



北京大学

本科生毕业论文

题目： 基于人脸识别技术的
轻量级社交平台

姓 名： 余俊峰

学 号： 1100012769

院 系： 信息科学技术学院

专 业： 计算机

研究方向： 移动互联网

导 师： 边凯归

二〇一五年五月

版权声明

任何收存和保管本论文各种版本的单位和个人，未经本论文作者同意，不得将本论文转借他人，亦不得随意复制、抄录、拍照或以任何方式传播。否则一旦引起有碍作者著作权之问题，将可能承担法律责任。

摘要

人际关系总是人们生活中非常重要的一环，人们扩展自己的交际圈的需求无所不在，在如今的互联网的时代下，移动应用成为了人们进行社交活动的重要方式。为了尽可能减少用户去扩展社交圈的障碍，令用户可以更加有效地获取新的人际关系，很多的移动互联网应用尝试用不同的方式去降低用户获取好友的难度。然而，这些现有的应用对于用户的信息利用不完全，往往不能提供令用户满意的效果。

本文提出了一种通过人脸识别的技术为主，用户的其他信息为辅助的方法来帮助用户去减少他们去扩展社交圈的难度，在本系统中，将用户的脸看做了一个非常重要的特征，来配合其他特征来进行匹配算法，为用户推荐好友。

针对以上的想法，本文实现了一种适合多移动平台，覆盖广泛的应用，并设计了不同的匹配算法，通过实验衡量其性能和用户体验。最终确定了一个比较优秀的设计成为了一个移动互联网应用，经实验证明该应用具有良好的体验，可以面向商用进一步推广。而更进一步的，由于图片流应用的火热和现在 computer version 技术的发展，可以根据人们在日常生活中所表现出的倾向进行学习和进一步的匹配，前景十分远大。

关键词：人脸识别，图片社交应用，移动设备，匹配算法

Light Social Application based on Face Recognition

Pluto She (Computer Science)

Directed by Prof. Kaigui Bian

Abstract

Personal relationship is always most important part in human's live, thus people are inclined to expand their relationship. In the internet age, mobile application become the most popular access to communicate with others. To facilitate people developing relationship, there are a myriad ways supporting by current application to find friends. However, lack of taking advantage of full of user information, the ways to find friends could not lead to a good result.

Therefore, regarding this problem, we introduce face recognition to facilitate user extending their social circle. Recommend friends to user mainly by the features of user's face, associating with other user information. Further we devise several matching friend algorithm, implement a application suited multiple mobile platform to realize it. And the user study shows that the social application based on face recognition has a good performance to extend user social circle.

Moreover, this matching algorithm could achieve more higher by the development of computer version technologies.

Keywords: Face recognition, photo sharing application, mobile application, matching algorithm

目录

第一章 引言	1
1.1 背景分析	1
1.2 方案概述	4
1.3 本文组织	5
第二章 相关工作	6
2.1 人脸识别技术	6
2.1.1 Face++	6
2.1.2 caffe	7
2.1.3 技术选用	7
2.2 移动应用	7
2.2.1 angularJS	7
2.2.2 ionic	8
2.2.3 技术选用	9
2.3 数据库调研	9
2.3.1 mongodb	9
2.3.2 数据来源	9
2.4 爬虫	10
第三章 系统设计	11
3.1 匹配算法设计	11
3.1.1 基本原理	11

3.1.2 基于人脸相似度的算法设计	11
3.1.3 机器学习算法设计	12
3.2 系统服务器架构设计	13
3.2.1 face++ SDK	14
3.2.2 数据库	14
第四章 系统实现	16
4.1 爬虫模块	16
4.2 应用端模块	18
4.2.1 状态查询及展示	18
4.2.2 信息源添加	19
4.2.3 匹配信息展示	19
第五章 实验和展望	21
5.1 系统测试	21
5.1.1 数据库用户分析	21
5.1.2 匹配算法实验	22
5.2 展望	24
本文总结	25
参考文献	26
致谢	27

第一章 引言

1.1 背景分析

随着网络和移动设备普及，人们越来越多地用移动设备来进行自己的社交活动，现在层出不穷的社交类应用一直在涌现，更为显著的是 wechat 其生态圈已经成为了中国用户一种特定的生活模式，所以移动应用在人们社交关系中起到的作用不言而喻。

并且在这个信息爆炸时代下，由于大量的信息涌到人们眼前，而不断兴起的社交媒体更是进一步将信息碎片化。不像文字一般，照片、图像作为不易切割的单位在传递信息过程中备受用户青睐。而图片类移动社交应用因为其独特性而成为了新一代移动应用的一个热点，比如国外的 instagram 就是图片分享类社交的先行者。而 flickr, snapchat，乃至于现在出现在中国的众多的移动社交应用都在充斥在人们的生活中。甚至有人断言到“分享相片是社交网络的将来”。



图 1.1: flickr



图 1.2: instagram



图 1.3: snapchat

上面的一组图展示了最近在图片分享应用中非常火热的几款应用。

其中图 1.1 是 Flickr，是雅虎旗下图片分享网站。为一家提供免费及付费数位照片储存、分享方案之线上服务，也提供网络社群服务的平台。其重要特点就是基于社会网络的人际关系的拓展与内容的组织。而随着移动应用的兴起，其在 2013 年也提出了 Android 版本，但因为其对于移动社交市场的响应滞后性，其占

到的份额已经变小。

图 1.2 是 **instagram** 是一款典型的图片型应用，而且最开始的专业的滤镜功能将一大帮摄影家囊括在了它的用户群之中，从而让摄影家有了扩展他们社交圈的方法，但之后由于用户的泛化，同样需要更进一步的好友推荐的方法。

图 1.3 是 **snapchat**，这是一款“阅后即焚”的图片型软件，由其发送图片的时效性而著名。

总结以上图片分享社交应用，与以往社交应用类似的是大多数的图片分享社交应用要么是熟人社交圈或者弱关系社交圈。而在这种关系社交类应用中，存在着用户难以找到令他们感兴趣的用户的不足。

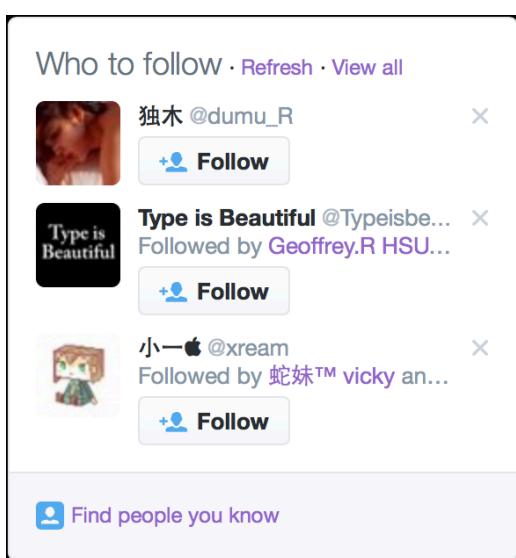


图 1.4: Twitter 推荐，
根据其已关注用户的关注关系推荐



图 1.5: weibo 推荐，
与 twitter 一样，同样由关注用户的
关系所推导

如今的社交应用都在尝试不同的方式帮助用户去寻找他们感兴趣的用户。比如微信的“摇一摇”、“漂流瓶”是一种具有随机性的推荐好友的方式，但除了一部分特定的人群，大部人的接受度并不高。而基于地点信息推荐的方式，很多社交应用都为用户设置了查看附近的人，但使用这项功能的人并不是很多。而基于用户

的兴趣，在类似于 twitter,weibo 这种弱关系社交中，根据关注人的和用户的某些特性，推荐好友的功能，这给予了用户一个很好的体验。在如今的 classification 中，deepwalk[1]，line[2]，node classification[3] 深度学习 [4] 等算法都是未来根据用户关系来分类，从而帮助用户去寻找他们感兴趣的人。

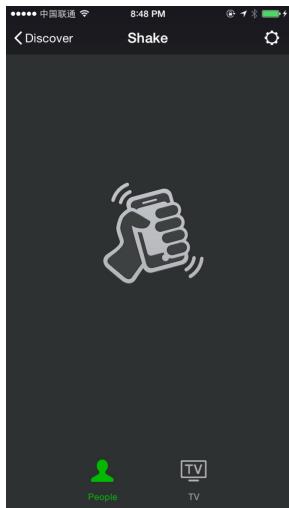


图 1.6: 摆一揺



图 1.7: 附近的人



图 1.8: 漂流瓶

上面一组图是 wechat 所做的各类关于增加用户好友关系的探寻。

图 1.5 是著名的“摇一摇”，他所做到的是利用即时性，和随机性为同时想扩展社交圈子的用户提供一个通道，但在现在中国已经成为一些人发展一些私密关系的工具。

图 1.6 展示的是 wechat 的“附近的人”，通过位置关系将可能成为朋友的人绑在一起。

图 1.7 是“漂流瓶”，与“摇一摇”不同的是它是非即时的社交方式，同样是以想扩充社交圈这种意愿为纽带。

虽然现在的方式各种各样，但无可否认的是至今不存在一种很好的方式提供给用户去拓展他们的社交圈子。

这就产生了本研究的一个最初的动机：

能否通过一种图片交流的方式提供图片类应用的用户去认识新朋友的渠道？

而最近由于人脸识别技术的火热，市面上不断在涌现着与人脸识别关系密切的应用，比如 face++ 与阿里巴巴在支付宝上联合推出的“刷脸”支付功能。

这些给予了笔者以灵感，笔者拟应用人脸的特征去提供一种独特的方式给用户去拓展他们的社交圈子，结合了传统的“夫妻脸”的概念，其中最基础的方式就是以人脸相似度去实现一个应用。之后通过实验的不断调整，加入其他特征去调整最后的好友推荐算法，得出最后的适合用户的匹配好友算法。

1.2 方案概述

本文提出基于人脸识别技术的图片社交平台，利用人脸识别的技术，不但能够提高人脸识别技术的通用性，并能提高用户对于社交应用的体验。根据不同的用户脸部特征，个人信息，本文对于基于用户特征值的匹配算法的可行性进行了系统的研究。基于已有的特征值，推荐给用户各项指标都符合其标准的好友，使得可以为用户提供一个良好的方式去扩展他们的社交圈。

因此本文是对于图片社交应用的一种基于人脸识别方式的好友扩展方式的研究，基本的工作可以总结为如下：

1. 了解现有的社交应用给予用户的扩展好友的方式，调研相关的人脸识别的技术和算法，提出一种基于人脸特征的好友推荐匹配算法
2. 基于 face++ 提供的 api，完善在 node.js 平台上 sdk 的支持，并调研和准备应用的数据，存储到数据库中。
3. 基于 angularJS+Ionic 框架，开发出一款能够应用在三个移动手机平台上的应用。该应用不但能有效的解决用户好友来源的问题，并且可以在好友推荐中，为用户推荐符合用户的好友。同时，根据用户的反馈能够及时的调整对应的匹配算法。
4. 对于基于不同算法的好友推荐算法，进行实际的用户效果的测试，通过实验数据的比较取得一种较高满意度的推荐算法。实验结果表明本应用的好友推荐算法具有较高的实用价值和扩展性。

1.3 本文组织

第一章包括背景和对研究的分析，及本文组织。第二章介绍论文相关技术和工作。第三章详细给出系统的设计。第四章给出系统的实现。第五章进行实验和展望。最后进行了总结。

第二章 相关工作

2.1 人脸识别技术

人脸识别 [5] 是一项热门的计算机技术研究领域，它属于生物特征识别技术，是对生物体（一般特指人）本身的生物特征来区分生物体个体。生物特征识别技术所研究的生物特征包括脸、指纹、手掌纹、虹膜、视网膜、声音（语音）、体形、个人习惯（例如敲击键盘的力度和频率、签字）等，相应的识别技术就有人脸识别、指纹识别、掌纹识别、虹膜识别、视网膜识别、语音识别（用语音识别可以进行身份识别，也可以进行语音内容的识别，只有前者属于生物特征识别技术）、体形识别、键盘敲击识别、签字识别等。

2.1.1 Face++

Face++ 是新一代云端视觉服务平台，提供一整套世界领先的人脸检测，人脸识别，面部分析的视觉技术服务。Face++ 旨在提供简单易用，功能强大，平台通用的视觉服务，让广大的 Web 及移动开发者可以轻松使用最前沿的计算机视觉技术，从而搭建个性化的视觉应用。

Face++ 同时提供云端 REST API 以及本地 API（涵盖 Android, iOS, Linux, Windows, Mac OS），并且提供定制化及企业级视觉服务。通过 Face++，可以轻松搭建云端身份认证，用户兴趣挖掘，移动体感交互，社交娱乐分享等多类型应用。Face++ 提供了相应的 http 请求的接口，并提供了 python, java, c++ 的 API。

在本项目中，作者对于 Face++ 对于 nodejs 的 API 进行了补全，能够很好的支持 nodejs 对于 face++ 的接口应用，方便之后的 nodejs 开发者能够有效的利用

该项技术。

2.1.2 caffe

Caffe (<http://caffe.berkeleyvision.org/>) 是一个清晰而高效的深度学习框架，其作者是博士毕业于 UC Berkeley 的贾扬清 (<http://daggerfs.com/>)，他目前在 Google 工作。Caffe 是纯粹的 C++/CUDA 架构，支持命令行、Python 和 MATLAB 接口，可以在 CPU 和 GPU 直接无缝切换。caffe 在图像处理方面有着自己独特的效果。

2.1.3 技术选用

通过比较了这两项技术之后，由于 Face++ 是专门做人脸识别的技术，他的各项评测也遥遥领先，所以在本文准备使用 face++ 的 API 来实现应用。

2.2 移动应用

移动应用 (Mobile application) 安装和运行在移动设备上。它随着智能手机的推出，由于其移动性和娱乐性，在近年来达到了全新的巅峰。目前有三大主流的移动应用平台，分别为 IOS, Android 和 Windows Phone。据三大平台统计，从 2010 年到 2013 年，移动应用的数目增加了 100 多万个。因此，移动应用是目前软件开发的一种最流行的模式，移动设备庞大的用户量和较小的开发经费也吸引了越来越多的自由开发者加入了移动应用的开发者行列之中。

2.2.1 angularJS

angularJS 是一款开源 JavaScript 函式库，由 Google 维护，用来协助单一页面应用程式运行的。它的目标是透过 MVC 模式 (MVC) 功能增强基于浏览器的应用，使开发和测试变得更加容易。函式库读取包含附加自定义 (标签属性) 的 HTML，遵从这些自定义属性中的指令，并将页面中的输入或输出与由 JavaScript 变量表示的模型绑定起来。这些 JavaScript 变量的值可以手工设置，或者从静态

或动态 JSON 资源中获取。AngularJS 是建立在这样的信念上的：即声明式编程应该用于构建用户界面以及编写软件构建，而指令式编程非常适合来表示业务逻辑。框架采用并扩展了传统 HTML，通过双向的数据绑定来适应动态内容，双向的数据绑定允许模型和视图之间的自动同步。因此，AngularJS 使得对 DOM 的操作不再重要并提升了可测试性。

设计目标：

1. 将应用逻辑与对 DOM 的操作解耦。提高代码的可测试性。
2. 将应用程序的测试看的跟应用程序的编写一样重要。代码的构成方式对测试的难度有巨大的影响。
3. 将应用程序的客户端与服务器端解耦。这允许客户端和服务器端的开发可以齐头并进，并且让双方的复用成为可能。

指导开发者完成构建应用程序的整个历程：从用户界面的设计，到编写业务逻辑，再到测试。

2.2.2 ionic

Ionic[6] 提供了一个免费且开源的移动优化 HTML、CSS 和 JS 组件库，来构建高交互性应用。基于 Sass 构建和 AngularJS 优化，并形成了一个强大的 HTML5 应用程序开发框架，可以帮助使用 Web 技术，比如 HTML、CSS 和 Javascript 构建接近原生体验的移动应用程序。

Ionic 主要关注外观和体验，以及和你的应用程序的 UI 交互，特别适合用于基于 Hybird 模式的 HTML5 移动应用程序开发。且它是一个轻量的手机 UI 库，具有速度快，界面现代化、美观等特点。为了解决其他一些 UI 库在手机上运行缓慢的问题，它直接放弃了 IOS6 和 Android4.1 以下的版本支持，来获取更好的使用体验。

Ionic 是现在 GitHub 上的最火的开源项目之一，具有超过 16,000 星及以上创建 600000 Ionic app。Ionic 遵循视图控制模式，通俗的理解和 Cocoa 触摸框架相似。在视图控制模式中，我们将界面的不同部分分为子视图或包含其他视图的子视图控制器。然后视图控制器“驱动”内部视图来提供交互和 UI 功能。一个很好的例子

就是标签栏（Tab Bar）视图控制器处理点击标签栏在一系列可视化面板间切换。

2.2.3 技术选用

相较于原生的 Android, IOS, Windows Phone 应用，于 angularJS+ionic 的框架实现的应用在三大平台都能够使用，存在了很强大的适用性，能够针对更多的用户，并且针对于现有先进的移动设备有了一个较好的实用性。

2.3 数据库调研

为了收集可用的用户信息，建立一个叫完整的图片数据库，我们需要调研一个基础的能够供我们对于用户的实验数据库。

2.3.1 mongodb

Mongo DB 是目前在 IT 行业非常流行的一种非关系型数据库 (NoSql)，其灵活的数据存储方式备受当前 IT 从业人员的青睐。Mongo DB 很好的实现了面向对象的思想 (OO 思想)，在 Mongo DB 中每一条记录都是一个 Document 对象。

Mongo DB 最大的优势在于所有的数据持久操作都无需开发人员手动编写 SQL 语句，直接调用方法就可以轻松的实现 CRUD 操作。MongoDB 的文档模型自由灵活，可以让你在开发过程中畅顺无比。对于大数据量、高并发、弱事务的互联网应用，MongoDB 可以应对自如。MongoDB 内置的水平扩展机制提供了从百万到十亿级别的数据量处理能力，完全可以满足 Web2.0 和移动互联网的数据存储需求，其开箱即用的特性也大大降低了中小型网站的运维成本。

2.3.2 数据来源

根据本文所要实现的应用的特性，是基于人脸的社交应用，而无疑最符合这个性质的社交应用就是婚恋市场的社交应用。而也只有婚恋的用户才会在大概率上将他们的真实人物照片提交到应用上，为了模拟这个应用的特性，本文实验的

数据库的来源也需要从类似的网站上抓取，排除掉国外的婚恋网站，作为国内的比较流行的婚恋网站有，百合网，珍爱网，世纪佳缘。

通过比较这几个网站的数据，发现百合网的会员资料的开放程度不及世纪佳缘，珍爱网。而世纪佳缘的用户活跃度和注册数远远优于珍爱网。所以本文拟用世纪佳缘的数据作为本文应用的数据库来源。

2.4 爬虫

Beatifulsoup

Beautiful Soup 提供一些简单的、python 式的函数用来处理导航、搜索、修改分析树等功能。它是一个工具箱，通过解析文档为用户提供需要抓取的数据，因为简单，所以不需要多少代码就可以写出一个完整的应用程序。**Beautiful Soup** 自动将输入文档转换为 Unicode 编码，输出文档转换为 utf-8 编码。你不需要考虑编码方式，除非文档没有指定一个编码方式，这时，**Beautiful Soup** 就不能自动识别编码方式了。然后，你仅仅需要说明一下原始编码方式就可以了。**Beautiful Soup** 已成为和 lxml、html6lib 一样出色的 python 解释器，为用户灵活地提供不同的解析策略或强劲的速度。

第三章 系统设计

3.1 匹配算法设计

3.1.1 基本原理

本文对于匹配算法的设计，主要的思想是探究出一种基于用户特征信息来推荐与用户志趣相投的人，所以其中必须要考虑到是现实用户对匹配算法实际效果的满意程度。之后对于不同的意见，匹配算法再不断进行对应的调整，从而得到一个相对最优的算法。故此，本文准备设计不同的匹配算法。最后对于不同的匹配算法进行实验，选取出实验效果最好的来当作最终实施在应用上的算法。

3.1.2 基于人脸相似度的算法设计

基于人脸相似度的算法设计的原理是提取用户提交的照片所提供的人脸特征，再将这一些特征与数据库中的图片的人脸特征结合起来，进行匹配。

Face++ 的人脸技术库所提供的人脸特征值包括

1. 左眼和右眼在相应脸上的相对坐标
2. 相应人脸的鼻尖在脸上的相对坐标
3. 相应人脸的左侧嘴角和右侧嘴角在脸上的相对坐标
4. 人脸的微笑程度分析结果，value 的值为 0 – 100 的实数，越大表示微笑程度越高
5. 包含眼镜佩戴分析结果，value 的值为 None/Dark/Normal，confidence 表示置信度

假设一个用户提供的图片所显示的人脸为 P , 特征的维度为 n , 则用户的人脸可被定义为

$$P = \{x_1, x_2, \dots, x_n\}$$

设 Face++ 提供的 API 的比较两张脸相似的算法为 F , 对于每个特征的损失函数为 L_i , 比较的人脸设为 X, Y , 则相似算法可表示为下式

$$F = \sum_{i=1}^n L_i(x_i, y_i)$$

设本文的数据库的人脸集合为 T , $T = T_1, T_2, \dots$, 则我们能给出的基于人脸相似度的匹配结果的算法的公式如下

$$G = \underset{G \in T}{\operatorname{argmax}} F(P, G)$$

3.1.3 机器学习算法设计

由于基于人脸相似度的算法仅仅考虑了两个人脸特征的相似度, 而基于一些美观上的经验, 我们可以用过人脸的特征对于不同阶段不同年龄的人的审美进行机器学习, 从而得到一个适合的算法。

但由于用户的不确定性, 我们针对于用户的猜测可能导致负相关, 因此这里的变量越少越好。由于现代人对于眼角位置的重视程度高于其他位置, 而他们审美习惯基于自己的眼角相对位置。所以在这里本文使用了眼角位置作为我们的机器学习算法所学习的变量。而这一个新的学习出的特征变量, 我们将此当作一个新的权重加到之前基于人脸相似度的匹配算法上, 从而达到优化匹配的目的, 设 r 为该特征变量, 则更新的匹配结果如下所示

$$G = \underset{G \in T}{\operatorname{argmax}} \lambda * r + (1 - \lambda) * F(P, G)$$

本文将该机器学习问题看做一个分类问题(或一个监督学习问题), 即根据用户本身图片的特征值和过去用户感兴趣的图片信息的历史特征值来对之后用户新

提交照片的感兴趣的人脸分类。

识别用户感兴趣的用户问题的目标是确定用户新提交的照片对应在数据库中的人脸的标签。在以往用户点击的感兴趣的图片中，我们收集了一组过去的用户图片以及配对感兴趣的图片，用 D 表示这个集合。每个用例 $d \in D$ 能被八个变量描述，即用户的左右眼角的相对坐标，和感兴趣的图片的人脸的左右眼角的相对坐标。

给定 D 和两张图片，一张图片为用户提供，一张图片为数据库提供，TapLock 系统需要确定密码的这一张数据库提供图片的标签是用户感兴趣与否的。这里使用一个简化的 k-最近邻分类 (k-Nearest Neighbor Classification, kNN)[7][8][9] 算法来解决这个问题。令 t 表示这两张图片的特征，去匹配在集合 D 中最近的向量。

1. 计算 t 和每个训练用例 $d_i \in D$ 的距离，这里简化为笛卡尔距离
2. 选出距离值离 t 最近的训练用例 $d_1 \in D$ 。
3. 同样地，选出距离值离 t 第二近的训练用例 $d_2 \in D$ 。
4. 如果两者标签相同，将选中的两个训练用例 d_1 和 d_2 的标签赋给 t 。
5. 如果两者标签不相同，选出距离值离 t 第三近的训练用例 $d_3 \in D$ ，把 d_3 的标签赋给 t 。

3.2 系统服务器架构设计

本文设计了具有如下性质的社交应用好友匹配系统。系统的服务端包括了应用 Face++API 的 SDK 程序以及连接数据库持久化存储的媒介，为用户提供对应的服务接口，并且为应用维护者提供管理和维护其数据库的方法，为用户提供查询信息以及应用匹配算法的方法

图 3.1 描述了具体的服务器的逻辑，数据库端因为应用端实现了用户系统和人脸图片的 post 系统，所以在数据库这边的 API 上需要提供对应的连接到数据库的接口，从而使得我们能够在应用端展示出我们在数据库端的数据信息。

而 Face++ API 部分经过我们与应用端，匹配算法的协调，舍弃了一些原本需要的功能，比如 Face++ 所提供的 Group 功能，而提供了与该系统相适应的 4 个请求接口。

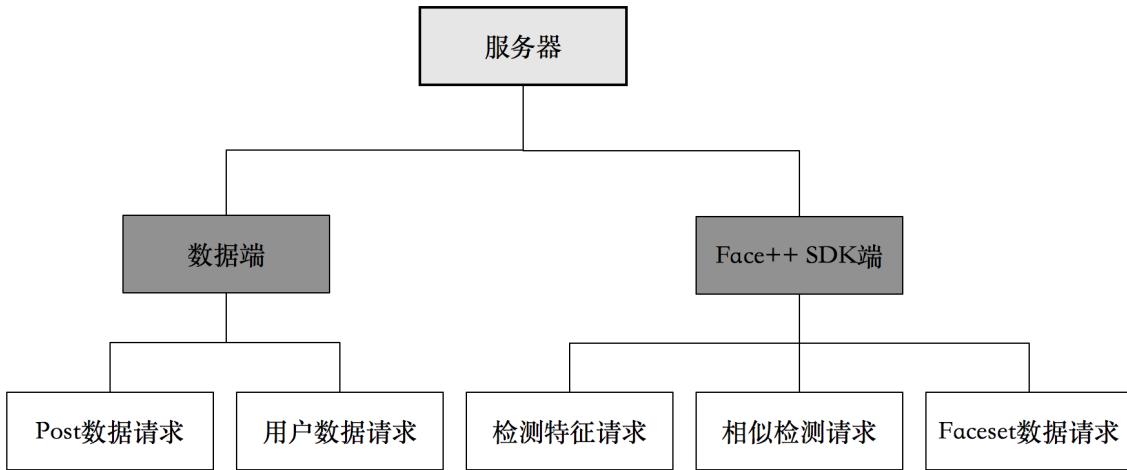


图 3.1: 服务器逻辑

3.2.1 face++ SDK

由于 Face++ 并没有提供对应的 SDK 给 node.js，所以本研究需要为 face++ 的 http 请求的 api 实现一个 SDK 给 node.js。

Face++ 的 API 可详见 [10]

如何实现 Face++API 在 nodejs 上具体的 SDK，因为和本文的研究关系不大，在此略过。

通过之前的描述，我们具体应用到的 Face++ 的部分仅为 face++ API 的关于 Detect, Compare, Face 和 FaceSet 数据请求这方面的功能。

因此在实际的服务器端，对于我们完成的 face++ SDK 的这四个接口给以在服务器端的接口，使得我们的应用端在提交人脸图片进行匹配算法的时候可以利用 Face++ 的人脸识别技术库完成我们需要的匹配算法模块。

3.2.2 数据库

为了完善我们的数据库的逻辑，因为 NOSQL 无法方便的进行系统的说明，故此本文对于数据库进行了用关系行数据库的描述对于 NOSQL 进行了比对。

图 3.2 表现了关系型数据库需要的表单，表现了三种需要表现的数据存储在数据库中，分别是用户信息，人脸信息，与对应的匹配信息。

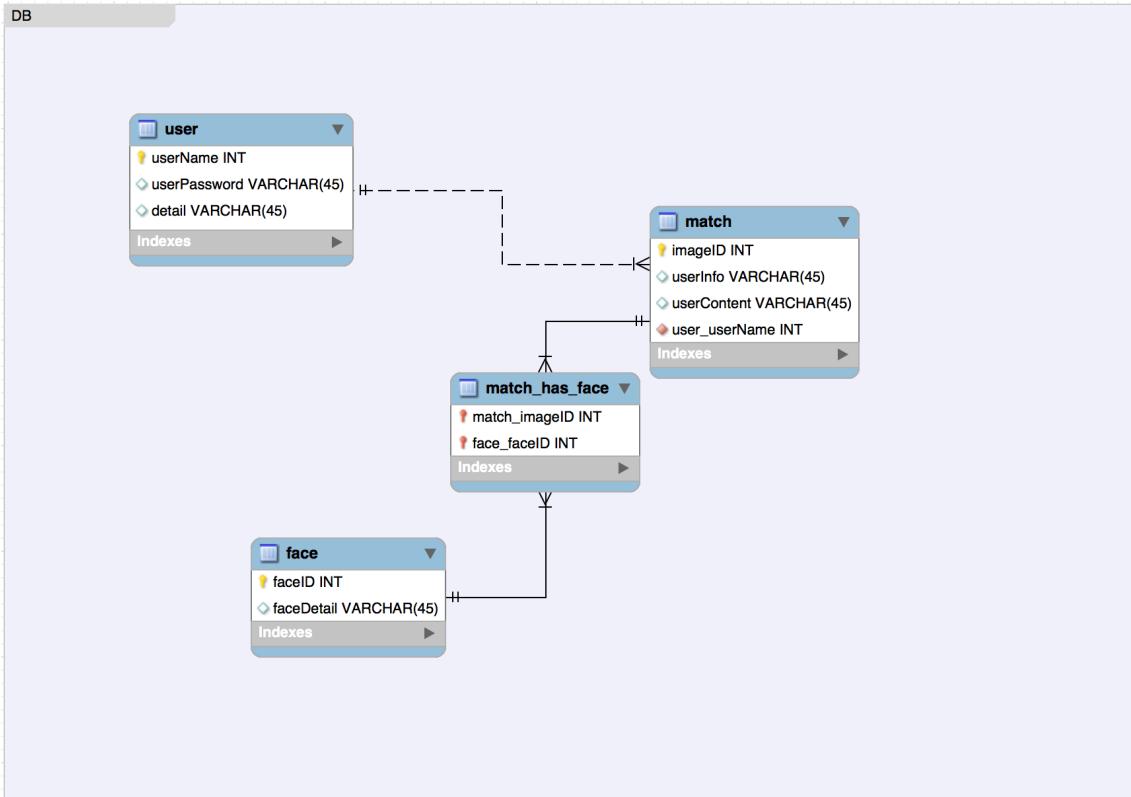


图 3.2: 关系型数据库表单抽象

经过我们数据库的逻辑和实际请求进行比对，我们发现服务器只需要实现 `post` 数据，用户获取数据以及更改的接口即可。

第四章 系统实现

4.1 爬虫模块

基于在本文第二部分中的数据来源和爬虫的分析，采用 beautifulSoup 技术来进行爬取数据。

The screenshot shows a user profile page from JiaYuan.com. At the top, there's a navigation bar with links like '首页', '我的佳缘', '俱乐部', '搜索', '新会员', '聊天', '游戏', '晒幸福', '红娘一对一', and '爱真心'. Below the navigation, there are tabs for '他的资料' (selected), '他的照片(6)', '他的礼物(1)', and '他的博客(2)'. A '下一个有缘人' button is also present. The main content area features a large profile picture of a man with glasses. To the right of the photo, the user's ID is listed as 'ID:39388193' and their status as '普通会员'. A '15 帅力值' badge is displayed. Below this, there's a brief bio: '29岁, 未婚, 来自浙江温州' with a '显示地图' link. Further down, detailed information includes education ('学历: 大专'), height ('身高: 175cm'), occupation ('购车: 登录后可见'), salary ('月薪: 登录后可见'), weight ('体重: -'), zodiac ('星座: 巨蟹座'), ethnicity ('民族: 汉族'), and blood type ('血型: -'). On the right side of the profile, there's a purple banner for '心电感应' with a '立即参与' button. At the bottom of the profile section, there are buttons for '发信', '打招呼', '送礼物', and '加关注'. To the right of the profile, there's a sidebar with a '喜欢他就发特快专递>>' button and a '注册世纪佳缘' button. A small note at the bottom left of the profile area says: '我决定为你付出自己的一切, 直到相爱的那一刻到来, 我的勇气和信仰, 是因为我知道你就是我一生在寻找的那个'.

图 4.1: 世纪佳缘用户信息页面

通过前期充分对于世纪佳缘网站的调研，对于用户信息的分析，针对用户数据的发掘，本文找到了一个基于用户 id 以及对应用户数据联系的网页，从图 4.1

可以看出，此网页 url 可以连接到用户 id 具体的用户信息，从而我们可以针对指定的用户进行必要的信息的爬取。

但在实际操作中，一些用户的图片和信息设置为成权限可见之外，我们因此在程序中需要排除掉此类非法数据。除此之外其他的数据对于爬虫都是友好的，并且在一定程度上提供了丰富的用户信息，以便于我们进行匹配算法的优化。

最后我们的爬虫脚本对于抓取到的数据信息，存储到到本文的用户人脸数据库之中。

我们的爬虫的算法可以抽象成如下的伪代码。

CRAWLER

```
1  for  $j \leftarrow$  to the Amount of extrace
2      do
3          URL  $\leftarrow$  crawler url
4          Data  $\leftarrow$  Raw data from URL
5          for  $k \leftarrow$  Rule of Elements to form Data
6              do
7                  info/gets information from Data regulated by k
8                  Insert Info into the sorted sequence Request.
9          Connect database submit Request to database
```

该算法大致的思路如下：

- 确定爬取 URL 的个数
- 循环，对于每个 URL，从网页端爬取原始信息。
- 对于每一个需求的信息的特征，进行不同的规则选取。
- 对于特定的规则，在原始信息中提取特征信息
- 将特征信息存入临时的表中
- 连接数据库，将临时数据提交至数据库

4.2 应用端模块

该应用的实现由三步组成: 状态查询, 信息源添加, 和匹配反馈。

4.2.1 状态查询及展示

为了满足用户自己查询感兴趣好友的需求, 应用需要提供用户在使用应用程序查看他人已经存在的 Post 信息。并为了便携性和可用性, 需完成进行对于用户 Post 信息的一个预览, 以及对于具体 Post 信息的查询。

从而可以满足用户的需求, 去查看其他用户 Post 具体的信息, 并可以支持用户随时对于需要的信息的刷新, 以及链接用户更进一步的个人主页的途径。



图 4.2: 主界面



图 4.3: 详细信息



图 4.4: 刷新

进入应用, 我们首先进入了主界面 (图 4.2 所示), 在这里我们可以看到用户 Post 的缩略图, 既省略了用户的 Post 详细信息, 可以达到快速浏览的目点, 又为用户提供了一个可以找到自己兴趣点的信息。

如果对一个用户感兴趣, 点击即可转到详细信息界面 (图 4.3 所示), 显示了用户 Post 的详细信息。

为了及时更新 Post，主界面补充了刷新功能（图 4.4 所示），可以对主界面的 Post 进行刷新。

4.2.2 信息源添加

用户关于自己信息的发布通常希望其发布的信息具有多种多样的形式，考虑到移动设备较难编辑和显示富文本信息，应用用程序被实现为能够编辑、存储和显示具有多种类型的字段，包括个人姓名、个人介绍、URL 和提交的匹配图片

并且基于用户照片来源的不同，分为从用户的图片库中采集和即时照相两种不同图片来源方式。并在此基础上实现了图片预览的功能。图 4.5 和 4.6 展示了这

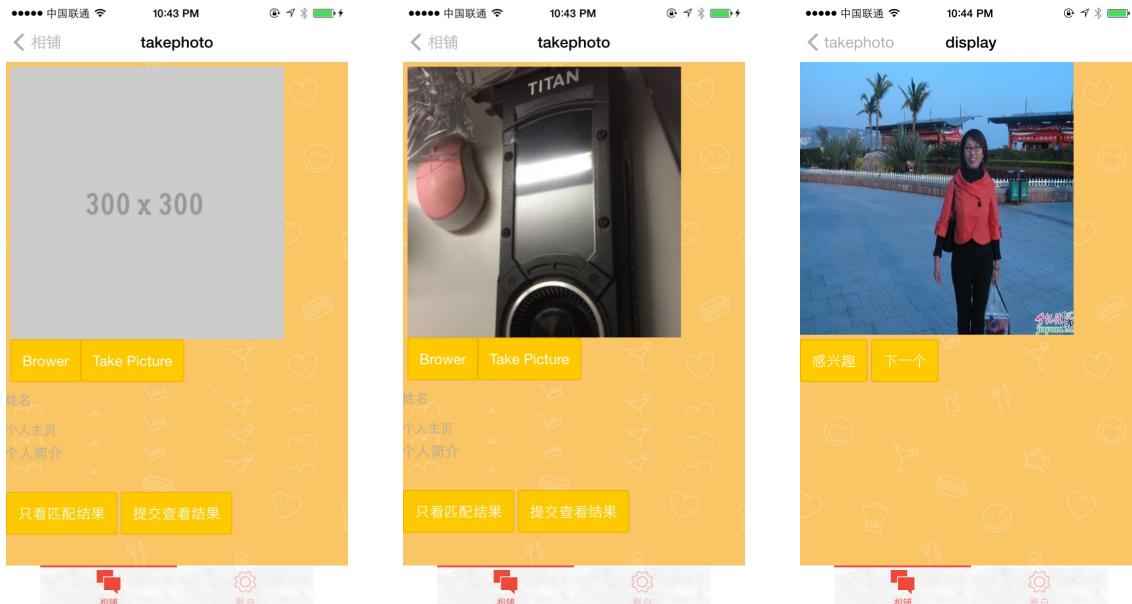


图 4.5: 提交界面

图 4.6: 预览提交的图片

图 4.7: 展示界面

两项功能，提供用户描述自己 Post 信息的通道，并使得用户能够通过不同的方式提供自己的人脸图片，并预览自己提交的图片。

4.2.3 匹配信息展示

由于服务端有客户端提交的图片以及文本信息直接调用对应匹配算法的接口，因此，应用端在用户在提交了自己的信息之后可以直接调用了此接口，服务端便

具有了足够的信息以匹配出对应的好友。

在服务端返回给客户端匹配出的结果后，客户端将详细信息显示在用用户界面上（图 4.7 所示）。这里具有具有多条匹配结果，客户端将按照匹配程度顺序将结果显示。当用户对该用户感兴趣时即点击“感兴趣”按钮进入查询详细信息。

第五章 实验和展望

5.1 系统测试

为了分析和评估本论文提出的基于人脸识别的匹配算法的有效性和可用性，我们使用了该算法编写的基于婚恋数据的匹配好友的应用进行了实验测试。本实验主要基于不同算法的匹配系统进行了分析评估。

本实验对于两个方面进行了分析和评估：

1. 数据库用户分析
2. 测试用户对于不同匹配算法的满意度

5.1.1 数据库用户分析

由于数据库的特性，对于数据库的用户进行分析是非常具有必要性的，具有相同属性的人成为好友的概率往往会非常的高。但由于本文的数据库的内容的一些缺失性，在世纪佳缘上的很多信息都不是必须填写的信息，所以本文只对于数据库的年龄属性这一必要的属性进行了分析。

通过数据库的采取的 3000 条数据进行分析，对于数据库的用户数据进行了统计。测试用户和数据库的用户的年龄分布如下图

通过图 5.1，我们可以看出在数据库中的用户的年龄都偏大，超过半数的用户都超过了 30 岁，而测试用户却相反的半数以上都为 30 岁以下的青年人。带来这种差异的原因主要是数据库数据来源的特性，由于是婚恋网站，所以大多数的用户年龄都偏大，以及本文所邀请的用户的局限性主要限定在大学周围。

由于年龄段的不同，用户有着不同的兴趣和审美观，而年龄差异过大的也并

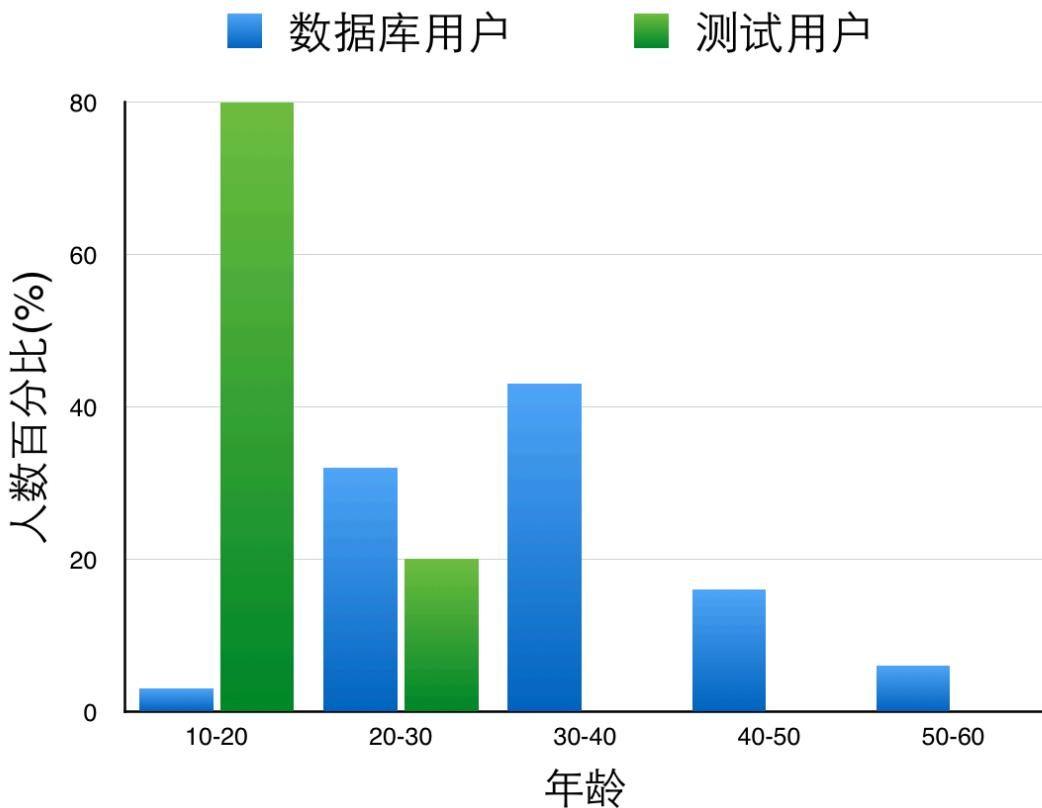


图 5.1: 年龄分布

不适合成为一个社交圈子里的人。这种差异给匹配算法带来了一些不确定性。

5.1.2 匹配算法实验

5.1.2.1 实验设计

本文对于匹配算法的评判的标准为用户给出对于匹配结果的满意程度，这里的满意程度鉴于用户密度和数据粒度的关系，设成了 1 ~ 10 十个不同的分数。

评选的方式为用户提交自己的照片，通过 10 张对于用户照片给出的匹配结果，用户统计一下自己对于这 10 个结果的满意程度，从而对于一个匹配算法给出一个最终的满意程度。

5.1.2.2 实验结果

总共 20 人参加了该实验的测试，该实验的满意度的分数为 10 级，如设计所述，测试用户对于一个匹配测试算法给出的 10 个推荐好友进行评分，作为最终的检测结果。而对于不同的匹配算法的满意度的结果如图 5.2 所示。从表中可以看

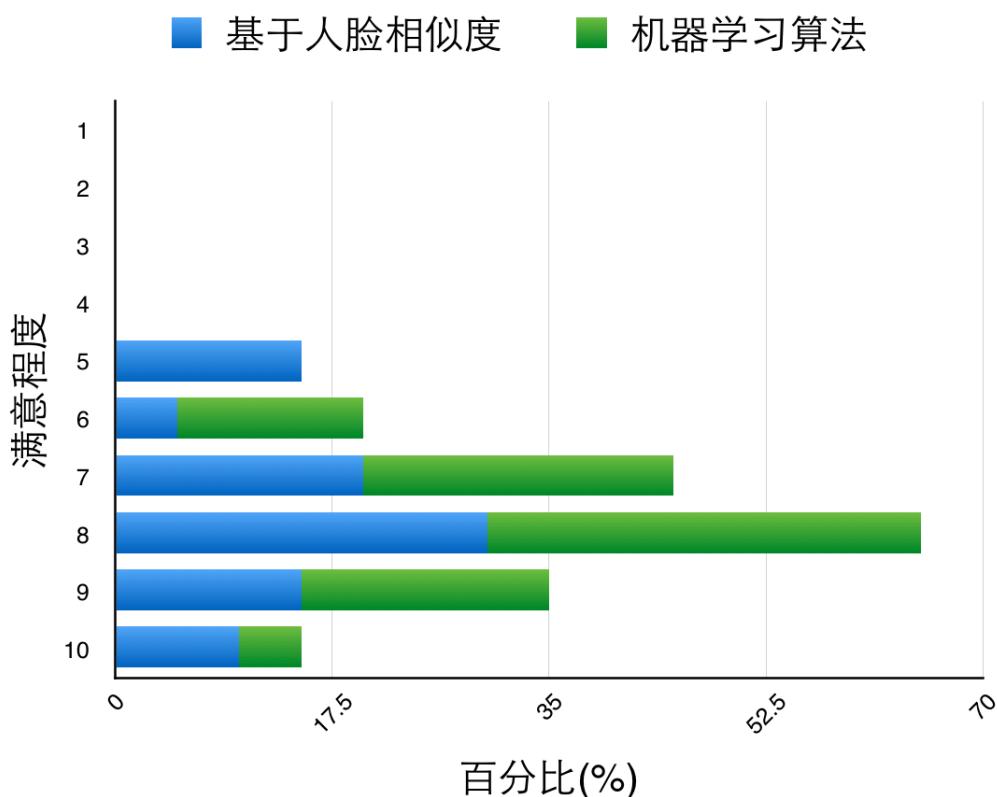


图 5.2: 满意度统计结果

出由于加入了更多的特征到了匹配算法中，机器学习算法的用户的满意度对于原有的匹配算法的效果有了一定的提升，而原有的匹配的满意程度也不低，说明了经过人脸识别技术的好友匹配算法有了一个比较好的效果。但由于个人主观的问题，还需要进一步时间去检验该不同匹配算法的优劣。但从整体结果来看，还是比较可喜的。

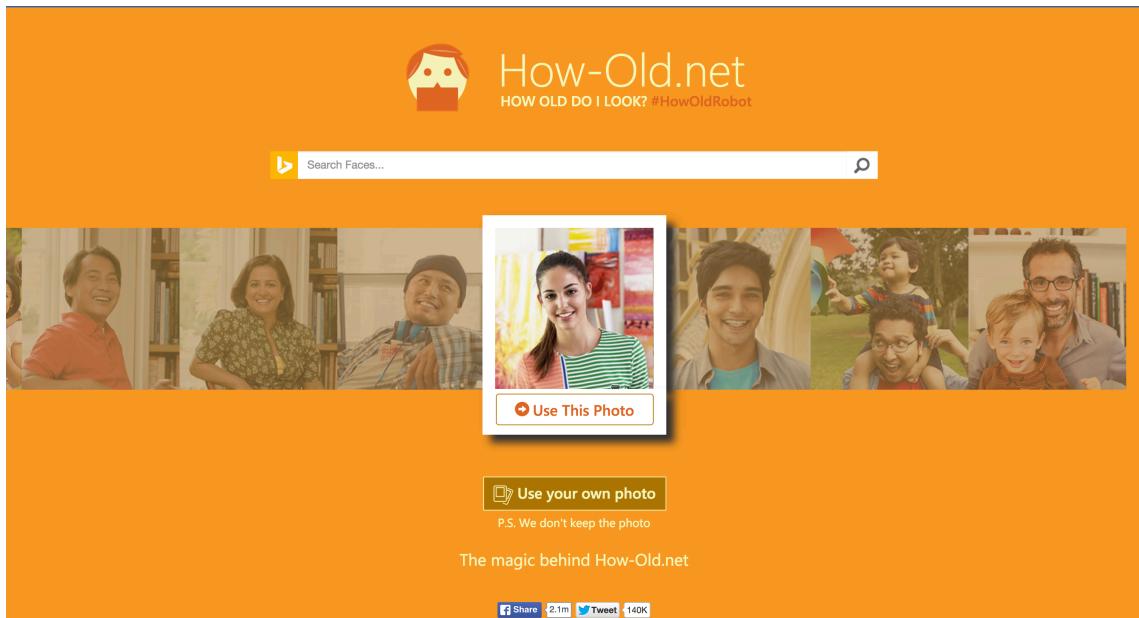


图 5.3: How old from Microsoft

5.2 展望

本文的匹配算法经过了一定量用户的检验，反映的效果比较良好，但在实验中对于用户的分析有一个缺点，是由于用户年龄的不协调性可能导致匹配到的结果效果并不好，但随着图片技术的发展（例如 Microsoft 的 How old，如图 5.2 所示，可以大致检测出人的年龄），使得我们可以在之后的工作中，为用户添加上这一特征值，从而进一步提供匹配算法的准确性，降低这种差异给我们算法带来的不确定性。

而更进一步的由于图片应用的火热和扩展性，可以不断扩充用户，并且不断通过更多的用户提交的信息的特征提取出来，来优化我们的匹配算法，达到更好的效果。

本文总结

本文提出了一种新型的图片社交平台，通过人脸识别技术为主导的好友推荐匹配来有效的扩展用户的社交圈以及提升用户活跃的程度。在本应用中用户可以通过他们的信息特征来寻找与他们兴趣相符的好友。

本文实现了基于 angularJS + ionic 的客户端应用程序，以及以世纪佳缘为数据源导入的数据库，以 Face++API 实现的 nodejs 的 SDK，以及对应的服务器程序。目前的实验表明本文的匹配算法具有良好的用户体验，能够有效的为用户提供适合用户特征的好友，具有良好的可扩展性和实用性。

但不可置否的是实验数据具有一定的随机性，且得到的数据较为主观，因此在之后为了更好的去探索人脸识别技术在图片社交平台上的应用，需要在之后考虑更广泛的实验，并且将此应用发布并推广出去获得更多的用户的反馈进一步来精确原算法的阈值，获得更多用户的特征数据来提升算法。

而在图片识别等技术的飞速发展的基础上，应用能够更多的通过图片来收集用户的信息，将用户的日常信息不断分类，从而更为精确的为用户匹配好友。我们可以期待图片社交应用更远大的未来。

参考文献

- [1] R. A.-R. B. Perozzi and S. Skiena. *Deepwalk: Online learning of social representations*. [IN PROCEEDINGS OF THE 20TH ACM SIGKDD INTERNATIONAL CONFERENCE ON KNOWLEDGE DISCOVERY AND DATA MINING, PAGES 701710. ACM], **2014**.
- [2] J. Tang *et al.* *LINE: Large-scale Information Network Embedding* [TO APPEAR IN WWW'15.], **2015**.
- [3] G. C. S. Bhagat and S. Muthukrishnan. *Node classification in social networks*. [IN SOCIAL NETWORK DATA ANALYTICS, PAGES 115148.], **2011**.
- [4] L. Deng and D. Yu. *Deep Learning*. Now Publishers Incorporated, **2014**.
- [5] R. Brunelli and T. Poggio. *Face recognition: Features versus templates*. [IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE], **1993**.
- [6] Ionic framework 文档. <http://ionicframework.com/docs/>.
- [7] D. Lin. *An Information-Theoretic Definition of Similarity*. [PROC. MACHINE LEARNING '98, PP. 296304], **1998**.
- [8] C. M. Bishop. *Neural Networks for Pattern Recognition*. **1995**.
- [9] M. Kearns and U. Vazirani. *Computational Learning Theory*. **1994**.
- [10] Face++API 文档. <http://www.faceplusplus.com.cn/api-overview/>.

致谢

感谢我的母校北京大学，燕园浓厚的学术氛围，培养了一代代社会精英，祖国栋梁。与良师益友交游，谈笑有鸿儒，往来无白丁，这里的寸寸土地，都充满了智慧与知识。感谢所有教授过我各类课程的老师。他们丰富的学术积淀和睿智博学的个人风采都让我崇敬仰慕。感谢北京大学网络与信息系统研究所移动网络组为我提供的良好的实验环境。感谢边凯归老师和严伟老师，他们在我的工具实现，实验完成和论文写作方面给予了我充分的指导和鼓励。在进入移动网络组两年的时间里，我切身感受到边老师与严老师亲切的师长风采与迷人的学术魅力，每一次和老师的交流，都能得到许多启迪和灵感，这都是我人生中最宝贵的财富。感谢实验室和与我同级的同学还有朋友们：毛景树，薛易清等等，他们在我的论文写作中，为我提供了充分的帮助，使我克服了许多困难，少走了许多弯路。还有很多参与了我毕设实验的亲人朋友还有同学们，正是有了他们对我的帮助，使我得以顺利完成本论文。感谢我的父母，无论寒暑春秋都给我最无私的支持和鼓励。感谢所有关心我的亲人和朋友以及所有给与我帮助的人，我的每一份收获都离不开你们。

北京大学学位论文原创性声明和使用授权说明

原创性声明

本人郑重声明：所呈交的学位论文，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，本论文不含任何其他个人或集体已经发表或撰写过的作品或成果。对本文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。本声明的法律结果由本人承担。

论文作者签名： 日期： 年 月 日

学位论文使用授权说明

(必须装订在提交学校图书馆的印刷本)

本人完全了解北京大学关于收集、保存、使用学位论文的规定，即：

- 按照学校要求提交学位论文的印刷本和电子版本；
- 学校有权保存学位论文的印刷本和电子版，并提供目录检索与阅览服务，在校园网上提供服务；
- 学校可以采用影印、缩印、数字化或其它复制手段保存论文；
- 因某种特殊原因需要延迟发布学位论文电子版，授权学校在一年/两年/三年以后在校园网上全文发布。

(保密论文在解密后遵守此规定)

论文作者签名： 导师签名： 日期： 年 月 日