清 华 大 学

综 合 论 文 训 练

题目：基于视觉的微表情分析

系　　别：电子工程系

专　　业：电子信息科学与技术

姓　　名：刘云帆

指导老师：陈健生　　助理教授

2015 年 5 月 21 日

关于学位论文使用授权的说明

本人完全了解清华大学有关保留、使用学位论文的规定，即：学校有权保留学位论文的复印件，允许该论文被查阅和借阅；学校可以公布该论文的全部或部分内容，可以采用影印、缩印或其他复制手段保存该论文。

(涉密的学位论文在解密后应遵守此规定)

签　名：_____导师签名：_____日　期：_____

中文摘要

表情是一种人类表达自身情感和内心活动的重要方式。到目前为止，计算机视觉领域已经对人脸表情进行了全面而深入的研究。然而一直以来，人们对于人脸面部表情的基于计算机视觉的研究集中于普通面部表情，对于人脸微表情的研究还不够全面。通常情况下，微表情是一种下意识的表情，能够显示出人物内心真实的、受到抑制的情感，所以具有很高的研究价值，在刑侦破案和医疗咨询等等方面都有着重要的应用。微表情具有持续时间短、运动幅度小等特点，很难用肉眼观察到微表情，为微表情的人工分析带来了很大困难。

针对上述问题，本文采用了基于计算机视觉的方法对微表情进行处理。针对目前已有的微表情数据库存在的问题，我们对已有的实验设计进行了改进，考虑了人物头部的宏观运动，采集了高帧率的全新的视频数据集。在该数据集上，我们成功地对微表情进行了检测，而且能够在此基础上进行进一步的微表情放大和识别的处理。

我们的算法将视频中的人脸部运动划分为三个部分：平面内运动、非平面内运动和微表情。在平面内运动中，人脸部所有像素都在图像所在平面内运动，运用视频帧间的低秩性可以将这部分运动去除；非平面内运动主要指的是人物头部的转动，人脸部分像素点的运动会脱离图像所在平面。通过计算帧间的光流场，我们可以得到像素的帧间位移，从而能够跟踪特定像素点的轨迹，并且利用宏观运动轨迹的低秩特性，将这部分运动和微表情进行分离，得到了参与微表情运动像素点的位置和运动轨迹。利用得到的微表情运动的相关信息，再结合图像变形算法和神经元网络，我们便能够对微表情进行放大和识别。

相对头部的宏观运动，微表情无论是在时域还是空域，都是一个相对微弱的信号，其检测和处理都有着较大的难度。本文所提出的算法虽然能够对参与微表情运动的点进行检测，对微表情加以放大并进行识别，但是其效果还有待提高。有些细节的问题，例如追踪点和阈值的选择，还有待于在之后的工作中加以解决和完善。

关键词：微表情；低秩性；检测；放大；识别

ABSTRACT

Expression is an important way for human beings to express their emotions and inner world. Broad and thorough study about expression has been done in computer vision area so far. However, computer vision based works about expression always focus on normal expressions, few study has been done on micro-expression. In most situations, micro-expression is a kind of subconscious expression that could reveal one's actual and suppressed emotion, so it possesses high research value in criminal investigation and medical consultation. The duration of micro-expression is short, and the magnitude of motion is subtle, therefore, it is hard for human being to detect micro-expression thereby brings difficulties to manual process of micro-expression.

In order to deal with the above-mentioned problems, this paper adopts computer vision based method to process micro-expression. To tackle with some problems in already existing database, we improved the experiment settings by including macro head motion and recording the videos using high frame rate video camera. We successfully perform micro-expression detection, and we do further processing including magnification and classification on the database we create.

In our algorithm, we divide the entire head motion into three parts: in-plane motion, non in-plane motion and micro-expression. As for in-plane motion, almost all pixels in facial region move within the plane where the image is located. This part of motion could be removed by making use of the 'low-rank' property between frames. In most situations, non in-plane motion refers to the rotation of head. When the head is shaking or nodding, some pixels in facial region will move out of the plane where the image is located. Non in-plane motion could be removed by tracking the traces of target pixels and separate those pixels into pixels that are involved in micro-expression and pixels that are not. Therefore, we successfully obtain the location and trace of the pixels that are involved in micro-expression. Basing on what we get, further processing such as magnification and classification are done by adopting image morphing and neural network respectively.

Comparing with the macro head motion, micro-expression is a relatively subtle change in both space domain and time domain, therefore, the detection and further processing is quite difficult. The algorithm proposed in this paper could successfully

perform detection, magnification and classification of micro-expression, but there is much improvement could be done. Some detail problems, like the selection of target pixels and threshold values, are still remain to be perfect.

Keywords： Micro-expression; Low-rank property; Detection; Magnification; Classification

目录

第 1 章 引　言

1.1 课题的提出

表情是一个人情感和心理活动的表露，是我们观察了解一个人内心世界的主要媒介，在人类社会的日常生活中扮演着不可替代的重要角色。人脸表情的相关处理，例如检测、分类和放大等，一直以来都是计算机视觉领域的重要研究课题。相关的工作和研究层出不穷，从人脸六种表情的分类（高兴、伤心、愤怒、恐惧、厌恶和惊讶）到用面部神经编码系统（FACS）对其进行详尽的标注，可以说人脸表情已经得到了全面而深入的研究。

然而直到目前为止，计算机视觉领域对于表情的研究还基本只停留在宏观的普通表情上，与之相对的微表情的研究才刚刚起步。微表情是一种持续时间短、运动幅度小，不容易被发现的表情[1]。同时根据心理学的研究，相对于普通表情，微表情能够更好地反映人们内心深处被压抑的真实情感。微表情的这种性质使得其成为了识别谎言的十分有效的线索，在医疗咨询、刑侦破案等方面有着广泛的应用。

不幸的是，由于微表情的持续时间短、运动幅度小，用肉眼很难通过观察发现微表情，而且未经过训练的人员很难准确地对微表情进行识别。人工在微表情处理上遇到的困难提示我们寻求更加精密的仪器和技术对微表情进行分析，在表情分析中得到广泛应用的计算机视觉技术便是不二选择。通过计算机视觉技术对微表情进行分析，实现微表情的检测，并在此基础上对其进行放大和识别等处理，便是本课题的主要目标。

1.2 课题意义以及实际应用

微表情于 1966 年第一次被 Haggard 和 Isaacs 观察到，他们认为这种微小的表情表现出了人们被内心压抑的真实情感，同时可能与自我防御机制有关[2]，然而他们的发现并未受到广泛关注。1969 年，Ekman 和 Friesen 在对一位抑郁症患者和医生谈话的录像进行分析时发现了微表情[3]。在这段录像中，患者在和医生谈话的过程中一直表现正常，有说有笑，并未显出任何异常。然而在对这段视频进行逐帧分析时，Ekman 和 Friesen 发现该患者在被问到关于未来生活的计划时，面部掠过了一个痛苦的表情，虽然短暂，但是十分明显。这位抑郁症病人最终承

认，其在与医生的谈话中掩盖了企图自杀的念头。

这个事例说明：微表情能够显示出一个人内心深处真实的情感。即使主观上想要抑制住这种情绪，然而只要这种情感足够强烈，总会有蛛丝马迹在脸上浮现，这种微小的线索就是微表情。

微表情的实际应用还远不止于此。在审讯室里，嫌疑人的微表情能够让审讯官掌握其真实的心理活动，从而能够做出正确的决断而不受到谎言的迷惑；在教室中，教室能够从同学们的微表情中了解到授课的效果，有哪些学生没有听懂，有哪些知识点没有讲清楚，一目了然；谈判桌上，代表们能够从对方的微表情中窥探到对方的真实想法，从而能够使得问题更快地得到解决。

如果能够在视频中对微表情进行检测并加以处理，使得我们能够高效而准确地微表情进行分析，得到其深层的意义，则将对我们生活带来极大的便利，这便是我们课题提出的意义。

1.3 相关研究工作

虽然人脸表情处理在计算机视觉领域已经是一个老生常谈的话题，但是该领域中微表情处理的研究才刚刚起步。经过文献调研，目前世界上有以下几个小组在进行基于计算机视觉的微表情的相关研究：

表 1.1 微表情研究团队及其主要相关工作

	研究团队	应用算法	应用	准确率
1	日本筑波大学 Polikovsky 团队	3D Descriptor + K-means Cluster	识别	95% (面向面部运动单元)
2	美国 USF 大学 Shreve 团队	Optical Strain + Threshold	检测	85% (宏观表情) 80% (微表情)
3	芬兰 Oulu 大学 赵国英团队	TIM + LBP-TOP + {RF, SVM}	识别	74.3% (检测) 71.4% (二分类)
4	中科院心理研究院 傅小兰团队	LBP-TOP + SVM	识别	63.41% (多类别分类)

日本筑波大学 Polikovsky 团队率先进行了基于计算机视觉的为表情研究[4]。他们运用了三维梯度描述子（3D Gradient Descriptor）作为微表情的特征，并运用 k 均值聚类算法（K-means Cluster）对特征进行了识别。最终他们的识别准确率达到了 95%，然而 Polikovsky 团队的工作是对于面部运动单元（AU, Action

Units）的分类，并非真正地对微表情进行分类。

美国 USF 大学的 Shreve 团队[5]运用光学应变（Optical Strain）对人脸的微表情变化进行描述，然后设置阈值来对光学应变的强度进行筛选，从而实现了宏观表情和微表情的检测，准确率在 80%以上。然而这个准确率是在 Shreve 团队自己构造的数据上取得的，在 Canal-9 节目录制的自然视频数据上，他们的算法只得到了 50%左右的准确率。

芬兰 Oulu 大学的赵国英团队[6]率先进行了自发的（Spontaneous）微表情的识别工作。他们首先设计了自发微表情诱发抑制实验（Spontaneous Micro-expression Inducement Suppress Experiment）并进行了数据采集。随后对于采集到的微表情数据进行了放缩校准，归一化等预处理，将面部的图像提取出来。接着他们对得到的图像序列进行了时域插值，他们运用了改进的局部二值模式（LBP-TOP）作为微表情的运动特征，最后运用随机森林（RF, Random Forest）和支持向量机（SVM, Support Vector Machine）对微表情进行了检测和识别，得到了 74.3%的检测准确率和 71.4%的识别准确率。值得注意的是，他们只对微表情进行了正面/负面或有感情/无感情的二分类。在他们的工作中还讨论了插值参数和不同帧率等要素对于实验结果的影响。

中科院心理研究院的傅小兰团队[7]进行了和赵国英团队相似的工作。在他们的实验中运用了更高帧率的数据采集模式（200fps）。和赵国英团队的工作不同的是，傅小兰团队实验中采取了多类分类的方法进行分类，他们将微表情分为惊讶、伤心、高兴、厌恶、恐惧、压抑和中性这 7 类，取得了 63.41%的分类准确率。

由于微表情的基于计算机视觉的研究刚刚起步，可供使用的数据还不够多，还没有足够成熟的数据库。上文中介绍的各个团队使用的都是自己采集的数据库，表 1.2 中列出了这几个研究团队所使用的数据库以及相关的信息。其中 Polikovsky 团队和 Shreve 团队的数据库中的微表情使用的是非自然的人为模仿（Posed）的表情。这种“微表情”虽然满足了持续时间短和幅度微小这两个特性，但是却不满足不受自主控制的条件，不符合微表情的诱发机理，所以其研究结果并不一定适用于自然的微表情。在赵国英和傅小兰团队的工作中，他们按照 Ekman 提出的微表情诱发方法[10]自行设计并实施了诱发微表情（Spontaneous）实验并进行了数据采集，取得了更好的效果。

通过对上述研究组的工作进行调研，我发现在他们的工作中存在着一个共同的不足之处：实验数据中没有考虑人物头部的宏观运动。在他们采集实验数据的过程中都要求被试的头部固定，几乎不能有任何的宏观运动。这个要求的提出是为了方便数据处理，但是只要稍微思考一下，就可以发现这个要求的不符实际之处：日常生活的应用中，几乎没有一处场景中的人物的头部是保持静止，没有宏

观运动的。所以我们在对微表情进行研究时，不应该做出头部没有宏观运动的假设，否则降低算法的实际适用性。针对这个问题，我们基于心理学文献中 Gemma Warren 等人的实验[8]，对现有数据采集实验设置进行了改进，考虑了人物头部的宏观运动，使得数据更加贴近日常生活中的实际情况，增强了算法的实际适用性。

相对于计算机视觉领域，心理学领域对微表情的研究开始得相对较早，研究得也更加充分。众多学者中，Paul Ekman[1][9][10]的贡献相对突出。Ekman 提出了微表情持续时间的范围（1/25 秒到 1/5 秒）、微表情与被抑制情感的关系和微表情的实验诱发条件等等都为本课题提供了丰富的理论基础。虽然心理学领域对微表情的研究有着重要的成果，但是与本文的主题内容并不直接相关，读者如果有兴趣请查阅相关专业文献。

表 1.2 微表情研究团队所使用的数据库信息

	研究团队	数据库名称	分辨率	帧率（fps）	特点
1	日本筑波大学 Polikovsky 团队	Polikovsky	640×480	200	Posed
2	美国 USF 大学 Shreve 团队	USF-HD	720×1280	29.7	Posed
3	芬兰 Oulu 大学赵国英团队	SMIC	640×480	100	Spontaneous
4	中科院心理研究院傅小兰团队	CASME2	640×480	200	Spontaneous

1.4 本课题要解决的问题

本课题要解决的问题主要分为两方面：数据方面和算法方面

数据方面，本课题中针对已有微表情数据所存在的问题：未考虑头部宏观运动，改进实验设置，考虑头部宏观运动，采集全新的数据集。力求使得新数据集能够更有效地代表日常生活中真实的微表情发生的情形。

算法方面，基于新采集的数据，提出在有头部宏观运动情形下的微表情检测算法，期望能够分离头部宏观运动和微表情。并且以此为基础，对分离出来的微表情进行放大和识别处理。

以上两方面便是本课题所要解决的主要问题。

第 2 章 算法原理

2.1 本章导论

本章讨论了我们课题中运用到的四个主要的算法：矩阵的低秩分解及其变形问题（Low-rank Sparse Decomposition）、光流法（Optical Flow Method）、图像变形算法（Image Morphing）和栈式自编码器理论（SAE, Stacked Auto-encoder）。其中矩阵的低秩分解算法用于微表情与头部宏观运动的分离，光流法用于提取像素在帧间的位移，图像变形算法用于微表情的放大，级联自动编码器用于微表情的识别。

2.2 节详细讲解了矩阵的低秩分解及其变形问题的解法；2.3 节介绍了光流法的基本原理；2.4 节对用于微表情放大的图像变形算法进行了简要介绍；2.5 节讨论了在微表情识别中用到的级联自动编码器的相关理论。

2.2 矩阵的低秩分解及其变形问题

矩阵的低秩分解主要解决的一个问题是如何将一个已知的矩阵分解成为一个低秩分量和一个稀疏分量的加和：

$$M = L + S \tag{2-1}$$

其中 M 一般是我们观测到的数据矩阵，可以是一幅图像，也可以是一个视频（将每一帧图像转化为向量作为 M 中的一行或一列）；L 是和 M 同样尺寸的低秩矩阵；S 矩阵是 M 和 L 之间的误差。

如果矩阵 S 作为误差矩阵，其中元素是幅度较小的、独立同分布的高斯变量，则传统的主成分分析算法（Principal Component Analysis，下称 PCA）就能够很好地解决式（2-1）中的问题[11]。PCA 将式（2-1）转化为下面这个最优化问题：

$$\text{minimize} \quad \|M - L\|_2 \tag{2-2}$$

$$\text{subject to} \quad rank(L) \leq k \tag{2-3}$$

其中，k 为低秩矩阵 L 的秩的最大值，$\|*\|_2$为 2 范数。上述最优化问题可以很方便地通过奇异值分解（SVD）进行求解。然而在很多应用场合，矩阵 S 中的元素

并不符合高斯分布（例如人脸图像中墨镜、帽子对于人脸的遮挡，或者图像中局部信息的缺失），所以传统的 PCA 算法并不能在这些应用场合中产生我们满意的结果，这也促使我们寻找更加泛用的、鲁棒的 PCA 算法。Ma Yi 和 John Wright 等人于 2009 年提出的鲁棒 PCA 算法（Robust PCA）较好地解决了这个问题[12]。本节中就将具体介绍该算法的具体求解过程，并且还会对变形后的相关问题进行讨论。

2.2.1 鲁棒 PCA 算法总述

鲁棒 PCA 算法中不再有对于误差矩阵 S 中元素分布的限制，即 S 中的元素现在可以取任意分布，同时其幅值也可以取任意值。但是在鲁棒 PCA 中，我们要求矩阵 S 中的值是稀疏的（Sparse），即矩阵 S 中的非零元素尽量少。在这种条件下，数据矩阵 M 的低秩分解就可以描述为：

$$\text{minimize} \quad rank(L)+\lambda\|S\|_0 \tag{2-4}$$

$$\text{subject to} \quad M=L+S \tag{2-5}$$

其中，$\|*\|_0$为 0 范数，代表矩阵 S 中非零元素的数目，λ为比例系数。上述最优化问题的求解存在着很大困难，因为$rank(L)$和$\|S\|_0$均为非凸函数，不能采用凸优化的方法进行求解。通过如下代换，可以将非凸函数近似转化为凸函数进行求解[11]：

$$rank(L)\rightarrow\|L\|_*,\ \ \|L\|_*=\sum_i\sigma_i(L) \tag{2-6}$$

$$\|S\|_0\rightarrow\|S\|_1,\ \ \|S\|_1=\sum_{i,j}|S_{ij}| \tag{2-7}$$

上述代换中，用矩阵的核范数$\|*\|_*$，即矩阵特征值的和代表矩阵的秩；用矩阵的 1 范数$\|*\|_1$来代替 0 范数。如此代换之后，原问题便近似地被转化为下面这个凸优化问题：

$$\text{minimize} \quad \|L\|_*+\lambda\|S\|_1 \tag{2-8}$$

$$\text{subject to} \quad M=L+S \tag{2-9}$$

本文中采用了拓展的拉格朗日乘子法（Augmented Lagrangian Multiplier method，下称 ALM）[12][13]对该凸优化问题进行求解，下节中我们将会对具体的求解方法进行介绍。

2.2.2 鲁棒 PCA 问题的求解

应用 ALM 对鲁棒 PCA 问题进行求解，首先要通过拉格朗日乘子将目标函数与约束限制条件结合起来，构造出拉格朗日函数，再进行迭代求解。对于我们的优化问题（式（2-8），式（2-9）），相应的拉格朗日函数为：

$$l(L,S,Y)=\|L\|_*+\lambda\|S\|_1+\langle Y,M-L-S\rangle+\frac{\mu}{2}\|M-L-S\|_F^2 \qquad (2\text{-}10)$$

其中 Y 是拉格朗日乘子，μ为正系数，$\|*\|_F$为弗罗贝尼乌斯范数（Frobenius norm）。传统的 ALM 算法中[14]一般是求解$(L_k,S_k)=\arg\min_{L,S} l(L,S,Y_k)$，即同时对变量 L 和 S 进行优化，然后再相应更新$Y_{k+1}=Y_k+\mu(M-S_k-L_k)$，不断迭代求解，直至 L 和 S 收敛。然而这种求解方法比较繁琐，文[12]中提出了一种相对简单的 ALM 近似求解算法，简化了求解步骤，避免了一系列优化问题的迭代求解，同时还保证了结果的准确性。

注意到优化问题$\min_L l(L,S,Y)$和$\min_S l(L,S,Y)$的解的形式十分简洁明了，我们不妨直接利用其对式（2-7）进行近似求解（由于解的形式推导过程较为复杂，为了简洁起见，这两个优化问题解的推导放在附录 B 中）。设软阈值函数（Soft Thresholding Function）为$S_\tau(x)=\mathrm{sgn}(x)\max(|x|-\tau,0)$，则$\min_S l(L,S,Y)$的解为：

$$\min_S l(L,S,Y)=S_{\lambda\mu}(M-L+\mu^{-1}Y) \qquad (2\text{-}11)$$

其中如果$S_\tau(X)$中 X 矩阵，则$S_\tau(X)$代表该操作作用于 X 中的每一个元素 x 中。对于问题$\min_L(L,S,Y)$，设奇异值阈值函数（Singular Value Thresholding Function）为$D_\tau(X)=US_\tau(\Sigma)V$，其中$X=U\Sigma V$为奇异值分解，则$\min_L(L,S,Y)$的解为：

$$\min_L l(L,S,Y)=D_\mu(M-L+\mu^{-1}Y) \qquad (2\text{-}12)$$

式（2-11）和（2-12）中列出了优化问题$\min_L l(L,S,Y)$和$\min_S l(L,S,Y)$的解，我们可以直接利用其对原始问题式（2-7）进行近似求解。主要思想是轮流对 L 和 S 进行优化，即先固定变量 S 的值对 L 进行优化，再固定变量 L 的值对 S 进行优化，最后对 Y 进行更新，直至 L 和 S 收敛。这种算法的优点在于用直接的代数运算（式（2-11）和（2-12））替代了传统算法中的优化问题求解

（$\left(L_k,S_k\right)=\arg\min_{L,S} l\left(L,S,Y_k\right)$），降低了运算的复杂度，同时保证了算法的收敛性和解的准确性[15][16]。拉格朗日函数求解算法的具体流程如下所示：

---

1 给定数据矩阵 $M\in\mathbb{R}^{m\times n}$,令 $\lambda=m^{-\frac{1}{2}}$；

2 初始化： $S_0=Y_0=0,\ \mu>0,\ k=0$；

3 while L and S does not converge：

4 计算 $X=M-S_k-\mu^{-1}Y_k$；

5 计算 $L_{k+1}=D_{\mu}(M-S_k-\mu^{-1}Y_k)$

6 计算 $S_{k+1}=S_{\lambda\mu}\left(M-L_{k+1}+\mu^{-1}Y_k\right)$

7 更新 Y， $Y_{k+1}=Y_k+\mu\left(M-L_{k+1}-S_{k+1}\right)$

8 $k=k+1$

9 end while

10 最终结果为 L，S

算法 2.1 拉格朗日函数求解算法

应用上述算法，我们求得了拉格朗日函数的极值 L 和 S。根据拉格朗日乘子法的相关理论知识，此处得到的 L 和 S 即为鲁棒 PCA 问题（式（2-8）和式（2-9））的解。

2.2.3 鲁棒 PCA 变形问题的求解

在有些应用场合下，我们已有的数据矩阵 M 并不具有低秩性，不能进行如同式（2-8）和式（2-9）中的分解。但是在有些时候，如果将 M 进行某种变换 τ，其得到的结果 M∘τ 则可以进行低秩分解[17]。如果 M 是一幅图像，则 τ 有可能是旋转，缩放或者剪切等图像变换。在这些场合中，我们依然可以借鉴鲁棒 PCA 中的思想，对变换之后的数据矩阵进行低秩分解。这种问题可以进行如下描述：

$$\text{minimize}\quad \|L\|_*+\lambda\|S\|_1 \tag{2-13}$$

$$\text{subject to}\quad M\circ\tau=L+S \tag{2-14}$$

不难看出，此处的优化问题和上一节中的鲁棒 PCA 中的优化问题有着很高的相似度，依然可以用相似的算法进行求解。不同之处在于每次迭代的时候，对变换

τ 也要进行更新。由于本优化问题的求解算法和上一节中的鲁棒 PCA 十分类似，在此就不再展开论述，详尽的步骤可以参考文[17]。

2.2.4 已知秩的情况下鲁棒 PCA 问题的求解

在前两节的讨论中，低秩矩阵 L 的秩都是未知量，是在优化问题中要最小化的量。然而在很多实际问题中，通过先验知识，我们已经知道了低秩矩阵 L 的秩的取值，本小节中我们就将对这种情况下矩阵低秩分解问题的解法进行讨论。

这种情况下的矩阵分解问题可以表述为[18]：

$$\text{minimize} \quad \|M-L-S\|_2,\ M\in\mathbb{R}^{m\times n} \tag{2-13}$$

$$\text{subject to} \quad rank(L)=k,\ \|S\|_{2,0}<p \tag{2-14}$$

这个问题中仍然含有两个待优化的变量，我们仍然可以将其一分为二，采用轮流优化的方法进行求解，即求解矩阵 S 的时候固定 L 的值不变，反之同理，如此迭代直至矩阵 L 和 S 收敛。该算法的具体流程如下：

---

1 给定数据矩阵 $M\in\mathbb{R}^{m\times n}$，矩阵 L 的秩 k，和 S 的范数 p；
2 初始化 $L_0=M,\ S_0=0$；
3 while L and S does not converge：
4 $M''=M-L$
5 计算 M'' 中每一行的 2 范数。将 2 范数最大的 p 行复制到 S 中相应的位置，并将 S 中剩余行的元素全置为 0；
6 计算 $M'=M-S$；
7 对 M' 进行奇异值分解，$M'=U\Sigma V$；
8 用最大的 k 个特征值及它们对应的特征向量构建 L
9 end while
10 最终结果为 L 和 S

算法 2.2 已知低秩矩阵的秩时的矩阵分解算法

2.3 光流法简介

光流法是计算机视觉领域中的一种常用算法，常用于表达图像中存在的运动。光流法的主要思想是通过考察像素在时域上的变化和帧间空域上的相关性，来找到帧间的物体运动信息。更具体地说，光流法就是通过研究各个像素灰度的帧间的运动，进而得到整个图像序列中的运动场。

光流法有如下几个重要的前提假设：

1) 空间中同一物体上的同一点所对应的像素，在相邻帧之间的灰度不变；

2) 相邻帧之间物体运动幅度足够小；

3) 邻近像素的运动应该一致；

其中，第一条假设是光流法求解的根据，而后两条假设是对光流法能够成功求解的保障。对于第 i 帧中特定的像素点 p_0，光流法实际上是在 i+1 帧中 p_0 位置的周围寻找灰度相同或相近的像素点 p’，然后认为向量 p_0p’就是像素点 p_0 在帧 i 和 i+1 间的运动矢量。由此可见，只有约定了运动前后像素灰度相近，才能够找到 p’；后两个假设保证了查找的效率和准确性。

现在假设灰度不变的前提成立，两帧图像之间变化如下图所示[19]：

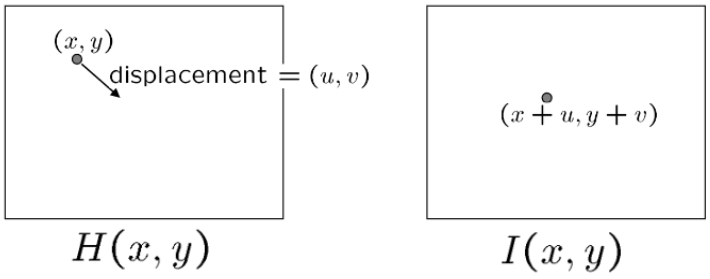


图 2.1 帧间运动示意图

基于运动前后灰度不变的假设，我们有：

$$I\left(x+u,y+v\right)=H\left(x,y\right) \tag{2-15}$$

由于 H 和 I 是相邻的两帧，所以在这个例子中，位移$\left(u,v\right)$表示的就是该像素的运动速度。我们将图像 I 以$\left(x,y\right)$为中心进行泰勒展开，我们有：

$$I\left(x,y\right)+\nabla I\cdot\left(u,v\right)-H\left(x,y\right)\approx 0 \tag{2-16}$$

如果我们定义两帧图像的差图像为I_t，则上式转化为

$$\nabla I\cdot(u,v)+I_t=0 \tag{2-17}$$

很显然上式所表示的方程是欠定的，我们还需要其他的条件对其进行限定。由于上式只是由灰度恒定假设推出的，我们不放考虑加上第 2 和第 3 个假设限定，即假设临近像素的运动向量也是(u,v)。设该点的邻域为$\Omega\in\mathbb{R}^{n\times n}$，则式（2-17）转化为：

$$\nabla I(\Omega)\cdot(u,v)+I_t(\Omega)=0 \tag{2-18}$$

将上式展开，我们有：

$$\begin{bmatrix} I_x(p_1) & I_y(p_1) \\ I_x(p_2) & I_y(p_2) \\ \vdots & \vdots \\ I_x(p_{n\times n}) & I_x(p_{n\times n}) \end{bmatrix}\cdot\begin{bmatrix} u \\ v \end{bmatrix}+\begin{bmatrix} I_t(p_1) \\ I_t(p_2) \\ \vdots \\ I_t(p_{n\times n}) \end{bmatrix}=0 \tag{2-19}$$

通过合理选择邻域大小，可以使得上式成为恰定方程组，从而可以通过矩阵求解的相关算法进行求解，最终得到该点的光流。将上述算法应用于图像中的每一点，就可以最终求出所有像素的光流，也就得到了图像的光流场。

2.4 图像变形算法简介

图像变形算法是一种通过两幅图像（源图像和目标图像）中已知的数个特征点，找到两组特征点之间的变换对应关系，然后利用这种关系对整幅图像进行操作的算法。

在计算两幅图之间的变换关系之前，我们首先应该求得两幅图像中的特征点。如果我们的图像是人脸，则特征点可能是嘴角或眼角点，如下图所示[20]：

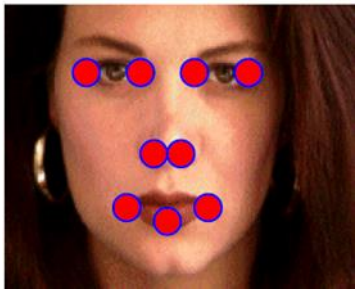


图 2.2 特征点示意图

得到特征点集后，一般要基于这些特征点用剖分的方法形成正方形或三角形的网格。然后将两幅图中的特征点进行对应，应用仿射变换计算从原图像到目标图像的变换矩阵 T。应用求得的变换矩阵 T 对网格的顶点进行变换，网格内部的点可以通过插值（线性插值，径向基函数插值等）得到。

如果我们知道图像中若干点的运动向量和起止坐标，那么运用图像变形算法可以得到这些点运动之后形成的图像。下图中是一个例子：

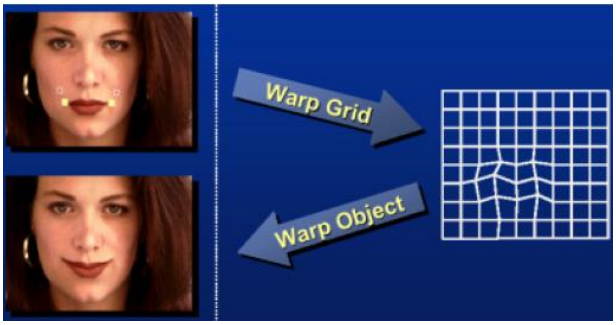


图 2.3 图像变换算法应用示意图

上图中，左上为原图像，即一幅没有表情的女性面部图像，图中的特征点位于嘴角处。若我们已知两特征点的运动轨迹：左嘴角特征点向左上移动，右嘴角特征点向右上移动，则通过应用图像变形算法可以得到上图左下的目标图像，即两嘴角上扬的微笑表情图像。

图像变形算法并不是本课题的主体内容，相关基础理论和数学推导请读者参阅文[20]。

2.5 栈式自编码器简介

栈式自编码器（Stacked Auto-encoder，下称 SAE）是神经网络的一种，通常用自编码算法对其进行训练（请注意自编码算法与 SAE 的区别）。自编码算法应用了神经网络领域常用的反向传播算法，并令数据的标签（也即网络的输出）与数据本身相等。假设我们现在有一个无标签的数据集$\{x_1, x_2, \ldots, x_n\}$，$x_i \in \mathbb{R}^n$，则自编码算法将每个样本的标签置为与数据相等，$x_i = \hat{x}_i$。

通过将目标输出设置为与输入相等，自编码算法本质上是要通过学习来实现一个恒等函数 $f(x_i) = x_i$。恒等函数乍看起来并没有什么学习的意义，但是当我们对于网络的结构（如图 2.4 所示[21]），例如隐层神经元数量等参数，进行限制后，自编码算法就会表现出一些重要的意义。举例来说，假设现在我们的网络输入为 10×10 的一张图像，则网络的输入和输出层都应该由 100 个神经元组成。

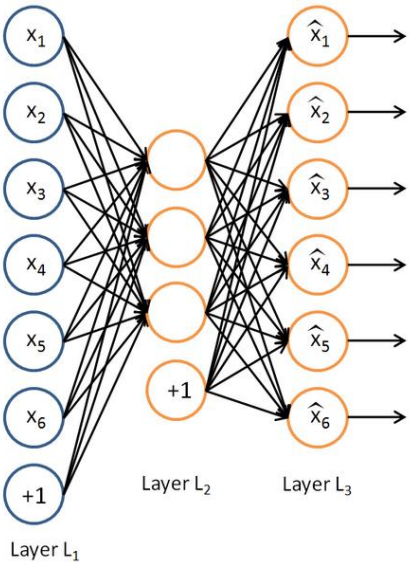


图 2.4 自编码算法网络结构示意图

如果我们将隐层神经元单元的数目设置为 50，少于 100 个，就会迫使自编码算法将输入数据进行压缩表示。自编码算法必须要从这 50 维的数据中以尽量小的误差恢复出 100 维的输入数据，这样我们就实现了对于输入数据的压缩，也可以说是提取了输入数据的低维特征。

SAE 就是一个由多层自编码算法网络级联构成的神经网络，也即前一级的自编码网络的隐层输出作为下一级的输入，其中每一层都要用自编码算法单独训练。通过这样的结构，我们可以提取输入数据的高阶特征，最终在网络最后一层应用 softmax 分类器，就可以实现对于输入特征的分类。一个典型的双隐层 SAE 的结构为[21]：

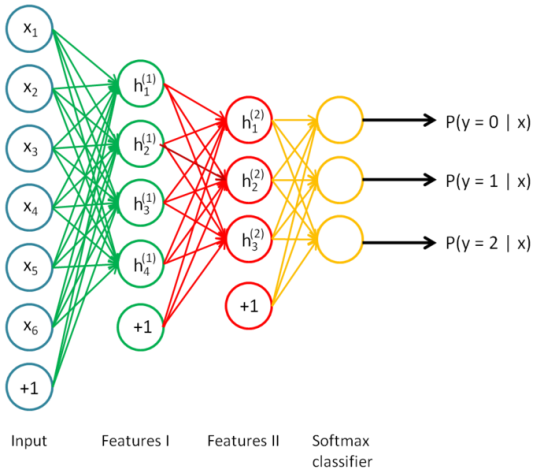


图 2.5 双隐层 SAE 网络结构示意图

图 2.4 中，Input 层是整个网络的输入层，Features 1 层提取输入数据的一阶特征。Input 层和 Features 1 层的训练正是采用了自编码算法。随后，Features 1 层的输出就成为了第二个自编码网络的输入，第二个自编码网络是由 Features 1 层和 Features 2 层构成的，其训练同样也采用了自编码算法。最后在网络的最高层应用 softmax classifier，使得整个网络能够对输入数据进行分类。

SAE 在本课题中被应用于微表情的分类，其相关数学基础并非是本课题的主要内容，详细的理论推导请读者参见文[21].

第 3 章 数据采集

3.1 本章导论

本章将主要介绍本课题数据采集的相关内容，并对数据采集实验的实验设计进行介绍。本课题的数据采集实验是基于相关心理学实验，针对现有微表情的计算机视觉研究的不足改进而成的。

3.2 节简要介绍了诱发微表情的心理学基础，这些理论对本实验的设计有着指导性的意义；3.3 节中对已有的微表情视觉研究中所用数据库的优缺点进行了讨论；3.4 节详细描述了实验设置；3.5 节展示了具体实验实施的场景和参数。

3.2 微表情的心理学基础简介

本节中将对诱发微表情的一些心理学基础进行讨论，这些理论对于实验设置有着重要的指导意义。

心理学家 Ekman[1]指出：“微表情是一种不受控制的，在人想要对情绪进行主观抑制时产生的表情。”也就是说，微表情是一种自发（Spontaneous）的表情，所以在我们的实验中，只能够通过实验设置去诱发被试产生微表情，而不能让被试去做出特定的表情。

Ekman[10]还对微表情诱发的条件进行了研究。研究结果表明，要想增加成功诱发微表情的概率，实验应该满足以下两个条件：

1) 外界给予被试足够的刺激。举例来说，当人看到十分搞笑的视频时，会忍不住发笑，那么这个视频就对此人产生了强烈的刺激。这种刺激是诱发微表情的必要条件。
2) 被试有足够的动机对表情进行抑制。举例来说，在审讯的时候，承认罪行所面临的严重惩罚为嫌疑人说谎提供了足够的动机，令其主动抑制表

情变化。这种抑制的动机是区别普通表情和微表情的重要条件。

综上所述，在设计实验诱发微表情时要充分考虑上述要点，这样才能增加产生微表情的概率。

3.3 已有微表情数据库的优缺点

第一章中已经对目前主要的微表情研究团队所用的数据库进行了简要介绍，在本节中我将会结合上节中介绍的微表情的心理学理论对这些数据库的优缺点进行讨论。

Polikovsky[4]和 Shreve 团队[5]所使用的数据库中的微表情使用的是非自然的人为微表情。这种微表情的产生方式是：被试被要求在保证头部固定的情况下，尽快做出细小的指定表情并恢复平静。这种“微表情”虽然满足了持续时间短和幅度微小这两个特性，但是并没有“外界刺激”和“抑制动机”这两个要素，所以并不能称为真正的微表情。

在赵国英[6]和傅小兰[7]团队的实验中，被试被要求观看一些经过挑选的，旨在引发特定微表情的视频片段。在观看视频之前，被试被告知实验的目的是尝试通过他们的面部表情推测出其正在观看的视频片段，如果成功猜中，则被试将会面临惩罚。这种实验设置既保证了视频能够成功诱发被试相应的情感，又给被试提供了足够的动机要在观看视频的过程中尽量压抑自己的表情，一定程度上满足了微表情产生的标准。

然而在他们的实验中，被试被要求尽量保证头部没有宏观运动。虽然这种要求给数据处理带来了便利，但是并不符合实际应用的要求。我们在实际应用中并不可能要求对象的头部没有任何除了微表情以外的运动。另外一个不足之处是，虽然在他们的文章中都提到了微表情和谎言紧密相关，是识别谎言的有力证据，但是在他们的工作之中都没有对这种关系进行基于计算机视觉的研究。

3.4 实验设置

针对上节的两点不足，我们提出了一种新的实验设置，将被试头部的宏观运动和微表情与说谎之间的关系考虑进来。为了寻求足够的心理学基础，我们对 Gemma Warren 等进行的心理学实验[8]（York Deception Detection Test，YorkDDT）进行了改编。

我们的实验设置如下：准备两段视频，一段视频记录了夏威夷海岸的美丽风景，意在引发被试放松、愉快的情感；另一端视频记录了牙科手术的场景，画面比较血腥，意在引发被试恐惧、厌恶的情感。这两段视频的选择在文[6][7]有相应

的根据。

实验一共分为两个部分：讲真话环节和说谎环节。在讲真话环节中，被试被要求观看两个视频中的任意一个，在观看过程中要用摇头和点头如实回答我提出的四个问题（详细调查问卷见附录 B），此过程中被试的头部运动将会被高速摄像机记录下来。四个问题分别为：

1) 此视频中是否有引发正面情感的物体或场景出现？
2) 此视频中是否引发了你如愉快、放松等正面情绪？
3) 此视频中是否有引发负面情感的物体或场景出现？
4) 此视频中是否引发了你如反感、厌恶等负面情绪？

在说谎环节中，试被被告知他的回答（点头、摇头）的录像将会拿给专业人员进行分析，专业人员从他的表情中推测出他所看的视频内容。如果成功推测，则将不会给予实验报酬，同时还将在实验结束后填写冗长的实验调查问卷。所以被试在回应我同样的四个问题时，将会尽量压抑自身的表情，回答过程中的面部运动也同样会被高速摄像机记录。

对比其他团队的实验，我们的这种实验设置具有以下创新点：

1) 放松了被试在采集数据时的约束，将头部的宏观运动考虑到微表情分析中来，有更高的实用价值；
2) 在实验中加入了说谎的要素，为深入研究微表情和说谎的关系提供了可能。

此外，我们的实验保证了微表情诱发的两个必要条件，同时还有详实的心理学实验基础。有关实验中视频选择、环节设置等细节的依据，请读者参阅文[6][7][8]。

3.5 图像采集

在本节中，我将对本实验的具体操作进行介绍。

本实验中，我们随机抽取了 10 名清华大学在校本科生作为被试。其中说真话和撒谎两个环节中各有 5 个人参加，两个环节中均有 3 个人观看了夏威夷风光视频，2 人观看了牙科手术视频。

我们使用的高速摄像机型号为索尼 NEX-FS700CK 摄录一体机，实验数据是在超慢录制功能下，以帧率 480fps，分辨率 1280×720 录制的。我们一共录制了 40 段视频，视频时长为 2.5~3.2 秒。帧率 480fps 保证了我们能够在微表情出现的 1/25~1/5 秒内记录足够的帧数，分辨率 1280×720 保证了我们能够捕捉到微表情那微小的变化。本实验的图像采集示意图和实际场景如图 3.1 和图 3.2 所示。

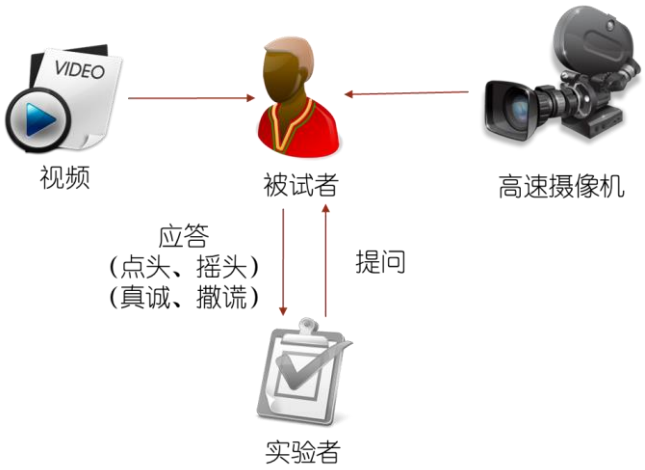


图 3.1 实验数据采集示意图

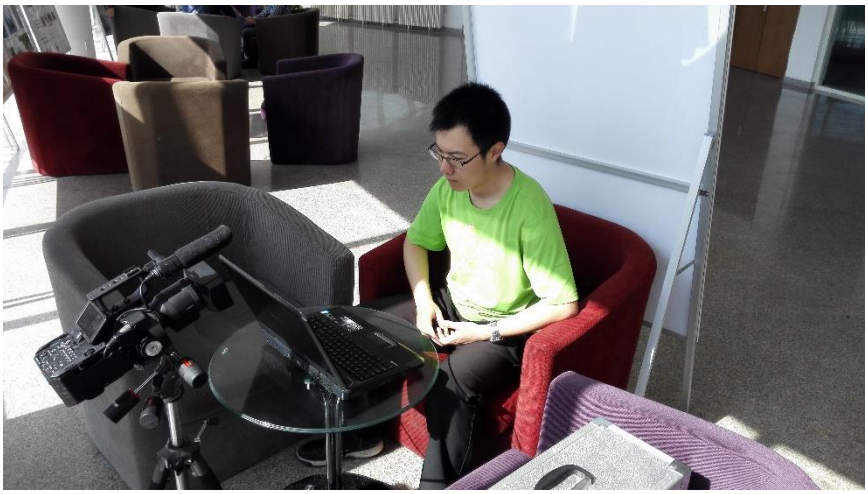


图 3.2 实验数据采集实际场景图

采集数据时，被试坐在椅子上观看笔记本电脑上播放的视频，摄像机架设在电脑后面，仔细调整摄像机位置和角度使镜头正对被试头部。在被试后方架设白板是为了尽量简化镜头中被试头部的背景。图 3.3 展示了两张实验中实际录制的画面。

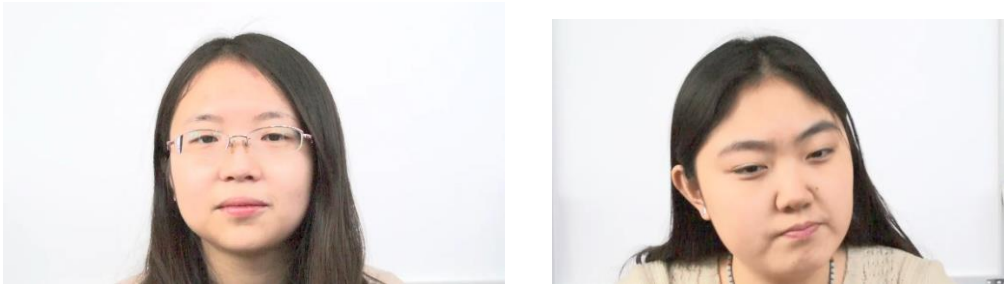


图 3.3 实际录制画面

第 4 章 具体算法流程和实验结果分析

4.1 本章导论

本章将详细阐述我们应用的算法是如何对微表情进行检测、放大和识别的。考虑到实际的微表情具有持续时间短，运动幅度小的特点，为了能够在文中更加清晰地展示出算法的效果，我将使用如下这段视频对算法进行说明（其中（1）至（5）为视频播放顺序，下图中只抽取了 5 帧对视频内运动进行描述）：

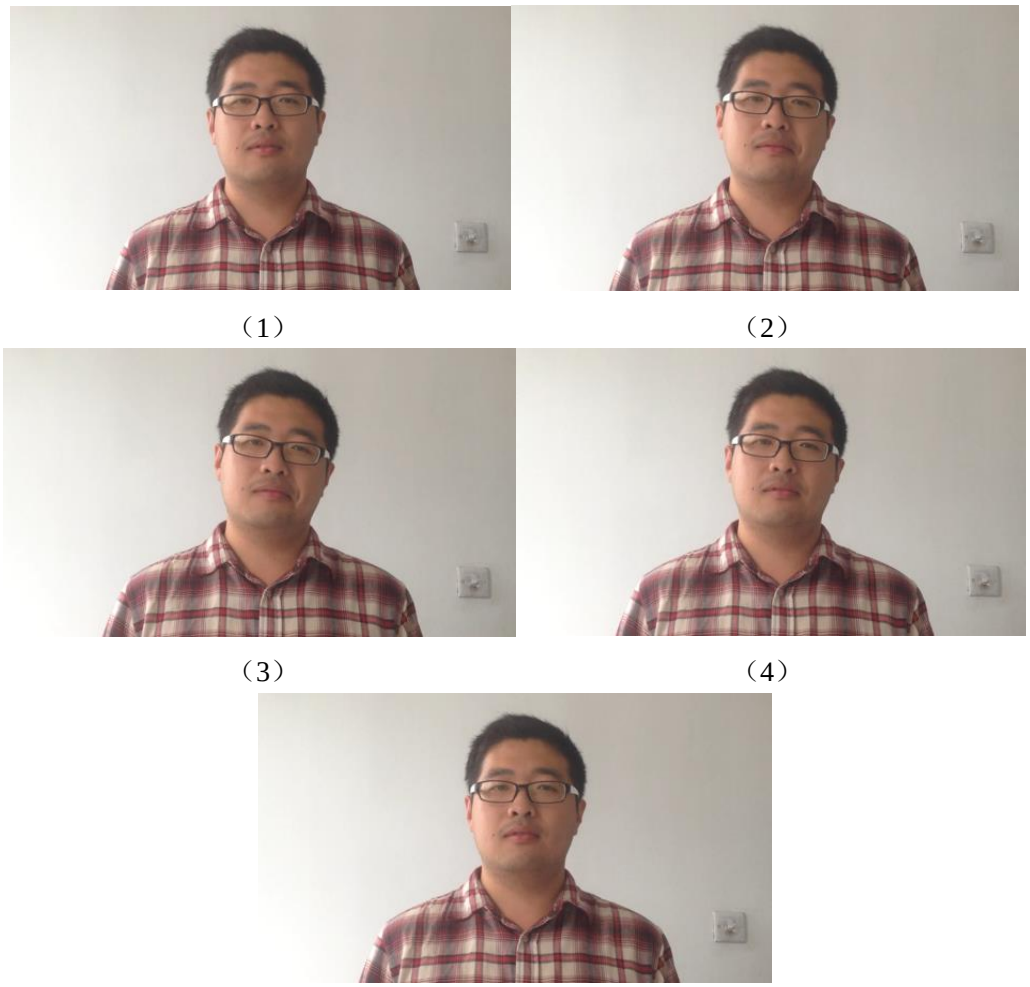


图 4.1 算法说明用视频样例帧

如图 4.1 所示，在整个视频播放过程中的男生的头部整体向右侧摆动，同时面部有一个微小的向右下转动的动作，另外右嘴角有一个向下撇的运动。图中所示的

表情并不是严格意义上的微表情，只不过由于其同时包含了人头部的宏观运动和微小但是可见的表情，便于我们展示算法的效果。所以本章在对算法进行讲解时将应用此视频，当然最终我们会将算法应用于自己录制的实验数据中。

本实验中所应用的平台为 MATLAB。MATLAB 作为脚本语言，可以方便调试，避免了编译的麻烦。另外 MATLAB 作为一款面向矩阵运算的语言，其中有很多方便调用的矩阵操作函数，例如 SVD 分解、求秩等操作，为我们的算法实现提供了很大的帮助。此外 MATLAB 自带的图像处理函数也为我们的实验提供了极大的方便。

我们将人头部的运动分解为平面内运动、非平面内运动和微表情，所以微表情的检测也就相当于从整体运动中将另外两种运动成分进行分离。4.2 节讨论平面内运动的去除办法；4.3 节阐述了平面内运动去除后，非平面内运动和微表情的分离方法；4.4 节中我们将算法应用于新录制的数据上，并对结果进行了详尽的讨论；4.5 节展示了微表情放大的结果；4.6 节讨论了微表情识别的相关问题。

4.2 平面内运动的去除

4.2.1 平面内运动的去除方法总述

如果我们将图 4.1 所示视频中的人脸部分截取出来：

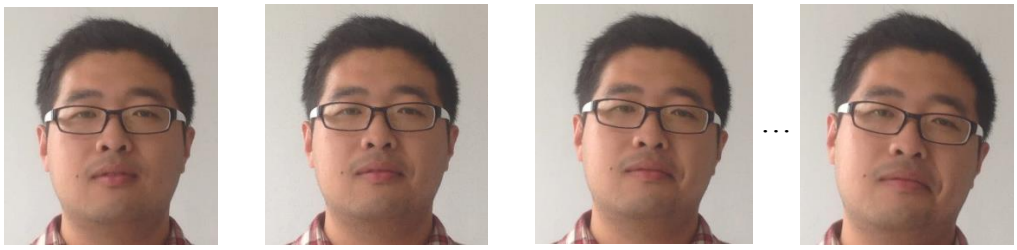


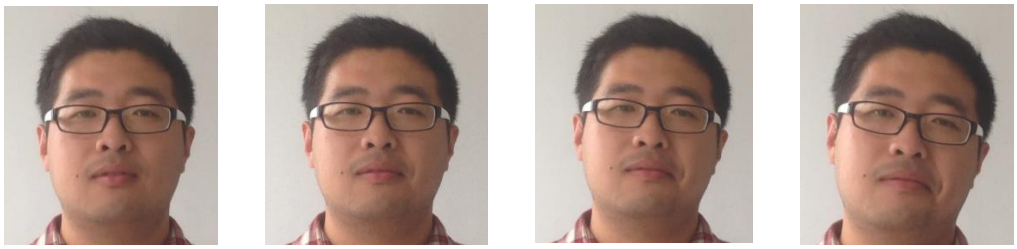
图 4.2 人脸部分截取图

不难发现，虽然每一幅图中的人脸的倾斜角度都不同，并且嘴角运动的幅度也有差异。但是如果我们把人脸作为一个整体，则其在每幅图中都保持了同样的结构特征（五官的外形，相对位置等等）。那么，如果我们能够将每幅图中人脸的平面内宏观运动消除，即使得每幅图中的人脸都呈同样的角度，再将这些图像矢量化后合并成一个矩阵，则这个矩阵应该具有什么特征呢？由于消除平面内运动后的图像中，除了右嘴角部分外的解决大部分都应该相似，所以这个矩阵应该具有**低秩性**。所以在我们的算法中，平面内运动去除的核心思想就是将人脸视为低秩结构，微表情作为稀疏误差，将平面内运动建模为图像变换，则这几部分的分解

可以运动第二章中介绍过的鲁棒 PCA 的相关算法求解。

4.2.2 具体算法

对图 4.2 中的每一帧进行数学建模：



$d_1 \circ \tau_1 = a + e_1$　　$d_2 \circ \tau_2 = a + e$　　$d_3 \circ \tau_3 = a + e$　　$d_n \circ \tau_n = a + e$

图 4.3 人脸部分截取图数学建模

其中，d 代表我们观测到的人脸部分的图像；τ 代表图像变换，在这里也就是人脸部的运动（旋转）；a 代表了几幅图像中统一的人脸结构；e 代表着由嘴角部分运动引入的误差。如果将图像矢量化后合并成矩阵，则有如下表示[17]：

$$D \circ \tau = [\mathrm{vec}(d_1 \circ \tau_1), \mathrm{vec}(d_2 \circ \tau_2), \ldots, \mathrm{vec}(d_n \circ \tau_n)] \quad (4\text{-}1)$$

$$A = [\mathrm{vec}(a_1), \mathrm{vec}(a_2), \ldots, \mathrm{vec}(a_n)] \quad (4\text{-}2)$$

$$E = [\mathrm{vec}(e_1), \mathrm{vec}(e_2), \ldots, \mathrm{vec}(e_n)] \quad (4\text{-}3)$$

其中，vec() 表示将一个 $n \times n$ 的矩阵矢量化成一个 $n^2 \times 1$ 的矢量。如上将各幅图像矢量化并合并成矩阵以后，该矩阵就代表了一个图像序列。则平面内运动的去除问题演变成如下的低秩矩阵分解的变形问题：

$$\text{minimize} \quad \|A\|_* + \lambda \|E\|_1 \quad (4\text{-}4)$$

$$\text{subject to} \quad D \circ \tau = A + E \quad (4\text{-}5)$$

我们的目标就是求解出 $D \circ \tau$，即消除平面内运动后的矩阵（图像序列）。注意这里的 A 和 E 虽然是 $D \circ \tau$ 分解得到的低秩成分和稀疏误差，但是由于 A 中非平面内运动的存在，E 还并不是单纯的微表情，还需要进一步处理。

在实际实验过程中，我们首先对每张图中的人脸特征点进行了定位，以两眼内眼角中点为基准截取出 150×140 大小的包含人脸以及周边区域的图像。这样

做是为了能够减小数据量，提高处理效率。另外为了初始化变换τ，我们进行了如下操作：

1) 在每幅图中运用人脸特征点定位算法，定位出两外眼角点的坐标；

2) 根据实际需要，设定变换后人脸图像的尺寸。本实验中设置为 120×95；

3) 根据实际需要，设定 2)中人脸图像规格中两外眼角点的坐标。

初始化完毕后，运用第二章中讨论过的鲁棒 PCA 及其变形问题的算法就可以对式（4-4）和式（4-5）中的问题进行求解。

4. 2. 3 去除效果

我们首先给出平面内运动去除的结果，为了提高算法运行效率，我们只对灰度图进行了操作。对于图 4.2 和图 4.3 中的四帧，去除平面内运动后的效果为：

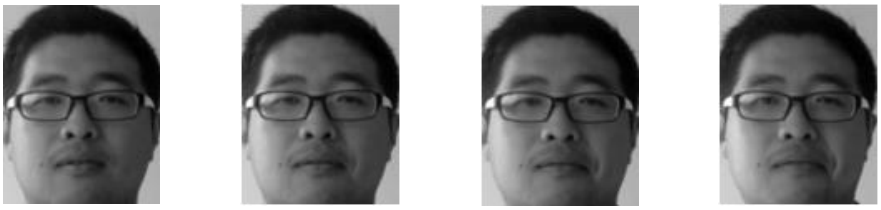


图 4.4 平面内运动去除效果

不难看出，上图中的人脸之前向右侧转动的运动已经被消除了，同时非平面内运动和微表情被保留了下来，为了说明这一点，我们给出上述四帧所对应的稀疏误差图像：

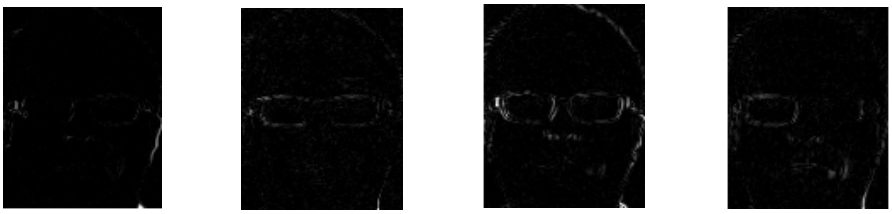


图 4.5 稀疏误差图像

图 4.5 中，黑色部分表示误差为 0，白色部分表示该处误差较大。误差指的是该幅图像和相应的低秩分量图像之间的差。低秩矩阵表示了消除平面内运动后的平均图像（人脸），所以白色部分就对应着该部分相对于平均图像的运动。理论上，如果稀疏误差中只有微表情存在，则在稀疏误差图像中应该只有嘴角部分有白色出现（对应着嘴角部分的运动）。然而图 4.5 中除了嘴角部以外，面部其他部分也有白色部分出现，这部分运动便是对应着视频中人脸的非平面内运动（人脸向右下方转动）。接下来我们要做的就是将这部分运动和微表情进行分离。

4.3 非平面内运动的去除

由于非平面内运动破坏了图像本身的低秩性，所以要去除这部分运动，只应用图像中像素的低秩性肯定是不够的。经过实验，我们发现虽然图像的低秩性被破坏，但是像素点运动的低秩性得以保留。像素点运动的低秩性指的是参与宏观刚体运动的像素点，其轨迹（位移）应该是相似的；如果我们将这些点的轨迹合并为一个矩阵，则这个矩阵应该具有低秩性。值得注意的是，此处矩阵的低秩性和 4.1 节中不同，4.1 节中矩阵的低秩性指的是图像序列中像素点的低秩性，而此处指的是某些像素点轨迹的低秩性。而参与微表情运动的点，其轨迹一定是有异于宏观运动点的，所以说可以作为稀疏误差加以分离。

用本实验所用视频来说，非平面内运动为人脸向右下转动。在此过程中，人脸上大部分点的轨迹都应该是向右下移动，而且轨迹应该是相似的。然而位于嘴角的点不单随着人脸向右下运动，同时还存在着相对运动（向下撇），所以其轨迹势必会与其他宏观运动点的轨迹存在差异。通过对这些差异进行检测，我们就能够分别出有哪些点参与了微表情运动。

运用轨迹的好处在于其不受图像所在平面的限制，即使运动并不在该平面内，参与宏观运动点的轨迹依然应该有相似性，其组成的矩阵依然应该具有低秩性，能够为我们的求解带来方便。

4.3.1 数学模型

假设视频一共有 n 帧，在这 n 帧中我们一共跟踪 k 个点的运动轨迹。跟踪点的选取和运动轨迹的计算都是运用的光流法。第 i 个跟踪点的轨迹都可以其坐标表示为：

$$t_i = \left[x_{1,i}, y_{1,i}, x_{2,i}, y_{2,i}, \ldots, x_{n,i}, y_{n,i} \right] \in \mathbb{R}^{1\times 2n} \tag{4-6}$$

其中 $x_{p,i}, y_{p,i}$ 代表该点在第 P 帧中的坐标。通过轨迹 t_i，我们可以计算该点在帧间的运动矢量（位移）：

$$t_{xi} = \left[\Delta x_{1,i}, \Delta x_{2,i}, \ldots, \Delta x_{n-1,i} \right] \in \mathbb{R}^{1\times (n-1)} \tag{4-7}$$

$$t_{yi} = \left[\Delta y_{1,i}, \Delta y_{2,i}, \ldots, \Delta y_{n-1,i} \right] \in \mathbb{R}^{1\times (n-1)} \tag{4-8}$$

其中t_{xi}和t_{yi}分别为该点在总共 n-1 帧中在 x 方向和 y 方向上的位移。将全部点的 x 方向和 y 方向上的位移合并起来写成矩阵形式，有：

$$t_x = \left[t_{x1}', t_{x2}', \ldots, t_{xk}'\right] \in \mathbb{R}^{(n-1)\times k} \quad (4\text{-}9)$$

$$t_y = \left[t_{y1}', t_{y2}', \ldots, t_{yk}'\right] \in \mathbb{R}^{(n-1)\times k} \quad (4\text{-}10)$$

这被追踪的 k 个点既包含参与宏观运动的像素点，也包含参与微表情运动的像素点，因此我们可以将其分解为两部分的加和（以t_x为例）：

$$t_x = tA_x + tE_x \quad (4\text{-}9)$$

其中tA_x代表参与宏观运动点的 x 方向上的位移矩阵，应具有低秩性；tE_x代表参与微表情运动点的 x 方向上的位移矩阵，应具有稀疏性。另外我们还知道，空间中刚体的运动，在平面上投影的点的运动轨迹矩阵的秩不会大于 3[18]，所以式（4-9）的求解就转化为已知低秩矩阵的秩时的鲁棒 PCA 求解问题：

$$\text{minimize} \quad \left\| t_x - tA_x - tE_x \right\|_2 \quad (4\text{-}10)$$

$$\text{subject to} \quad rank\left(tA_x\right) \le 3, \left\| tE_x \right\|_{2,0} < \alpha k \quad (4\text{-}11)$$

其中，k 为跟踪点的总个数，α 为被跟踪的参与微小运动像素点的占总述的比例，在我们的实验中取的是 0.05。对于 y 方向运动的t_y矩阵的分解同理。

在求解式（4-10）和式（4-11）的优化问题后，用tE_x和tE_y可以求出每个被追踪点稀疏误差分量的幅度。可以想象，参与微表情运动的点的稀疏误差分量的幅度一定会大于宏观运动的点，这样我们就对参与两种运动的像素点进行了区分，也即检测到了微表情。

4.3.2 实验结果

本步骤中，我们要处理的数据为消除了平面内运动之后的图像。首先我们通过求取第一帧和运动中某一帧之间的光流场来选择要跟踪的像素点（图 4.6）。为了减小误差，最好能够计算第一帧和表情运动幅度较大帧之间的光流）。计算得

到的光流场如图 4.7 所示：

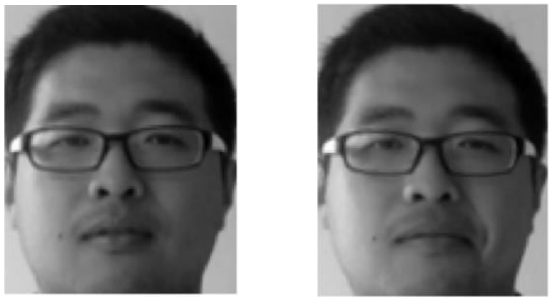


图 4.6 用于计算光流场的两帧图像

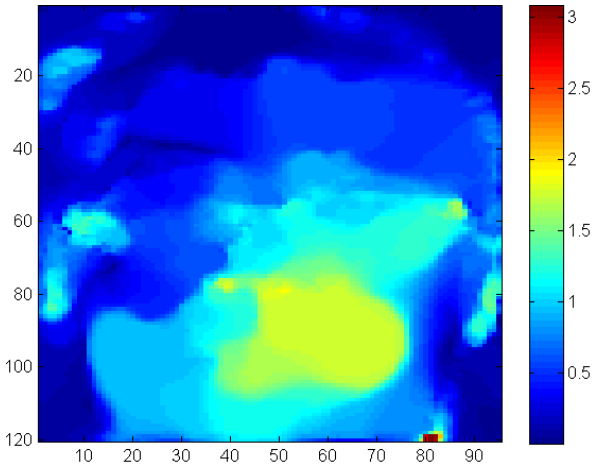


图 4.7 总光流幅度场

直观来看，右嘴角附近的点的光流幅度十分突出，这和之前我们做出的参与微表情运动的像素点轨迹会较为突出的假设是吻合的。而且整个面颊部分的光流幅度都相对较大，这是由于人脸在向右下转动时，面颊部分的光照变化较为明显导致的。通过设置阈值为 1.2，我们将光流幅度大于阈值的像素点作为要跟踪的像素点。

通过计算每相邻两帧之间的光流场，记录跟踪点的在 x 和 y 方向上的位移，就能够得到矩阵 t_x 和 t_y。在求解式（4-10）和式（4-11）的优化问题后，tE_x 和 tE_y 两个矩阵就已经可以求得。这两个矩阵的意义是：每个跟踪点的轨迹相对宏观运动轨迹的偏差，tA_x 和 tA_y 即为描述宏观运动的轨迹矩阵。分别计算每个跟踪点相对于宏观运动的偏差的幅度，幅度较大的前 αk 个跟踪点即为参与微表情运动的像素点。计算结果如下图所示：

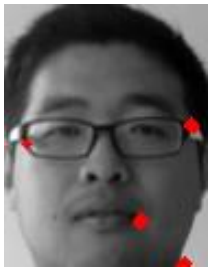


图 4.8 微小运动像素点检测结果

图 4.9 中，红色部分为最终检测到的参与微小运动的像素点。不难看出，除了嘴角位置点外，眼睛和颈部旁边也被归为参与微小运动的像素。经过分析，不难得知出现这种误差的原因：

1) 由于眼镜与人脸并不在同一平面上，所以当人的头部在转动的时候，眼镜会和人脸产生相对运动，这种运动与人脸的宏观运动有较大差异，故而被算法归为微小运动；
2) 当人脸向右侧倾斜时（平面内运动），右衣领与颈部距离变短，最终出现在了平面运动消除后的图像中（见图 4.4）。这个误差当然与宏观运动不相符，所以算法归为微小运动；

根据面部行为编码系统（FACS，Facial Action Coding System）的相关知识，人类的表情只能出现在若干个相对固定的位置上，常见的如眼角和嘴角等。运用这个先验知识，我们计算了每个被归为参与微小运动的像素与人脸特征点的距离（图 4.10 左），通过设定阈值，将与特征点距离较远的检测结果去掉，就得到了额最终的检测结果（图 4.10 右）。可以看出，我们的算法成功地检测到了参与微表情运动的像素点，同时我们之前还对这些点的轨迹进行了记录，所以说到此为止，我们已经将微表情和人脸的其他运动分离开来，达到了微表情检测的目的。

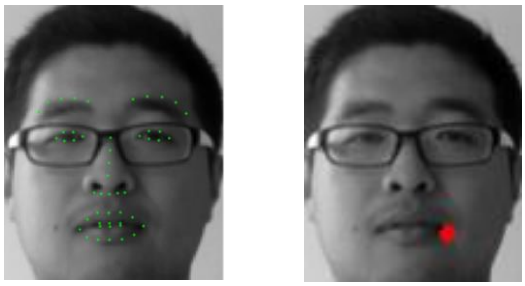


图 4.9 左：人脸特征点定位结果；右：最终检测结果

4.4 微表情检测算法在录制数据上的应用

在前几节中，为了能够清晰地展示算法的应用效果，我们使用的是表情运动幅度较为夸张的视频数据。通过视频中各帧图像的低秩性，我们消除了平面内的宏观运动；再运用宏观运动像素点轨迹的低秩性，我们对参与微小运动的点和参与宏观运动的点加以分别；最后利用表情相关的先验知识，我们精确地检测到了微表情。

在本节中，我们将前述算法应用于新录制的实验数据中。由于篇幅有限，同时也为了与之后的微表情放大和检测的结果相联系，我们在这里就只展示一个样例的所有结果。图 4.10 展示的是其中一个微表情中的按固定步长抽取的帧。可以看出，整个微表情持续时间很短，变化幅度也十分微小。

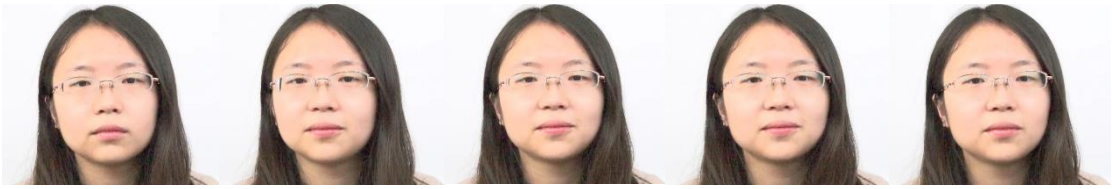


图 4.10 微表情示例

仔细观察的话，可以发现上图中女生在整个过程中嘴角微微上扬（微表情），同时头部左右摆动（宏观运动）。在消除平面内运动后，我们计算了第一帧和表情变化幅度最大帧（图 4.11）之间的光流幅度场（图 4.12），随后通过阈值（我们的实验中设为 5）筛选，确定了要跟踪的像素点。

图 4.11 计算光流场用到的两帧
（左图为第一帧图像，右图为表情变化幅度最大帧图像）

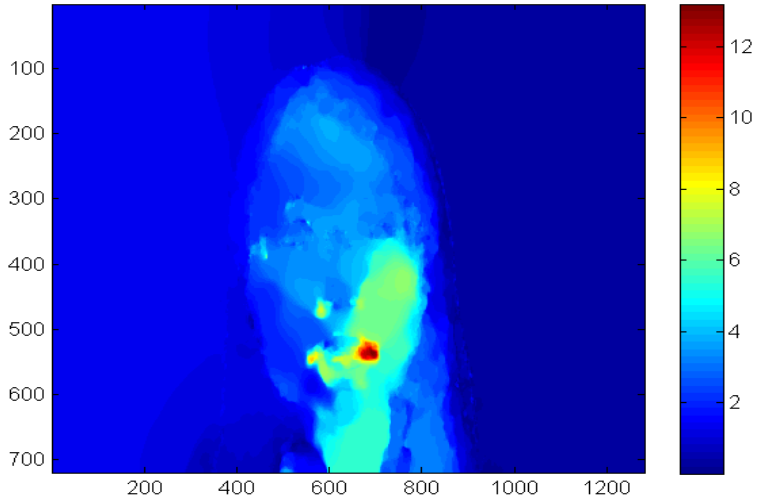


图 4.12 光流幅度场

通过对跟踪点的轨迹进行低秩分解，我们消除了微表情以外的宏观运动（此例中为头部的左右摆动），并运用人脸特征先验知识对微表情检测结果做了筛选，得到最终的检测结果（图 4.12 左）：

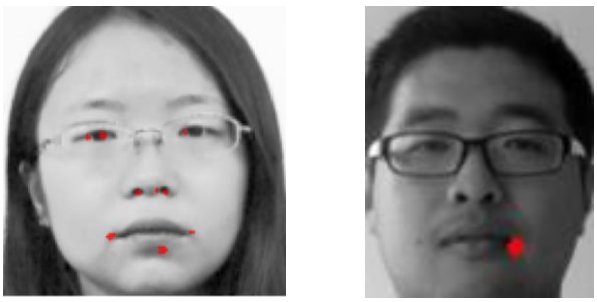


图 4.13 微表情检测结果比较

从图 4.12 中不难看出，与前例中检测结果（图 4.12 右）相比，本例中的检测结果中噪声较多：除了我们希望得到的嘴角点以外，眼白处、鼻孔处和下巴处均有像素点同样被算法识别为微表情运动点。结合图 4.10 中的示例帧可以分析得出噪声产生原因：

1) 由于视频中女生的头在左右摆动，同时眼睛一直在盯着正前方的电脑屏幕（实验中用于播放视频的电脑），所以眼球相对头部的宏观运动有着微小的相对移动，所以被算法为识别为参与微表情运动的像素点；
2) 女生在扬起左右嘴角的同时，势必会带动面部肌肉一同运动。由于鼻孔和下巴处的灰度较暗，所以在肌肉运动时的灰度变化相对较为明显，所以被算法为识别为参与微表情运动的像素点；

3) 结合我们的算法流程进行分析。由于我们的算法是根据各个跟踪点轨迹相对宏观运动轨迹的误差大小来进行判别的。在前例中（图 4.12 右所示男生例），嘴角处的运动幅度很大，用肉眼就能够轻松发现，说明嘴角处跟踪点相对宏观运动轨迹的偏离十分突出，所以我们的算法能够准确地对其进行检测。然而在本例中（图 4.12 左所示女生例），嘴角处运动极其微小，所以其相对宏观运动的偏离就不够突出，从而很难与其他部位由于非表情因素（光照，肌肉牵连运动等）带来的误差进行区分，最终产生了如图所示的噪声。

综上所述，我们的算法成功地对参与“微小运动”的像素点进行了检测。这里的“微小运动”既包括微表情，也包括前面提到的由非表情因素引起的变化。由于微表情的运动幅度实在过于微小，很难从变化幅度上对这两类变化（由微表情引起和由非表情因素引起）加以区分。这也是基于计算机视觉的微表情研究的根本困难所在：在高噪声条件下（大幅度宏观运动），对微小信号（微表情）进行检测。如何进一步提高这种低信噪比环境下的信号检测准确度，还需要再之后的工作中进一步分析。

4.5 微表情的放大算法

微表情的运动幅度极其微小，给人工识别带来了很大困难，即使是用计算机视觉技术进行处理，结果中也有存在着噪声。考虑到我们已经检测到了参与微表情运动的像素点的位置，同时我们已经求出了所有跟踪点的运动轨迹，那么我们能不能运用这些已有的信息对微表情运动进行放大，使之能够更加明显呢？通过实验我们发现，运用图像变形算法便可以实现微表情的放大。

假设由微表情检测算法得到的参与微小运动的像素点集为$\{x_E, y_E\}$，对应的运动轨迹为t_E；其他跟踪点（参与宏观运动）集合为$\{x_A, y_A\}$，对应的运动轨迹为t_A。在第四章中我们讲过，通过计算帧间的光流场，我们得到了跟踪点的运动轨迹t：

$$t = t_A + t_E \tag{4-12}$$

将微表情运动进行放大，就是将微小运动轨迹乘以放大系数λ，得到全新的运动轨迹t'：

$$t' = t_A + \lambda t_E \tag{4-13}$$

至此，我们已经得到了运动前的参与两种运动跟踪点的位置，以及微小运动放大后的总的运动轨迹，据此我们可以得到运动后各个跟踪点的轨迹。运用图像变形技术，我们可以计算出放大后的微表情图像：

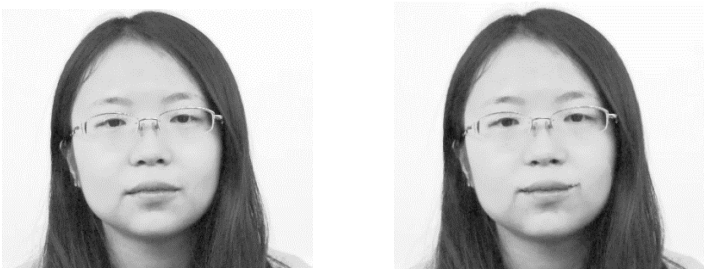


图 4.14 微表情放大结果
（左图为未放大的微表情运动最大帧，右图为放大后的图像）

从上图中我们可以看出，相对未放大的微表情，放大后的图像嘴角点有着更加明显的上扬，体现出更浓的笑意。通过对运动进行放大，我们将原本用肉眼难以发觉的微表情变得更加易于观察。用同样的算法，我们可以将任意两帧之间的运动进行放大，从而使得整个视频中的微表情运动更加明显，取得更好的效果。

4.6 微表情的识别

表情识别一直以来都是表情研究的重要课题，计算机视觉领域已经对其进行了大量的研究。一般来讲，表情识别分为以下几个步骤：

1) 特征提取。提取单张表情图像或者图像序列的特征用于训练和测试。常用的特征有 HOG 特征，LBP 特征和光流场等等；
2) 训练分类器。构建分类器，将上一步中提取的特征数据分为训练集和测试集，用训练集数据对分类器进行训练。常用的分类器有支持向量机（SVM），神经网络和随机森林等。
3) 测试分类器性能。将训练好的分类器用于测试集数据，计算识别准确率等数据。通常为了消除训练数据和测试数据选取的偏差，要运用交叉检验（cross validation）的方法进行测试。

考虑到微表情的视觉研究刚刚起步，微表情数据还不够成熟，我们决定在普通表情数据库上训练分类器，再将其运用到微表情的分类中。从视觉角度来讲，微表情和普通表情的主要的差别在于其运动幅度小，持续时间短，但是这并不会对其特征产生太大影响，所以理应也可以通过普通表情的分类器进行识别。

4.6.1 普通表情数据库简介

我们的实验中应用了 Extended Cohn-Kanade Dataset (CK+)数据库[22]。该数据包含了 123 个人的 593 个表情序列和相对应的已经标定的人脸特征点。每个序列都包含了从表情第一帧开始（中性脸，没有表情）到表情最大帧（最后一帧）的全部图像。按照 FACS Investigators Guide[23]的说明，我们对每个表情的标签进行了修正，将其归到六个常见表情中（愉快、伤心、愤怒、恐惧、厌恶和惊讶）。

4.6.2 数据预处理和特征提取

CK+数据库中的图像规格为 640×490 和 640×480 两种，数据纬度较高，不太适合直接用于特征提取。考虑到只需要脸部的图像数据进行特征提取，我们通过给定的人脸特征点将人脸部分的图像截取了出来，并通过降采样将其尺寸化为 37×36，大大降低了数据的维度。脸部图像截取和降采样方法如图 4.14 所示：

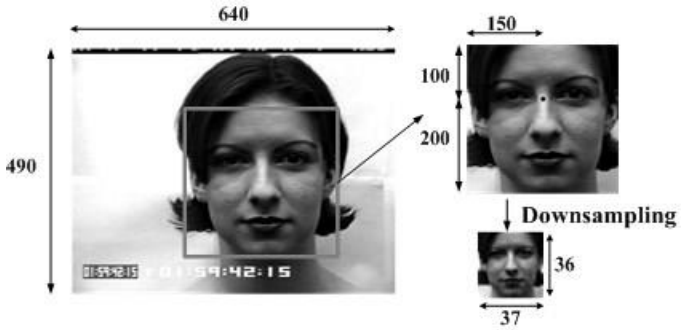


图 4.15 脸部图像截取和降采样

从图 4.14 中可以看出，我们通过人脸特征点定位了两内眼角点的中点。以该点作为基准，按照图中所示尺寸对面部表情进行了截取。接着我们对截取出的图像进行了 8 倍降采样操作。

为了能够将分类器用于微表情的识别，我们仍然用光流场的 x 和 y 方向分量作为表情的特征。为了扩大样本数，对于每一个表情序列，我们计算了第一帧和后三帧之间的光流场，并将他们标定为同一种表情。我们随机选择了 1392（493×3）个样本作为训练集，剩下的 150（50×3）个样本作为测试集。

4.6.3 分类器的训练和表现

我们采用了 SAE 作为分类器。我们设计的 SAE 有两个隐层，每个隐层含 300 个神经单元；隐层的稀疏性限制参数 ρ=0.1；梯度下降参数 λ=0.001;稀疏性惩罚参数 β=3.有关这些参数的具体意义，读者可以参阅文[21]。

分类器在测试集上的准确率如下图所示：

表 4.1 各表情类别分类准确率

	愤怒	厌恶	恐惧	愉快	伤心	惊讶
愤怒	24	0	0	0	3	0
厌恶	1	30	0	1	0	0
恐惧	1	0	15	0	0	0
愉快	1	0	0	23	0	0
伤心	3	0	0	0	21	3
惊讶	0	0	0	0	0	24
准确率	80.0%	100.0%	100.0%	95.8%	87.5%	88.9%

上表中，第一行表示标定的表情类别，第一列表示 SAE 分类输出的类别，表格中的数字表示相应情况下的样本数。举例来说，第一列数据表示测试集中一共有 24+1+1+1+3=30 个愤怒表情样本，有 24 个被 SAE 识别为愤怒，有 3 个被归类为伤心，而剩下 3 样本分别被识别为厌恶、恐惧和愉快。分类器的平均识别率为 91.3%，说明我们的分类器在 CK+数据库的普通表情上取得了不错的分类效果。CK+上的表情识别工作已经发表在 International Conference on Smart Computing 2014（SMARTCOMP 2014）上，感兴趣的读者可以参考文[24]。

4.6.4 微表情识别结果

在应用训练好的分类器对微表情进行识别之前，我们要先对微表情进行前述的预处理。首先，我们按照同样的方法截取出人脸局部图像（图 4.15），然后降采样到适合网络参数，接着计算出第一帧和微表情运动幅度最大帧之间的光流场的 x 和 y 方向分量作为特征（图 4.16），最后用分类器对特征进行识别。识别的结果如表 4.2 所示。

从图 4.15 中不难看出，嘴角对应部位的光流场幅值相对其他部位非常突出，这说明光流场敏锐地捕捉到了嘴角处的微小变化。表 4.2 第二行所示百分数代表分类器“认为”输入数据属于某一类别的概率，概率最大值对应的类别即为输入数据最终所属的类别；概率最大值越大说明分类器越“确信”输入数据属于该类别。本例中，输入微表情属于“愉快”类别的概率高达 90%，说明分类器成功地完成了微表情识别的任务。同时我们还可以看出，另外两项相对较大输出值对应的表情类别为“厌恶”和“惊讶”。这可以解释为这两种表情中都有两嘴角上扬，面部肌肉向上收缩的特征，与本例中的表情变化有相似之处，故这两种表情对应

的分类器输出相对高一些。

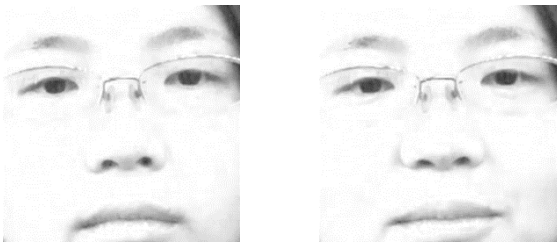


图 4.16 两帧中截取出的人脸局部图像

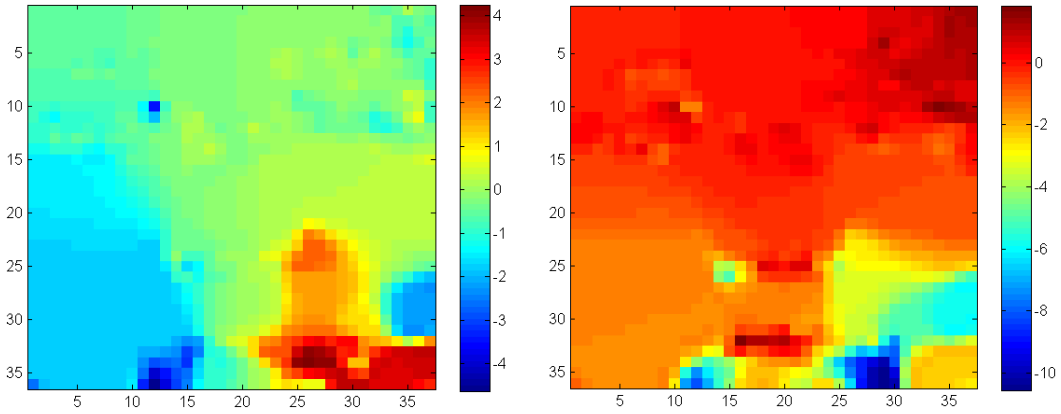


图 4.17 计算得到的 x 和 y 方向的光流场
（左图为光流场 x 分量，右图为 y 分量）

表 4.2 微表情分类结果

表情类别	伤心	伤心	厌恶	愉快	惊讶	恐惧
分类器输出(%)	0.008	0.17	5.13	**90.05**	4.58	0.062

第 5 章 总结

本课题针对表情研究领域新兴的课题——微表情进行了研究。针对现有相关工作中的不足（未考虑头部宏观运动），我们通过对心理学实验进行改进，自行采取了实验数据，初步建立了包含头部宏观运动和考虑被试心理状态（是否撒谎）的微表情数据库。接下来，我们将低秩性用于微表情的检测。视频中每一帧图像之间的低秩性被用于消除平面内的宏观运动；考虑到宏观运动像素点的轨迹也应该具有低秩性，我们对部分像素点进行了跟踪，并利用轨迹的低秩性将参与微小运动的像素点分离了出来，达到了微表情检测的效果。基于检测结果，我们应用了图像变形技术，对微表情进行放大，使得其变化更加明显，易于观察。最后我们在成熟的普通表情数据库 CK+上训练了分类器，不仅在其上取得了良好的分类准确率（91.3%），并且也成功地对微表情进行了识别。

本文尚存在以下几点不足：

1) 实验数据库不够完善。在人工对录制数据进行观察时，我发现其实大部分的视频中被试的表情基本没有任何变化。结合 Ekman 提出的微表情诱发条件，我们实验中的外部刺激和被试对于表情进行抑制的动机可能还不够强，一方面是我们的实验设置可能还有待改进，另一方面也可能是实验室条件本身就不利于诱发微表情。数据库的不完善也到至我们的算法很难在更多的视频数据上进行尝试。
2) 实验参数的确定方法强烈依赖于具体数据。我们的算法中有很多参数（筛选跟踪点的阈值等）得选择都是经过大量尝试得到的，如果换一个输入数据进行处理，则原有的各种参数可能都不再适用。

针对以上两点不足，我们相应地提出了可能的解决办法：

1) 建立更加完善的数据库，可能需要我们和相关专业人士合作，例如让心理学家帮助我们改进实验设置，启用更加专业的场地的人员录制数据，从而保证我们的数据足够典型、既满足心理学的理论基础，又能够让我们方便地对其进行处理。
2) 改进试验算法，使得参数尽量能够由输入数据求出，不再用经过尝试得到，这样也能够增加我们算法的稳定性和鲁棒性。

基于视觉的微表情研究是一个较新的领域，可以说是机遇与挑战并存。一方面，微表情有着深刻的心理学意义，在医疗咨询、刑侦审讯等各个方面都有着重要的应用价值；另一方面，微表情自身所具有的运动幅度小、持续时间短的特性，使之难以从其他运动中分离出来，为基于视觉的研究带来了很大的困难。本文率

先在考虑了宏观运动的情形下对微表情进行了检测，取得了较好的效果。其中低秩性的成功运用对于微小变化和宏观运动的分解问题有着很大的启示意义。我们对微表情进行的放大和识别实验说明了光流是描述微表情的有力方法。虽然微表情变化极其微小，但是运用光流我们依然能够敏锐地对其进行捕捉，这些结论为以后的微表情的视觉研究提供了基础。

插图索引

表格索引

参考文献

[1] Ekman Paul. 2001. Telling lies: Clues to Deceit in the Marketplace, Polities and Marriage. 2nd Ed. New York: Norton.

[2] Haggard E. A., Isaac K. S. 1996. Micro-momentary facial expressions as indication of ego mechanisms in psychotherapy. New York: Appleton-Century-Crofts, 154-165.

[3] Ekman Paul, Friesen W. V. 1969. Nonverbal Leakage and clues to deception. *Psychiatry*, 32: 88-97.

[4] Polikovsky S., Kameda Y., Ohta Y. 2009. Facial micro-expressions recognition using high speed camera and 3D-gradient descriptor. *3rd International Conference on Crime Detection and Prevention.*

[5] Shereve M., Godavarthy S., Goldgof D. et al. 2011. Macro- and micro-expression spotting in long videos using spatio-temporal strain. *The Ninth IEEE International Conference on Automatic Face and Gesture Recognition*.

[6] Pfister T., Li X., Zhao G. et al. 2011. Recognizing spontaneous facial micro-expressions. *IEEE International Conference on Computer Vision.* IEEE Press, 1449-1456.

[7] Yan W. J., Wu Q., Liu Y. J., et al. 2013. CASME database: A database of spontaneous micro-expressions collected from neutralized faces. *IEEE Conference on Automatic Face and Gesture Recognition.*

[8] G. Warren, E. Scherlter, and P. Bull. 2009. Detecting deception from emotional and unemotional cues. *J. November Behavior*, 33(1): 59-69.

[9] Ekman Paul, Sullivian M. O. 1991. Who can catch a liar? *American Psychologist*, 46(9): 913-920.

[10] Ekman Paul, Friesen W. V. 1974. Detecting deception from the body or face. *Journal of Personality and Social Psychology*, 29(3): 288-298.

[11] Zhou Z. H., Li X. D., John W., Emmanuel C., and Ma Yi. 2010. Stable Principal Component Pursuit. *IEEE International Symposium on Information Theory (ISIT)*.

[12] Emmanuel C., Li X. D., Ma Yi and J. Wright. 2011. Robust Principal Component Analysis? *Journal of the ACM*, volume 58, no. 3.

[13] Lin Z. C., Chen M. M., and Ma Yi. 2010. The Augmented Lagrange Multiplier Method for Exact Recovery of Corrupted Low-Rank Matrices. *UIUC Technical Report UILU-ENG-09-2214.*

[14] D. P. Bertsekas. 1982. Constrained Optimization and Lagrange Multiplier Method. *Academic Press.*

[15] Z. Lin, A. Ganesh, J. Wright, L. Wu, M. Chen, and Ma Yi. 2009. Fast convex optimization

algorithms for exact recovery of a corrupted low-rank matrix. I*n Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*.

[16] X. Yuan and J. Yang. 2009. Sparse and low-rank matrix decomposition via alternating direction methods. *Preprint*.

[17] Y. G. Peng, Arvind Ganesh, J. Wright, Wenli Xu, and Ma Yi. 2010. RASL: Robust Alignment by Sparse and Low-rank Decomposition for Linearly Correlated Images. *IEEE Conference on Computer Vision and Pattern Recognition*.

[18] Cui X, Huang J, Zhang S, et al. 2012. Background subtraction using low rank and group sparsity constraints. *European Conference on Computer Vision*.

[19] http://cs.nyu.edu/~fergus/teaching/vision_2012/13_opticalflow.pdf

[20] http://groups.csail.mit.edu/graphics/classes/CompPhoto06/syllabus.shtml

[21] http://ufldl.stanford.edu/wiki/index.php/UFLDL_Tutorial

[22] Lucey, Patrick, et al. 2010. The extended Cohn-Kanade dataset (CK+): A complete dataset for action unit and emotion-specified expression. *Computer Vision and Pattern Recognition Workshops*.

[23] Ekman Paul. 2002. Facial action coding system. *Salt Lake City: A Human Face*.

[24] Liu Y. F., Hou X. S. et al. 2014. Facial Expression Recognition and Generation using Sparse Autoencoder. *International Conference on Smart computing.*

致　谢

我谨在此对做毕业设计的数月以来对我提供方法和知识上帮助的陈健生老师，以及在组会讨论中给我提出宝贵意见的胡仲泽、李正钦同学和黄博、白高成学长。同时也非常感谢实验室能够为我提供十分专业的仪器设备和场地用于数据采集，这对本课题的顺利进行提供了很大帮助。另外还要感谢诸位积极参与并配合我数据采集实验的同学。

我在本科期间选修了陈老师的两门课程：数据与算法和视听信息系统导论。在这两门课上，陈老师严谨的治学态度和对学术科研的热情深深地影响了我。随后我报名了陈老师的 SRT 项目，取得了不错的成绩，在这过程中也学到了很多。最终我的毕业设计也使得到了陈老师的悉心指导，传授给我的一些科研和学术上的心得和理念也将在我之后的求学之路上给予我很大帮助。

本研究工作受到北京高校青年英才计划（项目编号 YETP0104）的资助，特此致谢。

声　　明

本人郑重声明：所呈交的学位论文，是本人在导师指导下，独立进行研究工作所取得的成果。尽我所知，除文中已经注明引用的内容外，本学位论文的研究成果不包含任何他人享有著作权的内容。对本论文所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确方式标明。

签　名：_____　　日　期：_____

附录 A 外文资料的调研阅读报告或书面翻译

面部自发微表情别

Tomas Pfister, Xiaobai Li, Guoying Zhao and Matti Pietik ¨ainen
Machine Vision Group, Department of Computer Science and Engineering,
University of Oulu
PO Box 4500, 90014 Oulu, Finland
{tpfister,lxiaobai,gyzhao,mkp}@ee.oulu.fi

摘　要：面部微表情是一种迅速的，自发的面部表情，往往揭示了一种被压抑的情感。就作者所知，在以往的所有工作中，并没有提出一种能够成功识别自发微表情的方法。在本文中，我们提出了一种结合了时域插值模型的自发微表情识别算法，并在微表情数据库上取得了准确的识别效果。我们还设计了一种诱发情感压抑实验来用高速摄像机来采集微表情数据库。本文率先提出了这种微表情识别算法，并且取得了比人工识别更高的准确率。

一、　引言

对于富含情感信息的宏观表情，人类可以迅速而准确地进行识别。然而，心理学研究发现人类的情感有时也会以微表情的形式出现。微表情是一种非常短暂（持续时间通常为 1/3 至 1/25 秒；至今尚未有准确的定义），下意识的面部表情，这种表情通常表现出了对象真实的，但是企图掩饰的情感。到目前为止，只有经过特殊训练的人能够识别微表情，但即使是经过专门的训练，其识别准确率也只有 47%。我们将给出一种结合了时域插值、多核学习(MKL)和随机森林(RF)分类器的微表情识别算法，该算法在一个新的自发微表情库上取得了比人工识别更高的，令人满意的结果。

微表情识别有着很大的应用前景。警察可以用微表情来检测异常的表现；当病人的信心不足时，医生可以通过其微表情来发现这种忧虑；教师可以通过微表情来看出学生的困惑并给予更多的指导；商人可以从客户的眉宇之间看出他们是否提供了合理的报价。正是因为微表情的人工识别准确率十分低下，我们迫切地需要一种能够准确识别微表情的方法。

微表情的识别主要有两个困难：短暂的持续时间和下意识性。短暂的持续时间意味着如果使用标准的 25fps 相机，则只有非常有限的几帧可供分析。为了解决这个问题，使用高帧率的相机势在必行。同时考虑到面貌的多样性，应用有监督的机器学习方法应该是一种比较好的解决问题的方法。但是由于微表情具有下

意识性，非自发的微表情（acted micro-expression）和自发的微表情有着很大的不同。因此，采集微表情数据库既需要心理学的理解，有需要耗时的实验来成功地诱发自发的微表情。

在本文中，我们提出了一种自发微表情识别的框架，这种方法得到了令人满意的结果。就我们所知，在我们之前并没有工作能够成功地对自发微表情进行了成功的识别。我们应用了时域插值算法来解决视频长度较短的问题，并且我们应用了时空局部纹理描述子来处理动态特征，最后我们使用了支持向量机、多核学习和随机森林来进行分类。我们和心理学家合作设计了感情诱发压抑实验来采集实验所需的训练数据。我们得到的自发微表情数据库(SMIC)是在 100fps 条件下录制的。我们使用的时域插值算法能够用在标准 25fps 条件下录制的数据产生同样的效果。

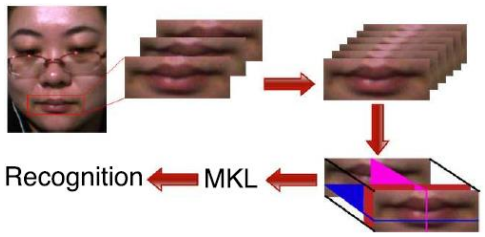


图 A.1 人脸微表情（左上）通过时域插值（右上）得到高帧率的图像序列，然后利用提取的时空局部纹理描述子（右下）进行训练和分类（左下）。

二、 相关工作

在本节中我们将介绍微表情的历史，并且对心理学的相关工作做一个总结，同时我们也会对非自发微表情的研究做一个简要的介绍。

在有关宏观表情的研究中，重要的研究工作集中于对于基础的 6 种表情的识别和对表情用 FACS 进行标记。在社会心理学中，微表情作为一种复杂的面部表情，已经被 Gottman 和 Ekman 进行了透彻的研究。Ekman 第一次发现微表情是在对一段心理疾病患者的采访中发现的。录像中该病人曾尝试掩饰其准备自杀的企图。通过在慢速模式下对该段视频进行观察，Ekman 发现了短暂且微小的，被微笑掩盖住的痛苦的表情。正是因为这种表情持续时间十分短暂，所以极其容易在正常观测条件下被人忽视。在之后的心理学研究中，微表情更多的被观察到，同时专门为提高微表情识别能力的训练课程也被设计了出来。

心理学上关于微表情的研究指出，人类天生就不善于识别微表情。Frank 用实际生活中的视频录像进行了一项微表情识别测试，实验结果说明了美国本科生和海岸警卫员在未经过训练的情况下的微表情识别准确率为 32%和 25%；在经过训练的情况下识别准确率为 40%和 47%。Ekman 的研究中指出，即使是将微

表情从背景中突出出来，如果不提供声音信息的话，人们还是难以对其进行识别。

到目前为止，绝大多数的关于人脸表情的研究使用的数据都是由被试特意摆出来的表情。研究表明这与人们日常生活中产生的，自然的表情有着较大的差异。最近的研究中心正在转移到诱发的，自然的表情数据上来。这些数据给研究者提出了新的挑战，因为诱发的表情自由度更高，录制条件更加不受控制。

仅有的几项关于微表情的研究都是基于非自然的（特意摆出来的）表情，并且没有一项研究公布出了他们所使用的数据供公众使用。Polikovsky 从 10 个大学生中收集了表情数据，并用梯度方向直方图描述子进行了处理。类似地，Shreve 等收集了 100 个非自然的面部微表情数据，并且运用张力模式作为特征描述子。被试被要求在观看包含有微表情的录像后对其进行模仿。

然而根据心理学研究的结果，微表情是一种下意识的行为，并不能够被特意模仿。不出所料的，Shreve 等在 Canal-9 政治辩论录像上的 24 个包含微表情的录像中只得到了 50%的识别准确率。虽然导致低识别率的原因可能是头部的运动和讲话所带来的额外噪声，但是可以看出，如果想要得到一个合理的，令人满意的识别准确率，一种不同的方法和一个足够大的数据库是必须的。在本文中，我们将给出一个更大的自然微表情数据库，同时还将提出一个成功的分类算法。

Micheal 等提出了一种自动通过对肢体语言进行分析进行测谎的算法。虽然他们在文章中简短地提到了微表情，但是在他们的训练数据中却没有相关的分析。

至今为止，很多成功的表情识别算法都用到了时空局部纹理描述子。LBP-TOP 是这类描述子中的一种，在表情分析中取得了最好的结果。

三、 算法

我们提出的人脸微表情识别算法结合了时域插值模型和最先进的表情识别方法。因为我们的算法是第一个成功地对微表情进行识别的算法，我们将会简要地对构建人脸微表情数据库的方法进行介绍。

1. 微表情识别算法

我们将会阐述如何结合时域插值模型(TIM)和最先进的机器学习方法来成功地以高准确率对微表情进行识别。算法 1 中给出了我们微表情识别算法的框架。

为了解决微表情的空域多样性的问题，我们将人脸部分的图像进行了裁剪和缩放。其基准是由 Haar eye detector 得到的眼睛的位置和由 Active Shape Model (ASM)得到的特征点。ASM 是通过迭代形变来逐步适应到对象上的统计模型。迭代是从一个尺寸和位置由人脸识别器所决定的平均形状开始的，该迭代不断重复直至收敛。

通过使用 68 个 ASM 特征点（如图 2 所示），我们计算得到了一个序列 i 中帧 $p_{i,1}$ 的局部加权平均变换(LWM, Local Weighted Mean)。LWM 通过将任一点(x,

y)的值设置成(1)中的值来计算所有多项式的加权平均，

$$f(x,y)=\frac{\sum_{i=1}^{N}V(\sqrt{(x-x_i)^2+(y-y_i)^2}/R_n)S_i(x,y)}{\sum_{i=1}^{N}V(\sqrt{(x-x_i)^2+(y-y_i)^2}/R_n)}$$

其中 $S_i(x, y)$是关于通过控制点(x_i, y_i)的 n 个参数和最近的 n-1 个参数的多项式，V 是权重向量，R_n 是在参考图中(x_i, y_i)距离最近的 n-1 个控制点的距离。接着，我们对一个图像序列的 s 帧中的 $p_{i,2}$，...，$p_{i,s}$ 应用这个变换。图 2 展示了从示例人脸到标准人脸的 LWM 变换。我们应用了 Haar 眼部检测的结果和 ASM 特征对人脸图像进行了裁剪和归一化，如算法 1 所示。

更进一步地，我们将所有微表情序列时域归一化到帧集 θ∈T 中。对于每一个微表情序列 i，我们计算得到了时域插值后的图像序列，

$$\xi_{i,\theta} = UM\mathcal{F}^n(t) + \xi_{i,\theta} \text{ for all } \theta \in T$$

其中 U 是奇异值分解矩阵，M 是一个方阵。随后，我们对视频录像提取了时空局部纹理描述子(SLTD)作为特征。值得注意的是 SLTD 的提取是对于视频的时长有着最短的要求。本文实验中我们应用的是 LBP-TOP 特征，半径 R 取 3，分块的面积在算法 1 中已经给出。这种参数设置要求序列中的前 3 帧和后 3 帧需要去掉，所以如果想要成功提取 1 帧的 LBP-TOP 特征，我们必须要求一段视频长度至少为 7 帧。如果使用标准 25fps 的摄像机进行录制，一段长为 1/3 至 1/25 秒的微表情长度大概为 1 至 8 帧，所以说为了更好地使用 STLD，我们需要一种能够产生更多帧图像的方法。如果我们的视频数据有更多帧的图像，则可以期望的是我们的算法将得到更好的结果。在 3.2 节中我们将给出一种时域图像插值方法。

在本文中，我们使用了多核学习(MKL)来提高我们的分类效果。对于一个给定的训练集 $H=\{(x_1,l_1)\ldots(x_n,l_n)\}$和核集 $K=\{K_1\ldots K_n\}$,其中 $K_k\in R^{nxn}$ 且半正定。多核学习通过在不同的定义域中优化能量方程 Z(K,H)来学习核函数的线性/非线性组合中权重 W 的大小。如算法 1 中所述，我们将 2 和 6 阶多项式核 POLY 与直方图交织核 POLY 相结合。对于所有的 p∈Γ，θ∈T，我们有，

$$\text{POLY}(q_{j,r}, q_{k,r}, d) = (1 + q_{j,r}q_{k,r}^{\mathrm{T}})^d$$

$$\text{HISINT}(q_{j,r}, q_{k,r}) = \sum_{a=1}^{b} \min\{q_{j,r}^a, q_{k,r}^a\}$$

其中 r=(m, θ, p)，b 是 $q_{j,r}$ 和 $q_{k,r}$ 中柱的数量。同样地，我们也可以选用随机森林和支持向量机作为分类器。算法 1 中分类器的参数是我们在之前的实验中事先进行优化过的。

在本实验中，分类算法分为两步进行。第一步(MKL)是对微表情的出现进行检测。在训练过程中，我们使用了时空归一化过的，同样大小的，经过标注的数据。标注的类别为{micro, ~micro}。

如果微表情监测的结果为真，则分类器将该微表情分类为标签集 $L=\{l_1,..,l_n\}$ 中的一类。将整个过程分为两步使得我们可以：1、分两步对系统进行优化；2、在保证第一步已经优化完成的前提下，对第二部进行参数优化。更进一步来讲，因为微表情的分类相对检测来说需要更进一步的分析，所以会面临着更大的困难。故而通过将两部分工作分开，我们可以避免第二步（分类）中的误差干扰第一步（检测）的结果。

算法 1：自发微表情识别算法。C 是图像序列 c_i 的集合，Γ是 SLTD 参数的集合，x, y, t 是计算 SLTD 特征时矩阵分块的行数，列数和页数。T 是序列 c_i 在时域插值后的帧集合。LWM 是局部加权平均变换。3.2 节中定义了时域插值的所需的变量。POLY 和 HISINT 分别是基于式 2 和 3 计算得到的多项式核和直方图交织核。MKL-1 和 MKL-2 应用多核学习分类器分别计算得到了检测(detection)和分类(classification)的结果。

微表情检测和识别算法(C)：

1、 初始化 Γ={8×8×1,5×5×1,8×8×2,5×5×2}, T={10,15,20,30}

2、 对于所有的 i, $c_i \in C$, 帧 $p_{i,1}\cdots p_{i,s}$

a）在帧 $p_{i,1}$ 中检测人脸 F_i;

b） 利用 ASM 方法提取 h 个脸部特征点 $\Psi=\{(a_1, b_1), (a_2, b_2),\cdots(a_h, b_h)\}$;

c）通过 LWM 方法计算变换ξ, 并利用该变换将检测到的人脸归一化到标准人脸上;

d）应用计算得到的变换ξ到帧 $p_{i,2}\cdots p_{i,s}$ 上

e）找到眼部位置 $E(F_i)=\{(x_{i,l},y_{i,l}),(x_{i,r},y_{i,r})\}$，设置距离为

$\delta_i=\sqrt{(x_{i,l}-x_{i,r})^2+(y_{i,l}-y_{i,r})^2}$

f) 截取面图像区域。左上角坐标为$(x_{i,l},y_{i,l})+0.4(y_{i,l}-y_{i,r})-0.6(x_{i,r}-x_{i,l})$，高为 $2.2\delta_i$，宽为 $1.8\delta_i$。

g) 对于所有的 $\theta\in T$，计算时域插值后的图像序列。

h) 对于所有的 $p\in\Gamma$，$\theta\in T$，提取 SLTD 特征集合

$$\mu_{i,p,\theta}(\xi_{i,\theta})=\{q_{i,p,\theta,1},\dots,q_{i,p,\theta,M}\}$$

其中 M 为 SLTD 特征的长度。

i) 计算核 POLY 和 HISTINT;

j) 如果检测（detection）的结果为真（即检测到微表情），则对该微表情进行分类。

2. 时域插值模型

在本节中我们将展示如何在微表情图像序列中任意位置插入帧。应用我们提出的这种方法就可以从持续时间很短的微表情中正常地提取我们所需要的特征。通过增加视频序列中的数据帧的数量，我们可以获得更加稳定的特征提取结果。Zhou 等在之前的工作中针对分析讲话中的嘴部的运动提出了相似的方法。据作者所知，这是这种方法第一次被用在面部表情识别中。

如图 3 中所示，我们将微表情看作是沿着曲线采集的一系列图像。通过将微表情录像看做一个图 P_n 中的 n 个顶点，我们得到了一个一个低维度的连续方程。顶点对应着微表情视频中的帧，而边对应着邻接矩阵 $W\in\{0,1\}^{nxn}$。其中当$|i-j|=1$时，$W_{i,j}=1$；$|i-j|\sim=1$ 时，$W_{i,j}=0$. 我们将 P_n 映射到一条能够最小化联通顶点之间距离的一条线上。设 $y=(y_1,y_2,\dots,y_n)^T$ 为该映射，为了求得该映射，我们将问题转化为最小化

$$\sum_{i,j}(y_i-y_j)^2W_{ij},\quad i,j=1,2,\dots,n$$

该问题等价于求解 P_n 的拉普拉斯图的特征向量。我们求得了拉普拉斯图的特征向量$\{y_1,y_2,\dots,y_n\}$，这时的我们将 y_k 看做由

$$f_k^n(t)=\sin(\pi kt+\pi(n-k)/(2n)),\ t\in[1/n,1]$$

所描述的一个点集，其中 t=1/n, 2/n, …, 1. 我们可以用计算得到的曲线

$$\mathcal{F}^n(t) = \begin{bmatrix} f_1^n(t) \\ f_2^n(t) \\ \vdots \\ f_{n-1}^n(t) \end{bmatrix}$$

来在任意一个微表情序列中的任意一个位置进行时域插值。为了找到该在图像空间中该曲线的对应，我们将图像帧映射到由 t=1/n, 2/n, …, 1 时曲线定义的点上，然后运用线性沿拓的方法来学习变换向量 w，使得能最小化

$$\sum_{i,j} \left(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i - \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_j\right)^2 W_{ij}, \quad i,j = 1,2,\ldots,n$$

其中 x_i 是图像和平均图像之间的差向量。He 等[9]用奇异值分解的方法求得了特征值问题

$$\boldsymbol{X}\boldsymbol{L}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{w} = \lambda' \boldsymbol{X}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{w}$$

的解。Zhou 等证明了我们可以通过

$$\boldsymbol{\xi} = \boldsymbol{U}\boldsymbol{M}\mathcal{F}^n(t) + \bar{\boldsymbol{\xi}}$$

来插入任意的一帧图像。这种方法成立的前提是各个矢量化的帧图像是相互独立的，这种前提在我们的数据库中是成立的。图 4 展示了时域插值是如何影响时域纹理的。可以看出，插入的图像帧很好地保留了原始图像的变化规律，同时使得整个变化过程显得更加平滑。

对于每一个微表情图像序列，我们计算得到了不同组合参数下的 SLTD 特征，从中选取了使得最终准确率达到最高的一组参数。

3. 数据库 1：约克谎言检测实验(York Deception Detection Test, YorkDDT)

作为心理学研究的一部分，Warren 等[17]设计了一个谎言检测实验，并录制了 20 段视频资料。被试被要求真实地/非真实地描述一段刺激性的/非刺激性的视频片段的内容。刺激性的视频片段描述的是一个外科手术的场景，而非刺激性的视频片段描述的是一个平和的沙滩的场景。在真实场景中，被试被要求描述他们真实看到的视频片段中的场景；在欺骗场景中，被试被要求描述他们没有看到的视频中的场景。作者指出他们在两个场景中都观察到了微表情的出现。这些录像的分辨率为 320×240.

我们得到了原始录制的视频数据资料，并将其中含有微表情的部分截取出来，然后按照实验设置对其进行标记。9 个被试(3 男 6 女)表现出了 18 个微表情：刺激性视频 7 个，非刺激性视频 11 个；真实场景 7 个，欺骗场景 11 个。持续时间最短的微表情在标准 25fps 情形下只有 7 帧。

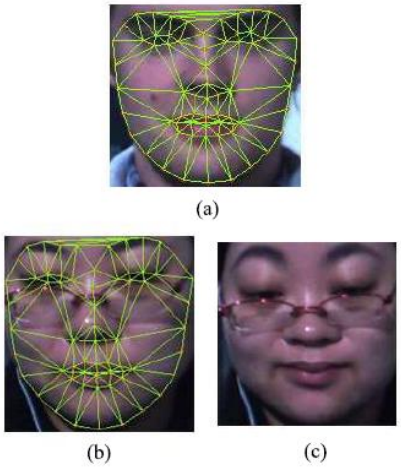


图 A.2 空域归一化到一个标准人脸。

(a)标准人脸以及相应的特征点；(b)示例人脸以及相应的特征点；

(c)经过特征点映射变换的示例人脸。

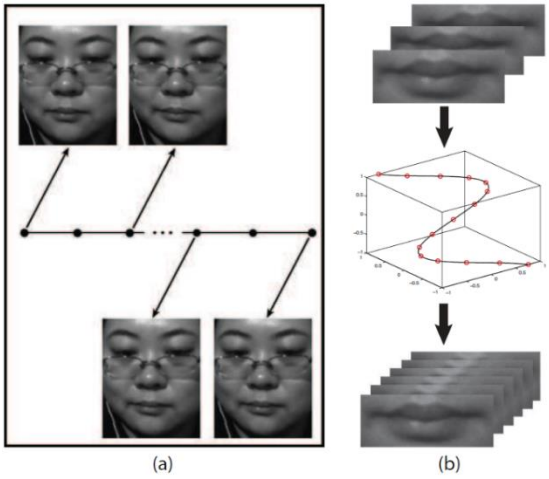


图 A.3 (a)中为微表情的图表示法；

(b)中表示的是视频帧被映射到曲线上，然后采样得到新视频的插值过程。

4. 数据库 2：100fps 自发微表情数据库(SMIC)

YorkDDT 数据的缺点在于数据容量小(small-training-sample-size, STSS)，并且分辨率较低。为了解决这些问题，我们构建了一个新的数据库。

我们使用了 100fps 的摄像机来构建我们的新数据库。我们共招募了 6 名被试(3 男 3 女)，共产生了 77 段微表情，其中有 4 名被试戴眼镜。

数据是在一个装扮成审讯室的房间内采集的。我们应用的是 PixeLINKPL-B774U 摄像机，帧率为 100fps，分辨率为 640×480。每个被试都被要求观看精心挑选的用于诱发厌恶、恐惧、高兴、悲伤和惊讶的视频。实验的步骤为：1、在仔细观看视频的同时尽量压抑自己的表情；2、研究人员将会通过观察面部表情来尝试猜出你所观看的视频是哪一段；3、如果研究人员猜出了被试所在观看的视频，被试将会被要求填写一份长而无聊的问卷作为惩罚；4、每观看完一段视频，被试将会填写一份调查问卷来报告自己观看视频的感受。

审讯室环境、惩罚的设计和强刺激性的视频的目的是为了设置一个高风险和紧张感的环境来诱发被试的感情同时抑制自己的面部表情。Ekman 在之前的文章指出这些条件是诱发微表情的理想条件。虽然我们实验中的高风险设置还不够理想，但是最终实验的结果证实了这些条件的组合成功地诱导了微表情的产生。

我们一共录制了 210 分钟的视频数据，共有 1260000 帧。根据被试的报告，这些数据被两位研究人员整理并进行标记。按照 Ekman 的建议，进行标记的研究人员先一帧一帧地对视频进行观察，再逐渐提升播放速度。持续时间最短的微表情大约有 1/9 秒钟长(100fps 条件下共 11 帧)，平均持续时间为 0.3 秒钟(29 帧)。我们目前正在致力于向数据库中添加更多的数据样本。

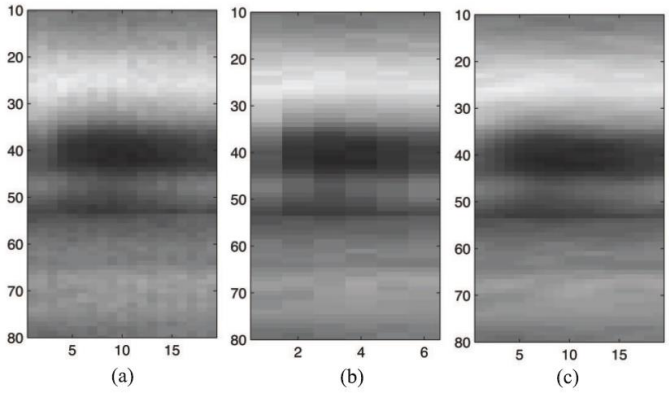


图 A.4 时域插值。图中表示的是原始的，降采样后的和插值后的微表情视频第 121 列数据的时域纹理。(a)中为原始的第 19 帧的纹理；(b)中为降采样过后的纹理；(c)中为(b)中数据在插值之后的结果。

四、 实验及结果

在两个数据集上，我们采用留一法对我们提出的微表情识别算法进行了测试。

我们选用了 LBP-TOP 特征作为 SLTD。多核学习中的参数在算法 1 中也已

经给出。非多核学习的分类结果是在 $SLTD_{8\times8\times1}$ 的条件下得到的，其中 8×8 是图像空域的分块的大小。SVM 方法中，我们应用了 6 阶多项式核。我们将会给出最佳的参数($p\in\Gamma$，$\theta\in T$)和分类器(SVM, RF, MKL)的组合。其中 RF 是随机森林分类器。通过应用滑动窗口方法，我们提出的算法可以完成微表情检测的任务。

1. 实验 1：YorkDDT 数据库

YorkDDT 数据库给微表情识别带来了很多困难和挑战。首先，该数据库中的微表情都是自发的，因此有着很高的多样性；其次，视频中的被试在不断地讲话，所以他们的面部运动就不再局限于微表情；最后就是该数据库中的帧率和分辨率都十分有限。尽管有上述提到的诸多困难，我们将说明我们在第三节提出的算法仍然成功地实现了独立于特定被试个体的微表情识别系统。表 1 展示了采用留一法得到的在 YorkDDT 上的实验结果。

在我们提出的算法的第一步中，我们将微表情和其他面部运动区分开来。为了达到这个目的，我们随机选取了 18 个不包含任何微表情，但是可以包含其他面部运动的视频片段。通过提取 $SLTD_{8\times8\times1}$ 特征并应用支持向量机分类器，我们取得了 65%的准确率。通过将每个视频片段时域插值至 10 帧并应用算法 1 中给出的参数进行多核学习，我们取得了 83%的留一准确率。将帧数提高到 10 帧以上并不会带来任何显著的提高。这种现象产生的原因可能是原始数据过少，插入过多帧的图像只会引入冗余的数据并且导致最终结果变差(纬度灾难)。但是我们可以看到将帧数提高到 10 帧带来了 15%准确率的提升。

实验第二部分的任务是识别微表情的种类。在 YorkDDT 数据库中，我们设置了两种类别：刺激性 vs 非刺激性(emo/¬emo)，欺骗 vs 诚实(lie/truth)。

在没有使用多核学习和时域插值时，我们在分辨欺骗与诚实的实验中使用了在 $SLTD_{8\times8\times1}$ 特征上训练的支持向量机，获得了 47.6%的正确率。通过将帧数插值为 10 帧并且运用多核学习方法，并选取最优的 STLD 参数和核函数，我们将结果提高至 76.2%。同样的，将帧数提高到 10 帧以上并不会带来额外的准确率的提高。在我们尝试的各种分类器中，多核学习总是能够给出最佳的结果。

通过将实验的两部分结合起来，我们能够首先检测到微表情的发生，随后对其进行分类。通过将帧数提高到 10 帧并应用多核学习，我们得到了 83%的微表情检测准确率。总体来看，我们分别在刺激性 vs 非刺激性(emo/¬emo)、欺骗 vs 诚实(lie/truth)的实验中分别得到了 63.2%和 59.3%的准确率。

2. 实验 2：SMIC 数据库

在我们新构建的 SMIC 数据库中，我们解决了 YorkDDT 数据库中存在的分辨率和帧率的问题。在表 2 中我们给出了相关的实验结果。

和 YorkDDT 数据库上结果的一个显著的不同点是，在时域插值仍然能够给出好的结果的同时，之前的分类器不再总能给出好的结果，然而随机森林却有时能够得到比之前更高的准确率。这说明最佳分类器总是和数据本身有关，同时不同类型分类器总是值得研究的。结合多核学习的话有可能会得到更好的效果。

和 YorkDDT 数据库上结果相似的一点是：将帧数插值到 10 帧依然能够给出很好的结果。即使是帧率从 25fps 提高到 100fps，这一点依然成立。事实上，将帧数通过插值提高到 10 帧等效于对原始视频序列进行了降采样。这说明更高的帧率带来的可能大部分是冗余的信息，会对分类器的表现造成不好的影响。

在第一步中我们将微表情和其他面部运动信息区分开来。我们随机选择了 77 段不包含微表情但是包含其他面部运动的视频序列。应用支持向量机，我们得到了 70.3%的检测准确率。应用多核学习能够略微提升一点准确率。通过应用随机森林分类器进行分类并将帧数插值到 10 帧，我们得到了最优的 74.3%的准确率。

在第二步中，我们将表情分为正面/负面这两大类，每一类分别有 18 和 17 个序列。应用支持向量机，我们只得到了较低的 54.2%的准确率。但是时域插值和多核学习将这一准确率提高到了 71.4%。

3. 实验 3：在标准 25fps 帧率数据上进行微表情识别实验

在理想条件下，自发微表情应该能够在标准 25fps 条件下不应用特殊硬件被识别。在本实验中，我们将说明时域插值方法将在标准 25fps 条件下识别微表情变为可能。

通过每 4 帧进行采样，我们将 100fps 的视频数据降采样至 25fps。这导致视频的长度只有 2 至 8 帧。

表 3 中展示了在降采样过的 SMIC 数据库上进行的实验结果。我们在第 3 节中曾经说过，SLTD 特征要求视频长度至少为 7 帧，所以时域插值势在必行。通过用多核学习取代纯支持向量机，我们将微表情检测的准确率提高到了 70.3%（5.3%的提高）。通过将帧数提高到 20 帧，并采用随机森林作为分类器，我们得到了最优的 78.9%的检测准确率。这比 Frank[5]等在他们文章中报告的人工检测准确率还要高。

在微表情识别环节中，通过应用多核学习并将帧数提高到 15 帧，我们得到了 64.9%的分类准确率，这比在 100fps 条件下得到的准确率要差一些（71.4%）。但是在检测环节中，我们可以看到 4 倍的降采样并没有使得准确率降低，一种合理的解释是检测并不需要太多的信息，一点点的面部变化就可以为检测提供充足的证据。只用很少的几帧就可以完成这项工作，更多的数据反而可能成为冗余，降低准确率。但是对于分类来说，更加细微的面部的时空变化是非常必要的，所

以高帧率通常能够带来更到的分类准确率。

更进一步地，我们探究了帧率和识别准确率之间的联系。表 5 中给出了 SMIC 在不同将采样率下的识别准确率。我们发现将帧数插值为 20 帧可以带来稳定的表现。在不进行插值的条件下，帧率的降低会显著地对准确率造成影响。总而言之，应用时域插值的办法，即使是较低的帧率，我们也可以准确地对微表情进行识别。

Phase	Classes	Method	Accuracy (%)
1	detection	SVM	65.0
1	detection	MKL	67.0
1	detection	MKL+TIM10	**83.0**
2	lie/truth	SVM	47.6
2	lie/truth	MKL	57.1
2	lie/truth	MKL+TIM10	**76.2**
2	emo/¬emo	SVM	69.5
2	emo/¬emo	MKL	**71.5**
2	emo/¬emo	MKL+TIM10	**71.5**

表 A.1 在 YorkDDT 数据库上的留一法实验结果。MKL 即为多核学习；
TIMn 指的是将视频序列插值到 n 帧；
emo/¬ emo 代表分类为刺激性微表情和非刺激性微表情

Phase	Classes	Method	Accuracy (%)
1	detection	RF+TIM15	67.7
1	detection	SVM	70.3
1	detection	RF+TIM20	70.3
1	detection	MKL	71.4
1	detection	RF+TIM10	**74.3**
2	neg/pos	SVM	54.2
2	neg/pos	SVM+TIM15	59.8
2	neg/pos	MKL	60.2
2	neg/pos	MKL+TIM10	**71.4**

表 A.2 在 SMIC 数据库上的留一法实验结果。MKL 即为多核学习；
TIMn 指的是将视频序列插值到 n 帧；
RF 意为随机森林分类器；neg/pos 表示负面和正面的微表情

Phase	Classes	Method	Accuracy (%)
1	detection	RF+TIM10	58.5
1	detection	SVM+TIM10	65.0
1	detection	MKL+TIM10	70.3
1	detection	RF+TIM15	76.3
1	detection	RF+TIM20	**78.9**
2	neg/pos	SVM+TIM10	51.4
2	neg/pos	MKL+TIM10	60.0
2	neg/pos	MKL+TIM10	60.0
2	neg/pos	SVM+TIM15	62.8
2	neg/pos	MKL+TIM15	**64.9**

表 A.3 在降采样至 25fps 的 SMIC 数据库上的留一法实验结果。
MKL 即为多核学习； TIMn 指的是将视频序列插值到 n 帧；
RF 意为随机森林分类器；neg/pos 表示负面和正面的微表情

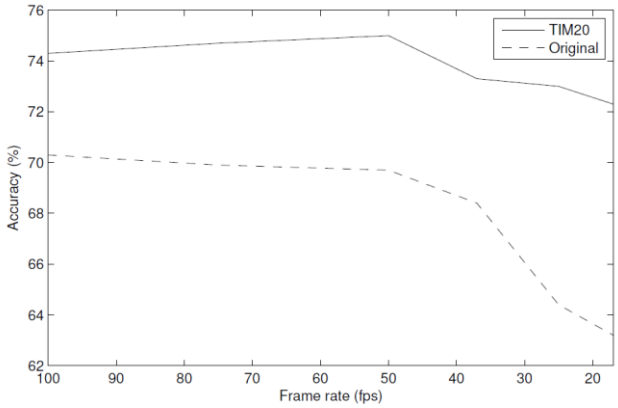


图 A.5 微表情检测准确率随帧率变化的关系。通过降采样倍数来控制帧率。
实线表示在多核学习和插值至 20 帧条件下留一法准确率；
虚线表示的是仅用多核学习的留一法准确率

五、 总结

在本文中我们应用了时域插值方法来解决低帧率的问题，使用了 SLTD 来做为提取的特征，并且使用了包括多核学习、支持向量机和随机森林在内的多种分类器。我们自行设计了表情诱发抑制实验，共有 6 名被试参与，共提取视频序列 77 段。我们在两个数据集上进行了实验，并得到了令人满意的结果。实验结果表明，应用时域插值算法，即使是在标准 25fps 数据库上，我们依然能够得到和 100fps 数据不相上下的微表情检测准确率。我们提出的算法是第一个自发微表情识别算法，取得了不输给人工识别的令人满意的识别准确率。

在未来的工作中，我们准备在以下几个方面进行深入研究：1、进一步扩充 SMIC 数据库中的数据样本；2、将我们算法的准确率和人工识别的准确率进行比较；3、实现微表情的实时处理和识别；4、进一步探索最优的分类器的组合。SMIC

数据库和微表情识别代码已经公布，以鼓励在此领域的相关研究。

翻译原文索引

Pfister T., Li X., Zhao G. et al. *Recognizing spontaneous facial micro-expressions*. IEEE International Conference on Computer Vision, Barcelona, Spain: IEEE Press, 2011. 1449-1456.

附录 B 相关数学推导及证明

B.1 鲁棒 PCA 中基本优化问题的解

在第二章中我们对于鲁棒 PCA 问题的求解进行了讨论，其中我们直接引用了优化问题 $\min_L l(L,S,Y)$ 和 $\min_S l(L,S,Y)$ 的解的形式，在这里我们对其进行推导。

拉格朗日函数为：

$$l(L,S,Y)=\|L\|_*+\lambda\|S\|_1+\langle Y,M-L-S\rangle+\frac{\mu}{2}\|M-L-S\|_F^2 \qquad \text{（B-1）}$$

为了求解 $\min_S l(L,S,Y)$，我们将与 S 无关的项去掉后得到：

$$l(L,S,Y)=\lambda\|S\|_1+\langle Y,M-L-S\rangle+\frac{\mu}{2}\|M-L-S\|_F^2 \qquad \text{（B-2）}$$

加上一个与 S 无关的项，有：

$$l(L,S,Y)=\lambda\|S\|_1+\langle Y,M-L-S\rangle+\frac{\mu}{2}\|M-L-S\|_F^2+\frac{\mu}{2}\left\|\mu^{-1}Y\right\|^2 \qquad \text{（B-3）}$$

运用类似于配方的思想，我们对上式进行如下化简：

$$l(L,S,Y)=\lambda\|S\|_1+\frac{\mu}{2}\left[2\mu^{-1}Y(M-L-S)+\|M-L-S\|_F^2+\left\|\mu^{-1}Y\right\|^2\right] \qquad \text{（B-4）}$$

通过观察，我们发现式（B-4）可以化成如下形式：

$$l(L,S,Y)=\frac{\lambda}{\mu}\|S\|_1+\frac{1}{2}\left\|S-\left(M-L-\mu^{-1}Y\right)\right\|_F^2 \qquad \text{（B-5）}$$

在求解式（B-5）之前，我们要先引用一个优化问题中常用的结论。假设我们有下面这个优化问题：

$$x_0 = \arg\min_x \varepsilon \|x\|_1 + \frac{1}{2}\|x-a\|_F^2 \qquad \text{（B-6）}$$

式（B-6）问题的解即为我们在第二章中提到的软阈值函数：

$$S_\varepsilon(a) = \text{sgn}(a) \cdot \max(|a| - \varepsilon, 0) \qquad \text{（B-7）}$$

这个结论可以通过对式（B-6）中的目标函数直接求导并令倒数等于 0 得到，推导过程比较简单，请读者自行推导。将这个结论带入到式（B-5）中，则可以直接得到解：

$$\min_S l(L,S,Y) = S_{\lambda\mu}\left(M - L + \mu^{-1}Y\right) \qquad \text{（B-8）}$$

即为我们在第二章中给出的解。

对于问题 $\min_L l(L,S,Y)$，我们可以用类似的方法进行求解。更加详细的讨论，读者可以参考 http://blog.csdn.net/abcjennifer/article/details/8572994。

B.2 完整的调查问卷

姓名		年龄		性别	
观看视频名称					
正面内容?			正面情感?		
负面内容?			负面情感?		
是否撒谎					
诱发情感	愤怒 高兴		悲伤 惊讶	恶心 恐惧	
强度等级	-5 -4 -3 -2 -1 0 1 2 3 4 5				
观看视频感受					

表格的设计参考了文[6][7]中的实验设计，感兴趣的读者可以再深入了解。

在学期间参加课题的研究成果

本文中在 Extended Cohn-Kanade Dataset (CK+)数据库上的工作已经发表在会议 International Conference on Smart Computing 2014（SMARTCOMP 2014）上，题目为 Facial Expression Recognition and Generation using Sparse Autoencoder。主要内容为基于稀疏自编码器的表情识别，除此以外还对表情进行了“转化”和“提纯”的操作，取得了较好的效果。本文也被 IEEE Digital Library 收录，链接为 http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7043849&filter%3DAND%28p_IS_Number%3A7043829%29。

综合论文训练记录表

学生姓名		学号		班级	
论文题目					
主要内容以及进度安排	指导教师签字： 考核组组长签字： 年　　月　　日				
中期考核意见	考核组组长签字： 年　　月　　日				

指导教师评语	指导教师签字：_____ 年　　月　　日
评阅教师评语	评阅教师签字：_____ 年　　月　　日
答辩小组评语	答辩小组组长签字：_____ 年　　月　　日

总成绩：_____

教学负责人签字：_____

年　　月　　日