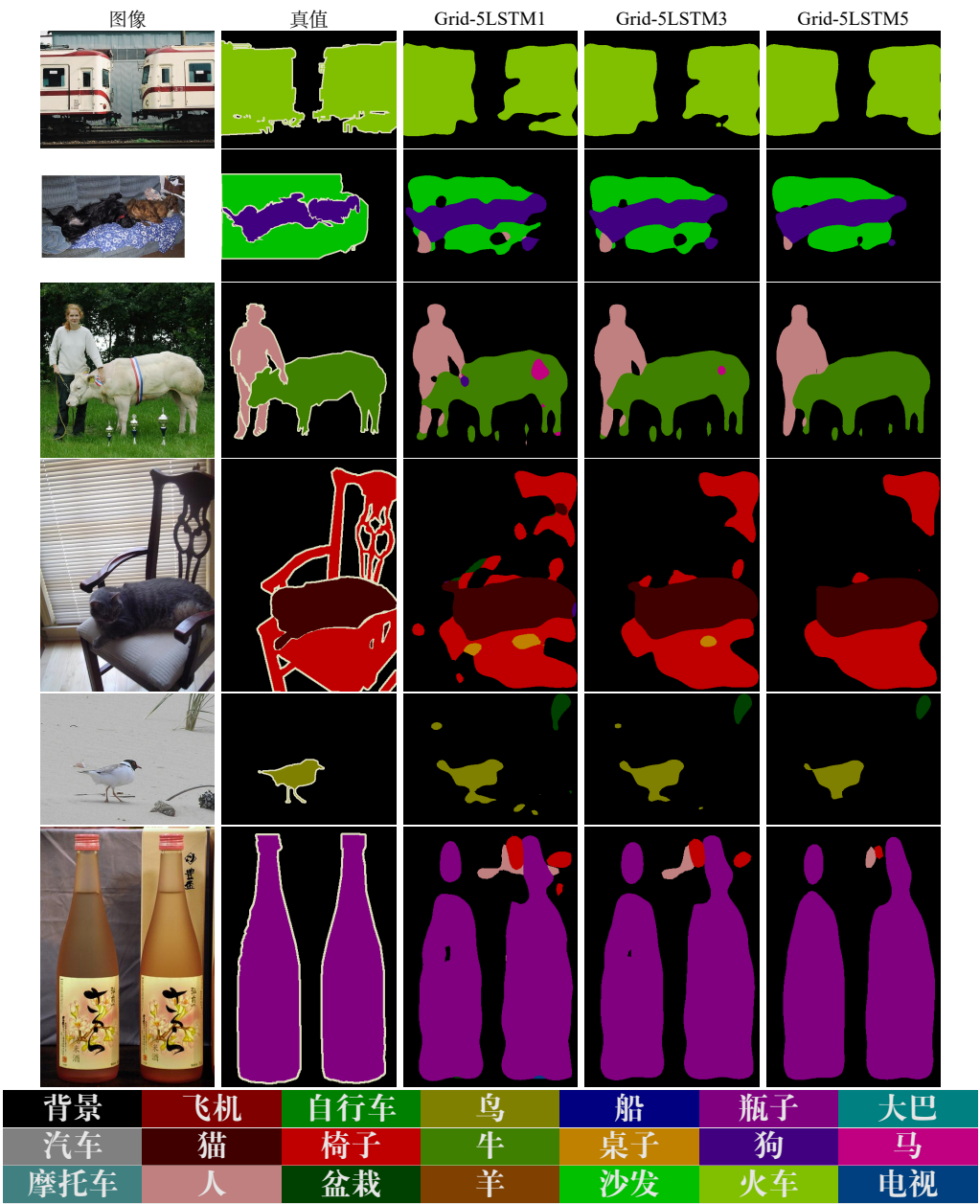


图 A.1 一个配有彩色表格的插图

致谢

四年时间转眼即逝，青涩而美好的本科生活快告一段落了。回首这段时间，我不仅学习到了很多知识和技能，而且提高了分析和解决问题的能力与养成了一定的科学素养。虽然走过了一些弯路，但更加坚定我后来选择学术研究的道路，实在是获益良多。这一切与老师的教诲和同学们的帮助是分不开的，在此对他们表达诚挚的谢意。

首先要感谢的是我的指导老师王大明教授。我作为一名本科生，缺少学术研究经验，不能很好地弄清所研究问题的重点、难点和热点，也很难分析自己的工作所能够达到的层次。王老师对整个研究领域有很好的理解，以其渊博的知识和敏锐的洞察力给了我非常有帮助的方向性指导。他严谨的治学态度与辛勤的工作方式也是我学习的榜样，在此向王老师致以崇高的敬意和衷心的感谢。

最后我要感谢我的家人，正是他们的无私的奉献和支持，我才有了不断拼搏的信心和勇气，才能取得现在的成果。

王小明

2022 年 3 月 27 日