**电 子 科 技 大 学**

UNIVERSITY OF ELECTRONIC SCIENCE AND TECHNOLOGY OF CHINA

**学士学位论文**

**BACHELOR THESIS**



论文题目 个性化人脸建模系统设计

学 院 自动化工程学院

专 业 自动化

学 号2015070909007

作者姓名林俊宏

指导教师 黄志奇 副教授

摘 要

人脸三维重建是计算机图形学中的一个重要研究领域，在许多行业领域都有着广泛的应用。自计算机图形学建立起来，一直都是一个研究热点。

本文以个性化建模系统设计为研究课题，探究使用单幅正脸照片进行人脸三维重建的方法。并使用该方法设计出一个原型系统，可以实现操作简便的简单的人脸重建。

在本文研究过程中，详细考察了3D可形变模型法（3DMM）在人脸重建方向上的使用方法和3DMM中参数化人脸表达。在此基础上，研究了一种精简了3DMM方法中人脸表达与匹配算法的径向基函数线性插值法。

本文详细研究了使用径向基函数线性插值法的3D可形变模型方法进行人脸三维重建过程中中所需要的各项关键技术，包括人脸特征点的标定方法、人脸三维模型的表征方法、径向基函数线性插值法、纹理贴图的映射方法等。并在此过程中使用OpenCV与OpenGL对图片与模型数据进行可视化渲染并输出图像。

之后依据各部分研究成果组合设计出原型系统，可以实现仅依靠输入单幅正脸照片对整个人脸的三维形状进行还原，将还原结果显示并作为模型文件保存。完成了基于单幅正脸照片的人脸三维重建功能。

**关键词：**人脸三维重建，三维模型形变，径向基函数离散插值法，三维模型纹理映射

ABSTRACT

Face 3d reconstruction is an important research field in computer graphics and has been widely used in many fields. Ever since the born of computer graphics, it has been a research hotspot.

This paper takes the design of personalized modeling system as the research subject, explores the method of 3d face reconstruction using a single positive face photo, and designs a prototype system using this method, which can make face reconstruction with simple operation.

During the process of research, the application of 3D morphable model in face reconstruction (3DMM) and parameterized face expression of 3DMM were investigated in detail. Based on this, a radial basis function linear interpolation method is studied, which simplifies the face expression and matching algorithm in the 3DMM method.

In this paper, the 3D deformable model method using radial basis function linear interpolation method is studied in detail to carry out various key technologies needed in the 3D face reconstruction process, including facial landmark detection, face expression of 3D model, linear interpolation method using radial basis function, texture mapping method and etc. In this process, OpenCV and OpenGL are used to visually render images and model data and output images.

The prototype system is designed according to the combination of research results of above. Which can make the 3D reconstruction model, only by inputting a single positive face photo, and display the restored results and save them as a model file. The 3d face reconstruction function based on single positive face photo is completed.

**Keywords:** 3D face reconstruction, face model morphing, Radial basis function, texture mapping

目 录

[第一章 绪 论 1](#_Toc8382896)

[1.1 研究工作的背景与意义 1](#_Toc8382897)

[1.2 人脸三维重建方法的国内外研究历史与现状 1](#_Toc8382898)

[1.3 本文的主要研究目的 2](#_Toc8382899)

[1.4 本论文的结构安排 2](#_Toc8382900)

[第二章 人脸三维重建方法 4](#_Toc8382901)

[2.1 人脸三维重建方法概述 4](#_Toc8382902)

[2.2 人脸特征点检测概述 5](#_Toc8382903)

[2.3 三维模型形变技术 6](#_Toc8382904)

[2.3.1 3DMM 6](#_Toc8382905)

[2.3.2 3DMM中人脸的表示方法 6](#_Toc8382906)

[2.4 3DMM人脸模型重建 8](#_Toc8382907)

[2.4 本章小结 9](#_Toc8382908)

[第三章 基于单幅正脸照片的人脸三维重建 10](#_Toc8382909)

[3.1 对于3DMM方法的精简 10](#_Toc8382910)

[3.2特征点标定与提取 10](#_Toc8382911)

[3.2.2 人脸特征点检测、标定方法使用 10](#_Toc8382912)

[3.2.3 特征点数据处理 11](#_Toc8382913)

[3.3 平均人脸模型及其处理 12](#_Toc8382914)

[3.3.1 SFM模型 12](#_Toc8382915)

[3.3.2 OBJ模型格式 13](#_Toc8382916)

[3.3.3 解析模型文件 14](#_Toc8382917)

[3.3.4 模型特征点标注 14](#_Toc8382918)

[3.4 三维模型插值形变 16](#_Toc8382919)

[3.4.1 径向基函数 17](#_Toc8382920)

[3.4.2 径向基函数插值方法 18](#_Toc8382921)

[3.5 由正脸图像到三维模型的纹理映射 18](#_Toc8382922)

[3.5.1　模型对齐 18](#_Toc8382923)

[3.5.2　纹理映射 19](#_Toc8382924)

[3.6 本章小结 19](#_Toc8382925)

[第四章 原型系统设计 20](#_Toc8382926)

[4.1 系统需求分析 20](#_Toc8382927)

[4.2 功能设计 21](#_Toc8382928)

[4.2.1 软件启动及初始化 21](#_Toc8382929)

[4.2.2 人脸特征点提取模块 22](#_Toc8382930)

[4.2.3 模型变形 23](#_Toc8382931)

[4.2.4 纹理映射 24](#_Toc8382932)

[4.2.5 数据输出 26](#_Toc8382933)

[4.3 功能测试 29](#_Toc8382934)

[4.4 本章小结 30](#_Toc8382935)

[第五章 总结与展望 31](#_Toc8382936)

[5.1 全文总结 31](#_Toc8382937)

[5.2 展望 32](#_Toc8382938)

[致 谢 33](#_Toc8382939)

[参考文献 34](#_Toc8382940)

[外文资料原文 35](#_Toc8382941)

[外文资料译文 37](#_Toc8382942)

第一章 绪 论

1.1 研究工作的背景与意义

处于信息时代，影视数码后期、游戏制作、虚拟现实重建、医疗和监控安防等行业、领域，对于三维人脸建模技术有着强烈的需求。三维人脸建模技术在商业上有非常优秀的应用前景。同时随着电子硬件、软件技术的不断进步，计算机图形学技术的迅速发展，有越来越多的研究单位和组织加入到三维人脸建模技术研究的领域中。眼下这正是一个十分热门的研究方向。

在数码影视行业，3DCG技术的流行使得电影特效与镜头语言跨上了一个前所未有的台阶，使用电脑合成人物影像来替代真人演员，可以执行更加高难度、富有视觉冲击力的表演，更能呈现出超现实的精彩演出。

在游戏制作领域和虚拟现实技术领域，产业正在迈入次世代，计算机硬件也在不停升级，使用者对于游戏画面的追求更在逐步提升。而在去年，英伟达推出了最新一代图灵架构的显卡，支持高效率的光线追踪运算，标志着光线追踪技术正式进入民用、商品化范围，这使得3D游戏和虚拟现实的画面表现力迈上一个更高的台阶。三维人脸建模作为3D游戏和虚拟现实的一个重要技术 ，也将产生更多更重要的作用。如何更快更节省成本地制作出更加精细的人脸模型，是现在制作者们面临的一个难题。

此外，在医疗、新闻、安防监控等其他行业、领域，也在借助人脸三维建模来辅助进行分析、视觉表现等。

在以前，较成熟的人脸三维建模通常是使用复杂的仪器、专业硬件和高度可控的光学环境来进行采样并分析建模。但在近些年，伴随着移动电子设备的流行和人脸数据化的普及，研究者们开始转向仅使用简单的采集、分析计算设备（如手机、家用电脑、自动售货机，售票机等），在非可控的光学条件下进行人脸的识别、采集、分析与建模。在这类条件下，数据的获取变得非常简单，但是数据所包含的信息量也大幅减少，这使得三维重建的难度陡增。并且，仅仅从照片、视频流等平面视角获取的人脸信息，是缺失深度数据的。由此进行的三维重建，如何获取完整的三维数据，是目前挑战的难点所在。

1.2 人脸三维重建方法的国内外研究历史与现状

人脸三维重建是一种计算机图形学技术。计算机图形学始于1963年Ivan Sutherland在麻省理工学院发表的《画板》博士论文及其系统。其后，在上世纪70年代，Frederic I. Parke提出了参数化人脸重建法，标志了计算机表示人脸技术的开端。

由于应用市场广阔，国外的许多公司、高校与研究机构很早就瞄准了相关应用市场并展开了大量的研究工作。微软、苹果、谷歌、脸书等科技公司在最近几年都投入了大量的财力物力研发人脸相关的计算机视觉技术，并落地了非常多的成熟产品到市场投入使用。

比较传统的人脸三位重建方法一般是以肌肉特征为基础，研究多边形网格模拟人脸下，收到肌肉应力产生形变的肌肉模型。其后常见参数化描述人脸特征，使用PCA法降低参数描述复杂度，同时保证足够数据特征差异的参数化人脸模型。Izadi、Kim、Hilliges等人通过Kinetic摄像机的深度扫描功能提取人脸特征点，并对普通人脸模型进行调整，可以实现基于单幅深度图片的人脸模型配准功能[1]。Blanz和Vetter的3DMM方法使用人脸数据库来参数化描述人脸，并使用迭代的方法逐渐逼近目标人脸[2]。Li、Chen、Liu等结合局部数据集与公共数据集，改进了3DMM的拟合方法，并使用卷积神经网络辅助模型形变，建模效果得到了有效的提高[3]。

国内的研究起步较晚，但近几年腾讯、百度、旷视等公司和许多高校与研究机构不断加大投入，全力追赶国际研究进度。

比较典型的有 Xiangyu Zhu等人提出的使用卷积网络求取3DMM相关系数并对齐三维人脸的方法[4]。

1.3 本文的主要研究目的

当前的行业的研究重心通常在使用卷积神经网络进行模型变形实现重建，或使用级联回归神经网络绕过模型变形，端对端地实现人脸三维重建。这些方法通常都以提高精度作为主要目的，使用数据库来训练神经网络。

本论文立足于功能实现，力求实现算法简单、数据依赖量小、速度快且具有较高精度的低成本的人脸三维重建方法，并利用该方法设计原型系统，实现基于单幅正脸照片的人脸三维重建功能。

1.4 本论文的结构安排

本文的章节结构安排如下：

第一章主要是介绍人脸三维重建的研究概况，包括研究工作的背景与意义、人脸特征点检测研究现状、人脸三维重建方法的国内外研究历史与现状，以及本文主要的研究内容；

第二章，是对人脸的三维重建方法进行详细的介绍与对比。重点介绍本文选择的三维形变模型方法的内容；

第三章是本文在研究实现基于单幅正脸照片的，使用三维形变模型的人脸三维重建方法中所使用到的技术：人脸识别与特征点提取、平均人脸模型的特征点标定、三维模型的形变算法、三维模型贴图映射等；

第四章是结合第三章中的研究结果，进行软件原型的设计，并对设计完成软件进行测试与分析；

第五章，对全文工作做出总结，并提出成果的不足与可改进的方向。

第二章 人脸三维重建方法

人脸三维重建是指在计算机中使用数据表征一个给定人脸的三维模型，即将给定人脸在计算机三维空间中重建出来。给定人脸可以是单幅或多幅照片、单目视角的视频、双目视角的视频等2维图像数据或经过专门仪器，如深度摄像机、核磁共振仪等，采集到的三维深度信息和图像数据。响应的，重建方法也有所不同。

本章节只针对2维图像数据，就由2维图像到三维模型的重建方法，进行简要介绍，并进一步关注基于模型的三维模型形变法，介绍其原理与重建方法。并且，作为重建中的重要一环，在本章节也会对人脸特征点检测研究状况进行简要介绍。

2.1 人脸三维重建方法概述

由2D图像到3D模型的重建方法，由大致三种类型组成：

传统人脸三维重建法；基于模型的三维人脸重建法；端到端的人脸三维重建。

传统人脸三维重建方法大多是直接使用图像信息，建立一个数学模型来描述人脸特征。

传统人脸重建方法中最著名的是肌肉模型，该方法使用多边形网格表征人脸，同时在形变上大量考虑了皮下骨骼、肌肉对于表面皮肤形状的影响。该方法在人脸的表情动画模拟上具有较大优势，但是如果定位不正确，容易产生不自然甚至不存在的表情。另外有一种比较著名的方法，叫做伪肌肉模型。该方法使用假想的操作点包围被控物体，利用空间点阵的操作实现形变，此方法没有天然地考虑肌肉、骨骼对模型表面的影响，比肌肉模型法操作简单，但控制细节相对缺失。

此外还有正交视图建模、多图建模、单目和双目视频序列建模等方法。核心通常都是基于不同视觉条件（如视角、光照等）下对人脸上同一点的空间坐标进行计算，最后得出一个表征目标模型的点云和其他相关信息，并重建出三维模型。

基于模型的三维人脸重建法，通常是使用一个基于统计学的平均人脸模型，再按照一定方法得到目标人脸的特定模型。

较早的，有J.Ahlberg提出的一种常用的通用人脸模型：Candide-3模型[5]。Candide-3模型顶点数量少，对人脸形状概括程度高，计算量小。但正由于顶点过少，所以重建结果不够细腻。

Blanz和Vetter提出的三维形变模型（3DMM）是最有代表性的做法[2]。他们使用一个基于大量人脸数据线性组和出的人脸参数模型，并使用数据库中的人脸数据作为形变插值量，与特定的人脸图像进行匹配，进而实现人脸三维重建。

这类方法运用统计学数据估计了二维图像的深度信息。对于大多数人脸来说误差在可接受范围之内。

端到端的人脸三维重建方法，是绕过人脸模型，直接使用自己设计的人脸表示方法，采用级联相关神经网络结构回归，端到端地建立起目标人脸表示形式（单幅照片、多幅照片、视频序列等）与目标人脸模型的回归方法，并重建人脸模型。此类方法需要大量数据进行训练，有良好的发展前景。

此外，还有在特定光学环境下对人脸光学特征进行采集从而判断人脸深度信息，进行三维重建的方法。这类方法对采集环境有很高的要求，同时使用仪器昂贵，计算难度大，适用范围狭窄，成本高。但在精度上可以达到相当高的级别。

2.2 人脸特征点检测概述

人脸检测（face detection）与特征点检测（facial landmark detection）是模式识别中的一个重要课题。

人脸特征点的检测是与人脸识别、人脸检测一体的，为了能够识别图像中的人脸，首先需要找到人脸在图像中的位置。因此，面部检测——在图像中定位面部并返回包含面部边界的矩形——是一个热门的研究领域[7]。

2001年，Paul Viola和Michael Jones在他们的论文《Rapid Object Detection using a Boosted Cascade of Simple Features》[8]中基本解决了这个问题。

人脸特征点检测是在定位出人脸位置的前提下，定位出人脸面部的关键区域位置描述，包括眼睛、鼻子、嘴巴、眉毛、脸部轮廓等，的方法。

人脸关键点的检测方法大致主要有：基于模型的主动形状模型ASM(Active Shape Model)和主动外观模型AAM(Active Appearence Model)；基于级联的姿势回归CPR(Cascaded Pose Regression);还有其它基于深度学习的方法。

ASM算法由Cootes等人提出于1995年，是基于特征点分布模型PDM(Point Distribution Model)的方法。在PDM中，可以用若干关键特征点(landmarks)的坐标一次串联而成的一个形状向量来表示一个几何形状。ASM就是针对人脸特征形状的PDM模型方法，它用一组形状向量描述人脸的形状，在训练集中从对齐的样本中建模，然后再形状向量未知的测试样本上，使用局部纹理模型搜索最契合统计描述的形状。ASM速度非常快，适合实时的跟踪性检测。

AAM是ASM的改进版，由Cootes在1998年提出。AAM不仅使用形状约束，而且又加入了脸部区域的纹理特征。即Appreance = Shape + Texture。

CPR算法由Dollar在2010年提出，用来预测物体的形状。对于人脸特征点检测，可以把这个问题看作是从人脸的表观到人脸形状的回归过程。通过反复迭代，可以把形状回归到最优的特征点位置上。

近几年人脸特征点检测在深度学习方面有了爆发式的进展，检测的精度与速度都有了显著的提升。不仅是特征点检测，多任务的检测方法也十分流行。较新的FAN人脸对齐网络结合了人脸特征点定位架构（landmark localization）和残差模块（residual block），使得网络深度达到一个非常强大的水平。

本文虽然需要使用人脸特征点检测技术，但并未对其做详尽的研究，仅是使用了现成的研究成果，因此在此只做简要介绍，不再详细论述其工作原理。

2.3 三维模型形变技术

形变模型（Morphable Model）来自计算机图形学中一种叫做“形变”（Morphing）的技术。Morphing是指将一个物体模型光滑过渡为另一个不同模型的技术。

形变技术需要解决的核心问题是如何建立两幅图像之间内容的相互映射关系。对于二维形变，只需要指定前后两幅图像中的特征要素（点或线）来表示前后的特征要素的对应关系，变换过程就能通过插值得到。对于普遍意义上的三维形变，针对两个不同拓扑结构模型之间的形状过度，由于不能直观地给出特征的对应关系，因此处理起来要困难的多。

此外，Marc ten Bosch研究了一种在四维模型中进行三维切片的方法[6]，如果四维模型处处连续，则一个连续切片过程为切片两端三维模型的一种光滑形变过程。

2.3.1 3DMM

瑞士巴塞尔大学的科学家Blanz和Vetter在99年的这篇文献《A Morphable Model For The Synthesis Of 3D Faces》[1]中，针对人脸的模型形变，提出三维形变模型法（3D Morphable Model,简称3DMM）。

三维形变模型法，是利用一个人脸数据库，构造出一个基于统计学的平均人脸模型，再使用模型数据库中的人脸数据与平均人脸模型进行线性组和，得到特定的人脸模型。

2.3.2 3DMM中人脸的表示方法

作者使用两种向量表征具有n个顶点的人脸模型。

形状向量（shape-vector），包含模型顶点坐标信息：

|  |  |  |
| --- | --- | --- |
|  |  | （2-1） |

其中 (i∈(1,n))是表示一个顶点坐标的三维向量，包含x、y、z三个分量。

纹理向量（texture-vector），包含颜色信息：

|  |  |  |
| --- | --- | --- |
|  |  | （2-2） |

其中 (i∈(1,n))为表示纹理颜色的三维向量，包含R、G、B三个分量。

假设建立的目标人脸三维模型可以由m个人脸模型线性组和而成，其中每个人脸模型都包含和两种向量(i∈(1,m))。那么目标人脸模型就可以用这样的方法来表示：

|  |  |  |
| --- | --- | --- |
|  |  | （2-3） |
|  |  | （2-4） |

其中为平均人脸模型顶点集，表示形状的PCA部分；为平均人脸模型纹理集，表示纹理的PCA部分。

由此，就可以使用已有的人脸模型的线性组和表示一个新的人脸模型。同时，改变系数、，便可以在已有人脸基础上生成更多不同的人脸。

在这个前提下，一个人脸需要由这几个部分表示：一个平均人脸模型、用于变形的大量人脸模型、每个人脸模型的比例系数。对于已有完整的人脸素材库的情况，只需要提供比例系数，就能表示一个完整的人脸。

文中作者做了图 2‑1中的演示：为一个参数化描述的可形变模型原型中添加大量的人脸数据并调整参数。图中数据代表从平均值中增加或减少原型与平均值的差值。标准的变形人脸就在原型和平均值的正中间（右上\*）。从平均值减去差值就能得到反脸（右下#）。

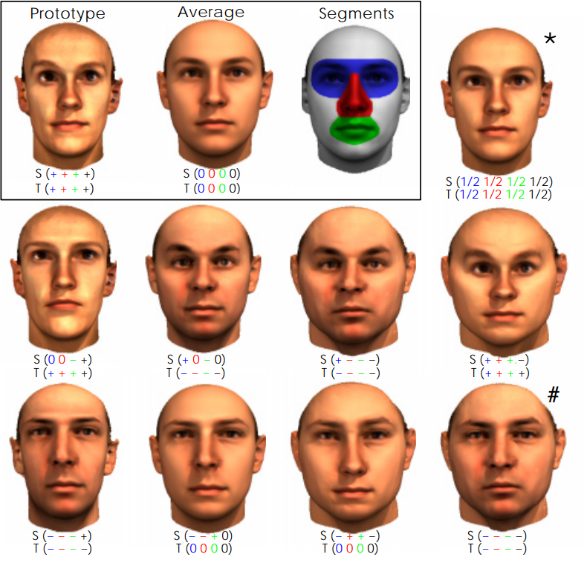


图 2‑1 对一个参数化人脸模型修改参数产生变化

同时将人脸分为四个部分：左右眼、鼻、口、脸部轮廓（篮、红、绿、黑），在四个部分上分别对形状和纹理进行相加或相减，就能产生许多不同的脸。这种分割相当于将一个完整的人脸的向量空间分为独立的子空间。再分别计算每个分割端的线性组和，并在边界处进行融合，就能还原成一个完整的三维人脸。

2.4 3DMM人脸模型重建

有了合理的人脸表征方法后，对于特定的人脸，对其进行三维重建的问题便转为了求取、值的问题。在这个问题中，对形状和纹理的求值问题处理过程是相似的，这里就只讨论形状。

对一张给定的目标人脸的照片，首先将模型与人脸照片进行配准。

按照相同的规则在模型与人脸照片上规定相同数目的特征点和。可知模型上的特征点是S（公式2-1中表示模型形状点集）的一个子集。于是新的面部形状的特征点的顶点坐标可以表示为：

|  |  |  |
| --- | --- | --- |
|  |  | （2-5） |

其中是平均人脸模型的特征点集，是使用的人脸库的特征点集。

求其在照片所在平面的二维投影：

|  |  |  |
| --- | --- | --- |
|  |  | （2-6） |

,是正交投影矩阵，是旋转矩阵，是位移矩阵。

通过对比与可得模型与目标人脸的差距。

所以三维重建问题转化为了求解满足下面能量方程的系数：

|  |  |  |
| --- | --- | --- |
|  |  | （2-7） |

式中添加了正则化部分：是主分量系数，对应形状系数，是对应的主成分偏差。

通过上式，可以求解出使得三维模型中的规定的特征点在照片所在的二维平面上的投影，与二维平面上的特征点的距离差最小的系数。

每次应用此法解出一个合适的3DMM模型表示参数，都可以使当前计算出的模型距离目标人脸更进一步。每次都以最新的人脸模型参数作为基础进行公式（2-7）的求解，便能不断逼近目标人脸的形状。反复迭代，当计算出的模型与目标人脸距离足够小时，即可认为这个模型是目标人脸的三维重建。纹理表示的计算工程相似。

如果参与模型表征的人脸数据越多，则需要迭代的次数越少，但由于参数变多，每次迭代的时长却会变长；并且如果设定的精度过于高时，有可能永远都无法迭代出距离足够小的计算结果，导致程序进入死循环，而由于本身计算速度比较慢，在运行时可能无法判断出陷入了死循环。这是这种方法明显存在的问题。

2.5 本章小结

本章简要介绍了人脸重建方法与人脸特征点检测方法，并详细介绍了三维可形变模型（3DMM）人脸重建法，包括3DMM算法、3DMM中人脸的参数化表示和使用3DMM方法进行重建的大致过程。

第三章 基于单幅正脸照片的人脸三维重建

出于本文立足于实现一个轻量化的系统的目的，此章节对一种使用精简化的三维可形变模型法的人脸三维重建方法进行探究，论述了此方法中使用到的各项技术及实验方法：人脸特征点提取、模型文件的解析、径向基函数插值法、纹理映射方法。

3.1 对于3DMM方法的精简

由于3DMM方法是完全使用参数与数据库来描述人脸，所以需要复杂的计算使得形状和纹理可以通过现有人脸模型的线性组和来表示。

但目前的条件下，已经获得了目标人脸的照片，所以纹理部分可以直接将照片纹理信息映射到已经与目标人脸配准的模型上。这样便可以节省大量的工作量，又由于直接使用目标人脸的纹理，在拟真程度上也比较高。

另外，原本3DMM方法的拟合过程受到了既有的人脸形状（纹理）库的约束，使得求解过程极为复杂，计算强度大、时间长。为了节省计算成本，本文采用了数学插值形变的方法，直接由特征点作为控制点进行插值运算拟合目标人脸特征点，再将求解出的运算方法应用到所有点上，便可得出目标人脸的三维模型。这种方法计算量小、速度快，且精度较高。缺点是缺乏原有3DMM方法形变的自由度，如果要对已完成的模型进行调整还需要另外的手段，不能简单地通过修改参数完成；也不能通过人脸的一些模糊的五官特征对人脸模型深度数据进行估计。

通过这样对问题进行简化，现在整个重建过程只需要重建目标的正脸照片和一个平均人脸模型即可。

因此本课题需要解决的问题被拆分成三个部分：特征点标定与提取、三维模型形变、纹理映射。

3.2特征点标定与提取

3.2.1 人脸特征点检测、标定方法使用

鉴于本课题立足于高效率、易实现的原则，为了保证模型重建精度与课题实现难易度，本文使用开放的人脸特征点检测工具来提取目标人脸的特征点。目前市面上开放的人脸检测工具通常准确率都在99%以上[17]，本文在考察了多个人脸特征点检测工具后，选择使用腾讯AI Lab的人脸与人体识别AI中的五官定位工具。

腾讯AI lab 使用的人脸检测算法Face R-FCN，是一种基于区域的人脸检测器。它通过在R-FCN框架上进行修改，使其更适用于人脸检测[10]。17年的时候它在WIDER FACE上冲到了榜首。

他们的AI工具提供了完整的说明文档[11]。通过规定的网络API，即可使用相应的功能：接入五官定位API，需使用Post方法通过https协议传输utf-8编码格式的数据请求。然后会获得json格式的响应结果。

调用AI技术接口之前，首先需要获取接口鉴权签名，并将签名附在请求数据之中。

签名计算需要四个步骤:

1.将键值对按照key进行字典升序排序，得到顺序键值对列表；

2.将列表中的参数对按照URL键值对的格式拼接成字符串；

3.使用创建的应用密钥生成的字符串拼接到上一步所得字符串末尾；

4.对步骤3得到的完整字符串进行MD5运算，然后将MD5值中所有的小写字符置换为大写字符。就得到了接口鉴权签名。

使用接口签名对API进行请求，即可获取json格式的响应结果。

3.2.2 特征点数据处理

通过腾讯AI lab获取的是json格式的字符串数据，需要得到里面的信息还需要解析。

对接口进行五官定位，将会得到如图 3‑1形式组成轮廓的88个点，包括眉毛（左右各8个点）、眼睛（左右各8个点）、鼻子（13个点）、嘴巴（22个点）、脸型轮廓（21个点）。



图 3-1 人脸特征点标记示范图片

其中眉毛和眼睛的特征点为自形状左右端，将之横向四等分、纵向匹配形状边缘的上下各五个端点（上下两层共用左右端点）；

鼻子为鼻尖、两鼻孔、鼻子最底端、左右侧鼻翼最外侧到两眼中点的四等分的五个端点（左右两边共用上端点）；

嘴为上下嘴唇的内、外侧轮廓，先根据左右端点和形状中点二分后，各自二等分，最后为四份的五个端点（共用左右端点）；

脸型轮廓为自上耳根起，到下巴轮廓上的人脸正中点，纵向距离上十等分的十一个端点（左右共用下端点）。

使用开源的jsoncpp库，可以方便地对json格式的字符串数据进行处理。

在这里依照点的分类和顺序依序读取保存，便可以得到人脸的特征点数据集：

|  |  |  |
| --- | --- | --- |
|  |  | （3-1） |

(i∈(1,n))表示照片二维平面上的坐标点。同时还获得照片的尺寸数据ImageWidth、ImageHeight。

对于照片的笛卡尔坐标系，原点在图片右上角，x轴指向左，y轴指向下。

3.3 平均人脸模型及其处理

3.3.1 SFM模型

在本课题中，简化了3DMM模型的表示方法，已不再需要一个人脸数据库。但一个用于重建的人脸模型，仍是必须的。

英国萨里大学视觉、语音和信号处理中心公开了一个使用C++编写的轻量级的3DMM库eos，其中包含了他们使用的一个低分辨率的、仅包含形状的3D可变形面部模型SFM（Surrey Face Model）[12]：有845个顶点、1650个面。他们也发布了该模型的高清版本：有3448个顶点、1.6万个顶点和2.9万个顶点的版本。模型形状都相同，只是精度不同。845顶点版本的模型如图 3-2 所示：

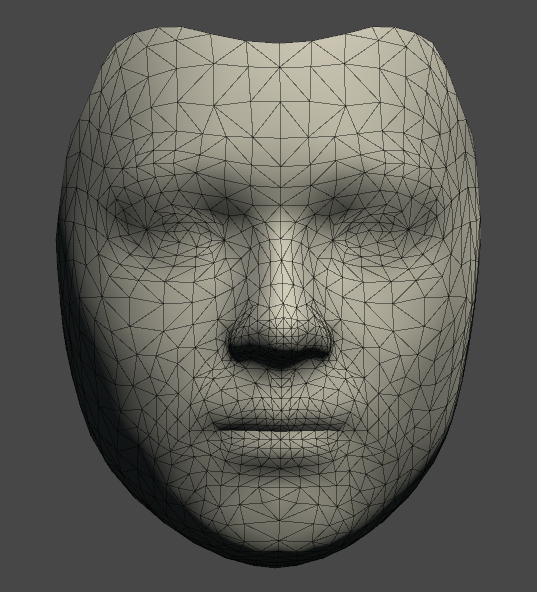


图 3-2 845顶点版本的模型

这是一个去除了性别与人种要素的平均模型，在表示亚洲人脸的表现上相性比其他常见的基于欧美人脸的平均人脸模型要高。

相同形状、不同精度的模型适合使用低精度的做开发与调试，在调试完成后再替换为高精度的模型使用。由于形状相同，因此算法相似，不需要做出很多改动，很适合用于实验性的开发。

另一个选择该模型的原因是，他们的模型采用obj格式储存。

3.3.2 OBJ模型格式

Obj模型格式最早是由[Wavefront Technologies](http://en.wikipedia.org/wiki/Wavefront_Technologies)为3D建模与动画软件Advanced Visualizer开发的一种文本化的模型储存标准，现在已是一种较为通用的模型储存格式。支持直线、多边形、面和自由形态曲线，支持法线和贴图坐标。可以直接用记事本或其他文本编辑方式进行读写修改，非常适合本课题的研究。

在obj文件中，数据由一行行文本组成。每行文本开头使用关键字开头，代表数据类型。在SFM的文件中仅有几何体顶点(Geometric vertices)数据“v”和面(Face)数据“f”，由于需要纹理映射操作，需要使用的数据类型还有纹理顶点 (Texture vertices)数据“vt”和法线顶点 (Vertex normals)数据“vn”。

几何体顶点“v”包含三个浮点数分量、、，代表这一个顶点在三维空间中所在的空间坐标，数字间用空格字符隔开；

纹理顶点“vt”包含两个0~1之间的浮点数、，代表这个点在贴图中的笛卡尔坐标系2个轴上的位置，数字间用空格字符隔开；

法线顶点“vn”包含三个浮点数、、，代表一个三维法向量，数字间用空格字符隔开；

“f”后跟三个数据块，每块包含三个整型数，数之间用“/”隔开，代表组成这个面的三个顶点对应的空间坐标索引号、纹理坐标索引号与顶点法线索引号，数据块之间用空格字符隔开。索引从1开始。

由此一个完整的，包含形状、颜色纹理和法线纹理的模型在obj文件中有如下形式表示的数据:

|  |  |  |
| --- | --- | --- |
|  |  | （3-2） |
|  |  | （3-3） |
|  |  | （3-4） |
|  |  | （3-5） |

其中：；j；；。

另外，由于使用了纹理贴图，还需要包含一个.mtl格式的文件指定贴图，以及一张图片文件作为贴图。图片文件使用目标人脸照片即可。

在.mtl中，通常还需要指定Ns值（高光系数）、d值（材质透明度）、illum值（照明模型）、Kd值（材质固有色diffuse color）、Ks值（材质高光色specular color）、Ka值（材质阴影色ambient color）等属性，用来表示材质的一些其他性质和光照模式。

此部分不在本文研究范围之中，具体的属性说明可以参考FileFormat.Info网站的说明[13]。

3.3.3 解析模型文件

使用C++的IO流，按行读取SFM文件内容，并通过关键字判断数据类型，并依次放入对应的容器中。得到两个容器vertrix和face。

由于face中的顶点序号从1开始，而C++程序中容器序号都是从0开始，为了能够使face中的数据能够正确地索引到对应的顶点上，需要遍历face中的每一个整型数，并使其数值自减1。

3.3.4 模型特征点标注

SFM模型是只包含形状数据的人脸模型，人脸信息包含不完整，无法使用人脸特征点检测工具准确提取到它的特征点。因此本课题选择使用人工标注关键点的方法。

为实现人工标注，需编写一个工具，用来输出模型图像，并可接收鼠标在图像上的点击，并且计算出鼠标点击的点在模型上的坐标。三维图形的渲染，使用OpenGL比较方便。

在OpenGL中比较重要的有5个坐标系[14]：

局部坐标系（Local Space）、世界坐标系（World Space）、观察坐标系（View Space）、裁剪坐标系（Clip Space）、屏幕坐标系（Screen Space）。

局部坐标系是以物体为中心的坐标系，默认与世界坐标系重合；

世界坐标系是绘制空间中固定的标准坐标参考；

观察坐标系可以认为是摄像机的空间坐标系，包含观察点和观察方向；

裁剪坐标系确定了观察的视野和观察的投影方式：透视投影(perspective projection)或者正交投影(orthographic projection)；

屏幕坐标系就是输出设备上的坐标，在电脑应用程序中就是窗口中的坐标，坐标原点位于窗口的左上角，x轴向右为正，y轴向下为正；

在程序方法中，使用矩阵来实现坐标系之间的的转换，局部空间到世界空间的变化矩阵是模型矩阵（model matrix），世界空间到观察空间的变化矩阵是观察矩阵（view matrix），观察空间到裁剪空间的变化矩阵是投影矩阵（projection matrix）。

在OpenGL建立的渲染对象中，存在一个坐标变换过程如图 3-3所示，用来决定物体在渲染对象中的位置、方向、大小、显示范围、观察投影方式、在窗口中的位置等等：

局部坐标系上的点，乘上模型矩阵，可转换到世界空间。再乘上观察矩阵，可以转换到观察空间。接着乘上投影矩阵，转换到裁剪空间。裁剪空间有单独的裁剪算法，用以决定显示范围等。然后点还需要经过视口变换和归一化计算，最后便能得到它在频幕空间上的位置。



屏幕坐标系

裁剪坐标系

观察坐标系

世界坐标系

本地坐标系

视口变换

投影矩阵

观察矩阵

模型矩阵

图 3-3 从物体坐标到屏幕坐标的变换过程

则标注一个点有如下步骤：

1. 对特征点进行人工标注，需要通过鼠标点击渲染窗口，获取点击点的屏幕坐标系位置。
2. 将屏幕坐标使用反向的视口变换（Viewport Transform）便可以得到裁剪坐标。归一化计算由OpenGL自动执行，不需要手动计算一次。
3. 解方程：观察坐标\*投影矩阵=裁剪坐标，无法得到一个在裁剪空间中的特解——坐标z值为不定值，那么设为0即可。
4. 解方程：世界坐标\*观察矩阵=观察坐标，可得世界空间中的位置。由于模型并没有进行过变换操作，因此世界坐标等于局部坐标。

由此得鼠标点击位置在模型中的坐标。注意此时得到的点，并不一定在模型上，因为在步骤3中，并没有求出模型z轴方向上的值对应在裁剪空间中的z值。

因此还需要遍历模型每个面，找出在x、y平面上包围目标点的面，再解出使目标点通过这个面的z值。

使用OpenGL渲染出模型正面正交视图的线框模型如图 3-4 所示。计算出模型的包围盒并储存其在x、y方向的四个二维坐标点。

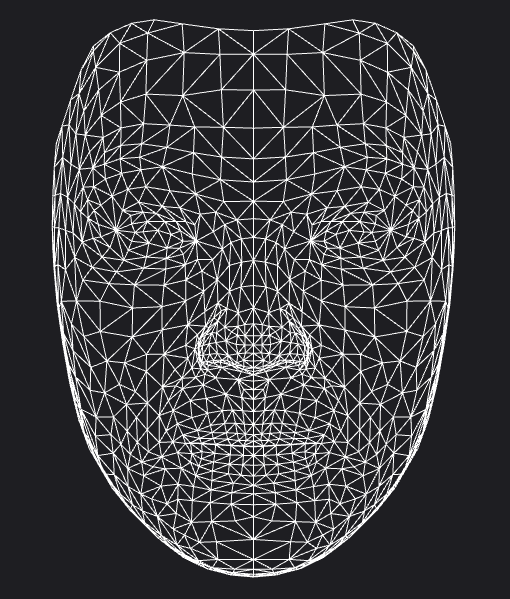


图 3-4 在窗口中输出的线框模型

利用此工具，可以通过人工，使用鼠标点击渲染出的图像中的点，来标定特征点。特征点的分布规则可见本文3.2.2一节。

经过标注，得到模型特征点：

|  |  |  |
| --- | --- | --- |
|  |  | （3-6） |

然后将标记出来的点，翻译成json文本格式储存为json文档。之后便可以利用之前的json解析工具，读取并解析其中的数据。

3.4 三维模型插值形变

3DMM模型的变形是参数式的模型变形: 使用人脸库中的模型数据作为插值量对平均人脸插值，改变输入的参数，便可直接完成变形过程。本文舍弃了需要大量采集数据的人脸库，因此使用自由变形（Free-Form Deformat）的方法，对平均人脸模型进行插值变形。

自由变形方法认为，对一个可变形物体，其变形行为可以使用少量控制点描述。通常是将其放到一个由控制点组成的假想的三维弹性网格中，当网格中的控制点发生移动时，网格形状发生挤压或拉伸产生变形，网格包裹的物体也依照相应的变形过程产生变形。这种方法使得可变形的物体的变形控制，不再依赖对于物体本身的描述，转而使用较少的数据量，模拟了整体的复杂变形。

对于人脸的变形，由于此方法不能考虑到面部骨骼、肌肉、脂肪等内部填充物对于外部轮廓的影响，因此不能很好的对人脸表情变化、皱纹、人脸体积随着肌肉变化的一致性进行有效的模拟。但在本课题中，仅使用普通正脸照片进行人脸模型的三维重建情况下，只是进行人脸形状的重建，因而这些模拟量可以不用考虑。

因此原二维映射逼近问题转化为多变量的散乱数据插值问题：已知初始人脸模型顶点集*P*、初始人脸特征点集以及初始人脸特征点移动后的点集*F’*，且 到*F’*与*P*到*P’* 遵循相同的移动规则，即*F’=f(F)*，*P’=f(P)*，求解*f(x)*，并带入*P*到*f(P)*计算出*P’*。

3.4.1 径向基函数

径向基函数（radial basis function）[15]是一个取值仅依赖于到原点距离的实值函数，即。可以用来表示很多径向函数的和来逼近某一个给定的函数。这种方法有很好的能量最小性质，且与网络拓扑结构无关，回避了普通三维形变方法会遇到的很多难题。并且该方法可以一次给出插值的结果，无需迭代优化，速度要比传统3DMM方法快上很多。

在自由变形方法中，对于一个非控制点，其位移量受所有控制点控制，这种状态正适合使用径向基函数描述。

常见的径向基函数有：

高斯函数：

多二次函数（multiquadric）：

逆二次函数（inverse quadratic）：

逆多二次函数（inverse multiquadric）：

多重调和样条（polyharmonic spline）：

薄板样条（thin plate spline，为多重调和样条的特例）：

在控制模型自由变形方面，常用的有高斯函数和薄板样条函数。

使用高斯函数的变形方法，控制效果随控制点与控制对象的距离增长而剧烈衰减。适合用于对局部地区进行高精度的变形。并且衰减参数可控。

使用薄板样条的变形方法，将会使整个模型所有的被控点较大地收到所有控制点的影响。比起高斯函数更偏向整体控制。

在本课题中，由于模型控制点只包含人脸下三分二部分，而额头没有控制点，如果使用高斯函数，则额头部分模型不容易受到调整。因此选择使用薄板样条函数作为插值方法。

3.4.2 径向基函数插值方法

令模型顶点*P*调整后的新位置为点*P’*，利用径向基函数计算*P’*相对*P*的位移：

|  |  |  |
| --- | --- | --- |
|  |  | （3-7） |

其中n为模型或照片中的特征点数量，即为特征点；为基函数；*表*示模型顶点点的移动距离；为基函数的系数。

由模型特征点与图片特征点均已知，因此。对于式中未知量，可带入所有特征点联立方程组解得。

带入求出的，可知，因此对模型中每个点求取，可得模型的插值量。插值后得模型顶点：

|  |  |  |
| --- | --- | --- |
|  | 。 | （3-8） |

3.5 由正脸图像到三维模型的纹理映射

材质贴图，又称纹理贴图，是一种把图像包裹到3D模型表面的技术。将图像与3D模型表面相关联起来的过程，就是纹理映射。

3.5.1　模型对齐

在obj文件中，使用纹理顶点vt储存一个图片上的uv坐标，值通常在[0,1]之间，表示以图片左下角为原点，x轴正向向右，y轴正向向上，横、纵方向上图片的百分比位置。

为求得模型映射到图片上的位置，首先需要使模型与图片大小、位置都对齐。对齐的依据是3.2节采集的人脸特征点和3.3节标注的平均人脸模型的特征点经3.4节变形后的坐标。理论上，两者在xy平面上具有完全相同的分布规律。

因此可建立变换矩阵T，并建立特征点矩阵F和*F’*，联立矩阵方程组：

|  |  |  |
| --- | --- | --- |
|  |  | (3-9) |
|  |  | (3-10) |

其中，

解方程组，即得T值。

因此对齐后得模型顶点：

|  |  |  |
| --- | --- | --- |
|  |  | （3-11） |

3.5.2　纹理映射

当模型与照片对齐以后，模型上的顶点的xy分量与照片上在人脸相同位置的点的坐标uv是相同的，因此可以使用模型顶点的xy分量直接表示这些顶点的uv值。

因此obj模型中，vt值的u、v量可以使用模型定点的x、y分量分别除以图片的长、宽表示。

3.6 本章小结

本章详细介绍了本文使用的人脸三维重建方法的各项技术的解决方案，从理论与数学层面解析了问题的解决方法。

第四章 原型系统设计

基于上一章节对使用径向基函数插值法的人脸三维重建方法中各项技术的单独研究，在本章节中，这些内容被结合起来，设计出了一个能够满足使用单幅正脸照片的三维重建功能的原型系统。在系统完成后，对其进行简单的测试。

4.1 系统需求分析

本课题的探究内容为“个性化人脸建模系统“，即设计制作一个人脸建模系统原型。系统需要能够输入符合规定的单幅正脸照片，输出一个完整的人脸三维模型。

系统设计需求如图 4-1所示，输入单幅正脸照片，经过个性化人脸建模系统，输出人脸三维模型。

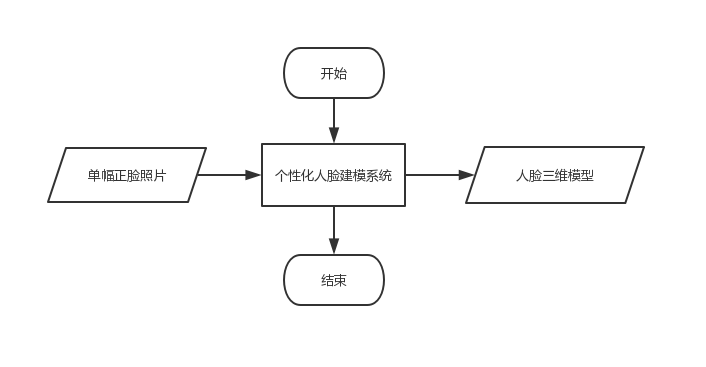


图 4-1 系统设计需求流程图

对系统功能拆解，分为以下功能需求：

1. 平均人脸模型读入内存，模型特征数据初始化；
2. 输入单幅正脸照片，输出目标人脸特征点数据；
3. 输入人脸特征点数据、模型特征点数据与平均人脸模型，使用径向基函数插值计算出目标人脸模型；
4. 输入目标人脸模型与目标人脸照片，进行纹理映射，输出带纹理的模型。

作为内容可视化的一部分，还有以下额外功能：

1. 显示人脸照片，展示标定的特征点；
2. 显示目标人脸模型；
3. 显示带纹理的目标人脸模型。

根据上述需求整理出系统流程图如图 4-2 所示：

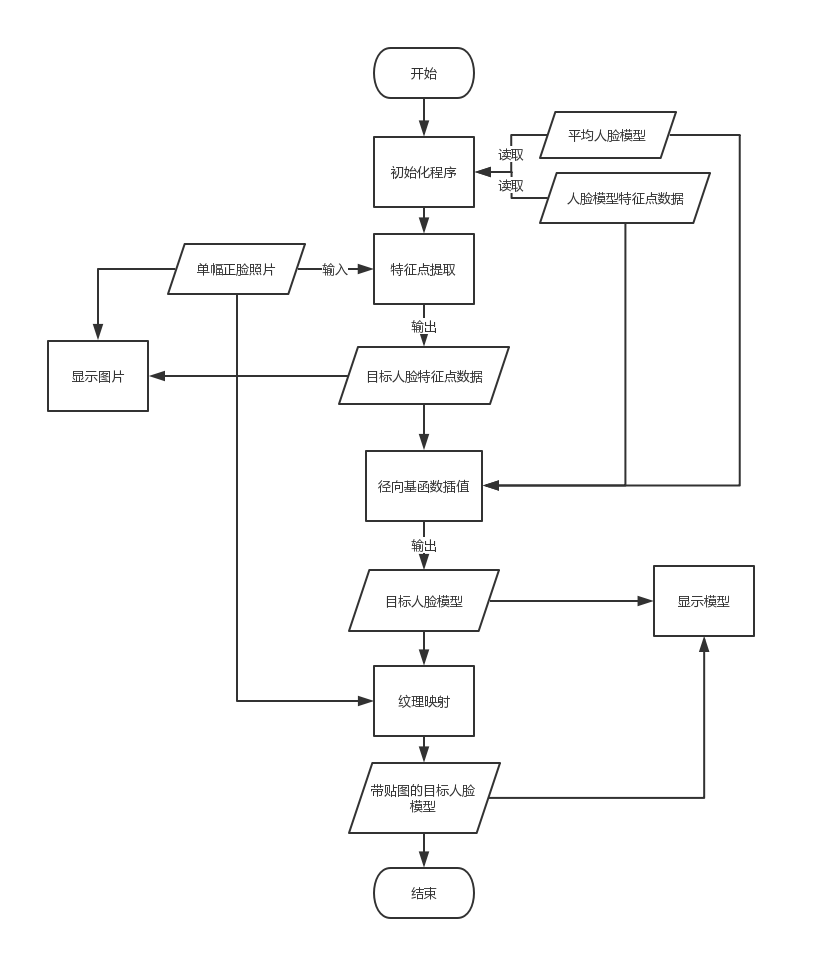


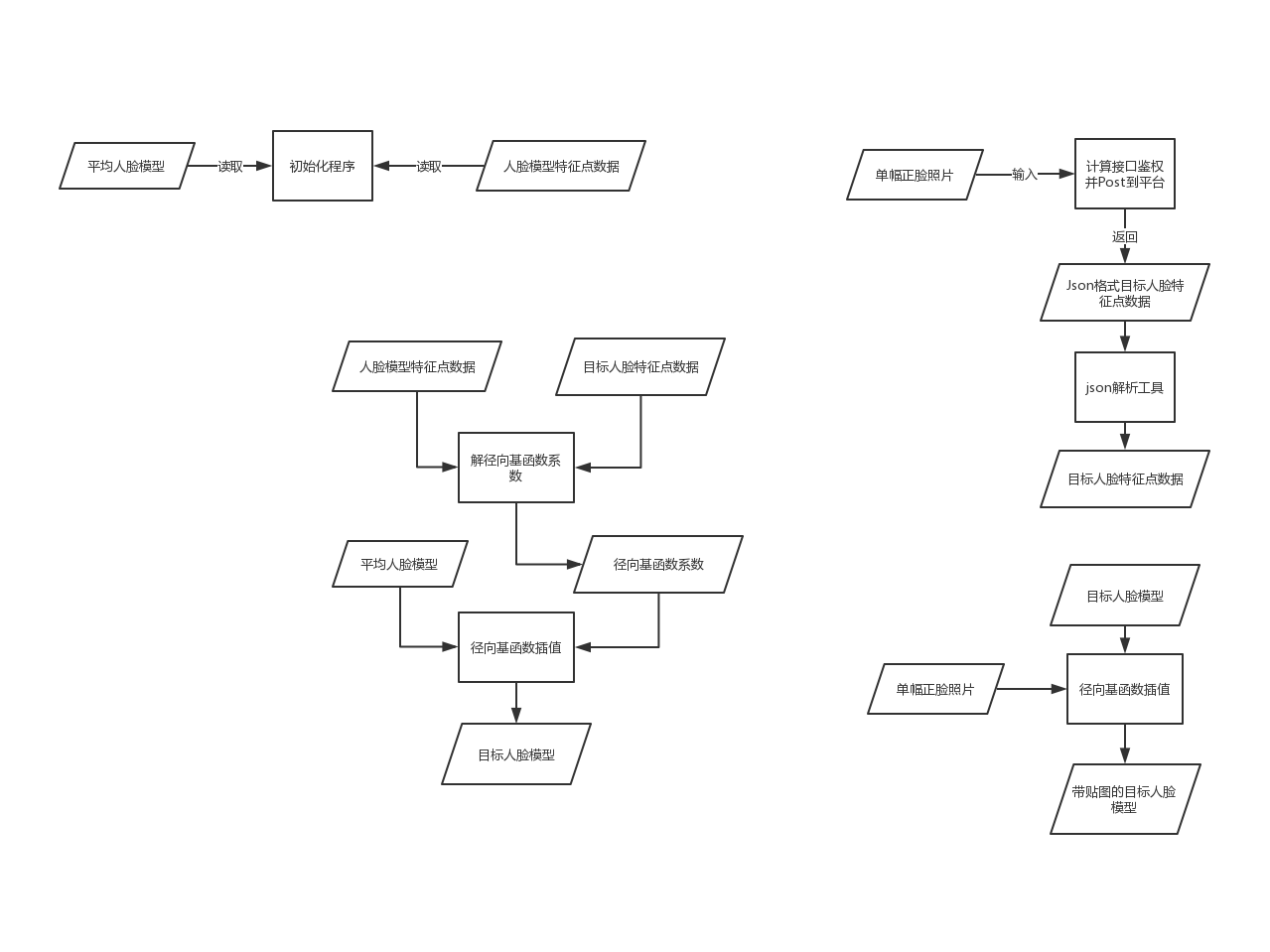
图 4-2 系统流程图

4.2 功能设计

作为一个原型系统，本文只要求系统可以完整并且正确地执行其功能，输出可靠的结果，因此系统设计上只追求功能完整，不考虑软件的美观与易用度。

4.2.1 软件启动及初始化

软件启动时，首先读取预存好的平均人脸模型与平均人脸模型特征点数据。这一模块流程图如图 4-3所示。



结束

开始

图 4-3 初始化模块内部流程图

平均人脸模型是单个的.obj格式文件，使用IO流逐行读入，如果遇到关键词“v”、“f”，就将本行剩余内容作为浮点数或整型数读入，并各自按照读取顺序推入向量中。并且在读取面数据时，由于模型文件中索引值从1开始，而C++程序语言中从0开始，因此读到的数据需要减去1。

最终得到平均人脸模型顶点集（参见公式（3-2））。

特征点数据使用.json格式的文本文件，使用IO流读入。Json数据中内容无换行符，所有的对象都是由 的形式储存的键值对。其中关键字只为字符串，值有json对象(使用花括号{}括起来)、字符串(使用引号””括起来)、数组（使用方括号[]括起来）三种类型，数据之间用逗号分隔，键值之间用冒号分隔。并且一段完整的json数据也是一个json对象而无关键字。

程序在读入.json格式的文件时，无法自动解析其内容。因而在这里使用一个开源的工具Jsoncpp。Jsoncpp可以将输入的字符串转换为json对象，使用者便无须再关心不同的括号类型对数据的划分。但json对象中的每个键值对，仍需人工编写算法识别并储存。

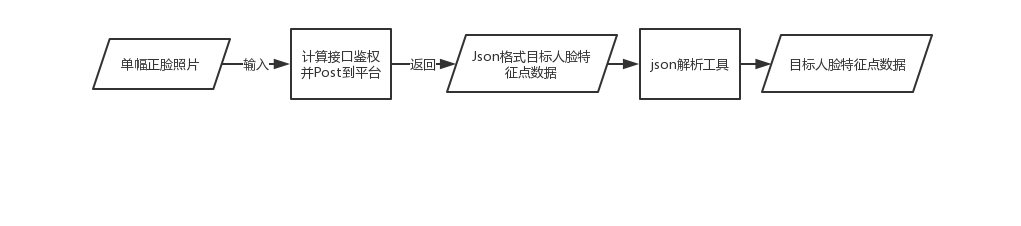
编写一个json解析工具，输入json对象，建立拓扑结构与json对象相同的结构体，使用遍历的方法遍历结构体的所有数据容器，并按照相同的容器结构访问json对象，读取其中数据并并输出。

通过这个工具，可以建立并储存与json数据对象格式完全相同的C结构体数据。

最终得到平均人脸模型特征点集Face（参见公式（3-5））。

4.2.2 人脸特征点提取模块

在本模块中，需要输入照片，将照片应用接口鉴权签名算法计算接口鉴权签名。然后将签名附在post请求中，得到返回的json数据。在这里再次使用json解析工具，得到存有人脸特征点的C结构体数据（参见公式（3-1））。本模块流程图如图 4-4 所示。



结束

开始

图 4-4 人脸特征点提取模块内部流程图

使用OpenCV实例化一个图像对象，并读入照片，例如图 4-5 所示。遍历人脸特征点数据，并在对象的特征点位置上绘制半径为0的圆。绘制完成后，输出图像如图 4-6 所示，可得照片的特征点标记图。



图 4-5 输入照片 图4-6 标记了特征的的照片

4.2.3 模型变形

经过上面两步，已获得平均人脸模型顶点集V，平均人脸特征点集Face，目标人脸特征点集,在这一模块需要计算出目标人脸顶点集V’。在这一模块需要根据人脸模型特征点数据与目标人脸特征点数据求解出对应此脸与平均人脸模型的径向基函数，在将径向基函数应用于平均人脸模型，求出目标人脸模型。本模块流程图如图4-7所示。

根据径向基函数插值法建立线性方程：

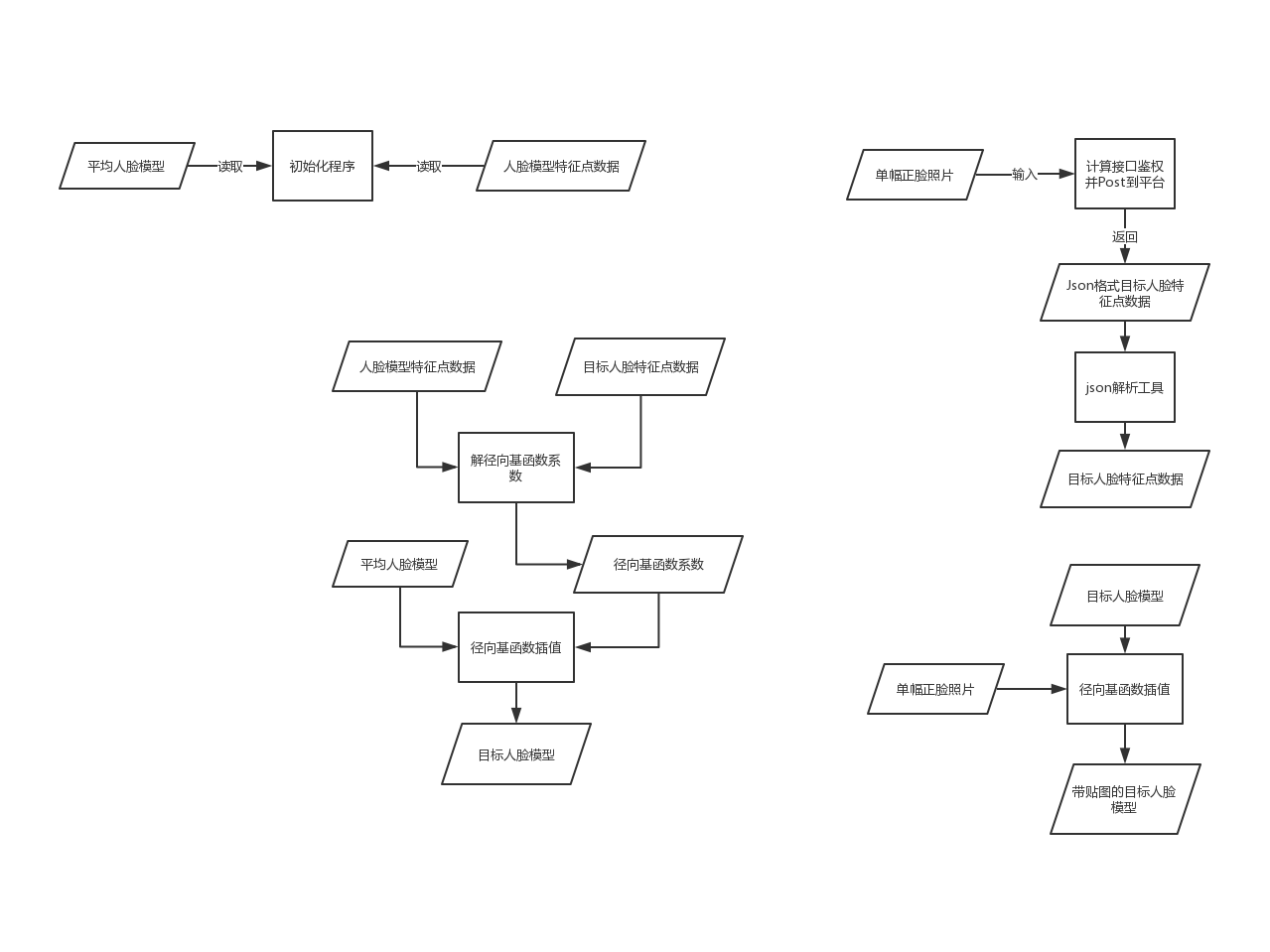
|  |  |  |
| --- | --- | --- |
|  |  | (4-1) |

为基函数的系数，是一个三维向量。带入总共88个特征点，可得线性方程组:

|  |  |  |
| --- | --- | --- |
|  |  | (4-2) |

式中C即为组成的矩阵。为-Face，表示特征点的移动量。K是由公式（4-1）建立的矩阵。

开始



结束

图 4-7 人脸特征点提取模块内部流程图

由于径向基函数是简单的径向和，不能产生多项式，无法规避目标点因与原点距离加大时变大的偏差值。在Pighin的文章中，他主动加上一个低阶多项式以表示整体变换（比如仿射变换），解决了这个问题[16]：

|  |  |  |
| --- | --- | --- |
|  |  | （4-3） |

其中M为仿射部分，t为一个简单的位移矩阵。

在此加上仿射变换约束条件：

|  |  |  |
| --- | --- | --- |
|  |  | （4-4） |
|  |  | （4-5） |

联立公式（4-3）、公式（4-4）、公式（4-5），建立线性方程组矩阵便可解出的值。

为便于计算，也为了加快计算速度，在此引入Eigen库。

Eigen是一个高层次的开源C++库，内含高效的线性代数、矩阵、矢量运算等相关算法。

首先将Face、、C全部转换为Eigen矩阵Eigen:: MatrixXd，然后建立线性方程组矩阵：

|  |  |  |
| --- | --- | --- |
|  |  | （4-6） |

使用Eigen内的LU分解法解出部分，提取、和。

将结果带入公式（4-3）：

|  |  |  |
| --- | --- | --- |
|  |  | （4-7） |

建立线性方程组矩阵：

|  |  |  |
| --- | --- | --- |
|  |  | （4-8） |

求出部分，即求出了模型每个点的径向基函数插值量ΔV。

应用式子V’=V+ΔV，求出目标人脸模型V’。

调用OpenGL方法glPolygonMode(GL\_FRONT\_AND\_BACK ,GL\_LINE )，并绘制物体，输出物体到窗口，可得以线框模型形式渲染的目标人脸模型如图4-8所示：

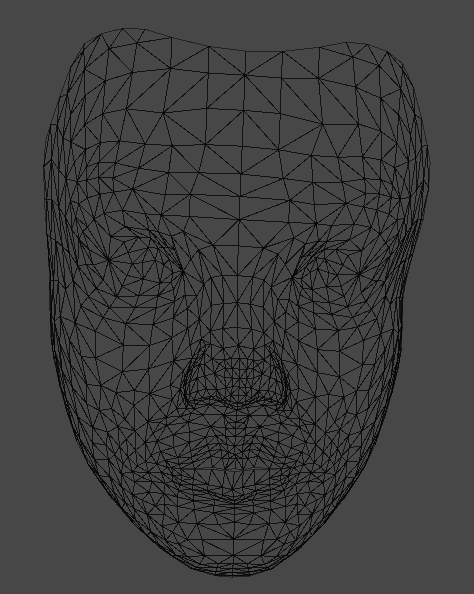
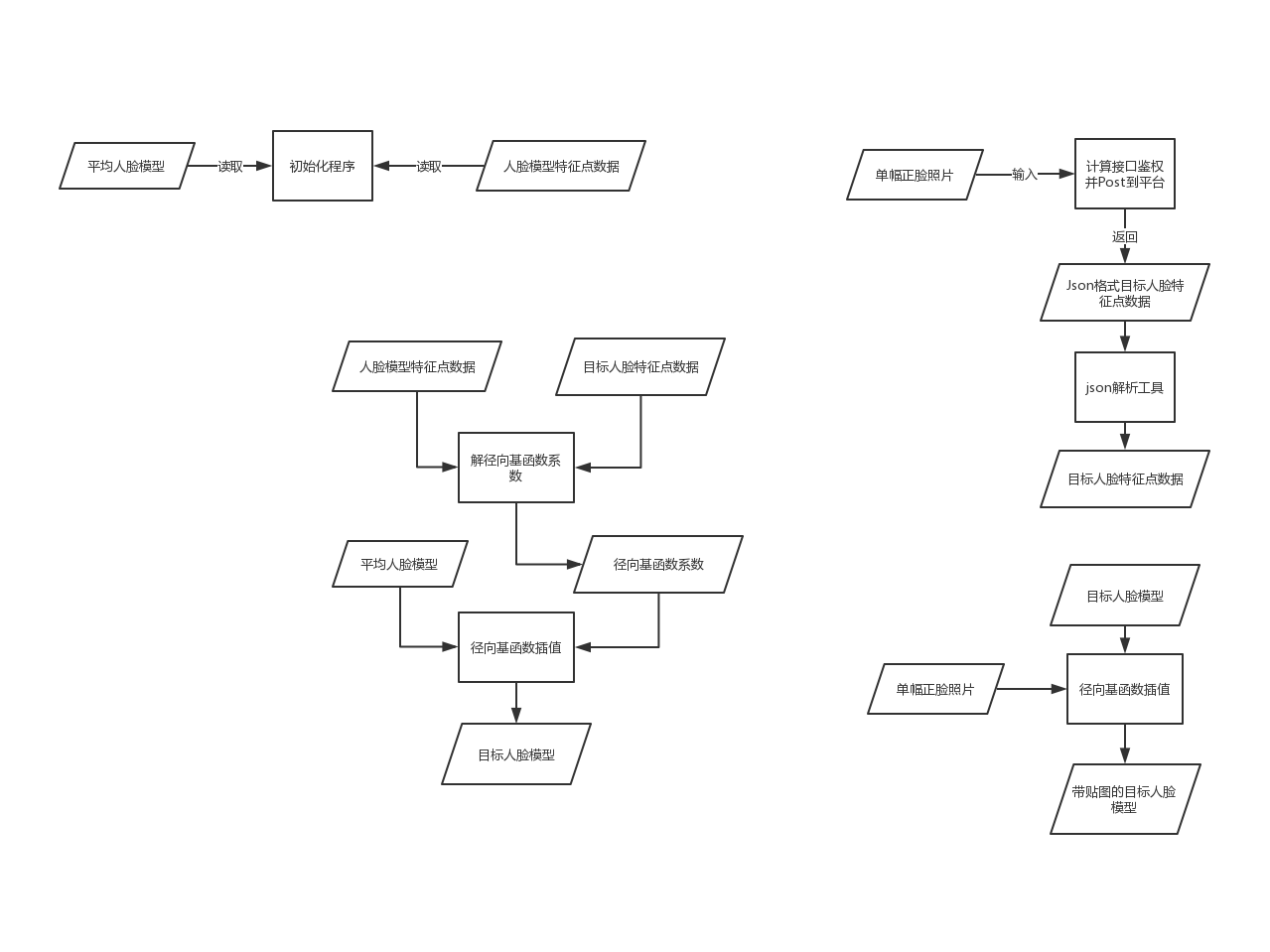


图 4-8 目标人脸模型的线框模型

4.2.4 纹理映射

本模块需要处理纹理映射模块的程序算法，系统流程图如图4-9所示。

经由4.2.3计算出来的人脸模型，由于使用了仿射部分用来消减误差量，并不是与纹理贴图对齐的，在这里需要先进行一次变换操作使模型与贴图对齐。



结束

纹理映射

开始

图 4-9 纹理映射模块流程图

输入目标人脸模型、人脸照片I、目标人脸特征点集，建立变换矩阵T[2,2],实例化一个投影矩阵将映射到照片所在二维平面。在此还需要目标人脸模型的特征点集，需在4.2.3中对F应用径向基函数插值，并左乘投影矩阵。

设线性方程组矩阵：

|  |  |  |
| --- | --- | --- |
|  |  | （4-9） |

再次使用Eigen内的LU分解法解出T的值。

通过P’T求出在xy平面上与照片对齐的模型顶点xy分量，并单独记录在一个向量容器中。

完成对齐操作后，对于模型的顶点i，顶点坐标值为，顶点对应的纹理uv坐标值为ImageWidth、ImageHeight。

此时还需计算模型的顶点法线。

在OBJ格式的文件中，默认可以不指定顶点纹理或顶点法线。但即使没有指定，在使用OpenGL进行绘制时，也需要再算一次。为了模型表现的统一和更方便、性能更高地使用模型，在这里还是计算出来。

顶点法线是用来帮助模型渲染时计算光线到达物体表面入射角的参数，含义为面（点）朝向的方向。一般情况下，顶点法线和面法线的方向是相同的，即针对每个面计算法线。在这种情况下，每个顶点处会有数个法线方向。

但在Obj格式的模型文件中，顶点法线储存是和顶点在一起的，如果针对每个面储存顶点法线，将会产生等同于面的数量的顶点法线数据，如果还要对法线进行平滑运算，数量还会乘三。如果针对每个顶点储存顶点法线，被不同面使用的同一个顶点，对应的是一个法线方向，产生的顶点法线数据量仅等同于顶点数，而产生的效果并无明显失真。并且计算的时候便已经完成了平滑运算操作。因此这里选择针对每个顶点进行法线运算，可以有效减少最终模型文件大小，也可以优化读写速度，有效改善模型性能。

对于一个由三个顶点A、B、C组成的三角形面，如何判断面的朝向，可以应用“左手原则”： 左手食指弯曲，指向顶点排布方向，竖起大拇指，即指向法线方向。对于该面上的一个顶点，可以这样计算：设A点为该点，则=，=，和都是法向量。简单的法向量判别方式可以参照图4-10所示：

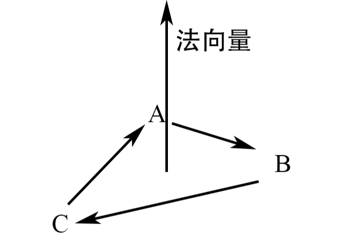


图 4-10 三角形面ABC的法向量

在模型中，对一个点所在的每个面的法向量进行加和取平均值，便可以得到这个点的法向量。

法向量的计算可以使用这个算法：

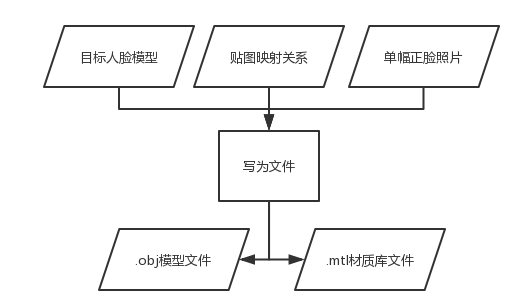
1. 实例化向量容器和暂存符号用于记录对应索引号顶点的法向量；
2. 按顺序取出一个面其中的顶点索引号。
3. 针对一个面的三个顶点，各自计算一次法向量，并分别加到对应索引号的元素中，计算顶点所在面的法向量和；
4. 中对应索引号的统计次数加一，表示参与加和的向量个数；
5. 在面数据取完之前，重复2、3、4步骤。
6. 遍历，使每个元素除等于相同索引号的元素值。

遍历完成后，就储存了对应顶点所在所有面的平均法向量，也就是目标人脸模型的法向量集。

为使法向量表现正常，还需要对求出的法向量进行规范化(normalised)。对于空间向量，规范化便是使向量各分量除以向量模长，等同于求取同向的单位空间向量。

4.2.5 数据输出

本模块需要将计算出来的模型数据写成文件，流程图如图4-11所示。



开始

结束

图 4-11 数据输出模块流程图

一个具有纹理贴图的obj格式模型，共有3个文件：.obj格式储存的模型文件、.mtl格式储存的材质库文件和贴图映射源的图片文件。图片文件使用输入照片即可，.obj模型文件和.mtl 材质库文件仍需使用算法生成。

先生成.obj文件：

使用IO流创建一个输出流，按步骤写入下面内容：

1. "mtllib "+材质库名+".mtl"，表示使用材质库；
2. 按照几何体顶点顺序，一行行写入"v "+顶点x坐标+“ ”+顶点y坐标+” “+顶点z坐标；
3. 按照纹理顶点顺序，一行行写入"vt "+顶点u坐标+“ ”+顶点v坐标；
4. 按照法线顶点顺序，一行行写入"vn "+顶点x坐标+“ ”+顶点y坐标+” “+顶点z坐标；
5. 在此之后，在写入面数据之前，需要指示使用的材质库文件中的材质库"usemtl "+材质库名；
6. 之后写入面数据，按行写入"f "+x+"/"+x+"/"+x+" "+y+"/"+y+"/"+y+" "+z+"/"+z+"/"+z ，格式与内容代表这个面的3个顶点x、y、z对应的几何体顶点坐标索引号/材质顶点坐标索引号/法线顶点索引号。

至此.obj模型文件写入完成，调用file.close()方法关闭并储存文件到指定目录下。

由于模型使用了纹理贴图，因此材质库只需要指定贴图文件就可以了。使用IO流创建一个输出流，在第一行实例化一个新材质库并命名，从第二行开始使用缩进符：

指定材质库的Ns值、d值、illum值、Kd值、Ks值、Ka值，最后写入map\_Kd+” “+贴图文件名。

至此.mtl材质库文件写入完成，调用file.close()方法关闭并储存文件到指定目录下。

将读入的输入图像一并写入到相同目录，便得到了一个完整的附带材质的人脸三维重建模型。

作为一个建模系统，依旧需要有一个渲染功能，能够不经过输出的文件直接查看生成的模型。鉴于模型已有纹理，可追加光照效果，这里使用一个模型加载库Asslmp，将本系统计算出来的目标人脸模型数据转换为OpenGL可以处理的VBO、EBO、纹理数据。将数据载入后，为保证渲染在合适的地方，对空间进行适当的变换，输出图像如图4-12所示：



图 4-12 输出的人脸模型

将输出算法设为一个等待循环，设置几个输入按键事件，循环响应按键事件：设置数字1为移动观察空间到人脸正前方，数字2、3分别在人脸左右，数字4在人脸正上方，并将移动后的图像重新输出，实现即时地变换观察视角，如图4-13所示。

由于经过对齐，模型尺寸与图片中的人脸尺寸相同，也可以将原图渲染在模型后面以供参考，如图4-14所示。





1. （b） （c）

图 4-13 （a）左侧脸视图；（b）右侧脸视图；（c）顶部视图





1. （b）

图 4-14 （a）输入图片；（b）将输入图片渲染在模型后面

可见模型与原人脸对齐比较精确，在脸部边缘有少许位置未能对准，漏出了原图，参照之前输出的特征点图可知照片上左脸边缘露出的部分为特征点标定时不够精确造成的，另外手工标记模型特征点时也存在一定的误差。模型颜色与照片颜色差别较大是因为光照系统没有调试准确：在这里应用了点光源，具有阴影与高光效果的渲染图像，而二维图片不会参与光照计算，造成了颜色差距。

4.3 功能测试

输入不同照片，记录每一步的输出结果，并记录程序除开等待输入文件的时间与模型渲染循环时间，仅就读取图片、通过网络API获取特征点、计算目标人脸模型并输出为文件的耗时。记录结果如表4-1所示：

表 4-1 5次测试的数据记录

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 输入照片 | 人脸特征点检测 | 网格模型 | 输出模型 | 耗时（ms） |
|  |  | /C:\Users\ljhwu\AppData\Roaming\Tencent\Users\735102786\TIM\WinTemp\RichOle\W@T@~VS0D{EF1`187@4{}ZG.png | /C:\Users\ljhwu\AppData\Roaming\Tencent\Users\735102786\TIM\WinTemp\RichOle\4%5~6K8CT_MBK4VANU}VLAL.png | 2089 |
|  |  |  |  | 2266 |
|  |  |  |  | 2371 |
|  |  |  |  | 2125 |
|  |  |  |  | 2269 |

经过测试，本程序重建效果无明显误差，重建平均耗时约2.2秒，速度满足即时的操作需求。

4.4 本章小结

本章详细分析了个性化人脸建模系统的功能需求，将系统功能需求拆解为各个功能的需求，并据此规划出系统的工作流程。按照工作流程顺序，本章节中依次对每个流程的设计内容展开详细的叙述，完成了整个原型系统的设计工作。最后对原型系统进行了简单的测试，验证系统可靠性。

第五章 总结与展望

5.1 全文总结

本论文的研究课题是“个性化人脸建模系统设计“。研究内容包括：人脸特征点提取与数据分析、三维模型的数据处理、插值变形、OpenCV 和OpenGL库的使用、C++IO流进行文件的输入与输出、Https通信、多语言交叉环境、编写计算机应用程序实现人机交互等。

以个性化建模系统为研究、设计目的，基于3D可形变模型的基本方法，本论文对一种基于原版3DMM方法的简化、改进方法：径向基函数插值法进行了研究。并在研究成果上设计了原型系统，成功使用系统实现了基于单幅正脸照片的人脸三维重建。验证了系统的可行性。

在本论文的研究过程中，还制作了几个工具：.Josn格式数据解析工具、.obj格式文件读写与数据解析工具、在OpenGL下使用鼠标点击标定模型表面点的工具、二维和三维向量的计算工具等，在本论文的研究过程中起到了较大的便利。

对人脸进行三维重建，是近半世纪以来，计算机图形学上的一个热门研究领域，也是计算机图形学的一个重要应用面。

对于人脸三维重建的方法的研究重心，自深度学习与神经网络流行起来，便从传统的基于算法和模型的方法，转向人工智能层面上。因此本文的研究内容并未跟上行业研究前沿。但对于传统的基于算法和模型的方法，由于其实现相对容易、成本低，不需要建立巨大的数据库并做人工标定，仍有一定的研究与使用价值。

在单幅正脸照片中，只有人脸横纵向的形状信息而缺乏直接的深度信息。但是人脸的形状是有规律的：窄脸一般深度差距明显，宽脸一般比较平，肥脸圆；眉眼距小，则眼窝较深，眉眼距大，眼窝较浅；厚嘴唇高，薄嘴唇平……应用这些规律与模式，研究者们提出了相当多的方法来估计深度值。本文研究方法的基础方法3DMM方法，便是应用一个人脸数据库，使用人脸数据本身作为规律来估计深度。

本文采用的基于径向基函数的模型插值方法，是使用一个基于概率学的平均人脸模型，通过特征点控制变形实现模型拟合原照片中人脸形状的方法。因此在深度信息上，十分粗暴地直接使用了一个标准解：平均人脸模型的深度信息。也就是说，对于通过本论文中设计的系统，计算出的人脸模型，使用的都是同一套深度信息分布。好在平均人脸模型本身在概率学上的代表性，使得在这种前提下计算出的人脸模型，通常与原本人脸不会有太大的误差，并且在正面形状分布上，模型具有较高的精度，使得误差距离进一步缩小。鉴于成本，可以接受。

5.2 展望

基于本论文因使用平均人脸模型，而造成普遍性的深度信息差异的问题，以及功能简陋的问题，可以通过以下方法进行改进：

1. 引入更多不同类型的平均人脸模型作为计算基础，即针对不同面部类型，如方脸、圆脸等，各自提供一个平均模型。在输入图像，获取人脸特征点分布时，判断目标人脸的类型，并提供匹配类型的平均人脸。也可以使用更高精度的人脸模型，提高模型拟真度。
2. 内置入人脸特征点识别方法，并可应用于平均人脸模型、可以在脱机情况下进行三维重建。在引入此方法后，就可以实现自动化的平均人脸库的扩展。也可能使用包含更多特征点的人脸特征提取，提高变形精确度。更进一步的，引入负反馈控制进行少量迭代，消除系统误差，由于系统本身具有比较快的运算速度，少量的迭代次数是可接受的。
3. 将人脸分区域变形。分眼、鼻、口、脸轮廓等部分分别进行变形，在变形完成后使用平滑算法连接各部分，可以在变形层面上更加精确匹配不同类型的人脸五官特征。
4. 人脸姿态判断。由于不能保证摄制人脸绝对正面，因此还需要对人脸的姿态进行判断，并在变形算法中做三维的特征插值。
5. 相机视角逆向。本文的计算过程是基于正交投影的，而原始照片通常是透视投影，因此重建出的模型可能会比原本的人脸更窄长。在数据获取中加入拍摄相机的镜头数据，计算出照片的投影矩阵并应用到模型映射上，可以进一步缩小误差。
6. 引入MPEG-4人脸表情描述模型和肌肉模型，使得输出人脸可以逼真地呈现不同表情，也可以识别出原照片中人脸表情，并使用对应表情的平均人脸进行变形操作。
7. 多角度照片的特征点识别与空间坐标点估算。使用不同角度的人脸照片，对于深度信息的估计便可以更加精确，可以有效提高模型的真实度。

致 谢

在完成毕业论文的过程中，我收到了许多人的帮助和关怀，是他们支持我坚持完成了本论文的研究内容！

感谢我的导师黄志奇老师对我工作的悉心指导，给迷惑不知方向的我指清了研究道路。

感谢同工作室的朋友们，赵泽源、易思源、何金权、崔玥、潘春贤、张世擎、谢榕泽，你们在我的研究工作与日常生活中给予了我莫大的帮助。

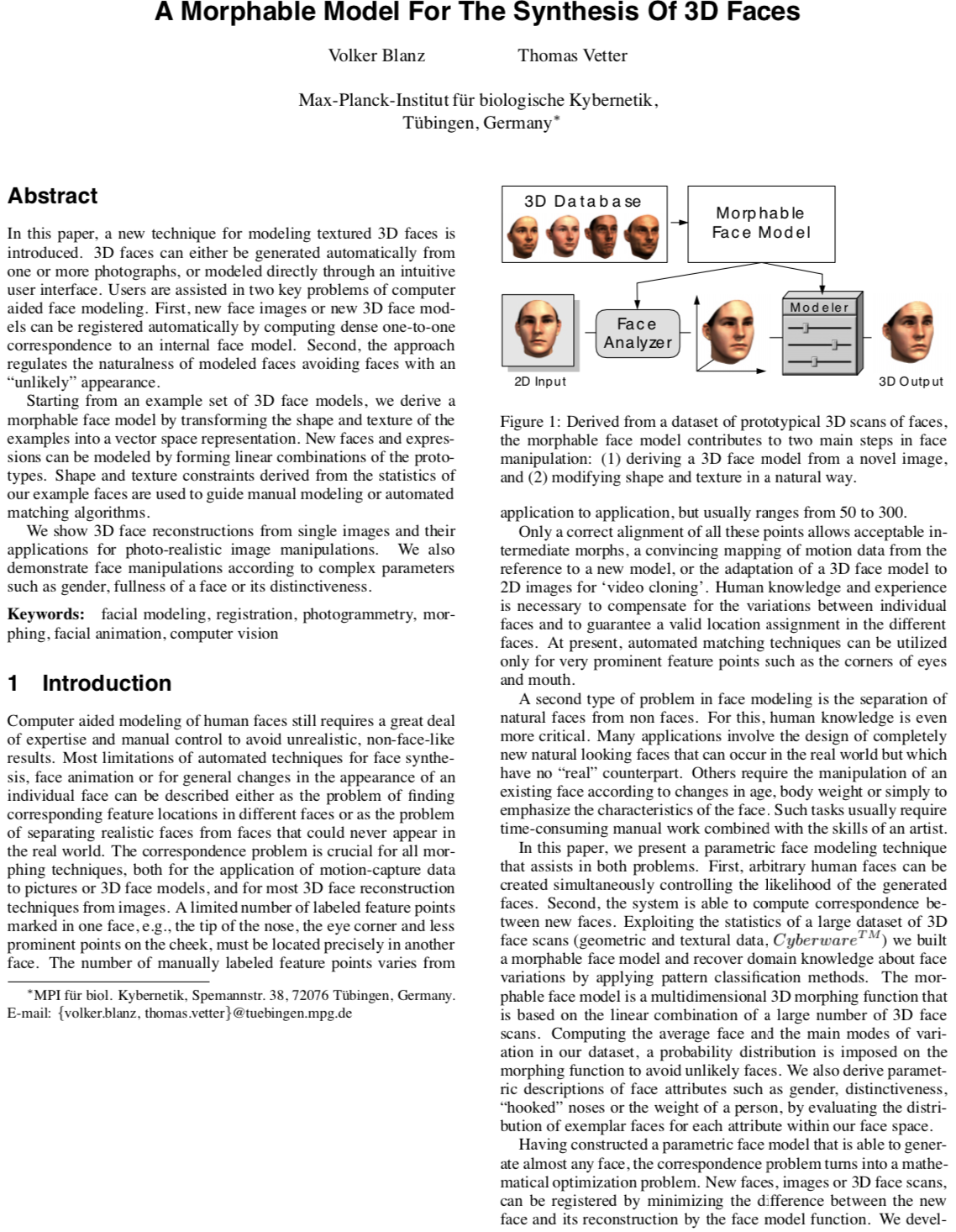
感谢自动化2015级9班的各位同窗，我的父母，以及格斗科技的各位前辈，在学习、工作与生活中对我提供的帮助与支持。

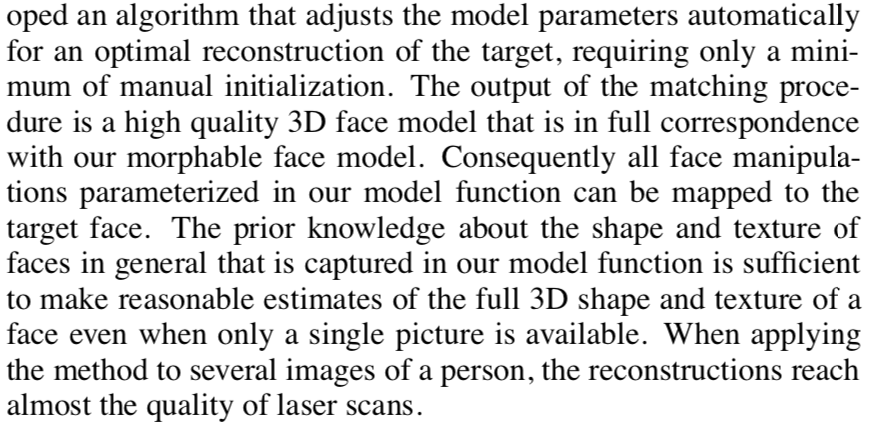
感谢各位，是你们的理解、帮助和支持让我能够专心学习、研究，顺利完成毕业设计内容。

参考文献

1. Izadi S, Kim D, Hilliges O, et al. KinectFusion: real-time 3D reconstruction and interaction using a moving depth camera[C]//Proceedings of the 24th annual ACM symposium on User interface software and technology. ACM, 2011: 559-568.
2. Blanz V, Vetter T. A morphable model for the synthesis of 3D faces[C]//Siggraph. 1999, 99(1999): 187-194.
3. Fangmin L, Ke C, Xinhua L. 3D face reconstruction based on convolutional neural network[C]//2017 10th International Conference on Intelligent Computation Technology and Automation (ICICTA). IEEE, 2017: 71-74.
4. Zhu X, Lei Z, Liu X, et al. Face alignment across large poses: A 3d solution[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 146-155.
5. Ahlberg J. Candide-3-an updated parameterised face[J]. 2001.
6. <http://marctenbosch.com/news/category/mathematics/>
7. <https://www.learnopencv.com/facial-landmark-detection/>
8. Viola P, Jones M. Rapid object detection using a boosted cascade of simple features[J]. CVPR (1), 2001, 1: 511-518.
9. https://api.ai.qq.com/fcgi-bin/face/face\_faceshape
10. Wang Y, Ji X, Zhou Z, et al. Detecting faces using region-based fully convolutional networks[J]. arXiv preprint arXiv:1709.05256, 2017.
11. <https://ai.qq.com/doc/faceshape.shtml>
12. <https://www.cvssp.org/faceweb/3dmm/facemodel/>
13. <http://www.fileformat.info/format/material/>
14. <https://learnopengl.com/Getting-started/Coordinate-Systems>
15. <https://zh.wikipedia.org/wiki/%E5%BE%84%E5%90%91%E5%9F%BA%E5%87%BD%E6%95%B0>
16. Pighin F, Hecker J, Lischinski D, et al. Synthesizing realistic facial expressions from photographs[C]//ACM SIGGRAPH 2006 Courses. ACM, 2006: 19.
17. <http://vis-www.cs.umass.edu/lfw/results.html>

外文资料原文





外文资料译文

绪论

在本文中，介绍了一种用于3D人脸纹理建模的新技术。 3D人脸可以从一张或多张照片中自动生成，也可以直接通过可视化的用户界面建模。计算机可以帮助使用者解决人脸建模中的两个关键问题。首先，通过计算与内部人脸模型密集的一对一对应关系，可以自动注册新的人脸图像或新的3D人脸模型。其次，该方法调节了建模人脸的自然性，避免了具有“不太可能”外观的人脸。

从一组示例3D人脸模型开始，我们通过将示例的形状和纹理转换为向量空间表示来导出三维可变形人脸模型。可以通过原型的线性组合来建模新的人脸和人脸数据表达。从我们的示例面的统计数据派生的形状和纹理约束可以用于指导手动建模或自动匹配算法。

我们展示了单个图像的3D人脸重建及其在照片般逼真的图像处理中的应用。我们还根据复杂的参数调整人脸（的特征），例如性别，人脸丰满度或其他独特特征。

介绍

人脸的计算机辅助建模仍然需要大量的专业知识和手动控制，以避免不切实际的非人脸结果。人脸合成，人脸动画或单个人脸外观的一般变化的自动化技术的大多数限制可以描述为在不同人脸中找到相应的特征位置的问题，或者作为将真实人脸与人脸分离这样的从来不会在现实世界里发生的问题。这些问题对于所有的成像技术都是至关重要的，无论是将运动捕捉数据应用于图像还是3D人脸模型，以及大多数对于图像中的3D人脸重建技术。在一个人脸上标记的有限数量的标记特征点，例如鼻尖，眼角和脸颊上较不突出的点，必须精确地定位在另一个人脸中。手动标记的特征点的数量因应用程序而异，但通常在50到300之间。

只有正确地对齐了这些点才能够允许可接受的中间变形：从参考到新模型的运动数据的令人信服的映射，或者将3D人脸模型适配到用于“视频克隆”的2D图像。 人类的知识和经验是必要的，以补偿各个面之间的差异，并保证在不同人脸的有效（特征点）的位置分配。 目前，自动匹配技术仅可用于非常突出的特征点，例如眼角和嘴巴。

人脸建模中的第二类问题是自然人脸与非人脸部分的分离。为此，人类的知识更为关键。 许多应用涉及设计全新的，可以在现实世界中存在的自然的人脸长相，但没有“真正的”对应物。其他的应用需要根据年龄，体重的调整或仅仅强调人脸特征来操纵现有的人脸。这些任务通常需要耗费时间的手工操作以及艺术家的技能。

在本文中，我们提出了参数化人脸建模技术，这有助于解决这两个问题。首先，可以同时创建任意人脸模型来控制生成的人脸的特征的相似性。其次，系统能够计算新面孔之间的对应关系。利用3D人脸扫描得到的大型数据集的统计数据，我们建立了一个可变形人脸模型，并通过应用模式分类方法恢复关于人脸变化领域的知识。可成像的人脸模型是一种多维3D变形功能，它基于大量3D人脸扫描的线性组合。计算数据集中的平均面和主要变量模式，对变形函数施加概率分布以避免不可能的人脸。我们还通过评估人脸空间中每个属性的样本面的分布，得出人脸属性的参数描述，例如性别，独特性，鹰钩鼻或人的重量。

构建了能够生成几乎任何面部的参数化人脸模型后，对应问题变成了数学优化问题。可以通过人脸模型功能最小化新人脸与其重建图像或3D面部扫描之间的差异来登记新的人脸。我们开发了一种算法，可以自动调整模型参数，以实现目标的最佳重建，只需要最少的手动初始化。匹配程序输出的是高质量的3D人脸模型，与我们的可变形人脸模型完全一致。因此，我们的模型函数中参数化的所有面部操作都可以映射到目标人脸。在我们的模型函数中捕获的关于人脸的形状和纹理的先验知识足以即使在仅有单个图片可用时也可以合理地估计面部的完整3D形状和纹理。当将该方法应用于人的多个图像时，重建几乎达到激光扫描的质量。