毕业设计论文撰写指南

学 生： 李琦

学 号： 14121161

指导老师： 朱晓强

专 业： 通信工程

完成年月： 2018年06月

（此页封面用统一模板，到时会通知学生）

摘要

本指南，是为通信与信息工程学院本科毕业生而编写，内容包括毕业设计论文撰写基本要求和打印基本格式。本文既可用作本科毕业设计论文撰写和打印基本格式要求的说明，又可直接作为Word文档模板使用。作为Word文档模板使用时，不可修改页面设置和格式中的样式。

**关键词：**毕业设计论文，撰写基本要求，打印基本格式

# Abstract

These instructions give you guidelines for preparing papers for B.E. degree*.* Use this document as a template if you are using Microsoft *Word* 6.0 or later. Otherwise, use this document as an instruction set.

Use this template as an aid in writing, but remember that you are entirely responsible for meeting the requirements outlined in the Direction For Preparing Bachelor’s Degree Theses, a current edition of which is available at the School. Use this template, not as a substitute but, in conjunction with the Manual.

**Key words:** About four key words or phrases, separated by commas.

目录

[摘要 2](#_Toc45190318)

[Abstract 3](#_Toc45190319)

[第一章 绪论 4](#_Toc45190320)

[第二章 虚幻引起过 5](#_Toc45190321)

[2.1 引擎简介 4](#_Toc45190322)

[2.2 虚幻引擎与SteamVR](#_Toc45190323) 4

[2.3 蓝图可视化与C++结合 5](#_Toc45190324)

[2.4 FBX静态网格物体通道 5](#_Toc45190325)

2.5虚幻引擎第三方库的引用

[2.4.1 图 5](#_Toc45190326)

[2.4.2 表 6](#_Toc45190327)

[2.5 参考文献 6](#_Toc45190328)

[第三章 打印基本格式 7](#_Toc45190329)

[3.1 论文题目 7](#_Toc45190330)

[3.2 摘要 7](#_Toc45190331)

[3.3 论文正文 7](#_Toc45190332)

[3.4 图表 8](#_Toc45190333)

[3.5 参考文献 8](#_Toc45190334)

[3.6 页面设置 9](#_Toc45190335)

[3.6.1 纸张和页边距 9](#_Toc45190336)

[3.6.2 装订线和页码 9](#_Toc45190337)

[3.7 论文撰写与装订顺序 9](#_Toc45190338)

[致谢 10](#_Toc45190339)

[参考文献 11](#_Toc45190340)

# 第一章 绪论

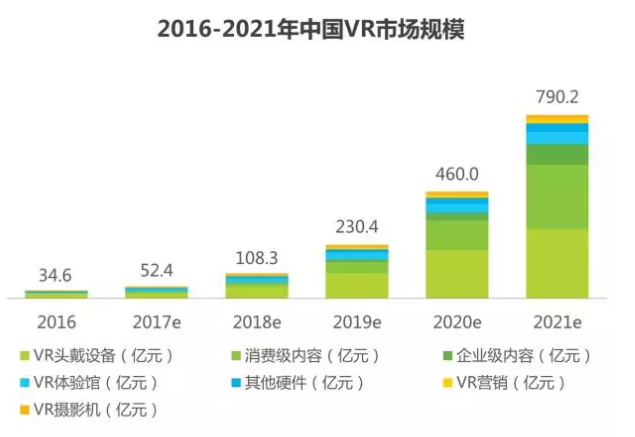
根据《上海大学毕业论文（设计）工作条例》[1]规定，我校学生毕业论文一律必须按照该条例附件所规定的基本格式用电脑打印成文。为方便我院学生掌握毕业设计论文撰写的基本格式要求，特制作本文档。本文既可用作毕业设计论文撰写基本格式要求的说明，又可直接作为Word文档模板使用。

## 课题研究背景

### 中国VR市场概况

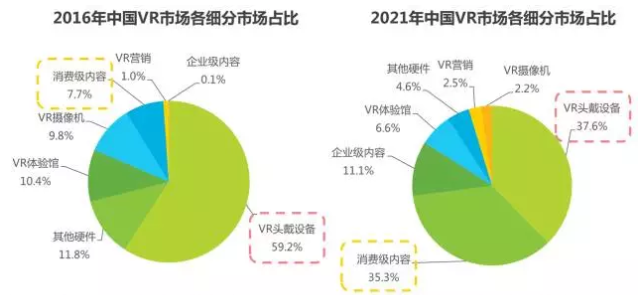
市场规模增速加快，中国将成为全球最大的VR市场

2016年中国VR市场规模为34.6亿元，尽管当前这一规模还处在比较小的水平，但是市场规模增长速度非常快，预计2018年中国VR市场将突破百亿元大关。未来五年中，VR市场的年复合增长率将超过80%。预计到2021年，中国会成为全球最大的VR市场，行业整体规模将达到790.2亿元。



## 内容市场将超过VR头戴设备，成为占比第一的细分市场

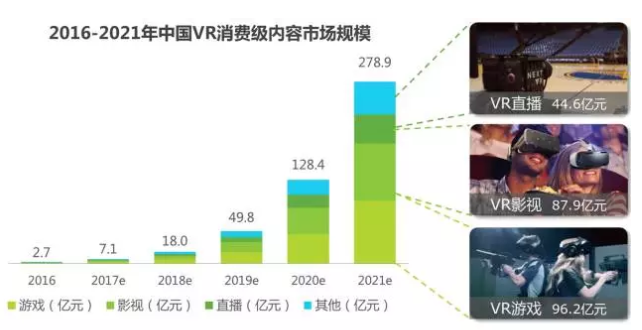
2016年，中国VR市场中规模最大的细分市场是VR头戴设备，以20.5亿元的规模占据整体份额的59.2%。而目前占比最小的VR内容市场（包括消费级内容、企业级内容和VR营销），会在未来五年快速增长，预计2021年的市场规模为386.4亿元，年复合增长率为163.4%。这也意味着VR内容市场将达到整体市场规模的近50%，成为VR市场中最大的细分市场。2015年开始兴起的VR体验馆，预计其市场规模将在2021年达到52.5亿元，排在头戴设备市场之后。



## VR内容市场规模预测和分析

消费级：内容制作商逐渐开始盈利，未来VR游戏市场最看好

消费级的VR内容包括游戏、影视、直播、其他四大类型。2019年将是消费级内容市场的转折点，在这一阶段，行业内主要的内容制作商会开始实现盈利。2021年，消费级内容市场的规模将达到278.9亿元，其中VR游戏的占比最近35%，市场规模为96.2亿元。紧随其后的影视内容在消费级市场中的比例也超过30%，到2021年VR影视的市场规模将达到87.9亿元。



企业级：市场规模年增长率超300%，教育培训需求推动行业发展

虽然目前企业级内容市场在VR市场规模中的比例不足1%，但其市场增长速度将始终保持在高位，年复合增长率为355.0%，预计到2021年，VR企业级内容的市场规模将达到87.7亿元。教育和培训的巨大需求是企业级内容市场快速增长的主要原因，国家政策层面高度重视教育信息化，政策对VR教育的利好势必会进一步推动市场规模的扩大。



## VR建模现状

Oculus Medium

现如今市场上主要的VR建模/绘画工具有两个，一个是Oculus Medium另一个是Tilt BrushVR绘画工具

Oculus Medium是一款专为Touch设计的沉浸式VR体验，用户可以在VR环境之中雕刻，建模，绘制和创建有形对象。Oculus最近添加了Move Tool，它允许用户抓拿，移动和重新设置雕刻作品的一部分。

Medium的设计灵感来源于让设计家和艺术家们从已有的2D平面中脱离出来，利用数字工具，借助计算机图形和色彩，并且可以以一种全新的方式观看和交互，

在VR模式下设计三维模型的最大优势是：我们可以免费获得2D平面中没有的立体感，比如大量的深度和声音先所以及立体渲染和头部追踪。也没有必要训练任何人应该如何移动摄像头，而这对显示器来说可能是一个复杂的任务。在VR中，摄像头就是我们的头，我们和三维模型处于同一个空间，关系自然，大小和谐。这些元素会让我们和模型产生自然的连接。

在VR中进行雕塑，操纵放置虚拟粘土，就像是在现实空间中一样，只是缺少了一些重力和其他物理限制。并且

## 虚幻引擎与SteamVR

在虚幻引擎4.8版本之后，虚幻引擎已经支持SteamVR，可以实现真正的即插即用，并且充分利用了VR，房间，输入，灯光以及SteamVR的最新的激光跟踪解决方案。SteamVR通过蓝图可视化脚本和本机代码完全继承到虚幻引擎中，因此可以在不需要依赖程序员支持的情况下构建项目。

安装好SteamVR后，我们将在电脑屏幕右下方看到下图



这表明SteamVR已经正常启用，并且手柄和头盔都被成功检测。

要在虚幻引擎中使用SteamVR，还需要启用SteamVR插件，该功能可以在虚幻引擎的Plugin中找到。做完上述步骤，我们就可以在游戏播放按钮下拉菜单中启用VRperview了，这意味着我们可以在VR模式下在游戏场景中移动。

第二步就是通过手柄在引擎中交互了。首先我们要设置一个新的VR蓝图角色。这将代替我们游戏中的默认角色，之后再蓝图中设置一个摄像机，在摄像机的右边属性栏可以将之连接到我们的头盔，接下来我们在人物组建中添加Motioncontroller（即我们的手柄），然后在每个手柄下添加一个模型，这样手柄的移动转向都可以在游戏中展示出来了。

完成上述步骤，VR人物就基本设置好了，如果要启用VR手柄交互功能，我们只需在VR人物的蓝图事件面板找到运动控制器（这是虚幻引擎中SteamVR手柄的名字）对应的输入就可以了。

在本次项目中主要使用了手柄的Trigger输入。

## 蓝图与C++相互调用

虚幻引擎为程序员提供了两套工具集，可以一起使用来加速开发的工作流程。新的游戏类，世界场景中的蓝图类中的组件可以为C++类，一些涉及到逻辑运算，数学处理的比较庞大的模块可以用C++语言来写，并且可以用Visual Studio编译后在虚幻编辑器中进行更新。而蓝图可视化脚本如概述中介绍的：可以在功能模块之间进行连线以及变量和属性设置在编辑器中记性创建。同时C++类可以作为蓝图类的基类来创建，这样的话，程序员就可以设置基础的游戏类，然后用可视化程度较高的蓝图来处理关卡设计和一些简单的逻辑处理，以及一些直接与游戏场景相关联的功能。  
C++和蓝图是相互配合的，不论创建项目时选择的是C++项目还是蓝图项目，都可以同时使用蓝图和C++分别处理不同的模块。

虚幻蓝图和C++相互调用一共有以下三种方法：

1. 蓝图调用C++变量函数或函数

首先在C++类的Public作用域中声明函数或变量，然后在声明之前添加虚幻引擎特有的属性声明。一般的，变量属性声明使用UPROPERTY，而函数使用UFUNCTION。在括号中可以添加枚举值表明我们想设置的属性。本项目主要使用了：BluePrintcallable（蓝图可调用），Category（蓝图分类目录），EditAnywhere(可任意编辑)。再次编译C++代码后即可在调用该C++的蓝图类中找到在C++中声明的变量或函数。

1. C++调用蓝图函数

首先创建一个C++游戏类，在该类的声明中添加UCLASS属性：BluePrintable（可被蓝图化），然后就可以在编辑器的内容浏览器中找到该C++类，基于该C++类创建一个蓝图类。

在该蓝图中创建一个函数（在Function下拉栏中），接着在该父类CPP文件中使用函数CallfunctionByNameWithArguments()，我们通过字符串将蓝图中刚刚创建的函数传递给C++，并且以空格形式隔开，在后面紧跟参数，即可完成在C++中对指定蓝图函数的调用。

1. C++中调用蓝图事件

在虚幻蓝图类中，有一个节点叫做自定义事件，它的作用和函数类似，都是执行一系列自定义行为。但是事件和函数不同的地方在于没有输入参数，也没有输出值。自定义事件更类似于一个火车头，作为某个功能模块的驱动事件。

首先在我们C++基类中做一个广播,声明变量类型，本项目使用了：DECLARE\_DYNAMIC\_MUTICAST\_DELEGATE,用定义的变量类型定义一个广播变量，即可在C++中调用该广播变量的成员函数（本项目中调用了广播函数），这个行为指定了该事件在何时被执行，接着可以在蓝图类的Event Dispatchers分类中找到该事件，我们可以在蓝图中填充该事件的具体内容，这样就实现了蓝图和C++的相互调用。

一般来说，蓝图效率比C++要低一半以上。所以在应用到比较复杂的模块处理时，程序员往往使用C++，本项目中蓝图只提供了获得数据和展示数据的功能，占总工程量的20%。

## 虚幻引擎的渲染

虚幻引擎 4 拥有全新的、DirectX 11 管线的渲染系统，包括延迟着色，全局光照，半透明光照，后处理以及使用矢量场的 GPU 粒子模拟。

虚幻引擎4中所有光照均是延迟光照，这点与虚幻引擎3有所不同，虚幻引擎3采用的是前置光照。材质将他们的属性写到缓存中，然后光照执行时读取材质的属性，根据一定数学运算对像素进行光照处理。

虚幻引擎一共有三条光照路径：可移动光源（适用于完全动态），固定光源（适用于部分静态），静态光源（适用于完全静态），这几个不同的工具在光照质量，性能和游戏中的实时性与可变化性具有不同的取舍，可以根据项目实际需要来选择。

可移动光源：产生完全动态的光照和阴影，可以实时地改变光源的位置，旋转角度，颜色，亮度，强度衰减，光照范围等属性，几乎所有光源的属性都可以被修改，并且可移动光源产生的不会烘焙到光照贴图中，也不会产生间接光照结果（由被光照的模型反射到别的物体上产生的光照效果）。可移动光源使用全场景动态的方式投射阴影，具有相当大的性能开销。性能消耗的程度主要取决于受到该光源影响的模型的数量，以及这些模型的三角面的数量。也就是说一个半径较大的可移动光源造成阴影的性能开销可能会几倍于一个半径较小的可移动光源。

固定光源：保持固定位置不变的光源，但可以在其他方面进行变更，例如亮度和颜色。这是它们与静态光源的主要不同之处，静态光源无法在游戏时以任何方式进行变更。然而，应该注意的是，在运行时对亮度进行修改仅会影响直接光照。间接（反射）光照由于是通过 Lightmass 进行预计算的，所以不会改变。

在三种光源的可移动属性中，固定光源具有最好的质量、中等的可变性，以及中等的性能消耗。

固定光源的所有间接光照和阴影都存储在光照贴图中。直接阴影被存储在阴影贴图中。这些光源使用距离场阴影，这意味着即使所照亮的物体的光照贴图分辨率很低，它们的阴影都可以保持清晰。

静态光源（Static Light） 是指在运行时不能以任何方式改变或移动的光源。它们仅在光照贴图中进行计算，一旦处理完成后，不会再有进一步的性能影响。可移动对象不能喝静态光源进行交互，所以静态光源的用处是非常有限的。

在三种不同的光源可移动性属性中，静态光源的质量中等、可变性最低、性能消耗也最少。

因为静态光源仅使用光照贴图，所以在游戏可玩之前它们的阴影就烘焙好了。这意味着它们不能给移动（动态）对象产生阴影，正如我们在示例中所看到的。但是，当照亮的对象也是静态的时，静态光源可以产生区域（接触）阴影。这是通过调整 光源半径 属性实现的。然而，应该注意的是，为了获得较好的阴影效果，接收柔和阴影的表面需要合理设置它们的光照贴图分辨率。

静态光源的主要应用场景是为低功率的移动平台准备的。

本项目采用的光源为固定光源，因为我们实时改变的三位模型会产生新的顶点和平面，这意味着我们要对新产生的平面和顶点计算光照，但是又不需要改变光源的位置，固定光源无疑是最好的选择。

## HalfEdge DataStructure

简介

表示多边形网格的一种常见方法是共享顶点列表和存储顶点的指针的列表。这种表示对于许多目的都是方便和有效的，但是在某些领域中证明是无效的。

例如，网格简化通常需要将边缘折叠成单个顶点。该操作需要删除边缘边缘的面部并更新在边缘的端点共享顶点的面部。这种类型的多边形“手术”要求我们发现网格的组成部分之间的相邻关系，例如面部和顶点。虽然我们当然可以在上面提到的简单网格表示上实现这些操作，但它们会花费开发者过多的时间；许多结果将需要我们搜索整个面部或顶点列表，或者甚至可能两者都需要。

多边形网格上的其他类型的邻接查询包括：

哪些面用到了这个顶点

哪些边用到了这个顶点

哪些面和这个边邻接

哪些边组成了这个面

哪些面和这个面邻接

为了有效地实现这些类型的邻接查询，已经开发了更复杂的边界表示（B-ReP），其明确地对网格的顶点、边和面进行建模，并在其内部存储附加的邻接信息。

这些类型的表示中最常见的一种是翼缘数据结构，其中边缘用指针指向它们接触的两个顶点，两个面邻接它们，以及指向从端点发出的四个边缘的指针。这种结构允许我们在恒定的时间内确定哪一个面或顶点与边缘相交，但是其他类型的查询可能需要更小号时间的处理。

半边数据结构是一个稍微复杂的B-ReP，它允许在上面的所有查询（以及其他）在恒定的时间（\*）中执行。此外，即使我们在面、顶点和边中包含邻接信息，它们的大小仍然是固定的（不使用动态数组）以及数据具有合理的紧凑。

这些性质使得半边缘数据结构对于许多应用来说是一个极好的选择，然而它只能代表歧管表面，在某些情况下，它被证明是无效的。歧管在数学上被定义为表面，每个点被一个具有圆盘拓扑的小区域包围。对于多边形网格，这意味着每个边都由两个完全的面接界，不允许T结点、内部多边形和网格中的断点。

更准确地说，每条信息收集的时间是固定的。例如，当查询与顶点相邻的所有边缘时，该操作将在与顶点相邻的边缘的数量上是线性的，但查询每个边缘的时间是恒定的。

## 结构

因为不是存储网格的边缘，所以我们存储半边，结构也被称为半边结构。顾名思义，半边是边的一半，并通过沿其边缘分裂边缘来构造。我们把两个半边称为一对边。半边是有向的，一对的两个边有相反的方向。

下面的图显示了三角形网格的半边表示的一小部分。黄色点是网格的顶点，而浅蓝色的条是半边。图中的箭头表示指针，尽管为了防止图表变得凌乱，有些图已被忽略。



正如上图所示，边框边的半边在其周界形成一个圆形链表。这个列表既可以顺时针或逆时针方向围绕脸部，只要使用相同的规则。循环中的每一个边沿都存储指向其边界的指针（图中未示出）、顶点在其端点（也未示出）和指向其对的指针。它可能在C语言中看起来像这样：

struct HE\_edge

{

HE\_vert\* vert; // vertex at the end of the half-edge

HE\_edge\* pair; // oppositely oriented adjacent half-edge

HE\_face\* face; // face the half-edge borders

HE\_edge\* next; // next half-edge around the face

};

半边数据结构中的顶点存储它们的x、y和z位置以及指向顶点的一个半边的指针，该顶点使用顶点作为起始点。在任何给定的顶点，我们可以选择一个以上的一半边，但是我们只需要一个。在C中，顶点结构看起来是这样的：

struct HE\_vert

{

float x;

float y;

float z；

HE\_edge\* edge; // one of the half-edges emantating from the vertex

};

对于半边数据结构的一个简单的版本，一个面只需要存储一个指针到一个半边沿。在一个更实际的实现中，我们可能也会在面结构中存储纹理、法线等信息。面中的半边指针类似于顶点结构中的指针，尽管每个面都有多个半边，但我们只需要存储其中一个，不必考虑哪一个。（因为半边之间有相邻关系，我们很容易由一个半边找到其余所有的半边）下面是C中的面部结构：

struct HE\_face

{

HE\_edge\* edge; // one of the half-edges bordering the face

};

# 

## 邻接查询

大多数邻接查询的答案直接存储在边缘、顶点和面的数据结构中。例如，边框或顶点边框可以很容易地找到这样的边沿：

HE\_vert\* vert1 = edge->vert;

HE\_vert\* vert2 = edge->pair->vert;

HE\_face\* face1 = edge->face;

HE\_face\* face2 = edge->pair->face;

一个稍微复杂的例子是在与面相邻的半边上迭代。因为面周围的半边形成了一个循环链表，并且面结构存储了指向这些半边中的一个的指针，所以我们这样做：

HE\_edge\* edge = face->edge;

do {

// do something with edge

edge = edge->next;

} while (edge != face->edge);

类似地，我们可能对迭代在与某个顶点相邻的边或面感兴趣。回头看一下图表，可以看到除了在面的边界周围的循环链表之外，指针还围绕顶点形成循环。迭代过程对于发现相邻的边或顶点到顶点是相同的；这里是在C语言中的实现：

HE\_edge\* edge = vert->edge;

do {

// do something with edge, edge->pair or edge->face

edge = edge->pair->next;

} while (edge != vert->edge);

注意，在这些迭代示例中不包含空指针的检查。这是因为对曲面的限制是多方面的，为了满足这一要求，所有的指针都必须是有效的。

通过这些例子可以快速找到其他的邻接关系。

在本项目中使用了经过拓展的HalfEdge Data Structure，功能更加完整，具体将在下一节做出介绍。

# 第二章 撰写基本要求

## 2.1 论文题目

论文题目应言简意赅，方便检索，不超过20个汉字，英文题名不超过10个实词为宜。

## 2.2 摘要

以第三人称撰写，必须在300字左右，要求无缩略语和特殊术语，内容应包括目的、方法、结果(主要数据)和结论。

在中、英文摘要后，必须提供3—5个关键词。

## 2.3 论文正文

论文正文部分主要包括：绪论、方法、实验结果、讨论、结论。

绪论应包括该毕业设计的目的和该毕业设计与其他相关研究的关系。

方法描述应尽量做到简明扼要 ，但应让其他有经验的研究者能够重复该方法进行验证。完全新的方法应该详细描述。有关方法的改进只有在必须重复该实验的前提下才需给出详细的论述。

实验结果应尽量用图表表示。实验结果的讨论要简明，应集中对所得的结果做出解释而不是重复的叙述。

## 2.4 图表

### 2.4.1 图

图应有标题和图标注释，以使其容易被读者理解。在有分图的时候，所有数字、字母和符号必须一致，并清楚标明测量单位。

### 2.4.2 表

表应有标题并有足够的信息使读者不去查阅正文即可理解该表的内容。表内每一栏均应有表头，表内的缩写应在表注中说明，文中表格输入在正文中该出现的地方。

## 2.5 参考文献

按GB7714-87《文后参考文献著录规则》采用顺序编码制著录，依照其在文中出现的先后顺序用阿拉伯数字加方括号标出。

作者应在自己的文章中，对国内同行已发表的相关研究论文，给予充分的反映。论文摘要集；未正式公开发表的论文、著作、数据等，以及“私人通讯”不能作为文献引用。

参考文献中的作者，第1—3名全部列出，3名以上只列出前3名，后加“等”（英文用“et al”）。外文期刊名称可用标准缩写，中文期刊用全名。

# 第三章 打印基本格式

## 3.1 论文题目

中文一号黑体，英文Times New Roman 26。

## 3.2 摘要

“摘要”标题打印格式要求：中文黑体小二号，英文Times New Roman 22。

摘要正文打印格式要求：中文宋小四号，英文Times New Roman 14。

关键词打印格式要求：中文宋小四号，英文Times New Roman 14。

## 3.3 论文正文

论文正文，中文用宋小四号，英文用Times New Roman 12。叙述性文字间不要有空格；数字与单位之间需空一格。行距为1.5倍行距。文章中的大小写、正斜体、上下角、希文等，均需特别留意区分。

每个标题以不超过15个字为宜，题末不加标点符号。

一级标题，例如“第一章 绪论”、“第二章 撰写基本要求”、“第三章 打印基本格式”，题序和标题用黑体小二号字（即选择样式“标题1”），居中编排。

二级标题，例如“2.1 论文题目”，题序和标题用黑体三号字（即选择样式“标题2”），编号数字之间用下圆点“."隔开，如“2.1”，取左顶格编排格式。

三级标题，例如“2.4.1 图”，题序和标题用黑体四号字（即选择样式“标题3”），也取左顶格编排格式，编号数字之间用下圆点“."隔开，如“2.4.1”。二、三级标题后的内容另起一行。

三级标题以下的内容，可用(1)，(2)，(3)区分段落，如仍有小标题，则标题与正文之间空一个汉字距。



图1 打印基本格式要求（字体）

## 3.4 图表

图表应有序号和标题。中文用宋小四号，英文用Times New Roman 12。图的序号和标题应打印在图的正下方，如图1所示。表的序号和标题应打印在表的正上方，如表1。

表1 字体与字号要求

|  |  |  |
| --- | --- | --- |
| 题序和标题 | 字体 | 字号 |
| 论文题目 | 黑体 | 一号 |
| 第一层次（章） | 黑体 | 小二号 |
| 第二层次（节） | 黑体 | 小三号 |
| 第三层次（条） | 黑体 | 四号 |
| 第四层次（正文） | 宋体 | 小四号 |

## 3.5 参考文献

“参考文献”标题打印格式要求：中文黑体小二号，英文Times New Roman 22。

中文用宋小四号，英文用Times New Roman 12。按GB7714-87《文后参考文献著录规则》采用顺序编码制著录，依照其在文中出现的先后顺序用阿拉伯数字加方括号标出。

引用期刊的格式：[序号]，作者姓名(名与名之间不空格). 题目. 期刊名称(斜体，缩写)，年份，卷(期)号：起止页码。例如：

[1] 严壮志.立体视差测距中摄像机参数的一种测定方法.电子学报, 1996, 24(6):37-40

[2] ROCHA A, TONG F, and Yan ZZ. A computer aided consultant system for mammogram diagnosis. Journal of Shanghai University, 1999,3(4): 293-298

引用书籍的格式：[序号]，著者(或加In：编者). 书名， 版本(第1版不著录),出版地：出版者, 出版年：起止页码。例如：

[3] Yan ZZ, Eiho S and Tanaka H. A-Map: A knowledge model of human brain atlas for CT interpretation. In: K.C. Lun et al (eds). Medinfo92, Netherlands: Elsevier Science Publishers, 1992:635-641

引用论文集的格式：[序号], 作者.论文名.主编.论文集名.出版地: 出版者, 出版年: 起止页码

引用学位论文的格式：[序号], 作者. 题名.[博士或硕士论文].保存地点:保存单位,年份,起止页码

引用技术标准的格式：[序号], 标准编号. 标准名称.

引用专利文献的格式：[序号], 专利所有者.专利题名.专利国别,专利文献种类, 专利号. 出版日期

引用电子文献的格式：[序号], 电子文献名. 电子文献出处或可获得地址, 发表或更新日期

## 3.6 页面设置

## 3.6.1 纸张和页边距

纸张设定为A4，页边距为：上下边距同为2.54厘米，左边距为2.5厘米，右边距为2厘米。（作为Word文档模板使用时，不用修改页面设置）

### 3.6.2 装订线和页码

装订线为0，位置在左边；页码一律用小五号居中标明。

## 3.7 论文撰写与装订顺序

封面中文摘要英文摘要论文目录论文正文致谢参考文献附录一、附录二等等【包括-- 英文文献 ，翻译中文(2500～3000中文字)， 有关图纸等】

# 致谢

本文中图1和表1由通信工程系陈杰老师制作，在此表示感谢。

# 参考文献

[1] 上海大学教务处编. 上海大学毕业论文（设计）工作条例. G6.1-4