

论文相关素材归档文档

基于眼动数据的立体图像质量评价论文文档

归 档 人_____马宝岩_____

学 号_____1130349147_____

导 师_____周军老师_____

专 业_____电子与通信工程专业_____

归档日期_____2015 年 12 月 20 日_____

第一章 文档内容

这是马宝岩同学在攻读上海交通大学硕士期间关于眼动实验设计、数据处理以及特征提取、训练的文档。文档的归档将基于一个工程、一件工作对应一个文件夹的原则整理，每个文件夹下将包括与该工程相关的代码、文件、数据、图片以及关于如何使用该工程的 README 文件。此文件是全局文件，将对各工程的内容进行概括与总结。

为了方便获取文档相关内容，本人会将相关代码、数据保存在个人的 `gitHub`¹ 上(相关数据会做加密处理)

1.1 眼动数据与主观评分文件预处理

1.1.1 眼动过程分离与滤波代码

此工程位于 `EyetrackingDataFilter/ExtraEyetrackerDataFeature` 文件夹内，是基于 C++ 实现的，工程的最终调试是在 Xcode 上完成的（前期在 VS2010 上用过，可能需要极个别调整）。工程的目的是将给定的眼动数据文件夹下的所有文件进行眼动过程分离与滤波——即对源文件中的数据标注出注视点与扫视点部分，并剔除掉其中的眨眼部分的数据。最后的结果是生成一个与源文件一一对应的同名文件，因此，需要在程序开始是设定输出文件的目录，防止源文件被覆盖。二是针对 3D 众包网站上下载的单刺激主观评分进行处理，程序开始时，需要配置文件的输入路径与输出路径。此工程具体的配置可见 `EyetrackingDataFilter` 文件夹下的 `README.md` 文件。

1.1.2 单刺激主观评分

这里还包括了一个针对 task26 的眼动实验的单刺激评分文件，其内为 24 个被试对 77 幅图像的评分，由于文件的写入方式的问题不能直接使用，上述程序可以将该文件进行归类整理，形成可直接使用的文件。

源文件：`3dvqa_Shifted_Image_Test.csv`;

处理后的文件：`ss_subjectAssessmentScore_aferProcessed.csv`;

读者在使用本工程材料时可以用上述程序处理下载好的文件，也可以使用已处理好的文件。

¹<https://github.com/huangturen/EyetrackerProject>

1.1.3 眼动数据

本实验涉及三部分眼动数据——与视敏度检验过程相关的数据 (task6), 立体图像的眼动实验数据 (task26), 3D 校正实验数据 (task28), 所有的数据打包归档为文件: *OrigEyetrackData.tar.gz*。直接解压即可得到三个实验的数据。Mac 下双击即可解压。或者在终端中首先进入文件所在位置, 输入: `tar -zxvf OrigEyetrackData.tar.gz` 即可解压。

本实验处理了 task26 (立体图像眼动实验) 的数据, 分别依据左右眼进行了滤波与眼动过程分离, 数据归档为文件: *task26_processed.tar.gz*。直接解压后可以得到分别以左右眼为参考的眼动数据处理结果。

1.1.4 实验图像

前面提到了本次研究设计了 3 个实验, 视敏度检验相关实验, 立体图像眼动实验, 3D 校正实验, 这里给出三个实验所用的图像。图像打包整理为文件 *OrigImage.tar.gz*

需要说明的是本文件夹中的 *OrigEyetrackData.tar.gz*、*OrigImage.tar.gz*、*task26_processed.tar.gz* 存储在百度云盘上。

链接: <http://pan.baidu.com/s/1dEvrGK5>

这里的链接以及密码请在 Trello 上 3DVQ crowdsourcing 中查找。

实际使用时请现在百度云盘上下载眼动数据, 并在 *EyetrackingDataFilter/ExtraEyetrackerDataFeature* 下的 main 函数中修改原始眼动数据的路径。

数据预处理文件夹 *EyetrackingDataFilter* 文档布局如代码 1.1 所示。

代码 1.1 数据预处理文件夹布局

```
—— 3dvqa_Shifted_Image_Test.csv
—— ExtraEyetrackerDataFeature
—— ExtractEyetrackDataFeature.cpp
—— eyetrackFeatureFactory.cpp
—— eyetrackFeatureFactory.h
—— fixationfilter.cpp           //滤波类
—— fixationfilter.h
—— main.cpp
—— preprocesseyetrackingdata.cpp //去掉眨眼过程
—— preprocesseyetrackingdata.h
—— readeyetrackingexperimentdatafile.cpp //读取眼动数据类
—— readeyetrackingexperimentdatafile.h
—— readfile.cpp                 //读文件基类
—— readfile.h
```

```
—— readsubjectqualityassessmentfile.cpp      //读取主观评分文件
—— readsubjectqualityassessmentfile.h
—— stdafx.cpp          // 自定义struct等
—— stdafx.h
—— ExtraEyetrackerDataFeature.xcodeproj
—— OrigEyetrackData.tar.gz
—— README.md
—— ss_subjectAssessmentScore_aferProcessed.csv
—— task26_processed.tar.gz
```

1.2 眼动数据特征提取

眼动数据特征提取工程基于 Python 开发, Python 有强大的图形与数据处理能力, 这里首先要进行 Python 模块配置。本工程依赖三个独立的 Python 库:

- Numpy——良好的矩阵处理、数据运算;
- Matplotlib ——类 Matlab 作图工具, 可以做出与 MATLAB 类似的图形;
- scikit-learn¹ ——SVM 训练工具;

三个库都可以通过 Python 的安装工具安装, 其命令为: `pip install xxx`, 表示安装 xxx 模块。本工程的内容布局可以用代码 1.2 来表示:

代码 1.2 眼动数据特征提取文件夹布局

```
—— EyetrackerData.py
—— EyetrackingDataBlinkFeatures.py
—— SubjectAssessmentData.py
—— UserInfo.py
—— corestats.py
—— eyetrackerDataProcess.py
—— participantInfo.csv
—— ss_subjectAssessmentScore_aferProcessed.csv
—— README.md
```

眼动特征提取还依赖两部分内容: 主观评分和用户信息。

- 主观评分生成特征训练的标签;
- 用户信息在生成主、辅眼特征时标记主辅眼。

SubjectAssessmentData.py 处理主观评分, 并将结果用在 eyetrackerDataProcess.py 中;

¹<http://scikit-learn.org/stable/install.html>

`UserInfo.py` 处理了被试的信息，特别是用户的主辅眼信息，最终以字典的形式将主辅眼的信息保存。

`EyetrackerData.py` 类提供了针对单个文件（一个人在看一幅图像时）获取特征的接口和获取视差角的接口。

`eyetrackerDataProcess.py` 整合了整体特征，得到 77 幅图像对应的 15 个特征。通过本文件的调用可以获得 77 幅图像对应的特征，并将其保存在用户指定的文件中。本文件的调用还可以获取每个眼动数据文件中计算的视差角的值，并将其以文件的形式保存在用户指定的文件夹。

本工程的特征提取工程中保存了自己处理的结果，分别保存在 `Disparity` 文件夹和 `Features` 文件夹中，读者可以选择使用。

程序的具体执行可以参考文件夹/`EyetrackingDataFeatures` 下的 `README.md` 文件。

1.3 主观评分处理过程

本工程用来处理主观评分，主要工作由 `SubjectAssessmentData.py` 来完成。本工程的布局如代码 1.3 所示：

代码 1.3 主观评分处理文件夹布局

```
—— README.md
—— SubjectAssessmentData.py
—— ss_subjectAssessmentScore_afterProcessed.csv
```

本工程可以提供一下功能：

- 得到不同移轴值下主观评分的表现；
- 得到同一图像主观评分的表现；
- 获取每个用户的 `kurtosis` 系数；
- 获取主观评分 95% 的置信区间；
- 获取整体 `MOS` 值的分布；
- 获取用户评分的绝对偏差；
- 获取用户评分的相对偏差；

功能利用 Python 执行脚本 `SubjectAssessmentData.py`，程序的执行细节与方法参考/`SubjectScoresProcess` 下的 `README.md` 文件。例如：

Python `SubjectAssessmentData.py` -p -Disparity

可以得到不同移轴值下主观评分的 `MOS` 值分布图。

代码 1.4 眼动数据特征训练文件夹布局

```
—— Features
—— features_left.csv
—— features_main.csv
—— features_multilevel.csv
—— features_nomain.csv
—— features_right.csv
—— README.md
—— corestats.py
—— corestats.pyc
—— featuresTrainAndAnalysis.py
```

1.4 眼动数据特征训练以及结果呈现

本工程是对提取的特征基于 SVR 进行回归训练，所用到的素材为工程 1.2 生成的 mos-features 文件，利用 Python 执行脚本 featuresTrainAndAnalysis.py 以及相关的选项就可以得到想要的结果。本工程的布局如代码 1.4 所示 本工程执行时需要预先在 Features 文件夹中保存想应的特征提取文件（工程 1.2 的工作），如果觉得不方便，也很容易对 featuresTrainAndAnalysis.py 中的 main 函数进行改造，设置灵活的文件路径。

此脚本提供了以下几种功能：

- 获取前 %x 的视差角特征与 MOS 值相关度最大的 x；
- 获取一些低相关度的特征在模型中的增益情况；
- 针对基于不同眼睛数据提取的特征进行训练，并打印出训练结果；
- 针对基于不同眼睛数据提取的特征除去低增益的特征后进行训练，并打印出训练结果；
- 获取特征集修正前后的效果对比；
- 获取所有特征与 MOS 值的相关度；

例如，执行：

```
$:Python featuresTrainAndAnalysis.py -t -fm right
```

便可以得到基于右眼数据滤波后的特征进行训练的结果。更多地功能，请参考 SVM-Train/README.md

1.5 大论文素材

本部分包含了大论文撰写过程中的所有素材，理论上只要将 **Latex** 配置正确，执行 Thesis 下 **thesis.tex** 文件便可以得到原始大论文。

Thesis 工程本人是在 **Mac** 上完成的，原始工程为上海交通大学学术论文模板¹。本地配置的基本的要求为：**MacTex 2015**，另外，需要安装依赖的字体（共八种，本工程将其打包归类，方便他人使用，安装方法很简单，只需要双击相应字体，系统便会自动安装），该模板使用的 **tex** 引擎为 **xelatex**，所以，开始编译时务必将默认的 **tex** 引擎由 **LaTeX** 改为 **xelatex**。

编译环境更多地问题可以参考“官方”文献 **SJTUThesisReadMe.pdf**，或在线查看 **gitHub** 上得帮助文档。

本工程的布局为：

代码 1.5 眼动数据特征训练文件夹布局

```
—— LICENSE
—— Makefile
—— SJTUThesisReadMe.pdf
—— bib
    —— thesis.bib
—— figure
    —— chap1
    —— chap2
    —— chap3
    —— chap4
    —— chap5
—— make.sh
—— pdf
    —— authorization.pdf
    —— original.pdf
—— sjtuthesis.cfg
—— sjtuthesis.cls
—— sjtuthesis.log
—— tex
    —— abstract.tex
    —— ack.tex
    —— app_cjk.tex
```

¹[urlhttps://github.com/weijianwen/SJTUThesis](https://github.com/weijianwen/SJTUThesis)

- app_eq.tex
- app_log.tex
- app_setup.tex
- chapter01.tex
- chapter02.tex
- chapter03.tex
- chapter04.tex
- chapter05.tex
- chapter06.tex
- id.tex
- patents.tex
- projects.tex
- pub.tex
- pubnoname.tex
- summary.tex
- symbol.tex
- texcount.pl
- thesis.pdf.blg
- thesis_tex

工程对于有 LaTeX 基础的读者来说并不复杂,值得说明的是大论文中使用的所有结果图都保存在 figure 文件夹中,且按章节进行了分类,读者可以直接使用。

读者在使用本工程过程中如果有不明白的其他问题,可以联系:huangturen@sjtu.edu.cn。

