学校代码：10286
分 类 号：TP393
密　　级：公开
U D C：621.39
学　　号：160916

东南大学

SOUTHEAST UNIVERSITY

工程硕士学位论文

基于人脸防伪的移动智能终端安全性研究

（学位论文形式：应用研究）

研究生姓名：**魏一鸣**

导 师 姓 名：**宋宇波**

刘　凯

申请学位类别 **工程硕士**　　学位授予单位 **东 南 大 学**

工程领域名称 **电子与通信工程**　　论文答辩日期 **2019 年　月　日**

研究方向 **信息安全**　　学位授予日期 **20　年　月　日**

答辩委员会主席＿＿＿＿＿＿　　评　阅　人＿＿＿＿＿＿

＿＿＿＿＿＿

20　年　月　日

東南大學

硕士学位论文

基于人脸防伪的移动智能终端安全性研究

专 业 名 称：电子与通信工程

研究生姓名：魏一鸣

导 师 姓 名：宋宇波

刘 凯

RESEARCH ON INTELLIGENT TERMINAL SECURITY BASED ON FACE ANTI-SPOOFING

A Thesis Submitted to

Southeast University

For the Academic Degree of Master of Engineering

BY

Wei Yiming

Supervised by

A.Prof Song Yubo

and

Dr. Liu Kai

School of Information Science and Engineering

Southeast University

2019/08/16

东南大学学位论文独创性声明

本人声明所呈交的学位论文是我个人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得东南大学或其它教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

研究生签名：_____日期：_____

东南大学学位论文使用授权声明

东南大学、中国科学技术信息研究所、国家图书馆、《中国学术期刊（光盘版）》电子杂志社有限公司、万方数据电子出版社、北京万方数据股份有限公司有权保留本人所送交学位论文的复印件和电子文档，可以采用影印、缩印或其他复制手段保存论文。本人电子文档的内容和纸质论文的内容相一致。除在保密期内的保密论文外，允许论文被查阅和借阅，可以公布（包括以电子信息形式刊登）论文的全部内容或中、英文摘要等部分内容。论文的公布（包括以电子信息形式刊登）授权东南大学研究生院办理。

研究生签名：_____导师签名：_____日期：_____

摘要

人脸识别技术应用于手机、平板电脑等移动智能终端的现象越来越普遍。然而通过打印照片、重播视频以及制作掩膜等方式能够伪造人脸绕过移动智能终端的识别机制，从而对其身份识别系统造成极大的威胁。因此，如何检测虚假人脸是目前移动智能终端人脸防伪技术的研究热点。传统的人脸防伪技术以人工设计特征（Hand-crafted Features）作为区分人脸真伪的依据，由于加入了过多的人为限制，通常适用于防范打印照片攻击的模型难以用于抵御重播视频攻击，存在着对于不同种类攻击通用性差的问题。为了提高算法的通用性，将深度学习引入人脸防伪领域。然而，现有的基于深度学习的模型存在三大问题：一是，对于人脸姿态、表情和光照条件变化的适应性差；二是，缺乏明确的监督信息，模型学习到的并不是区分人脸真伪的关键特征，导致检测的准确度不高；三是，对于人脸视频的采集设备和方式等比较敏感，对不同数据集的泛化性差。针对上述问题，本文提出了一种静态特征和动态特征相融合的人脸真伪检测方案。主要工作和创新点如下：

1. 本文所提的静态特征和动态特征相融合的人脸真伪检测方案首先通过移动智能终端的摄像头采集人脸视频，然后对连续帧进行采样；提取其中第一帧人脸的深度图作为静态特征；再引入光流引导特征，通过分析采样后所有帧中人脸的动态变化得到动态特征；最后将两者融合并以此作为依据区分人脸的真伪。该方案通过设置融合系数 λ，控制静态特征和动态特征的相对重要程度，综合了静态特征和动态特征各自的优势，使得融合特征同时具有较强的通用性、适应性和泛化性。

2. 针对现有算法对不同人脸姿态、表情和光照条件适应性差以及检测准确度不高的问题，本文提出了一种基于 3D 点云图和深度图的人脸静态特征提取方法，先重建 3D 点云图，再基于此提取人脸深度图。为了增强对于不同人脸姿态、表情和光照条件的适应性，该方法采用 PRNet 算法，通过设置专门的 UV 空间映射，重建图像中人脸的 3D 点云图。为了提取对于真伪人脸区分度高的特征，进而提高检测的准确度，该方法将 3D 点云图归一化，以此作为真实标记指导深度网络模型进行训练，引入了明确的监督信息，提取人脸的深度图作为静态特征。

3. 针对现有算法对不同数据集泛化性差的问题，本文提出了一种基于时序关系和注意力机制的人脸动态特征提取方法。该方法采用光流引导特征（Optical Flow Guided Features）表示连续帧中人脸的短期动态变化，采用 CGRU 模块对短期动态变化进行积累得到长期动态变化，使得在获得连续帧时序关系的同时，又保留了人脸的空间信息；同时采用注意力机制根据重要程度对不同时间点的视频帧加权组合得到人脸的动态特征，对不同数据集具有较强的泛化性能。

4. 为了验证本文所提方案的有效性，本文利用 CASIA-MFSD、Replay-Attack 和 SiW 三个数据集进行了数据集内部测试和数据集交叉测试，并且对单一特征和融合特征的效果进行了对比。实验结果表明，基于静态特征或动态特征的人脸真伪检测的错误率都不同程度地低于现有算法的错误率；静态特征在数据集内部测试时表现较好，动态特征在数据集交叉测试时表现较好；而融合特征相较于单一的静态特征或动态特征错误率更低，其在数据集内部测试的平均分类错误率最低为（0.18±0.23）%，在数据集交叉测试的半错误率最低为 17.0%，能够更好地用于区分人脸真伪。

关键词：移动智能终端、人脸防伪、深度图、注意力机制、融合特征

Abstract

Face recognition technology is applied to the mobile phone of tablets and other mobile intelligent terminal is becoming more popular. However, printing pictures, replaying video and making masks can forge face recognition mechanism of mobile intelligent terminal, which poses a great threat to its identification system. Therefore, how to detect the false face is a hotspot of mobile intelligent terminal face anti-counterfeiting technology.Traditional face anti-counterfeiting technology in hand- crafted features as the basis of authenticity that makes one face different from another, because of the added artificial restrictions, usually applied to prevent printing photos attacks model is difficult to to resist replay attack, the video there are for different kinds of poor universality problems in order to improve the algorithm of generality, deep learning is used in face security domain.However, there are three major problems with existing models based on deep learning: First, poor adaptability to changes in facial posture and lighting conditions. Secondly, there is a lack of clear supervision information. What the model learns is not the key feature to distinguish the human faces, which leads to the low accuracy of detection. Third, it is sensitive to face video collection equipment and methods, and poor generalization of different data sets. To solve the above problems, this paper proposes a face authenticity detection scheme combining static features and dynamic features. The main work and innovation points are as follows:

1. The face authenticity detection scheme combining static features and dynamic features proposed in this paper firstly collects face video through the camera of mobile intelligent terminal, and then samples the continuous frames. The depth map of the first frame is extracted as the static feature. Then the optical flow guidance feature is introduced and the dynamic features are obtained by analyzing the dynamic changes of human faces in all frames after sampling. The solution is to use the fusion coefficient lambda to control the relative importance of static features or dynamic features. It integrates the advantages of static features and dynamic features, making the fusion features have strong universality and generalization at the same time.

2. In view of the existing algorithms adapted to different face postures,expressions and light conditions are poor and the problem of detecting accuracy is not high, this paper puts forward a kind of based on 3 d point cloud image and depth map of static face feature extraction method, the reconstruction of 3 d point cloud picture first, then based on the extracted face depth map in order to enhance to different facial expressions and the adaptability of light conditions, the method adopts the PRNet algorithm, by setting up

special UV space mapping, in the face of 3 d point clouds reconstruction images In order to extract features with high degree of discrimination for true and false faces, and thus improve the accuracy of detection, this method normalizes 3D point cloud maps, which are used as real markers to guide the deep network model for training, introduces explicit supervision information, and extracts the depth maps of faces as static features.

3. In view of the existing algorithms for different data sets with poor generalization problem, this paper proposes a mechanism based on temporal relationship and attention of the face is the method of dynamic feature extraction method using optical flow guide features consecutive frames in the face of short-term dynamic changes, said of the short-term dynamic changes by CGRU module accumulation get dynamic change for a long time, has given in successive frames of temporal relations at the same time, and keep the face of the space information; At the same time, the attention mechanism is used to weighted video frames at different time points according to the importance to obtain the dynamic features of the face, which has a strong generalization performance for different data sets.

4. In order to verify the effectiveness of the scheme, this paper based on CASIA-MFSD, Replay Attack and SiW three data sets within the data set tests and cross test data sets, and the characteristics of single feature and fusion compares the experimental results show that the effect of face false detection based on static or dynamic characteristics of error rate is lower than the existing algorithm of error rate in different degrees. The static features are better in the data set internal test and the dynamic features are better in the data set cross test. Compared with the single static feature or dynamic feature, the fusion feature has a lower error rate, with the lowest average classification error rate (0.18±0.23) % in the internal test of the data set and the lowest semi-error rate 17.0% in the cross-test of the data set, which can be better used to distinguish face authenticity.

Key words: mobile intelligent terminal, face anti-spoofing, depth map, attention mechanism, fusing feature

目录

第1章 绪论

1.1 课题背景及研究意义

近年来，随着移动终端设备和互联网技术的飞速发展，人们的生活方式发生了极大的变化。从人们的衣食住行，到投资理财都可以通过互联网完成，而以手机、平板电脑为代表的移动智能终端因为其便携性，成为了最常用的入网设备之一。这些移动智能终端上存储了用户的大量的个人信息和隐私数据，一旦泄露，可能会给用户的个人隐私、财产安全各个方面造成难以估量的损失。

随着人工智能的高速发展，生物识别技术[1]渐渐取代了原来的密码认证，成为了主流的移动智能终端身份认证方式。所谓生物识别技术，是指利用人类的生物特征进行身份认证，通常包括指纹识别、人脸识别、虹膜识别等。因为人的指纹、脸、虹膜这些生物特征通常是终身不变的，通过专门的设计，这些特征可以作为身份的唯一标识。其中，人脸认证在生物识别技术中最常用而且安全性较高。

然而，移动智能终端上的人脸认证并非绝对安全，也会受到各种伪造身份手段的威胁，其中威胁最大的是人脸欺骗攻击（face spoofing attacks）[2][3]，在很多文献里也叫表示攻击（presentation attacks）[4][5]。为了和研究内容更加贴合，本文一律采用“人脸欺骗攻击”来表述。所谓人脸欺骗攻击，是指利用伪造的人脸身份来通过人脸认证系统，它具有如下特点：

(1)攻击成本低：打印攻击和重放攻击是两种最为常见的人脸欺骗攻击。前者将真实用户的人脸图像打印成高清彩色纸质图片，在认证系统的摄像头前展示；后者把真实用户人脸拍摄成视频，在认证时重新播放。这两种攻击利用常见的打印设备和播放设备就可以实施，成本较低。

(2)攻击目标广：常见的搭载了人脸认证系统的移动智能终端都可能受到攻击，不受设备的系统和型号等的限制。比如 iPhone 手机、三星手机、iPad 平板电脑、MacBook 笔记本电脑等。

为了保护移动智能终端安全，国内外的学者们提出了很多针对人脸欺骗攻击的防范手段，统称为人脸防伪（Face Anti-Spoofing）技术[6][7]。人脸防伪技术通常先提取图像或视频中的人脸特征，这些特征在真实人脸和人脸欺骗攻击场景下具有较大的不同，所以能够作为区分人脸真伪的依据。因此，如何提取区分度高的特征是人脸防伪技术的关键所在。

近年来，深度学习（Deep Learning）的出现为人脸防伪中提取人脸特征提供了一种崭新的思路。深度学习通过多层神经网络的深度模型的设计，能够对数据进行表征学习，得到表征原始数据的各种特征，用于解决不同的任务，它在图像识别、图像分类、语义分割、机器翻译、视频理解等众多领

域都取得了重大成功。深度学习能够提取更有泛化性、更加鲁棒的特征，为人脸防伪带来了新的思路，对提升移动智能终端的安全性有重大意义。

将深度学习理论应用到人脸防伪技术中，提高人脸防伪技术中对人脸特征提取的准确性，是本课题尝试解决的问题。通过对这一课题的研究，能够提升人脸防伪技术的性能，优化人脸识别认证方式，营造安全可靠的移动智能终端使用环境。

1.2 国内外研究现状

本文的研究目标是将人脸防伪技术与深度学习算法结合，在移动智能终端背景下研究性能更好的人脸真伪检测手段。人脸防伪技术和深度学习算法两者目前都是国内外研究的热点，存在不少经典的理论研究，下面分几个方面介绍：

（1）传统的人脸特征提取算法

为了防范人脸欺骗攻击，学术界和工业界提出了大量的防伪算法来区分真假人脸。传统的提取人脸特征的算法有局部二值模式（LBP，Local Binary Pattern）[8][9]、方向梯度直方图（HOG，Histogram of Oriented Gradient）[10]、尺度不变特征变换（SIFT，Scale-Invariant Feature Transform）[11]等。

部分研究文献分别将 LBP、HOG、SIFT 这些含有图像纹理信息的人工设计特征（Hand-crafted Features）作为关键信息，通过训练简单的分类器来区分真伪[12][13]。这几类方法存在的缺点如下：首先，LBP 对图像的方向信息比较敏感，HOG 对图像的噪声比较敏感，SIFT 的生成过程缓慢，都有一定的缺点；其次，它们都属于人工设计特征，将人脸的纹理等信息提取出来作为特征，需要根据不同的实验数据库、不同的应用场景等人为地设计和选择，主观性较强，经验性较强，过程比较麻烦；再者，由于加入了过多的人为限制，通常适用于防范打印攻击的模型难以用于抵御重放攻击，对于不同种类攻击的通用性差。

另一部分研究文献则从活体检测的角度进行了设计，利用人脸的动作信息来判别。例如把眨眼动作作为线索[14][15]，或者依据嘴唇动作区分真伪[16]，但它们过于依赖眼部或嘴部动作这些局部信息，只适用于检测打印的人脸照片的场景，无法防范重放视频和 2D、3D 面具的攻击。而且由于采用的是人工设计特征，是针对实验的数据集特别制作的，对其他数据集适应性较差，泛化性能上的劣势导致它们很难应用到工程实践中。

人工设计特征主要关注人脸的局部信息，对不同的人脸欺骗攻击通用性较差，不适合作为通用的人脸真伪检测框架。检测框架应该能够提取真假人脸图像的可区分性特征，从而根据这些特征来辨别人脸真伪，作为检测框架的基础，全面、有效的真伪人脸特征至关重要，而人脸的纹理、眨眼和嘴唇动作等特征包括的局部信息不足以作为区分真伪的本质特征。

（2）基于深度学习的人脸特征提取

近年来，深度学习也逐渐被应用到了人脸识别领域当中。深度学习通过多层神经网络的深度模型的设计，能够对数据进行表征学习，得到表征原始数据的各种特征，用于解决不同的任务。深度学习中最常用的算法模型是卷积神经网络（CNN，Convolutional Neural Network）和循环神经网络（Recurrent Neural Network）。CNN 是一种专门用来处理具有类似网络结构数据的神经网络，通过对内部结构的特定设计可以用来解决不同的任务。CNN 可以粗略地理解为一个从输入到输出的复杂数学函数，比如就分类任务而言，输入是各种动物的图片，输出为动物的类别，CNN 通过对大量数据的训练来拟合，将输入图片直接映射到输出类别的复杂函数。RNN 是一种用于处理序列数据的神经网络，与 CNN 类似，也可以自动学习数据的特征，只是 RNN 更加关注于数据随时间变化的信息。比如，输入数据是一个人物的视频，RNN 通过学习视频连续帧中人物肢体动作的变化，来对人物的行为进行估计和分类。

为了解决传统人工设计特征存在的问题，部分研究文献[17][18] [19]中提出了基于卷积神经网络（CNN）“端到端”的深度学习方法，旨在使用 CNN 自动学习真假人脸图像特征，并且能够广泛地适用于不同的场景。虽然基于 CNN 的方法用自动学习特征取代了人工设计特征，使得特征提取更加方便，并且在一定程度上提高了算法对于不同种类攻击的通用性，但是这些方法并没有考虑不同数据集采集人脸视频时人脸的姿态、表情和光照条件差异很大，只是将人脸防伪作为简单的二分类问题，真脸为 1，假脸为 0，训练简单的以 softmax 为目标函数的神经网络做分类。因此，基于 CNN 学习特征的方法学习的过程仍然存在需要改善的地方：一方面，对于人脸姿态、表情和光照条件变化的适应性差；另一方面，这些方法没有明确的监督信息，学习到的特征很可能与人脸真伪关联性不大，导致检测的准确度不高。因此，提取适应性强的特征、设计出具有明确监督信息的模型是目前亟待解决的重点问题。

将长短期记忆（Long Short-Term Memory，简称 LSTM）单元[20]引入 CNN，可以组成混合架构的神经网络，来学习能够描述人脸时间域信息的动态特征，从而辨别人脸的真伪。LSTM 单元可以通过使用输入门、输出门、遗忘门来控制修改访问和存储内部状态，从而从输入序列中发现长期的时间关系。该方案利用了 LSTM 的这些特点，从输入的视频帧序列中学习时间域信息从而描述人脸的动作模式，得到人脸的动态特征。然而，基于 LSTM-CNN 架构学习特征的网络，容易丢失图像中的空间信息，导致输出的特征对于不同数据集人脸视频的采集设备和方式等比较敏感，对不同数据集的泛化性差。虽然将 LSTM 单元引入网络架构可以很好地描述和记录视频帧的时间信息，但是空间信息也很重要，所以网络设计还要进一步优化。因此，需要改进 LSTM-CNN 网络，使得其在捕捉连续帧的时间域信息的同时，又不丢失空间域信息，提高动态特征对于不同数据集的泛化性能。

从国内外的研究现状可以看出，目前人脸识别技术仍有许多需要完善的地方，还存在传统人工设计特征通用性差、CNN 算法模型缺乏监督且适应性差、动态特征泛化性差等问题，这些都是本文的主要研究内容。

1.3 研究内容与结构安排

本课题的题目为“基于人脸防伪的移动智能终端安全性研究”，主要目标为将深度学习与人脸防伪相结合，针对现有人脸特征提取研究中存在的不足，设计通用的人脸真伪检测框架，提高智能终端环境下的人脸检测性能。

本文完成的主要工作包含以下五部分：

（1）对移动智能终端环境下人脸防伪的研究背景进行介绍，并对国内外的研究现状进行调研和总结，明确本课题需要解决的关键问题；

（2）设计了基于融合特征的人脸真伪检测方案。采用静态特征和动态特征相融合的设计理念，介绍了方案的整体架构，同时对特征融合模块和分类模块进行了介绍；

（3）设计了人脸静态特征的提取方法。采用 3D 点云图和深度图技术，对静态特征提取方法进行设计，并分别介绍框架的各个子模块的功能与具体实现；

（4）设计了人脸动态特征的提取方法。利用卷积门控单元提取视频帧的光流引导特征，并以此作为动态特征提取方法进行了设计，同时介绍了子模块的实现方案；

（5）对检测框架进行实验验证与结果分析。完成检测实验的具体训练细节和参数设置，并对检测框架的功能和性能进行比较和评估，确保检测框架能够满足人脸防伪的性能需求。

全文组织上，本论文一共分为七个章节，各个章节内容安排如下：

第一章为本文的绪论部分。该章节主要介绍移动智能终端人脸防伪技术的研究背景与课题意义，接着对国内外关于人脸特征提取、人脸防伪领域中深度学习的应用情况进行了总结概括，最后对本文的主要研究内容和论文组织结构进行介绍。

第二章为原理概述部分。本章主要对人脸防伪课题涉及到的背景技术进行介绍，包含了人脸防伪中需要应对的主要攻击方式与人脸真伪检测的具体内容。同时也对深度学习中的卷积神经网络与循环神经网络算法进行了介绍。

第三章为基于融合特征的人脸真伪检测方案设计部分。本章首先从整体角度介绍了检测方案的组成，随后依次介绍了特征融合模块和分类模块的具体设计。

第四章为人脸静态特征提取方法的设计部分。本章首先从整体角度介绍静态特征提取方法的整体架构，随后依次介绍基于深度图技术的各个子模块的具体功能和实现方案。

第五章为人脸动态特征提取方法的设计部分。本章首先从整体角度介绍动态特征提取方法的整体架构，随后分别介绍各个子模块的具体功能和实现方案。

第六章为检测框架的实验验证部分，本章首先对检测评估标准进行介绍，随后对不同检测方案的实验细节与参数进行设置，并对实验的结果进行分析和评估，同时对静态特征、动态特征和融合特征的实验结果进行了对比分析，验证融合特征的有效性。

第七章为总结和展望部分。本章主要对本课题主要完成的工作进行总结和回顾，并对研究中存在的不足与后续的改进方向进行介绍。

第2章 背景技术介绍

人脸防伪与深度学习涉及大量的理论知识，本章将着重对论文涉及的关键原理和算法进行介绍，为后续的具体方案设计提供原理基础。本章首先对常见的人脸欺骗攻击方式和主要的人脸防伪方案进行介绍，随后分别对深度学习中卷积神经网络和循环神经网络两种模型的原理和流程进行介绍，为后续的人脸真伪检测方案实现提供理论支持。

2.1 人脸防伪简介

常见的人脸欺骗攻击有三大类，分别是打印攻击（print attack）、重放攻击（replay attack）和掩膜攻击（mask attack）。根据不同类型的攻击方式，学术界和工业界提出了一系列人脸防伪（face anti-spoofing）方案。

2.1.1 人脸欺骗攻击

通常的人脸欺骗攻击有三种：一是，打印用户的 2D 照片进行攻击；二是，在电子设备上重放人脸视频进行攻击；三是，制作用户的 3D 掩膜进行攻击。

（1）打印攻击

随着网络社交媒体的兴起，获取合法用户的照片十分容易，所以打印攻击也成为了最常见的人脸欺骗攻击手段。而且，在先进的照相设备的帮助下，窃取到的用户照片分辨率和质量都很高，使得打印攻击对人脸识别造成了很大的威胁。如图 2-1，虽然打印出来的通常只是静态的正脸照片，但是可以通过弯曲照片来模拟侧脸的情形，或者通过抠除眼睛部位让攻击者能够模拟用户眨眼，从而骗过人脸识别系统。

图 2-1 打印攻击

（2）重放攻击

重放攻击，指的是在电子设备上播放目标用户的人脸视频，因为视频包含了用户动作、姿态、

表情变化等信息，相较于静态图片更加与真人贴近，所以与打印攻击相比，重放攻击更具有威胁性。如图 2-2 所示，重放攻击凭借高清的播放设备更具有迷惑性。

图 2-2 重放攻击

（3）掩膜攻击

掩膜攻击分为 2D 和 3D 两种。2D 掩膜类似于图 2-1，但与照片不同，掩膜与具有纹理的人脸皮肤更贴近。由于 3D 打印技术的出现，3D 掩膜的制作成为了可能，3D 掩膜与真实人脸非常接近，已经达到了以假乱真的效果，如图 2-3，对人脸识别来说是一个很大的挑战。

图 2-3 3D 掩膜攻击

2.1.2 人脸防伪方案

学术界和工业界提出了很多人脸防伪的方案，因为实验效果的优越性受到了广泛的关注，按照它们依据的特征的不同，大致可分为基于静态特征的和基于动态特征的两大类。

基于静态特征[21]的方案主要关注人脸的局部信息，将具有图像纹理信息的人脸特征作为关键信息进行真伪识别。静态特征通常又可以具体分为人工设计特征类[22]和学习特征类[23][24]两种方案。人工设计特征类方式简单，通常采用人工方式根据数据库和具体的场景进行选择和设计，具有较大的主观性和经验性，缺乏通用性，且当实际场景和数据库场景的光照环境有区别时，并不具备很好的适应性。学习特征类采用自动学习取代了人工选择，使得特征提取更加方便，并且在一定程度上提高了泛化性能。但是学习特征类缺乏监督信息，无法得知学习到的特征是否与人脸真伪有关。静态特征方案通常用于防范打印攻击，应对重放攻击和掩膜攻击比较困难。

基于动态特征[25]的方案遵循的是另外一种思路。考虑到很多情况下欺骗攻击是通过重播录制视频进行的，所以真实人脸的视频和虚假人脸的视频在视频帧序列中动作模式可能不同。比如，录制

的人脸照片的视频帧中没有眨眼和嘴巴张合等面部动作。如果攻击设备是手持的，那么在重播的攻击视频中不可避免地会出现额外的动作模式，比如手抖，这些运动线索对于区分攻击意图非常有价值。同时，人脸区域与背景之间的相对运动也有很大的参考价值，通过获取连续帧的时间域信息，进而提取出动态特征，从而描述人脸在视频中的运动模式，也有利于人脸真伪检测。一般在应对重放攻击和掩膜攻击时，会采用动态特征的方案。

2.2 深度学习简介

深度学习通过多层神经网络的深度模型的设计，能够对数据进行表征学习，得到表征原始数据的各种特征，用于解决不同的任务。深度学习的出现为人脸防伪中提取人脸特征提供了一种崭新的思路。常见的深度学习网络结构有卷积神经网络和循环神经网络，本节将对两者分别进行介绍。

2.2.1 卷积神经网络

卷积神经网络[26]是深度学习中一种经典的层次模型，它支持多种原始输入数据，比如 RGB 图像、原始音频数据等。卷积神经网络通过卷积（convolution）操作、池化（pooling）操作和非线性激活函数（non-linear activation function）映射等一系列操作的层层堆叠，将高层语义信息逐层地从原始数据输入层中抽取出来，层层抽象，这个过程称为前馈运算（feed-forward）[27]。在这个过程中，不同类型的操作在卷积神经网络中通常称为“层”，即卷积操作对应“卷积层”，池化操作对应“池化层”等。最终，卷积神经网络的最后一层将整个目标任务（分类、回归等）形式化为一个目标函数（objective function）。进而通过计算预测值与真实值之间的误差（loss），利用反向传播算法（back-propagation algorithm）[28]，把误差从最后一层逐层向前反馈，更新每一层的参数，并在更新参数后再次前馈，如此往复，直到网络模型收敛，完成整个模型训练的过程。

因此，卷积神经网络由以下几个重要的基本组件组成：卷积层、池化层、激活函数和目标函数等。

（1）卷积层

在卷积神经网络中通常仅涉及离散卷积的情形，下面以二维离散卷积运算为例作简要说明。

若假设输入图像（数据）为 5×5 矩阵 B，其对应的卷积核（convolution kernel）为一个 3x3 的矩阵 A，假设每进行一次卷积操作，卷积核移动一个像素位置，即卷积步长（stride）为 1。

$$A=\begin{bmatrix}1&0&1\\0&1&0\\1&0&1\end{bmatrix},B=\begin{bmatrix}1&2&3&4&5\\6&7&8&9&0\\9&8&7&6&5\\4&3&2&1&0\\1&2&3&4&5\end{bmatrix} \tag{2.1}$$

第一次卷积操作从输入图像的左上角像素开始，由卷积核参数与对应位置图像像素逐位相乘后累加作为一次卷积操作的结果。在步长为 1 时，卷积核按照步长大小在输入图像上从左至右从上到下将卷积操作进行下去，最终输出 3×3 大小的矩阵 C，即卷积特征（convolution feature），并将该结果作为下一层的输入。

$$C=\begin{bmatrix}27&28&29\\28&27&16\\23&22&21\end{bmatrix} \tag{2.2}$$

相似地，在三维情况下，假设输入图像的高为 H，宽为 W，则卷积层 L 的输入张量（tensor）可表示为$x^l \in \mathrm{R}^{H^l\times W^l\times D^l}$，该层卷积核为$f^l \in \mathrm{R}^{H\times W\times D^l}$。三维输入时卷积操作实际上是在所有通道上（即 D^l）对应地进行二维卷积，最终该位置的卷积结果为所有 HWD^l 个元素卷积后的和。进一步地，如果有 N 个类似于f^l的卷积核，则同位置上卷积输出的维度为 1x1x1xN，N 为第 l+1 层特征的通道数。因此，第 l+1 层与第 l 层的卷积关系如下：

$$y_{i^{l+1},j^{l+1},d}=\sum_{i=0}^{H}\sum_{j=0}^{W}\sum_{d^l=0}^{D^l} f_{i,j,d^l,d}\times x^l_{i^{l+1}+i,j^{l+1}+j,d^l} \tag{2.3}$$

其中，$f_{i,j,d^l,d}$是第 l 层学习到的权重（weight），(i^{l+1},j^{l+1})是卷积结果对应的位置坐标，各自对应的范围如下式所示：

$$0\le i^{l+1}<H^l-H+1=H^{l+1} \tag{2.4}$$

$$0\le j^{l+1}<W^l-W+1=W^{l+1} \tag{2.5}$$

因为 i,j,d 都是变量，则可以看出在不同位置不同通道所有输入的权重都是一致的，这就是所谓的“权值共享”（weight sharing）特性。此外，卷积核大小和卷积步长是卷积操作的两个非常重要的超参数。通过在训练过程中反复调整和优化，可以给最终模型的性能带来提升。

从以上的叙述中可以看出，卷积是一种局部操作，通过一定大小的卷积核作用于局部图像区域获得图像中的局部信息，然后每一块的局部信息组成整张图像的语义信息，在经过层层抽象得到高层语义信息，最终作为图像的特征信息。

（2）池化层

通常情况下，卷积神经网络中的池化操作分为两种，平均值池化（average-pooling）和最大值

池化（max-pooling）。与卷积操作不同的是，池化层不需要学习参数，池化操作包含的超参数与卷积类似，包含有核大小和池化步长。

第 l 层池化的核可以表示为$p^l \in \mathrm{R}^{H\times W\times D^l}$，那么平均值（或最大值）池化就是将池化核覆盖范围内的所有数值的平均值（或最大值）作为操作的结果，如下所示：

平均值池化：

$$y_{i^{l+1},j^{l+1},d}=\frac{1}{HW}\sum_{0\le i<H,0\le j<W}x^{l}_{i^{l+1}\times H+i,j^{l+1}\times W+j,d^l} \tag{2.6}$$

最大值池化：

$$y_{i^{l+1},j^{l+1},d}=\max_{0\le i<H,0\le j<W}x^{l}_{i^{l+1}\times H+i,j^{l+1}\times W+j,d^l} \tag{2.7}$$

其中，$0\le i^{l+1}<H^{l+1}$, $0\le j^{l+1}<W^{l+1}$,$0\le d<D^{l+1}=D^l$。以最大值池化为例，矩阵 B 经 16 次操作后得到 4×4 的矩阵 D,即池化特征。

$$D=\begin{bmatrix}7&8&9&9\\9&8&9&9\\9&8&7&6\\4&3&4&5\end{bmatrix} \tag{2.8}$$

由上可以看出，池化操作在功能上等同于降采样（down-sampling），是对人的视觉系统的一种模拟，是一个对视觉输入的对象进行降维和抽象的过程。池化操作主要有以下三种作用：

一是特征不变性（feature invariant）。池化操作使模型更加关注是否存在某些特征而非具体位置。可以看作是一种很强的先验知识，使得特征有一定的自由度，能够容忍微小的位移变化，即平移不变性。

二是特征降维。由于池化操作相当于降采样，将原输入数据的一个子区域对应为一个元素，缩减了原空间范围的维度，因此模型可以提取更广阔范围的特征。

三是防止过拟合（overfitting）。对原数据的降采样，同时也减小了池化后下一层的输入大小，进而减小计算量和参数个数，防止过拟合，提高泛化性能。

（3）激活函数

激活函数（activation function）层又称为非线性映射层[29]，众所周知，单纯的线性操作层的堆叠仍然只是线性映射，无法组合成更加复杂的数学函数。因此，激活函数的引入能够提升整个网络的表达能力。激活函数有很多，如 Sigmoid、tanh、ReLU、ELU、Maxout 等等，最常用的是 Sigmoid 型激活函数和 ReLU 激活函数。

直观上，激活函数是对生物神经元特性的一种模拟，即接受输入信号并对应产生输出。在神经

科学中，生物神经元常常有一个控制状态的阈值，当某一神经元得到的输入信号的累积效果超过了这个阈值，该神经元就被激活成兴奋状态；否则，就呈现抑制状态。Sigmoid 型激活函数的广泛应用正是因为模拟了这一生物过程。

Sigmoid 型函数也可以称为 Logistic 函数，数学表达式如下所示：

$$\sigma(x)=\frac{1}{1+\exp(-x)} \tag{2.9}$$

函数和函数梯度图如图 2-4 所示。可以看出，在 Sigmoid 型函数作用后，输出响应的值域为[0,1]，0 对应的是神经元的“抑制状态”，1 则对应其“兴奋状态”。然而，在 Sigmoid 函数两端，在大于 5（或小于-5）时，无论值多大（或多小）都会被置为 1（或 0）。由此就会产生一个很严重的问题，即梯度的“饱和效应”。在 Sigmoid 函数的大于 5（或小于-5）的部分梯度趋向于 0，这样会导致采用 Sigmoid 函数的神经网络在误差反向传播的过程中，极易发生梯度消失（Vanishing Gradient）现象，导致网络难以训练。

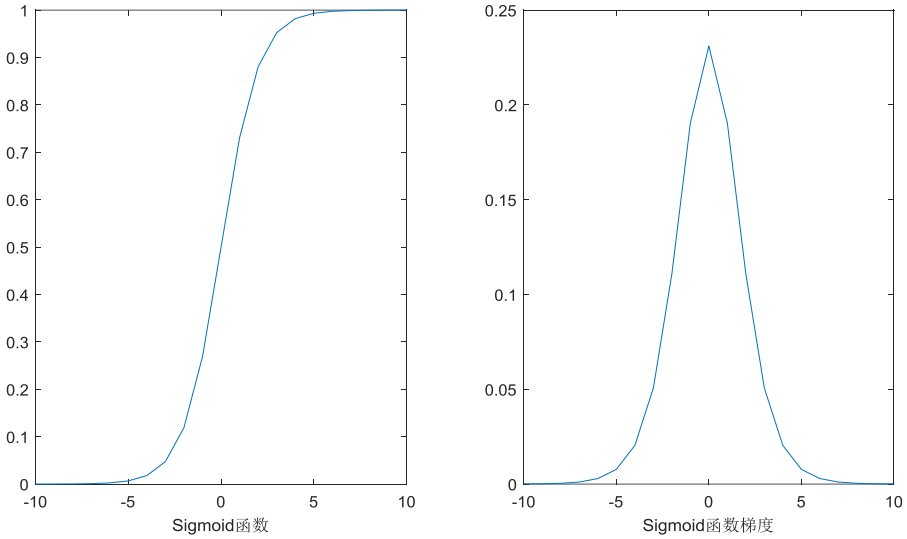


图 2-4 Sigmoid 函数及其梯度示意图

为了避免发生梯度饱和现象，修正线性单元（Rectified Linear Unit， ReLU）广泛地引入到神经网络中。ReLU 函数实际是一个分段函数，定义如下：

$$rectifier(x)=\max\{0,x\}=\begin{cases}x, x\geq 0\\ 0, x<0\end{cases} \tag{2.10}$$

从图 2-5 可以看出，在 $x\geq$时 ReLU 函数的梯度始终为 1，很好地解决了 Sigmoid 的梯度饱和问题，有助于模型训练的收敛。因此 ReLU 函数目前已经是卷积神经网络等深度学习算法模型激活函数的首选之一。然而，当 $x<0$ 时梯度直接被置为 0，那么在训练过程中参数将无法更新，也就是出现了“死亡”节点。这种情况下网络难以收敛到比较好的结果。

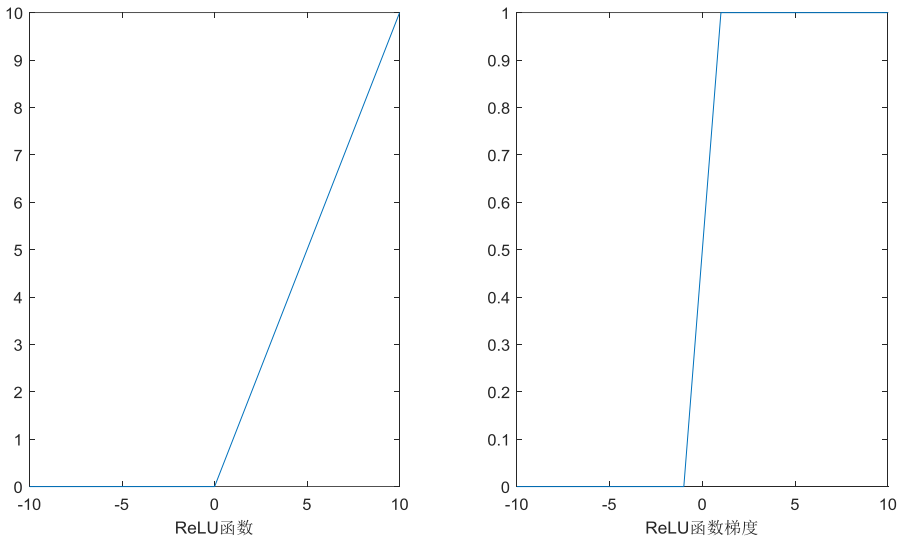


图 2-5 ReLU 函数及其梯度示意图

针对 ReLU 函数存在的不足，Leaky ReLU 和 PReLU 函数可以较好地应对。Leaky ReLU 函数定义如下：

$$F_{\text{LeakyReLU}} = \begin{cases} x, x \geq 0 \\ 0.1x, x < 0 \end{cases} \tag{2.11}$$

当 $x<0$ 时，Leaky ReLU 的梯度为 0.1，参数仍然会更新。与之相似的，PReLU 函数定义如下：

$$F_{P\text{Re}LU} = \begin{cases} x, x \geq 0 \\ px, x < 0 \end{cases} \tag{2.12}$$

其中 p 是一个可变参数，由训练得到。由上可知，Leaky ReLU 和 PReLU 能够较好解决节点“死亡”的问题。

（4）目标函数

深度学习中的目标函数是网络训练的关键，通过反向传播输入样本的预测结果与其真实标记产生的误差来指导网络参数学习与更新。预测任务通常分为分类和回归两大类，本文将介绍在人脸识别领域这两大类任务的几个常用目标函数。

在分类任务中，假设共有 N 个训练样本，x_i 表示网络分类层第 i 个样本的输入特征，与之相应的真实标记为 $y_i \in \{1,2,\ldots,C\}$， $h=\{h_1,h_2,\ldots,h_C\}^{\mathrm{T}}$ 是网络的最终输出，即样本 i 的预测结果，其中 C 为分类任务的类别数。

交叉熵损失函数，又称 softmax 函数，常用于分类任务。首先基于 softmax 将网络的输出归一化，使得各输出的和为 1，这样就转换成概率分布的形式，如下所示：

$$S_{y_i} = \frac{e^{h_{y_i}}}{\sum_{j=1}^{C} e^{h_j}} \tag{2.13}$$

因此交叉熵损失函数为:

$$L_{cross-entropy-loss}=-\frac{1}{N}\sum_{i=1}^{N}\log\left(\frac{e^{h_{y_i}}}{\sum_{j=1}^{C}e^{h_j}}\right) \tag{2.14}$$

大间隔交叉熵损失函数，在传统交叉熵损失函数的基础上做了改进。具体而言，网络的输出结果 h，实际上是全连接层参数 W 与该层特征向量 x_i 的内积，将 $h=W^Tx_i$ 代入式(2.14)中，传统交叉熵损失函数还可表示为

$$L_{soft\max-loss}=-\frac{1}{N}\sum_{i=1}^{N}\log\left(\frac{e^{W_{y_i}^{T}x_i}}{\sum_{j=1}^{C}e^{W_j^{T}x_i}}\right) \tag{2.15}$$

根据内积的定义，上式可以转换为

$$L_{soft\max-loss}=-\frac{1}{N}\sum_{i=1}^{N}\log\left(\frac{e^{\|W_{y_i}\|\|x_i\|\cos(\theta_{y_i})}}{\sum_{j=1}^{C}e^{\|W_j\|\|x_i\|\cos(\theta_j)}}\right) \tag{2.16}$$

其中，$\theta_j(0\le\theta_j\le\pi)$为向量 W^T 和 x_i 的夹角。

与传统交叉熵损失函数相比，大间隔交叉熵损失函数引入了正整数 m 来放大类间间隔，并且 $cos(\theta_{yi})$变为 $\phi(\theta_{yi})$，则有

$$\phi(\theta)=\begin{cases}\cos(m\theta),0\le\theta\le\dfrac{\pi}{m}\\ D(\theta),\dfrac{\pi}{m}<\theta\le\pi\end{cases} \tag{2.17}$$

式中 $D(\pi/m)=\cos((\pi/m))$。

大间隔交叉熵函数不仅要求分类正确而且扩大了类间的间隔，这可以防止模型过拟合，提高模型的泛化性能。

在回归任务中，假设网络的第 i 个输入特征 x_i 的真实标记为 $y^i=(y_1,y_2,\ldots,y_M)^{\mathrm{T}}$，$M$ 为标记向量的维度，则网络回归预测值与其真实标记的预测误差（也称残差）在 t 维度上可表示为：

$$l_t^i=y_t^i-\hat{y}_t^i \tag{2.18}$$

常用于回归任务的损失函数有 l_1 和 l_2 损失函数。对于 l_1 函数，其定义如下：

$$L_{l_1}=\frac{1}{N}\sum_{i=1}^{N}\sum_{t=1}^{M}\left|l_t^i\right| \tag{2.19}$$

其中，N 为样本数。

与之类似，l_2 损失函数定义如下：

$$L_{l_2} = \frac{1}{N}\sum_{i=1}^{N}\sum_{t=1}^{M}\left(l_t^i\right)^2 \tag{2.20}$$

一般情况下，l_1 损失函数与 l_2 损失函数训练效果相似，但在一些特定的情况下，l_2 损失函数在回归精度和收敛速度方面会略优于 l_1 损失函数。

（5）卷积神经网络的经典结构

将上一节讨论的卷积层、池化层、激活函数和目标函数这几个基本组件连接到一起，就能组成最常见的卷积神经网络了。事实上，第一个卷积神经网络 LeNet 就是这样产生的，它主要用于手写邮政编码数字的识别。真正让卷积神经网络在计算机视觉领域一鸣惊人的是 2012 年提出的 Alex-Net，在当年的 ImageNet 竞赛中，它以超越第二名 10.9 个百分点的成绩夺得了冠军。之后，各种各样的卷积神经网络如雨后春笋般不断涌现出来。总的来说，网络结构逐渐向“更深”和“更宽”的方向发展，最具代表性的是 VGGNet 和 GoogLeNet，通常深度的增加比宽度更能有效地提升网络的复杂性。然而，对着网络深度的增加，训练变得愈加困难。主要原因是在基于随机梯度下降的网络训练过程中，误差信号的多层反向传播非常容易引发梯度“消失”或者“爆炸”现象。虽然可以用批规范化（batch normalization）等策略帮助训练、加速深度网络的收敛，但与此同时训练误差并没有随着深度的增加而降低反而升高了，也就是所谓的“网络退化”现象。为了解决上述的这些问题，何恺明等人提出了残差网络（residual network, 简称 ResNet）。

残差网络通过残差模块的设计，在浅层网络和深层网络之间搭起了“高速公路”般的捷径，有利于梯度在不同的网络层之间传播，避免了梯度消失和爆炸这类问题的发生。假设 X 和 $H(X)$ 分别是残差模块的输入和输出，F 是该层所需要学习的映射函数，则残差模块的数学表示为：$H(X) = F(X) + X$。则 ResNet 所需要优化的部分就是残差项 $H(X) - X$，因此 $F(X)$ 被称为残差函数。如下图所示，残差模块有两个分支，其一是左侧的残差函数，其二是右侧的输入的恒等映射，这两部分通过对应元素的相加，在经过一个非线性激活函数 ReLU 后，组成了一个残差模块，残差网络由多个残差模块堆叠而成。

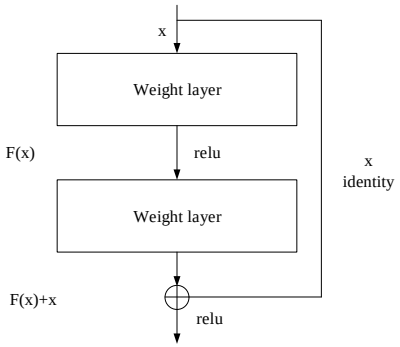


图 2-6 残差模块结构示意图

下图是两种不同形式的残差模块，左图是经典的残差模块，它由两个 3×3 卷积堆叠而成，适用于比较浅的网络，但是随着网络层数的加深，这种模块结构在工程实践中并不实用。因此，右图进行了改进，称为瓶颈残差模块，由 3 个 1×1，3×3 和 1×1 的卷积层构成。其中 1×1 卷积能够起到升维和降维的作用，使得 3×3 卷积能够从比较低的维度上进行计算。这样的设计结构，能够大大地减少参数数量，提高计算效率，有助于搭建很深的网络。

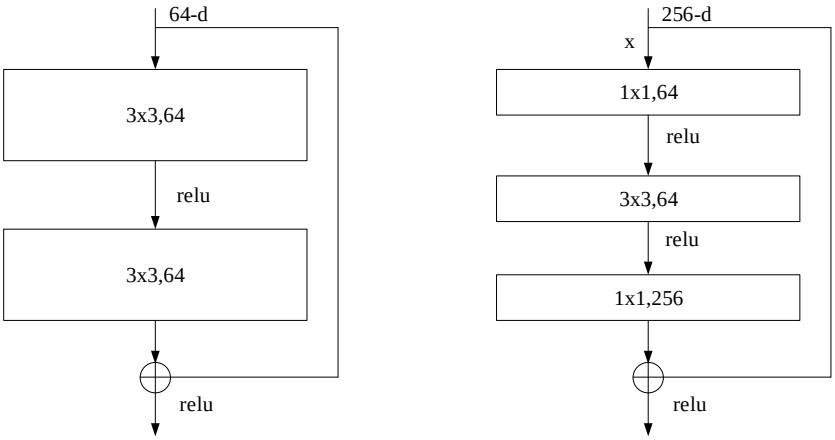


图 2-7 两种不同的残差模块结构

2.2.2 循环神经网络

循环神经网络（Recurrent neural networks，简称 RNN）[30]是深度学习中又一经典算法，它在处理时序数据时十分成功，常被用于机器翻译、视频理解等领域。本节一方面通过对 RNN 原理的解释，进而分析 RNN 时序有效性的本质；另一方面通过阐述 RNN 普遍存在的问题，进而介绍它的优化网络。

（1）循环神经网络结构

近些年，深度学习模型在处理有非常复杂内部结构的数据时十分有效。例如，卷积神经网络（convolution neural networks，简称 CNN）广泛应用于处理图像数据的像素之间的二维空间关系，在图像识别和分类等任务上表现出了卓越的性能。然而，时序数据（sequential data）以变长序列的形式作为网络的输入，序列与序列之间的时序关系非常重要，为了探索这种时序关系，循环神经网络（recurrent neural networks，简称 RNN）应运而生。

RNN 的关键点之一是它可以将前一时间点的信息用于解决当前时间点的任务。以视频为例理解，视频流中，前后一段时间内的连续帧通常具有很大的联系，比如连续的几帧能够完整地描述一个人拿起水杯喝水的整个动作过程，但如果只能单单观察其中的几帧可能仅仅获得人拿着水杯这个语义信息，可能人会把水杯里的水倒掉，又或者人拿水杯是为了去接水。也就是说，从连续帧中获取到

的动作信息帮助我们理解了视频中人拿水杯这个动作的目的。

通常情况下，RNN 是由一个个重复的单元前后或上下连接组成的一个向四周延展的网络。如果用下标 t 表示输入时序序列的不同时刻，用 h_t 表示在时刻 t 的系统隐层状态向量，用 x_t 表示时刻 t 的输入。t 时刻的隐层状态向量 h_t 依赖于当前输入 x_t 和前一时刻的隐层状态向量 h_{t-1}，数学表达如下：

$$h_t = f(x_t, h_{t-1}) \tag{2.21}$$

其中 f 是一个非线性映射函数。通常将 x_t 和 h_{t-1} 先进行线性变换后，再通过一个非线性激活函数 tanh，数学表达如下：

$$h_t = \tanh(W_{xh}x_t + W_{hh}h_{t-1}) \tag{2.22}$$

其中 W_{xh} 和 W_{hh} 是可学习的参数矩阵，激活函数 tanh 作用于输入的每个元素上。单层的 RNN 结构示意图如下所示。

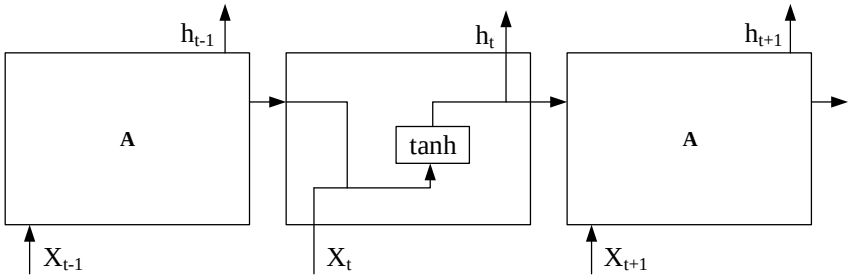


图 2-8 标准 RNN 的单层结构

因此，对于当前时间点 t 而言，隐层状态向量 h_{t-1} 可以认为存储了网络的“记忆”，RNN 学习的目标就是让 h_{t-1} 记录当前时刻 t 与之前若干时刻的输入信息之间的联系，在当前时刻 t 的数据 x_t 输入时，通过 h_{t-1} 和 x_t 得到当前时刻的隐层状态向量 h_t。

（2）梯度爆炸和梯度消失

虽然理论上 RNN 可以捕获时间序列之间的关系，但是在实际应用中存在两个比较大的问题：梯度爆炸（gradient explosion）[31]和梯度消失（vanishing gradient）[32]。

考虑一种最简单的状况，当激活函数为恒等变换时，则

$$h_t = W_{xh}x_t + W_{hh}h_{t-1} \tag{2.23}$$

于是，在进行误差反向传播（back propagation）时，当目标函数 L 对于 t 时刻隐层状态变量 h_t 的偏导数已知时，利用链式求导法则，可以计算得到目标函数 L 对 t 时刻隐层状态变量 h_0 的偏导数为

$$\frac{\partial l}{\partial h_0} = \left(\frac{\partial h_t}{\partial h_0}\right)^T \frac{\partial l}{\partial h_t} \tag{2.24}$$

由 RNN 对于时间序列的特性，沿时间维度展开计算，即

$$\frac{\partial h_t}{\partial h_0}=\prod_{i=1}^{t}\frac{\partial h_i}{\partial h_{i-1}}=\prod_{i=1}^{t}W_{hh}=W_{hh}^{t} \tag{2.25}$$

也就是说，在误差方向传播的时候，需要反复地乘以参数矩阵 W_{hh}。对 W_{hh} 进行奇异值分解（Singular Value Decomposition，简称 SVD），可得

$$W_{hh}=U\sum V^{T}=\sum_{i=1}^{r}\sigma_i u_i v_i^{T} \tag{2.26}$$

其中 r 是矩阵 W_{hh} 的秩。代入(2.25)可得

$$\frac{\partial h_t}{\partial h_0}=W_{hh}^{t}=\sum_{i=1}^{r}\sigma_i^{t} u_i v_i^{T} \tag{2.27}$$

再代入(2.24)，则我们需要计算的目标

$$\frac{\partial l}{\partial h_0}=\sum_{i=1}^{r}\sigma_i^{t} v_i u_i^{T}\frac{\partial l}{\partial h_t} \tag{2.28}$$

当时间序列很长，即 t 很大时，上式的结果取决于参数矩阵 W_{hh} 的最大奇异值 σ_1。若 σ_1 大于 1，则结果会非常大，产生梯度爆炸现象；若 σ_1 小于 1，则结果会很小，产生梯度消失现象。具体如下：

1）梯度爆炸

当 σ_1>1 时：

$$\frac{\partial l}{\partial h_0}=\sum_{i=1}^{r}\sigma_i^{t} v_i u_i^{T}\frac{\partial l}{\partial h_t}\approx\infty\cdot v_1 u_1^{T}\frac{\partial l}{\partial h_t}=\infty \tag{2.29}$$

此时偏导数会变得很大，在误差反向传播的过程中，会出现值溢出的错误，导致网络难以训练，甚至不收敛，这种现象叫做梯度爆炸。

2）梯度消失

当 σ_1<1 时：

$$\frac{\partial l}{\partial h_0}=\sum_{i=1}^{r}\sigma_i^{t} v_i u_i^{T}\frac{\partial l}{\partial h_t}\approx 0\cdot v_1 u_1^{T}\frac{\partial l}{\partial h_t}=0 \tag{2.30}$$

此时偏导数非常接近于 0，在误差反向传播的过程中，梯度更新前后没有什么变化，使得网络的学习能力退化，这种现象称为梯度消失。

梯度爆炸问题通常可以采用梯度裁剪（gradient clipping）来解决，简单地说，就是当梯度超过某一阈值时或被强制赋值为这个值，也就是梯度有一个最大值，不会出现无限趋近于无穷大的问题。

梯度消失问题解决起来比较困难，它是 RNN 甚至是整个深度学习领域普遍存在的问题。因此，如何缓解梯度消失是 RNN 和其他深度学习网络研究的关键所在。近几年提出的 ResNet 等网络架构，

很好地解决了卷积神经网络（CNN）的梯度消失问题；而对于 RNN，采用诸如 LSTM 等的门（gate）来控制内部信息流动，是一种比较有效的解决方式。

（3）长短期记忆网络

由上一小节可知，当时间序列很长时，RNN 很容易出现梯度爆炸和梯度消失，这个现象被称为长期依赖（Long Dependency）问题[33]。长短期记忆（Long Short-Term Memory）网络能够学习长期依赖信息，从而解决这个问题。

LSTM 对标准的 RNN 单元内部结构做了改进，如下图所示。LSTM 网络同样由重复的单元组成，但是单元的内部结构发生了变化。不同于 RNN 单元的单一神经网络层，LSTM 单元内部有四个不同的层，信息在四个层中交互传递。

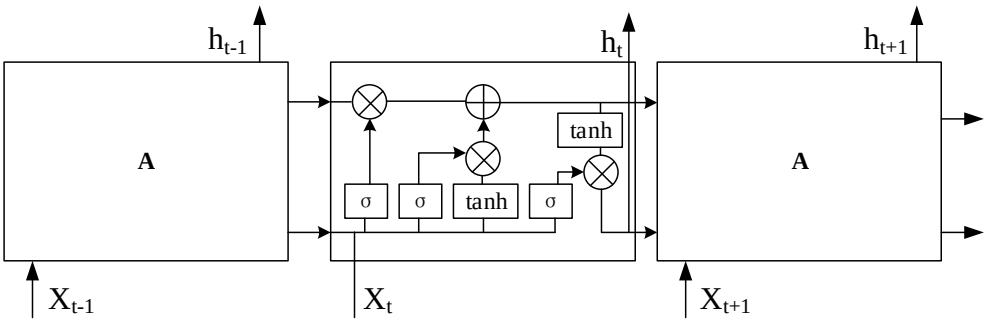


图 2-9 LSTM 的单层结构

LSTM 单元有三个“门”用来控制本单元的状态，分别是遗忘门、输入门和输出门，各自的组成和计算细节如下：

1）遗忘门

遗忘门（Forget Gate），顾名思义，用来控制信息是否应该丢弃，如下图所示。遗忘门有两个输入，分别是 t 时刻的输入信息 x_t 和上一时刻 t-1 的隐状态 h_{t-1}，遗忘门的输出是一个介于 0 和 1 之间的数值，1 表示“完全保留”，0 表示“完全丢弃”。

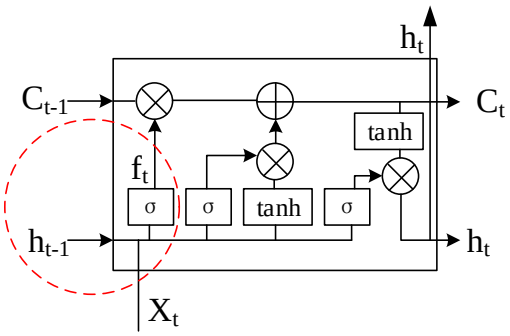


图 2-10 遗忘门

图中 f_t 表示遗忘门的输出，数学表示如下：

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{2.31}$$

其中 σ 表示 Sigmoid 激活函数。

2）输入门

输入门（Input Gate）有两步操作：一是，通过 Sigmoid 输出一个介于 0 和 1 之间的值来决定是否更新单元的状态；二是，tanh 函数生成一个候选状态向量$\tilde{C}$，如下图所示。

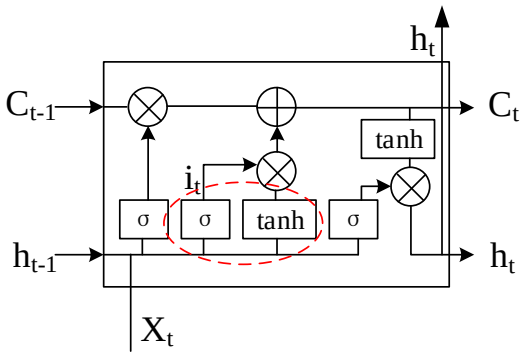


图 2-11 输入门

图中的 i_t 和 C_t 计算公式如下：

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{2.32}$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{2.33}$$

单元状态的更新情况可由以下式子来表达：

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \tag{2.34}$$

其中*表示像素相乘，示意图如下：

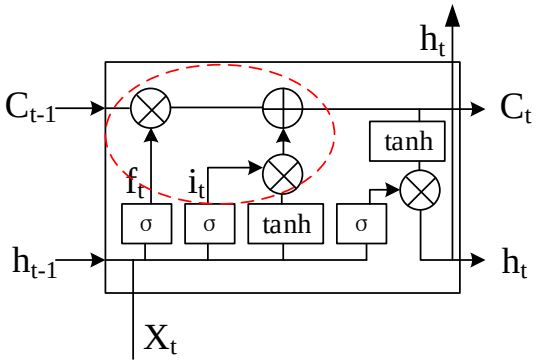


图 2-12 单元状态的更新

3）输出门

输出门（Output Gate）决定单元的输出信息。类似地，输出门首先将 h_{t-1} 和 x_t 通过 Sigmoid 函数得到一个权值，然后将 C_t 用 tanh 函数处理，最后将两者相乘作为当前时刻的隐状态，如下图所示。

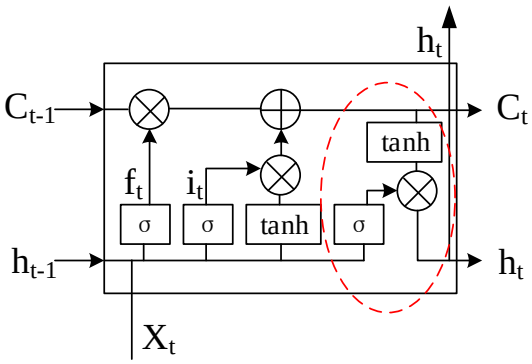


图 2-13 输出门

将上图的计算过程用公式表达如下：

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \tag{2.35}$$

$$h_t = o_t * \tanh(C_t) \tag{2.36}$$

综上所述，LSTM 单元通过遗忘门、输入门和输出门对当前的输入信息 x_t、上一时刻的隐状态 h_{t-1} 和上一时刻的单元状态 c_{t-1} 进行不同程度的丢弃和保留，得到当前时刻单元状态 C_t 和隐状态 h_t，并作为下一时刻单元状态和隐状态的影响信息。由此可以看出，LSTM 能够处理输入序列在时间上的联系，并且通过精妙的“门”的设计，使得网络具备了“记忆”功能，能够根据以往时间点的信息推测当前时刻的信息，通过“记忆”的累积，从而解决传统 RNN 的长期依赖问题。

2.3 本章小结

本章首先介绍了常见的人脸欺骗攻击方式及其特点，同时对静态特征和动态特征这两种人脸防伪方案的原理和缺点进行了总结。随后本章对卷积神经网络和循环神经网络这两种经典深度学习算法进行了介绍。在卷积神经网络中，除了对其原理、流程进行了详细说明，还列举了卷积神经网络的经典结构。在循环神经网络中，除了对其结构进行介绍，还对其在实际应用中存在的梯度爆炸和梯度消失现象机理进行了分析，并对可以解决这些问题的长短期记忆网络 LSTM 进行了介绍。本章的原理部分为后续的人脸真伪检测框架设计提供了理论基础。

第3章 基于融合特征的人脸真伪检测

针对上文中提到的人脸静态特征和动态特征在人脸防伪中表现出的优势，本章将设计一种静态特征和动态特征相融合的检测方案。首先将介绍方案的整体架构，然后将阐述融合特征的设计细节，最后说明分类模块的设计思路。

3.1 检测框架的整体架构

上文所述的人脸静态特征有效地利用了打印照片、重放视频中真脸深度与假脸深度的明显差异，可以有效地对其进行防范。人脸动态特征从连续帧的动作信息的角度切入，通过探索连续帧中人脸的动作模式，提取视频帧序列中包含人脸时间域和空间域信息的动态特征，作为检测人脸真伪的依据。综合静态特征和动态特征的优势，本文将两种特征进行融合，将人脸深度信息和脸部动作信息结合起来，可以设计出一种表达能力更强、鲁棒性更高的融合特征。基于融合特征的检测方案整体架构设计如下图所示：

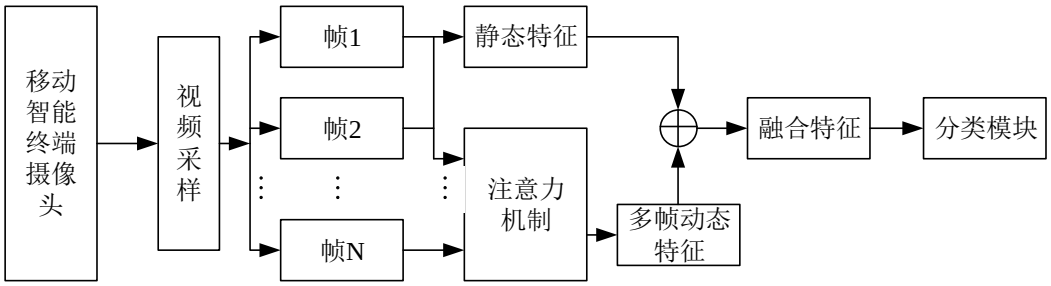


图 3-1 基于融合特征的检测方案的整体架构

基于融合特征的人脸真伪检测方案首先通过移动智能终端的摄像头采集人脸视频，然后对视频进行采样。随后对采样后视频的第一帧进行处理，提取其静态特征；接着分析采样后全部的 N 帧中人脸的动态变化，利用注意力机制提取出这 N 帧的动态特征。最后将第一帧的静态特征和 N 帧的动态特征相融合，用分类模块基于融合特征进行人脸真伪判别。

3.2 融合特征设计

假设单帧图像的静态特征为 f_s，相邻两帧图像经过 OFFR 模块处理后的短期动态特征为 f_d。那么注意力生成网络的输出其实是一段时间内的多个 f_d，在注意力机制的作用下加权组合而成。令 f_d 为某个时间段内短期动态特征的序列，则有

$$F_d = (f_{d_1}, f_{d_2}, ..., f_{d_n}) \tag{3.1}$$

其中 n 为序列长度。设 f_a 为 f_d 经注意力生成网络处理的加权组合特征，则有

$$f_a = a_1 \cdot f_{d_1} + a_2 \cdot f_{d_2} + ... + a_n \cdot f_{d_n} \tag{3.2}$$

f_a 即是注意力网络的输出特征，也是一个时间段内的连续帧按照重要程度加权组合而成的动态特征，也就是长期动态特征。最后，将长期动态特征 f_a 和静态特征 f_s 进行融合，融合特征 f_{fusion} 定义如下：

$$f_{fusion} = \lambda \cdot f_a + (1-\lambda) \cdot f_s \tag{3.3}$$

其中 $\lambda \in [0,1]$ 称为融合系数，是网络训练的一个参数，反映的是长期动态特征和静态特征的相对重要程度。

下图为融合特征产生过程的示意图。D 表示单帧图像生成的人脸深度图，T 表示从连续帧中提取的人脸的长期动态特征，两者进行融合，得到融合特征 F。

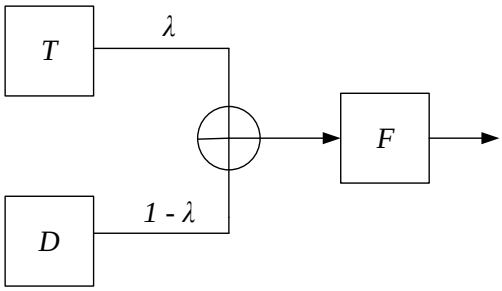


图 3-2 融合特征

3.3 分类模块设计

在深度学习中，全连接层（Fully Connected Layers，简称 FC）常常作为“分类器”与 softmax 函数一起连接在整个神经网络的尾部。由前文可知，神经网络中的卷积层、池化层和激活函数等操作将原始的输入数据映射到隐层的特征空间，全连接层则是将网络学习到的原始数据分布式特征表示映射到样本的标记空间。

在实际的使用中，全连接层可以由卷积操作实现，具体分为两种情况：

（1）如果前一层是卷积层的话，全连接层可以转化为全局卷积，即卷积核尺寸为 $h \times w$，h 和 w 分别为前一层卷积结果的高和宽；

（2）如果前一层仍是全连接层，全连接层可以转化为卷积核为 1×1 的卷积。

虽然全连接层的参数比较冗余，但是它在模型迁移学习的过程中能够起到“防火墙”的作用[34]，

特别是当源域和目标域的数据相差比较大的时候，通常有不俗的性能表现。因此，本节采用全连接层依据得到的融合特征进行分类，从而保证模型对于不同人脸防伪数据库的鲁棒性。

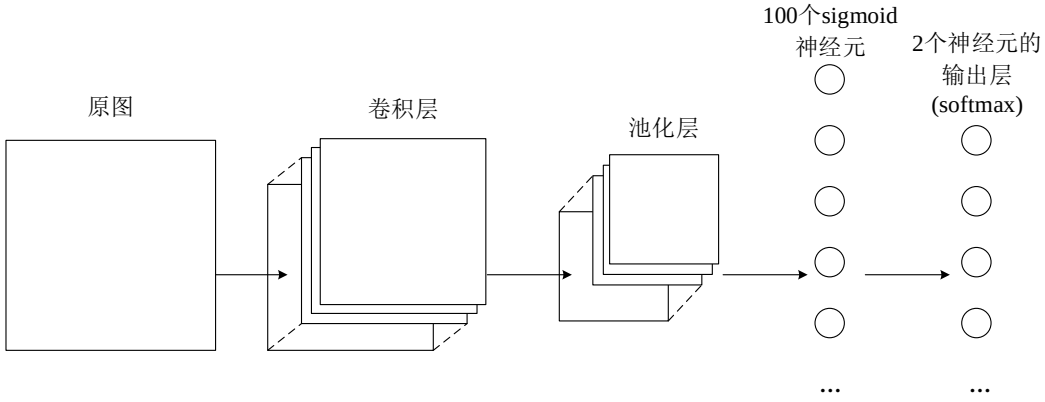


图 3-3　全连接层

如上图所示，本节设计了两个全连接层将融合特征转换为一个一维向量用于最后的人脸真伪分类。两个全连接层对特征尺寸和维度的变换，如下表所示。

表 3.1 全连接层特征的变化

层		FC1	FC2
输入	尺寸	32 × 32	100
	维度	3	1
输出	尺寸	100	2
	维度	1	1

由上表可知，两个全连接层将网络学习到的融合特征先转化为 100 × 1 的列向量，再转换为 2 ×1 的列向量，对应人脸检测的真或假两种类别。众所周知，通常预测类别的最后输出结果为概率分布，按照每个类别对应的概率大小决定预测为哪一类。因此，最后用 softmax 函数对全连接层的输出归一化，得到两种类别的概率分布。softmax 函数表示如下：

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}}, j = 1,...,K \tag{3.4}$$

其中，z_j 为全连接层的输出，在人脸真伪分类的场景下式中 K=2。分类的示意图如下所示：

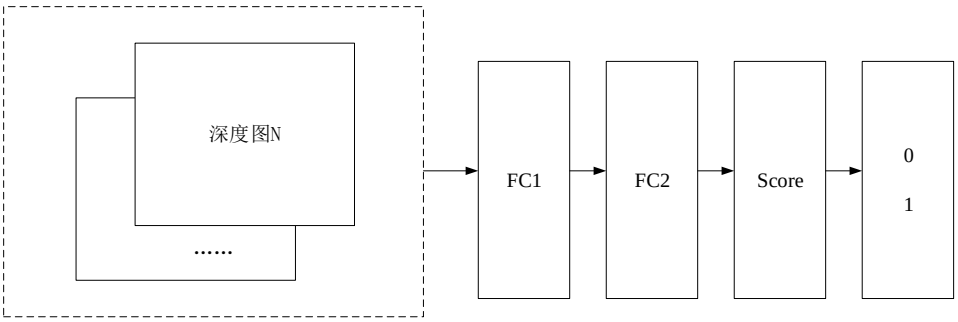


图 3-4 分类示意图

其中得分 Score 即为 softmax 函数输出的概率分布，0 表示假脸，1 表示真脸。

3.4 本章小结

本章针对上文中提到的人脸静态特征和动态特征在人脸防伪中表现出的优势，设计了一种静态特征和动态特征相融合的检测方案。该方案通过设置融合系数 λ，控制静态特征和动态特征的相对重要程度，综合了静态特征和动态特征各自的优势，使得融合特征同时具有较强的通用性、适应性和泛化性。

第4章 人脸静态特征的提取方法

针对前文中提到现有的基于深度学习的人脸特征提取方法存在的问题，本章将提出一种人脸静态特征的提取方法。该方法针对现有算法对于人脸姿态、表情和光照条件变化的适应性差的问题，采用 PRNet 重建人脸 3D 点云图；针对现有算法缺乏明确的监督信息、模型学习到的并不是区分人脸真伪的关键特征、导致检测的准确度不高的问题，该方法利用人脸深度图引入了明确的监督信息，指导深度网络进行训练，进而提取人脸的静态特征。本章将首先介绍方法的整体架构，然后将对深度图真实标记生成模块进行介绍，最后将对基于深度图的特征提取网络模块的设计细节进行阐述。

4.1 方法的整体架构

下图描述了人脸静态特征提取方法的整体架构，由深度图真实标记的生成模块和基于深度图的特征提取网络模块两大部分组成。首先，深度图的真实标记分为两类，一类是真脸的标记，另一类是假脸的标记，在深度图真实标记生成模块中分别用不同的方法生成；其次，在特征提取网络模块中用深度图的真实标记来指导卷积神经网络训练参数，得到能够提取泛化性能优秀的深度图网络模型；最后，将包含人脸全局信息的深度图作为人脸的静态特征，用于辨别人脸的真伪。

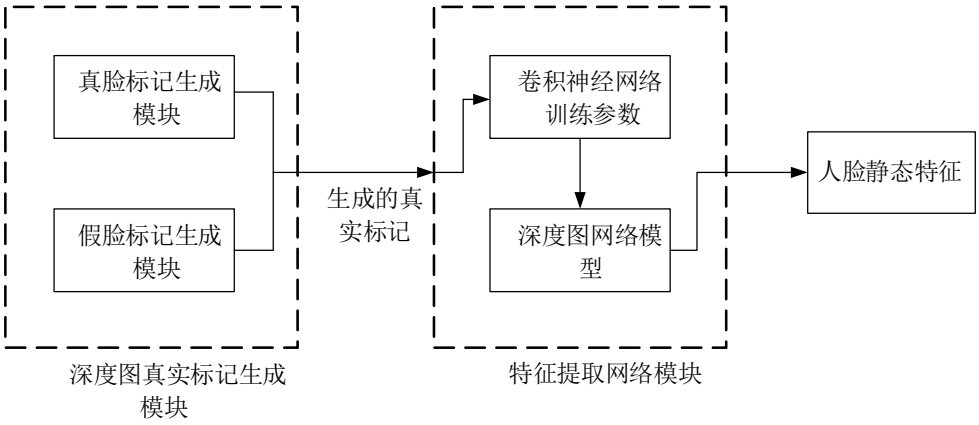


图 4-1 静态特征提取方法的整体架构图

4.2 深度图真实标记的生成模块

前文已经对卷积神经网络做了简要的介绍，真实标记（ground turth）在网络训练过程中扮演着不可或缺的角色。当网络的输入是人脸图像时，通过卷积层、池化层和激活函数的层层抽象，到达网络输出层时，目标函数计算输出结果与真实标记的误差，并且把这个误差回传，通过逐层反馈更

新每一层的参数，从而达到网络训练的目的。因此，真实标记可以理解为“答案”或者是“参考标准”。为了得到能够很好地表征人脸信息进而区分人脸真伪的深度图，首先需要人脸深度图的真实标记作为监督信息训练神经网络。

事实上，从移动智能终端观测的角度，真伪人脸的深度信息存在着很大的不同[35]，如下图所示。其中左图表示的是真脸的深度信息，右图表示的是电子设备上播放人脸视频时的深度信息。可以看出，当从智能设备摄像头进行观测时，在真脸情况下，鼻子和脸颊的深度是不同的；在重放攻击场景下，由于电子设备屏幕的反射，人脸的各部分深度都是一样的。因此，根据真假人脸深度信息的不同，本节分别采取两种方式生成深度图的真实标记。

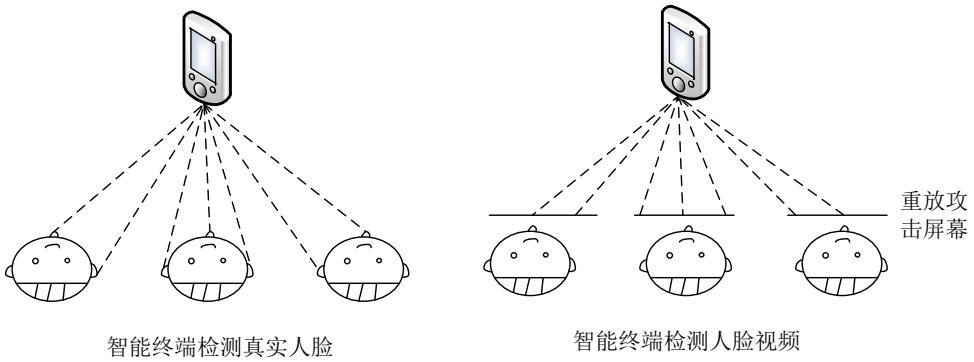


图 4-2 真假人脸深度信息的对比

对于真脸深度图真实标记，根据从鼻子到脸颊面部各位置深度的不同，先用神经网络预测输入 2D 图像的 3D 人脸点云图，再依据 3D 人脸点云图做深度的估计和归一化。而对于假脸，因为脸部各位置深度一致，只需要简单地归一化处理。真脸深度图的生成分为两步，首先是生成 3D 人脸点云图，然后是人脸深度的估计和归一化。4.2.1 节将详细阐述真脸的 3D 点云图的生成方案，4.2.2 节将分别对真假脸的深度进行归一化处理。

4.2.1 真脸的 3D 点云图生成方案

本节采用开源的位置图回归网络（Position Map Regression Network, 简称 PRNet）[36]来生成真脸的 3D 点云图。与其他生成 3D 人脸的方法相比，PRNet 有以下三大优势：

一是，PRNet 将 3D 人脸重建和密集人脸关键点对齐联合起来，摆脱了传统的 3D 形变模型（3D Morphable Model, 简称 3DMM）对人脸形状的束缚，能够得到更精细、更真切的人脸 3D 模型；

二是，通过专门的 UV 映射空间的设置，将完整的 3D 人脸的点云图映射到 2D 空间同时保持每个位置的语义，能够在人脸遮挡、姿态变化和光照改变等情况下表现出很好的鲁棒性；

三是，通过精细的网络结构和相应的损失函数的设计，训练得到的网络模型参数量较少、计算时长短，能够在 9.8ms 内完成一帧的处理。

以下将从 3D 人脸表示方式以及网络和损失函数设计两个方面，阐述 PRNet 的设计细节。

（1）3D 人脸表示方式

要生成人脸深度图，首先要将 3D 人脸模型和密集的面部关键点信息从 2D 图像中还原（或重建）出来。所以，需要设计一种合适的 3D 人脸表示方式，这种表示方式实际上是原来的 2D 图像到 3D 人脸模型的一个映射，可以用神经网络来实现。

比较常用的方法是将 3D 人脸中所有的点坐标用一个向量表示[37][38][39][40][41]，输入神经网络进行预测。但是这种三维到一维的直接转换，一方面需要用参数量更多全连接层来预测输出，增加了训练的难度；另一方面从三维压缩到一维，忽略了点与点之间的位置关系，丢失了原来图像中相邻点之间的邻接信息。

本节采用 UV 映射作为 3D 人脸的表示方式[42][43][44][45]，具体过程如下图所示。首先通过 UV 纹理映射得到输入图像的 2D 纹理图像，称为 UV 纹理图；然后用 *x,y,z* 坐标取代纹理图中的 *r,g,b* 值，得到 UV 位置图。UV 位置图是一种二维图像，记录了 3D 人脸模型中各个点的三维位置信息。在得到包含人脸 3D 空间和语义信息的 UV 位置图之后，再由 UV 位置图还原出 3D 人脸点云，即 3D 人脸模型。在左手直角坐标系中，坐标系的原点在输入图像的左上角，当 3D 人脸投影到 *x-y* 平面时，正好与输入图像的人脸区域对应。

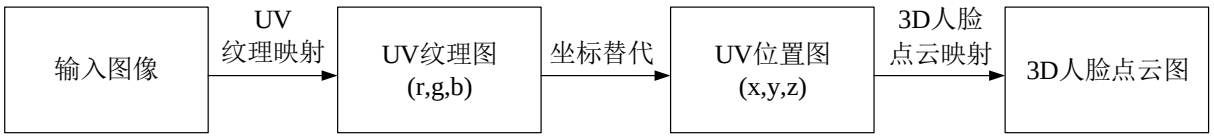


图 4-3 3D 人脸点云图生成过程

上述的整个过程可以用 CNN 实现，首先在大规模的 2D 图像和 3D 人脸点云配对的数据集（300W-LP）上进行训练，然后对参数调优得到泛化性能强的网络模型。接下来对 PRNet 具体的网络结构进行介绍。

（2）网络架构和目标函数设计

网络设计的目标是把输入的 RGB 图像转换成 UV 位置图，使用“编码—译码”结构[46]，来学习映射函数。因此，网络可分为编码部分和译码部分。

编码部分，第一层是卷积层，后面连接着 10 个残差块，可以将 256×256×3 的输入图像压缩为 8×8×512 的特征图，完成输入图像的“编码”任务。译码部分，包含了 17 个反卷积层，将编码部分的特征图作为输入，产生预测结果为 256×256×3 的 UV 位置图。所有的卷积核与反卷积核的尺寸都设置为 4×4，采用 ReLU 函数激活。最后，再根据原 2D 图像人脸的姿态，将 UV 位置图还原成 3D 点云图。

为了对网络参数进行优化，设计了一个目标函数来测量网络的输出结果与 3D 人脸点云图的真实标记（来自数据集 300W-LP）之间的误差。假设网络输出的 UV 位置图为 $P(x,y)$，其中(x,y)代表每个像素点的坐标；对于 3D 人脸图的真实标记，用$\tilde{P}(x,y)$表示；$W(x,y)$表示人脸不同位置（眼睛、鼻子、嘴巴等）关键点像素的权重。那么，网络的目标函数可以表示为：

$$Loss = \sum \left\| P(x,y) - \tilde{P}(x,y) \right\| \cdot W(x,y) \tag{4.1}$$

4.2.2 真假脸的深度归一化处理

真脸的 3D 点云图中各位置的点 z 坐标的值不尽相同，因此直观上可以用 z 值来估计真脸的深度。为了简化运算，对深度进行归一化，压缩到[0,1]范围内。归一化的数学表达如下：

$$d_i = \frac{z_i - z_{\min}}{z_{\max} - z_{\min}}, i = 1, 2, ..., N \tag{4.2}$$

其中 N 表示点云图中点的总数，z_i 表示真脸 3D 点云图 i 点的 z 坐标，z_{min} 表示所有点中 z 坐标的最小值，Z_{max} 表示所有点中 z 坐标的最大值，d_i 表示归一化后该点的人脸深度值。也就是说越接近脸颊的点深度值越高，越接近 1；反之，越接近 0。

而对于假脸来说，面部各点的深度不发生变化，因此把假脸各点的深度值设置为 0，以此作为假脸深度图的真实标记。

综上所述，对于真脸和假脸本文分别采用两种方法生成对应的深度图真实标记。对于真脸有两个步骤，首先通过 PRNet 得到输入 2D 图像中人脸的 3D 点云图，然后对 3D 点云图进行深度归一化，得到真脸深度图的真实标记。对于假脸比较简单，直接把输入 2D 图像各像素点的深度置为 0，得到假脸深度图的真实标记。

4.3 基于深度图的特征提取网络模块

除了 3D 掩膜攻击，其他已知的人脸欺骗攻击，如打印攻击和重放攻击，与真人人脸相比，深度都有着明显的不同。因此，基于深度图的人脸真伪检测方案对于防范人脸欺骗攻击十分有效。在上一节中，先通过 PRNet 生成了真脸的 3D 点云图，再用深度归一化分别得到了真脸和假脸的真实标记。本节以真假脸的真实标记为深度监督信息，训练 CNN 对输入的 2D 图像估计其图像中人脸的深度图，并以此作为人脸的静态特征。

4.3.1 模块架构

特征提取网络模块架构如下图所示，整个全卷积网络由 3 个子模块组成：层级特征提取模块、特征重组模块和深度图估计模块。

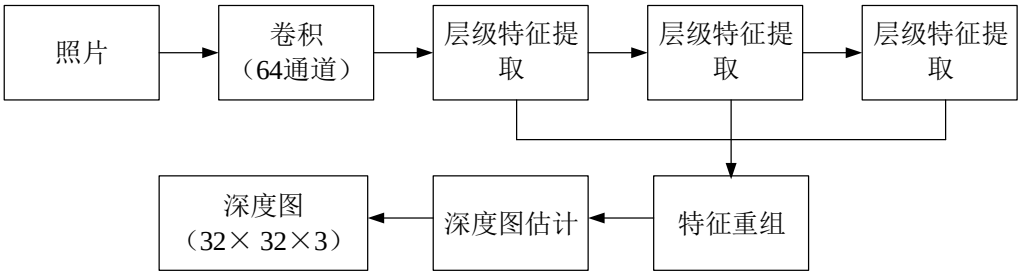


图 4-4 特征提取网络模块

（1）层级特征提取模块

层级特征提取模块（Hierarchical Feature Extraction Block，简称 HFEB），由 3 个 HFEB 子模块组成，每个模块都有 3 个卷积层，1 个池化层，1 个 ReLU 层和 1 个批归一化层，如下图所示。输入的原始图像依次经过 3 个 HFEB 子模块的处理，输出不同尺寸大小的响应图，代表着由浅层到高层的不同层次的图像语义信息。

图 4-5 HFEB 模块的结构

（2）特征重组模块

特征重组模块（Feature Recombination Block，简称 FRB），将 3 个 HFEB 子模块输出的的层级特征作为输入，分别将响应图的尺寸重新调整为预定义的 32×32，然后将其组合起来，如下图所示。这个模块借鉴了 ResNet 的原理：因为浅层网络富含空间信息，所提取的特征大多数和物体边缘、线条等低层信息相关；而深层网络含有更多的高层语义信息，而在空间信息上比较匮乏，所以不同层级的特征组合能够更好地表示图像的特征信息。

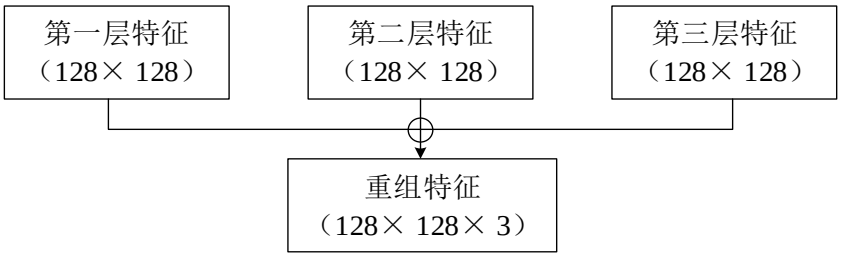


图 4-6 FRB 模块的结构

（3）深度图估计模块

深度图估计模块，将组合特征作为输入，经过 3 层卷积，得到尺寸为 32×32 的深度图。再将网络预测（或估计）的深度图与该深度图的真实标记（ground truth）比较，计算损失，作为误差通过网络反向传播，循环往复训练网络。

综上所述，整个特征提取网络的结构细节和各层输出的尺寸，如表 4.1 所示。

表 4.1 网络的结构细节和各层输出的尺寸

层	核/步长	输出尺寸 （高×宽×通道数）
Conv-1	3 × 3/1	256 × 256 × 64
Conv-2	3 × 3/1	256 × 256 × 128
Conv-3	3 × 3/1	256 × 256 × 196
Conv-4	3 × 3/1	256 × 256 × 128
MaxPooling-1	2 × 2/2	128 × 128 × 128
Conv-5	3 × 3/1	128 × 128 × 128
Conv-6	3 × 3/1	128 × 128 × 196
Conv-7	3 × 3/1	128 × 128 × 128
MaxPooling-2	2 × 2/2	64 × 64 × 128
Conv-8	3 × 3/1	64 × 64 × 128
Conv-9	3 × 3/1	64 × 64 × 196
Conv-10	3 × 3/1	64 × 64 × 128
MaxPooling-3	2 × 2/2	32 × 32 × 128
Conv-11	3 × 3/1	32 × 32 × 128
Conv-12	3 × 3/1	32 × 32 × 64
Conv-13	3 × 3/1	32 × 32 × 1

4.3.2 目标函数设计

对于输入图像 I（256×256×3），若其深度图的真实标记为 D，则对于任意一张输入图像 I_i，有

$$\theta_D' = \arg_{\theta_D} \min \sum_{i=1}^{N_D} \left\| CNN_D\left(I_i;\theta_D\right) - D_i \right\|_1^2 \tag{4.3}$$

其中 θ_D 是深度图估计网络（Depth Map Estimate Network，DMENet）的参数，N_d 是训练图像的数目，D_i 表示图像 i 的深度图真实标记， 表示 DMENet 新一轮的参数。

如上式所示，图像的深度作为监督信息训练 DMENet，目标是使得 DMENet 对输入图像深度图的估计值与真实标记的误差尽可能的小。DMENet 通过每一轮训练得到新一轮的网络参数，如此循环往复，当误差足够小并趋于稳定时停止训练。此时，训练好的 DMENet 可以在误差允许范围内，准确地估计输入图像的深度图，并且以此作为图像的深度信息特征，用于下一步真伪人脸的分类。

4.3.3 激活函数选择

神经网络中的激活函数又称为非线性映射层，用来增加网络的非线性因素。因为单纯的线性模型的堆叠相当于还是一个线性映射，无法组合成复杂的数学函数，不足以表达现实场景中的输入数据和输出结果间的对应关系。因此，激活函数的引入可以强化网络的表达能力。正如前文所述，神经网络中常用的激活函数有 Sigmoid（S 型）函数、tanh（双曲正切）函数和 ReLU（修正线性单元）函数等。

Sigmoid 函数将网络层的输出压缩为[0,1]之间的一个实数，可以看成是所属类别的概率，常用于二分类问题，在类别相差比较复杂或者相差不大时，效果比较好。然而，Sigmoid 函数存在以下几个问题：一是，它不是以 0 为中心的，若某个神经元的输入一直为正或者为负，则其权重的导数也一直为正或负，这样在误差反向传播时，权重就会一直正或负方向更新，形成“捆绑效应”，导致网络收敛缓慢；二是，当输入信号绝对值较大时，Sigmoid 函数的梯度几乎为零，在反向传播的过程中，对不同的输入不会有任何变化，很容易发生梯度消失现象，导致较深的网络难以训练；三是，函数本身是幂函数，计算不够简便，运算时间长。

$$Sigmoid(x)=\frac{1}{1+e^{-x}} \tag{4.4}$$

tanh 函数将输出映射到[-1,1]之间，可以看成是 Sigmoid 函数的一种变换，同样适合于分类问题，随着网络层对特征的循环计算，它会放大类别之间特征的差距，所以在特征相差比较明显时效果比较好。tanh 函数解决了 Sigmoid 不是以 0 为中心的问题，但是仍然存在梯度消失的问题，另外同样也是幂函数，计算复杂。

$$\tanh(x)=\frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}=2*Sigmoid(2x)-1 \tag{4.5}$$

ReLU 函数取的是 0 与输入 x 的最大值，当输入大于 0 时，其梯度恒定为 1，很好地解决了梯度消失问题。对于原始数据来说，其特征具有稀疏性，用大多数元素为 0 的稀疏矩阵表示特征，往往具有不错的效果。实际上神经网络的反复计算，就是一个不断尝试如何用一个稀疏矩阵来表达数据特征的过程。因为 ReLU 函数设计简单、具有稀疏性又能防止梯度消失，所以其训练效果较好，收敛速度也要快于 Sigmoid 和 tanh 函数。

$$\mathrm{Re}\,LU(x)=\max(0,x) \tag{4.6}$$

综上所述，本文基于训练效果、收敛速度和计算量方面的考虑，采用 ReLU 函数作为神经网络的激活函数。

4.3.4 超参数设置

神经网络网络中有很多超参数，用于控制网络的训练过程，它们的选择和设置对网络的性能有重要的影响，以下对深度图估计网络的各个超参数进行详细说明。

（1）参数初始化方法

搭建好神经网络模型后，在训练开始时需要先对网络参数进行初始化，为了达到较好的训练效果，一般要使得网络中每层的输入输出数据满足同样的分布。常见的初始化方法有随机初始化、Xavier 初始化、MSRA 初始化、预训练初始化等，各种初始化方法根据任务不同、网络结构不同自行选择。本章对于深度图估计网络（DMENet）采用随机初始化，将参数初始化为服从均值为 0、标准差为 0.02 的正态分布的较小随机数。

（2）训练轮数

训练轮数（epoch）一般根据模型在验证集上的表现来确定，模型收敛就可以停止迭代。当模型不收敛时，训练轮数是最大的迭代轮数。在实际操作中，如果连续几轮模型的损失（loss）都没有降低，就认为模型已经收敛，可以停止训练了。本章将 DMENet 的 epoch 设置为 30。

（3）学习率

在误差反向传播更新网络参数时，有一个系数控制模型学习的进度，称作学习率。学习率太大，可能会导致网络不收敛；学习率太小，则会导致训练速度慢。通常训练时的初始学习率设置较大，随着训练轮数的增加，减小学习率优化模型。本章将 DMENet 网络的初始学习率设置为 0.003，10 轮之后减小为 0.0003，20 轮后再降为 0.00003。

（4）批尺寸

批尺寸（batch size）表示在训练过程中，每次迭代（iteration）从全部样本中抽取的小批样本数量，当迭代次数和批尺寸的乘积达到样本总数时，就完成了一轮（epoch）训练。在显卡内存允许的范围内，批尺寸应该尽可能设置得大，这样可以最大程度地提高内存的使用效率，并且批尺寸越大，确定的误差梯度的下降方向越准确，模型效果越好。然而，批尺寸设置太大也有弊端，这样会导致每次迭代的计算时间过长，导致模型训练难度增大。在多次实验后，本章将批尺寸设置为 16。

4.3.5 网络训练策略

网络的训练过程由前向传播和反向传播反复循环形成，描述如下：

（1）输入图像通过层级特征提取模块、特征重组模块和深度图估计模块，实际上是各模块卷积层、池化层、激活函数的处理，到达输出层得到结果，完成一次前向传播过程；

（2）前向传播的计算结果与实际的真实标记（ground truth）有误差，通过目标函数（损失函数）的计算，将误差反向传播直到输出层，并且更新网络各层的参数；

（3）重复上述过程，不断迭代，直到网络最终收敛。

训练过程的流程示意如下图所示，当输入数据固定时，反向传播算法根据神经网络输出的敏感度计算神经网络中的参数。它计算输出结果 f 对所有参数 w 的偏微分，即$\partial f/\partial w_i$，f 是神经元的输出向量，w_i 是 f 的第 i 个参数。这样就可以计算出 f 对于各网络参数的梯度∇f，使用梯度下降法训练网络，即每次沿着梯度减小的方向$-\nabla f$移动一小步，循环往复，直到误差不再减少。

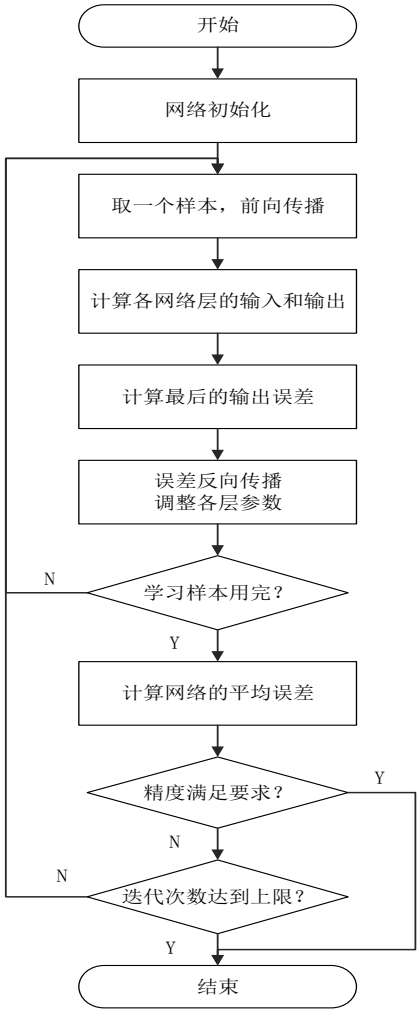


图 4-7 训练流程示意图

4.3.6 优化算法选择

在调整模型更新参数的方式时，通常会选用一些优化算法来使模型的效果更好，优化算法的目的是通过改善训练的方式来最小化损失函数 $E(x)$。损失函数 $E(x)$用于计算测试集中目标值 Y 的真实值和预测值的偏差程度。比如说，权重(W)和偏差(b)就是损失函数的内部参数，在训练神经网络模型

时有着重要作用。目前常用的优化算法有随机梯度下降（SGD）、动量（Momentum）和自适应时刻估计方法（Adam）等。

随机梯度下降(SDG)：对每个样本的训练进行参数更新，每次执行算法都更新一次。频繁的参数更新使得参数间具有较大的方差，损失函数会产生不同强度的波动。这样有助于发现更优的局部最小值，使得整体的训练效果更佳。

动量（Momentum）：SGD 方法中的高方差可能导致网络震荡而难以收敛，所以有研究者提出了可以通过优化相关方向的训练和弱化无关方向的振荡，来加速 SGD 的训练。也就是说，将 SGD 中更新向量的分量 γ 添加到当前更新向量。通常将动量项 γ 设置为 0.9 或相近的值，这里的动量与物理学中动量的概念是一致的。形象地说，比如从山上投出一个球，在球下落过程中小球的速度不断增加。当其梯度与实际移动方向一致时，动量项 γ 变大；当梯度和实际移动方向相反时，γ 减小。这意味着动量只对相关项的参数进行更新，减少了不必要的更新，从而使得网络更快地收敛，减少了振荡的过程。

自适应时刻估计方法（Adaptive Moment Estimation，Adam）：Adam 优化算法是随机梯度下降算法的扩展方式，近来其广泛用于深度学习中，尤其是计算机视觉与自然语言处理等任务。Adam 算法与传统的随机梯度下降算法存在很大的不同。随机梯度下降在训练时维持着单一的学习率，更新所有的参数权重。而 Adam 算法通过计算梯度的一阶矩估计和二阶矩估计为不同的参数匹配自适应的性学习率。Adam 不仅基于一阶矩均值计算适应性的学习率，同时还充分结合了梯度的二阶矩均值。

Adam 在深度学习领域内是十分流行的优化算法，相对于其他的优化算法具有很大优势。因此本文搭建的深度图估计网络模型中也采用 Adam 算法作为优化算法。

4.4 本章小结

本章提出了基于深度图的人脸真伪检测框架，整个检测框架分为两个部分。首先是深度图真实标记的生成方案，分别生成真假脸深度图的真实标记来指导下一步网络模型的训练；然后是深度图提取的网络模型，通过训练网络，提取包含人脸全局信息的深度图，并以此作为人脸的静态特征，用于区分人脸真伪。该检测框架能够有效解决静态特征中存在的没有明确的监督信息和泛化性差等问题，具有更好的通用性和适应性。

第5章 人脸动态特征的提取方法

前文提到现有的基于深度学习的特征提取方法中存在对于人脸视频的采集设备和方式等比较敏感、对不同数据集的泛化性差的问题，为了解决这一问题，本章将提出一种人脸动态特征的提取方法，该方法包含光流引导特征残差模块、卷积门控循环单元模块、注意力机制模块等部分，本章将依次对各个子模块功能和具体设计进行介绍。

5.1 方法的整体架构

如下图所示，动态特征检测框架的整体架构分别由层级特征提取模块、光流引导特征残差模块（Optical Flow Guided Feature-Residual Block，OFF-ResB）、卷积门控循环单元（Convolution Gated Recurrent Unit，CGRU）和注意力机制模块四个部分组建而成。人脸动态特征提取方法的架构更加复杂，将静态特征提取方法的部分模块作为子网络，利用 OFF-ResB 模块提取连续帧中蕴藏的光流引导特征，以其作为 CGRU 模块的输入之一，进一步捕获一定时间段内连续帧的时空域信息，并采用注意力机制权衡各帧的影响权重，从而得到代表了人脸运动模式的多帧动态特征，最后将其与单帧静态特征进行融合，用来区分人脸真伪。因此，整个框架巧妙地将光流对动作信息的表示、CGRU 对时间信息的把控以及注意力机制对帧与帧之间关系的诠释结合起来，为区分人脸的真伪提供了有力的依据。

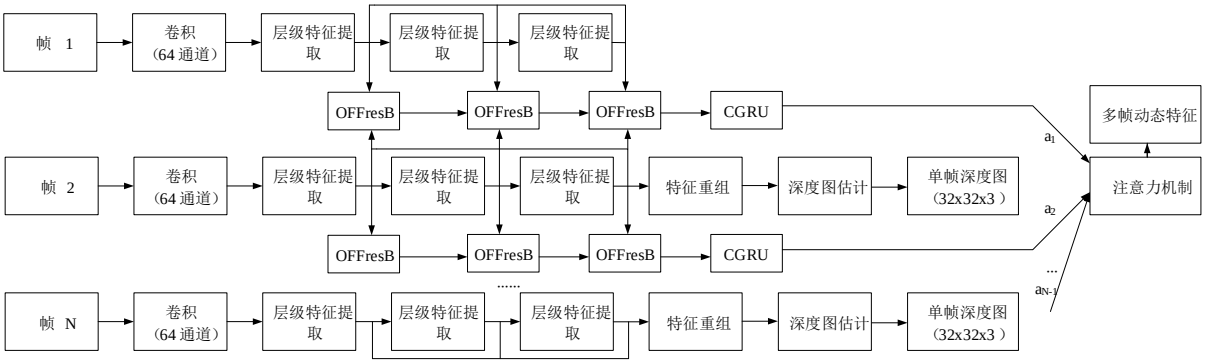


图 5-1 动态特征提取方法的整体架构

5.2 光流引导特征残差模块

光流引导特征残差模块（OFF-ResB）由光流引导特征模块（Optical Flow Guided Feature Block，OFFB）与残差模块（Residual Block，ResB）组成，如下图所示。OFFB 子模块对不同卷积层的层级特征进行处理后，输入到 ResB 子模块中，经 ResB 模块处理后，再作为下一个 OFF-ResB 模块的输

入。

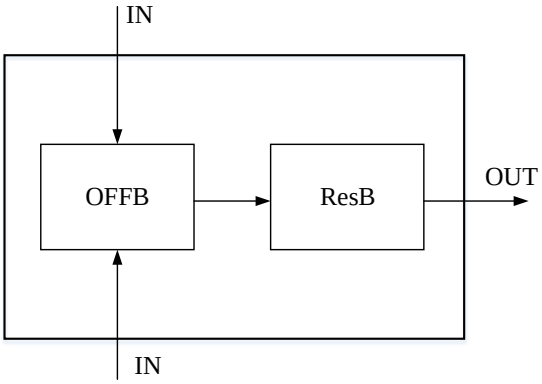


图 5-2 OFF-ResB 模块

5.2.1 光流引导特征模块

光流引导特征模块（OFFB）的作用是提取光流引导特征[47]，作为连续帧动作信息的一种表达方式。光流引导特征来源于传统的光流[48]的概念，著名的亮度常数约束条件表示如下：

$$I(x,y,t)=I(x+\Delta x,y+\Delta y,t+\Delta t) \tag{5.1}$$

其中 $I(x, y, t)$表示某一帧在时刻 t 位于位置(x, y)处的像素点的亮度大小，Δx 和Δy 分别表示该像素点在时刻 t 到时刻$(t+\Delta t)$过程中，其在 x 轴和 y 轴上的空间像素位移（spatial pixel displacement）。表明帧中的任意像素点从时刻 t 到$(t+\Delta t)$的过程中，虽然移动了位置，但是其亮度保持不变。因为特征图本质上也是二维图像，由一个个像素点组成，所以基于这一假设，可以在特征图上特到类似的结论，表示如下：

$$f(I;w)(x,y,t)=f(I;w)(x+\Delta x,y+\Delta y,t+\Delta t) \tag{5.2}$$

其中 f 表示从图像（帧）I 提取特征的映射函数，w 表示的是该映射函数的参数。当然，在本文中映射函数就是前文介绍的层级特征提取模块（HFEB），由堆叠的卷积层、池化层、ReLU 函数组成。

根据光流的定义，本节假设在某一帧的特征图中位置（x,y）处的像素点为 $p=(x, y, t)$，则有：

$$\frac{\partial f(I;w)(p)}{\partial x}\Delta x+\frac{\partial f(I;w)(p)}{\partial y}\Delta y+\frac{\partial f(I;w)(p)}{\partial t}\Delta t=0 \tag{5.3}$$

对上式两边同除以Δt，可得：

$$\frac{\partial f(I;w)(p)}{\partial x}v_x+\frac{\partial f(I;w)(p)}{\partial y}v_y+\frac{\partial f(I;w)(p)}{\partial t}=0 \tag{5.4}$$

其中(v_x, v_y)表示特征点 p 处的瞬时二维速度（instantaneous two dimensional velocity），$\frac{\partial f(I;\ w)(p)}{\partial x}$ 表示$\partial f(I;\ w)(p)$在 x 轴上的空间梯度（spatial gradient），$\frac{\partial f(I;\ w)(p)}{\partial y}$表示$\partial f(I;\ w)(p)$在 y 轴上的空间

梯度，而$\frac{\partial f(I;\ w)(p)}{\partial t}$是在时间轴上的时间梯度（temporal gradient）。当且仅当 $f(I;\ w)(p) = I(p)$时，$f(I;\ w)(p)$直接表示像素点 p，在这种特殊情况下，(v_x, v_y)被称为光流。通过求解具有式(4.4)的约束条件的最优化问题，可以得到每个 p 点的光流。

从式(4.4)中可以看出，向量$(\frac{\partial f(I;\ w)(p)}{\partial x}, \frac{\partial f(I;\ w)(p)}{\partial y}, \frac{\partial f(I;\ w)(p)}{\partial t})$是与特征流向量$(v_x, v_y, 1)$正交，将向量$\vec{F}(I;\ w)(p)$定义如下：

$$\vec{F}(I;w)(p)=\left[\frac{\partial f(I;w)(p)}{\partial x},\frac{\partial f(I;w)(p)}{\partial y},\frac{\partial f(I;w)(p)}{\partial t}\right] \tag{5.5}$$

不难发现，如果特征流向量发生变化，由式(4.4)，则向量$\vec{F}(I;\ w)(p)$也相应改变，反之亦然。于是，向量$\vec{F}(I;\ w)(p)$引导着特征流（特征层的光流）的改变，因此$\vec{F}(I;\ w)(p)$被称为光流引导特征（OFF）向量。因为光流表达了连续帧在时间维度上的动作信息，作为光流的引导特征，OFF 实际上相当于编码了连续帧的时空信息。

OFFB 在卷积网络中的具体实现如下图所示，一个 OFFB 模块有 3 个输入和 1 个输出。三个输入来自于时间和层级两个不同的维度，第一个输入是层级特征提取模块（HFEB）提取的 t（即 t_1）时刻帧的特征$F_l^o(t)$，第二个输入是 HFEB 在（t+Δt）（即 t_2）提取到的$F_l^o(t+\Delta t)$，第三个输入是上一个层级的 OFFB 提取到的特征 $OFFB_{l-1}(t)$。前两个输入通过 1×1 的卷积之后，一方面进行元素相减（element-wise subtraction）得到时间梯度$F_l^T(t)$，另一方面分别利用 Sobel 算子得到各自的空间梯度$F_l^S(t)$和$F_l^S(t+\Delta t)$。$F_l^T(t)$、$F_l^S(t)$、$F_l^S(t+\Delta t)$再与第三个输入 $OFFB_{l-1}(t)$进行组合，组合特征通过 3×3 卷积之后得到输出 $OFFB_l(t)$，并作为 ResB 模块的输入。

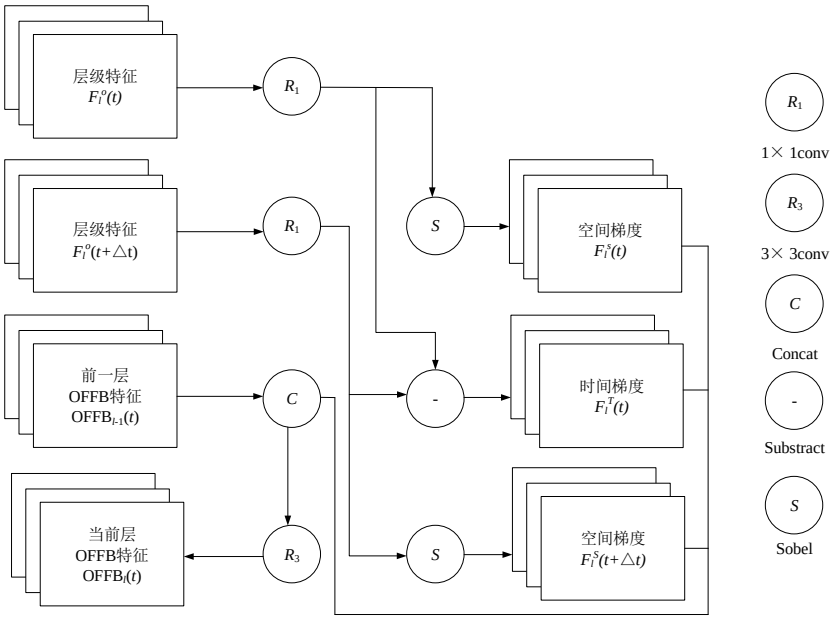


图 5-3 OFFB 的内部结构图

以下是 OFFB 模块内部的几个运算操作的具体细节：

（1）1×1 卷积

1×1 卷积可以调整（升维或降维）输入图像（或特征图）的通道数目，用于解决网络架构中各模块间关于通道数的适配问题。本节将 1×1 卷积的通道数设计为 128，即 1×1×128 的卷积核，将输入 OFFB 模块的特征图通道数统一调整为 128，方便计算。

（2）Sobel 算子

Sobel 算子是一种离散一阶差分算子，通过与图像的卷积，可以计算图像亮度函数一阶梯度的近似值。本节设计的 Sobel 算子用两个 3×3 的矩阵表示，分别计算特征图在横向（x 轴）和纵向（y 轴）的空间梯度。两个算子如下所示：

$$S_x=\begin{bmatrix}-1 & 0 & 1\\ -1 & 0 & 1\\ -1 & 0 & 1\end{bmatrix},\ S_y=\begin{bmatrix}1 & 1 & 1\\ 0 & 0 & 0\\ -1 & -1 & -1\end{bmatrix} \tag{5.6}$$

（3）空间梯度

用 $f(I)$表示图像 I 的基础特征图，$f(I,c)$表示特征图 $f(I)$在第 c 通道上的特征。根据式(5.4)，定义 F_x 和 F_y 分别为光流引导特征 OFF 在 x 方向和 y 方向上的空间梯度，数学表示如下：

$$F_x=\left\{\begin{bmatrix}-1 & 0 & 1\\ -1 & 0 & 1\\ -1 & 0 & 1\end{bmatrix}*f(I,c)\middle|c=0,...,N_c-1\right\} \tag{5.7}$$

$$F_y=\left\{\begin{bmatrix}1 & 1 & 1\\ 0 & 0 & 0\\ -1 & -1 & -1\end{bmatrix}*f(I,c)\middle|c=0,...,N_c-1\right\} \tag{5.8}$$

其中*代表卷积操作，N_c 表示特征 $f(I)$通道的总数。

（4）时间梯度

由式（5.4），光流引导特征 OFF 的时间梯度可以由 t 时刻的特征 $f_t(I,c)$与(t-Δt)时刻的特征 $f_{t-\triangle t}(I,c)$进行元素相减得到，数学表示如下：

$$F_t=\left\{f_t(I,c)-f_{t-\Delta t}(I,c)\middle|c=0,...,N_c-1\right\} \tag{5.9}$$

综上，在计算得到空间梯度 F_x、F_y 和时间梯度 F_t 之后，根据式（5.5）可以得到像素点（x,y）在 t 时刻于当前层级 l 的光流引导特征向量为$\vec{F_1}(t)=(F_x，F_y，F_t)$ 。为了作区分，这里把 OFFB 子模块当前层的输出向量表示为OFFB$_l(t)$。则光流引导特征向量 $l(t)$再与上一层级 l-1 的OFFB 输出OFFB$_{l-1}(t)$进行合并，得到当前层 l 的 OFFB 模块的输出向量 OFFB$_l(t)$，用©表示合并操作，则数学表示如下：

$$OFFB_l(t) = l(t) \copyright OFFB_{l-1}(t) \tag{5.10}$$

这样，可以用上一层级的 OFFB 向量和当前层级的空间、时间信息来计算当前层级的 OFFB 向量。

5.2.2 残差模块

将得到的 OFFB 向量输入到残差模块（ResB）中，对 OFFB 向量进一步处理，作为下一个 OFFB 子模块的输入。ResB 模块的结果如下图所示，由于残差模块的特性，能够在网络加深的同时，防止网络退化，提高网络性能。

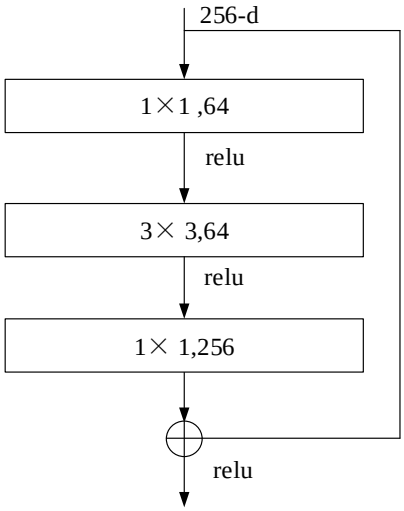


图 5-4 ResB 模块内部结构

ResB 模块内部采用的是三层的残差学习单元的结构，首先 1×1 的卷积单元将输入的 OFFB 向量的通道数转变成 64，再通过激活函数 ReLU；然后 3×3 的卷积能够提升网络的表达能力，再经过 ReLU 函数进行非线性变换；最后再用 1×1 卷积进行升维，将特征的通道数转变成 256。

总的来说，由 HFEB 模块提取的各层级特征输入到 OFFB 模块，处理后得到光流引导特征并与上一层级向量合并后得到 OFFB 向量，再通过 ResB 模块进一步提高特征的表达能力，得到 OFF-ResB 向量，作为 CGRU 模块的输入。

5.3 卷积门控循环单元模块

卷积门控循环单元（CGRU）是循环神经网络的一种衍生体，能够捕获时间序列数据的长期依赖（long-term dependency）。本节利用 CGRU 来处理不同时间段 OFF-ResB 模块的输出，从而探索连续帧图像之间的动作关系。CGRU 来源于 GRU，而 GRU 则是前文介绍的长短期记忆（LSTM）网络的

一种变体。

5.3.1 门控循环单元

门控循环单元（Gated Recurrent Unit， GRU）与 LSTM 一样能够解决 RNN 的长期依赖问题。相较于 LSTM，GRU 将输入门和忘记门合并为一个更新门，减少计算量和参数数量的同时，又保持了 LSTM 的处理效果，因此 GRU 组成网络的计算成本和收敛速度都要优于 LSTM。GRU 与 LSTM 的对比情况如下表所示：

表 5.1 GRU 与 LSTM 的对比

类别	LSTM	GRU
门数量	3	2
激活函数	Sigmoid	Sigmoid
输入量	h_{t-1},c_{t-1},x_t	h_{t-1},x_t
输出量	h_t,c_t	h_t

LSTM 引入了输入门、遗忘门和输出门三个门函数来控制输入值、记忆值和输出值，而 GRU 中只有两个，分别是更新门与重置门，如下图所示。图中 z_t 表示更新门，r_t 表示重置门。GRU 利用更新门来控制前一时刻的状态信息 h_{t-1} 在当前时刻状态 h_t 中的影响程度，更新门的输出值越大则说明其在当前状态的影响越大，带入的信息越多。重置门控制 h_{t-1} 在当前状态候选状态$\tilde{h}(t)$上的影响程度，重置门的输出越小，则前一状态信息影响越小，写入得越少。

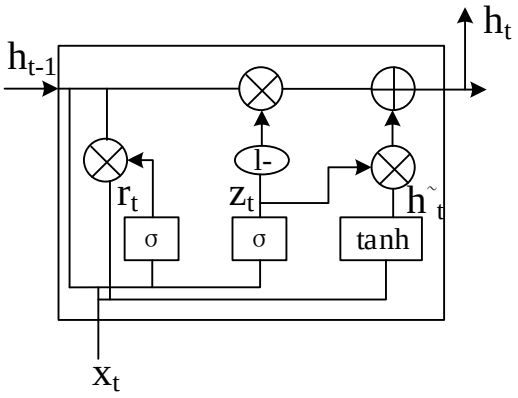


图 5-5 GRU 的内部结构

根据上图，可以得到 GRU 单元的前向传播（forward propagation）公式，如下：

$$\begin{aligned} r_1 &= \sigma(W_r \cdot [h_{t-1}, x_t]) \\ z_t &= \sigma(W_z \cdot [h_{t-1}, x_t]) \\ \tilde{h}_t &= \tanh(W_{\tilde{h}} \cdot [r_t * h_{t-1}, x_t]) \\ h_t &= (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t \\ y_t &= \sigma(W_o \cdot h_t) \end{aligned} \tag{5.11}$$

其中[ ]代表两个向量拼接，*表示的是矩阵的元素相乘。

在训练过程中，GRU 组成的网络需要学习的参数是 W_r、W_z、W_h、W_o。对于权重矩阵 W_r，其作用的输入量$[h_{t-1},x_t]$是由 h_{t-1} 和 x_t 两个向量拼接而成，因此求解时需要分离开来分别计算，W_z 和 W_h 也与之类似，表示如下：

$$\begin{aligned} W_r &= W_{rx} + W_{rh} \\ W_z &= W_{zx} + W_{zh} \\ W_{\tilde{h}} &= W_{\tilde{h}x} + W_{\tilde{h}h} \end{aligned} \tag{5.12}$$

而对于 W_o，它是输出层的参数矩阵，输出层的输入量$y_t^i = W_o h$ ，输出量为$y_t^o = \sigma(y_t^i)$ 。

对于由多个 GRU 单元组成的网络而言，在输出层输出最后前向传播的结果后，根据输入量对应的真实标记（ground truth）就可以计算误差（损失），然后用误差反向传播（error back propagation）算法更新各单元的参数，直到整个网络收敛。下面以单个 GRU 单元为例，推导训练时各参数的求解过程：

假设单个样本在时刻 t 的输出量的真实标记为 y_d，则 GRU 对于该样本在 t 时刻的预测误差可以设为：

$$E_t = \frac{1}{2}(y_d - y_t^o)^2 \tag{5.13}$$

则该样本在所有时刻的损失为：

$$E = \sum_{t=1}^{T} E_t \tag{5.14}$$

那么，可得损失函数对 7 个参数的偏导数为：

$$\begin{aligned} &\frac{\partial E}{\partial W_o} = \delta_{y,t} h_t, \frac{\partial E}{\partial W_{zx}} = \delta_{z,t} x_t \\ &\frac{\partial E}{\partial W_{zh}} = \delta_{z,t} h_{t-1}, \frac{\partial E}{\partial W_{\tilde{h}x}} = \delta_t x_t \\ &\frac{\partial E}{\partial W_{\tilde{h}h}} = \delta_t (r_t \cdot h_{t-1}) \\ &\frac{\partial E}{\partial W_{rx}} = \delta_{r,t} x_t, \frac{\partial E}{\partial W_{rh}} = \delta_{r,t} h_{t-1} \end{aligned} \tag{5.15}$$

其中，各个中间参数可以由已知量计算出，公式如式 5.16 所示。在计算出各个偏导数之后，利用反向传播算法，就可以更新各个参数，然后依次迭代，直到损失收敛。

综上，单个 GRU 中各个参数在训练时的更新过程已经明了，而对于多个 GRU 组成的大型网络，其内部的参数更新可以看成是以单个 GRU 为单元的逐个传导和层层迭代直到整个网络收敛的过程。

$$
\begin{aligned}
&\delta_{y,t} = (y_d - y_t^o)\cdot\sigma' \\
&\delta_{h,t} = \delta_{y,t}W_o + \delta_{z,t+1}W_{zh} + \delta_{t+1}W_{\tilde{h}h}\cdot r_{t+1} + \delta_{h,t+1}W_{rh} + \delta_{h,t+1}\cdot(1-z_{t+1}) \\
&\delta_{z,t} = \delta_{t,h}\cdot(\tilde{h}_t - h_{t-1})\cdot\sigma' \\
&\delta_t = \delta_{h,t}\cdot z_t\cdot\phi' \\
&\delta_{r,t} = h_{t-1}\cdot[(\delta_{h,t}\cdot z_t\cdot\phi')W_{\tilde{h}h}]\cdot\sigma'
\end{aligned}
\tag{5.16}
$$

5.3.2 卷积门控循环单元

普通的 GRU 虽然能够得到输入序列的时序关系，但是忽略了输入的空间信息。本节利用卷积操作能够反映相邻像素点位置关系的特性，将卷积操作引入 GRU 单元，形成卷积门控循环单元（Convolution Gated Recurrent Unit）来提取输入连续帧的时间特征和空间特征（即时空特征）。

$$
\begin{aligned}
&R_t = \sigma(K_r \otimes [H_{t-1}, X_t]) \\
&U_t = \sigma(K_u \otimes [H_{t-1}, X_t]) \\
&\hat{H}_t = \tanh(K_{\hat{h}} \otimes [R_t * H_{t-1}, X_t]) \\
&H_t = (1-U_t) * H_{t-1} + U_t * \hat{H}_t
\end{aligned}
\tag{5.17}
$$

其中 X_t，H_t，R_t，U_t 分别表示 t 时刻的输入、隐层状态、重置门输出和更新门输出，K_r,K_u, $K_{\hat{h}}$表示对应的三个卷积核，⊗ 表示卷积操作， *表示矩阵的元素乘积，σ 表示 sigmoid 激活函数。以上所示的公式组和原始 GRU 公式相比，相当于用张量卷积代替了向量乘法。CGRU 训练时的前向传播和误差反向传播公式推导与 GRU 类似，在此不再赘述。

CGRU 模块对含有光流引导特征的 OFF-ResB 向量进行了处理，在获得连续帧时序关系的同时，还保留了帧本身的空间信息，得到了含有原始帧时间和空间信息的特征，简称时空特征。大量的 CGRU 模块通过逐个连接和层层传导组成了 CGRU 网络，能够对不同时间段的连续帧进行处理，得到帧与帧之间的短期依赖（short-term dependency）和长期依赖（long-term dependency），从而提取到不同时间段连续帧的时空特征。

以上通过 OFF-ResB 和 CGRU 模块提取到了连续帧的时空特征，下一小节将利用注意力机制对不同时间段视频帧进行进一步处理。

5.4 注意力机制

注意力机制（Attention Mechanism）最早起源于生物学，借鉴了人类视觉注意力的基本原理。具体来说，人在观察图像时，通过快速的扫描全局图像，获取需要重点关注的区域，然后将注意力焦点集中在此，对该区域投入更多的注意力资源，从而观察到该目标区域更多的细节信息，抑制其他

区域的无用信息。人的视觉注意力促使人能够利用有限的注意力资源，快速获取更多有效的、有价值的信息，提高了视觉处理信息的准确性和高效性。

注意力机制[49][50][51][52]在图像分类、图像分割、图像理解等方面的成功应用，使得注意力模型在某些场景下已经成为一种优化网络模型、提高网络性能的常用手段。比如，有的研究文献中提到的SENet[53]在特征通道层面采用了注意力机制，通过学习的方式对每一个特征通道的重要程度进行重新排布，然后根据重要程度去增强对当前任务的有效特征并抑制无效特征。

不同于 SENet 在特征通道维度的处理，本节在不同时间段的连续帧层面采用了注意力机制，目的是区分不同时间段连续帧的重要程度，按照重要程度对每一小段视频帧的时空特征进行加权，得到表达能力更强、更加有效的连续帧的时空特征。

5.4.1 注意力机制的流程

目前很多文献中的注意力机制采用了编码—译码的架构，如图 5-6 所示。首先，在编码部分，编码器通过非线性变换对输入向量进行重新编码，得到中间层向量；然后，在译码部分，译码器将中间层向量重新译码输出；通过输入与输出的对应关系用 softmax 函数计算对应的权值，对原始输入在空间或通道等维度按重要性程度重新赋予权值，得到相应的注意力分布，如图 5-7 所示。

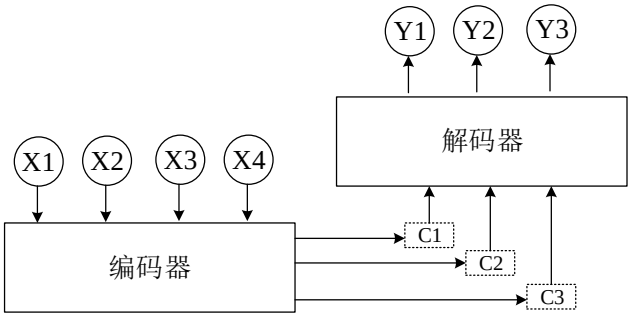


图 5-6 注意力机制架构图

然而，注意力机制的本质并不局限于编码—译码这种形式。因此，本节将注意力机制的本质思想从编码—译码架构中抽离，用更加宽泛的流程加以说明。

注意力机制的目的是将原始输入序列按照对结果的影响程度重新排布，得到注意力分布概率。本节将注意力机制的流程定义如下：

（1）如下图所示，假设源域（Source）中有一系列的<key, value>数据对，对于目标域中某个指定元素 query， 计算 query 与源域中每个 key 的相关性；

（2）按照 query 与每个 key 相关程度大小，赋给各个 key 所对应的 value 的权重系数；

（3）对每个 value 按其权重系数进行加权求和，得到引入注意力机制后的输出结果。

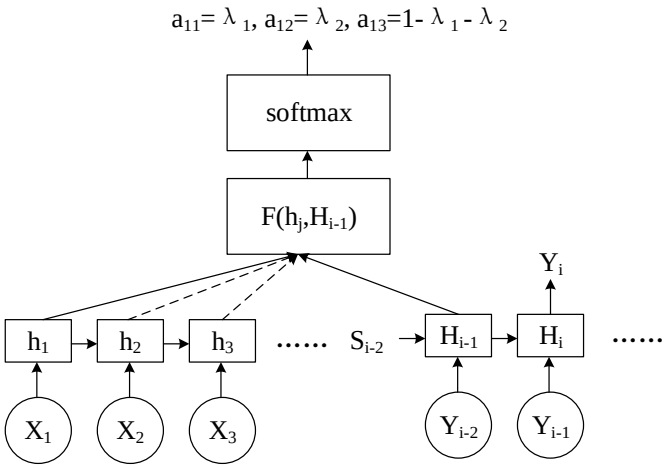


图 5-7 注意力分布概率计算

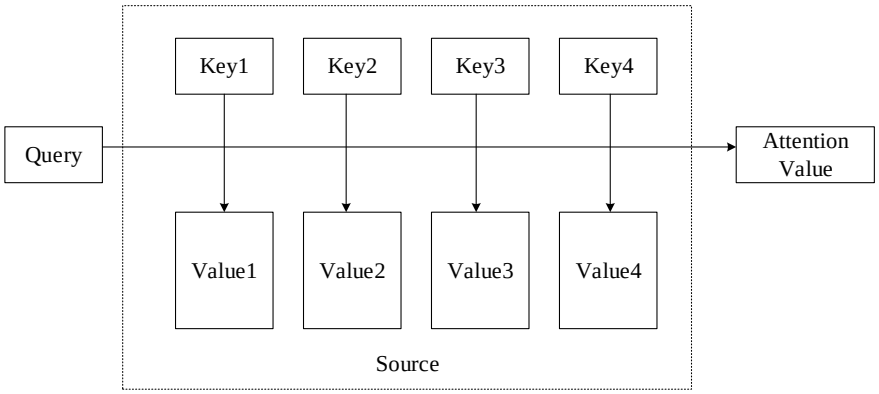


图 5-8 注意力机制的流程

由图 5-8，可以将注意力机制可以理解为，从大量的信息中选择性地筛选出少量重要的信息并聚焦到这些重要的信息上，忽略大多数不重要信息。聚焦的这一过程体现在权重系数的计算上，权重越大表明聚焦程度越高，即权重表示了信息的重要程度，而 value 就是对应的信息。

注意力机制的计算过程可以分为三个阶段：第一阶段，根据 query 和 key 计算权重系数；第二阶段，对权重进行归一化处理；第三阶段，依据权重系数对 value 进行加权求和。整个计算过程如图 5-9 所示。

在第一阶段，对于权重系数的计算函数 $F(Q,K)$有不同的选择，常用的由以下点乘方式（dot）、一般方式（general）和连接方式（concat）三种，如下式所示：

$$score(h_t,\bar{h}_s)=\begin{cases} h_t^T\bar{h}_s, dot \\ h_t^T W_a\bar{h}_s, general \\ v_a^T \tanh(W_a\left|h_t;\bar{h}_s\right|), concat \end{cases} \tag{5.18}$$

分别对应点乘、权值网络映射和组合映射。因此，计算得出的权值大小根据选用的方式不同其数值的取值范围也不同。

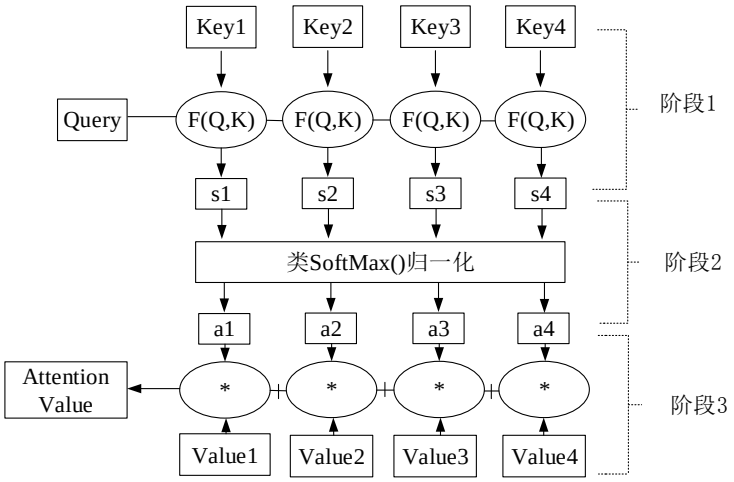


图 5-9 注意力机制的计算过程

在第二阶段，由于第一阶段采用不同方式计算得到的权值范围不同，所以需要对权值进行归一化处理，将原始的分值转换成所有元素权重之和为 1 的概率分布，而且 softmax 函数的内在机制也会突出重要性高的权重。数学表示如下：

$$a_i = soft\max(Sim_i) = \frac{e^{Sim_i}}{\sum_{j=1}^{L_x} e^{Sim_j}} \tag{5.19}$$

在第三阶段，将第二阶段的计算得到的各个归一化权值 a_i 和对应的 $value_i$ 加权求和，得到最终的输出，即 Attention 数值，表示如下：

$$Attention(Query, Source) = \sum_{i=1}^{L_x} a_i \cdot value_i \tag{5.20}$$

其中 L_x=||Source||表示 Source 中 key（或 value）的数量。

5.4.2 注意力生成网络

如图 5-10 所示，本节设计的注意力生成网络也是编码—译码架构。其中编码部分是 CGRU 网络，由 CGRU 单元组成；中间的 Attention 模块是注意力生成网络的核心部分；译码部分同样也是 CGRU 网络，编码—译码的网络结构借鉴了 U-Net 的设计思想，能够对图像的特征进行很好地诠释和表达。

OFF-ResB 模块输出的 OFF-ResB 向量是注意力生成网络的输入向量，同样也是编码部分 CGRU 网络的输入。若干个 OFF-ResB 向量组成输入序列，用 x 表示，则有$x =$ （$x_1, \ldots, x_{T_x}$）。假设译码部分 CGRU 网络的输出序列为 y，则 y 可以表示为$y =$ （$y_1, \ldots, y_{T_y}$）。注意这里的下标指的是时间点，编码部分的输入 x 和译码部分的输出 y 并不是一一对应的。

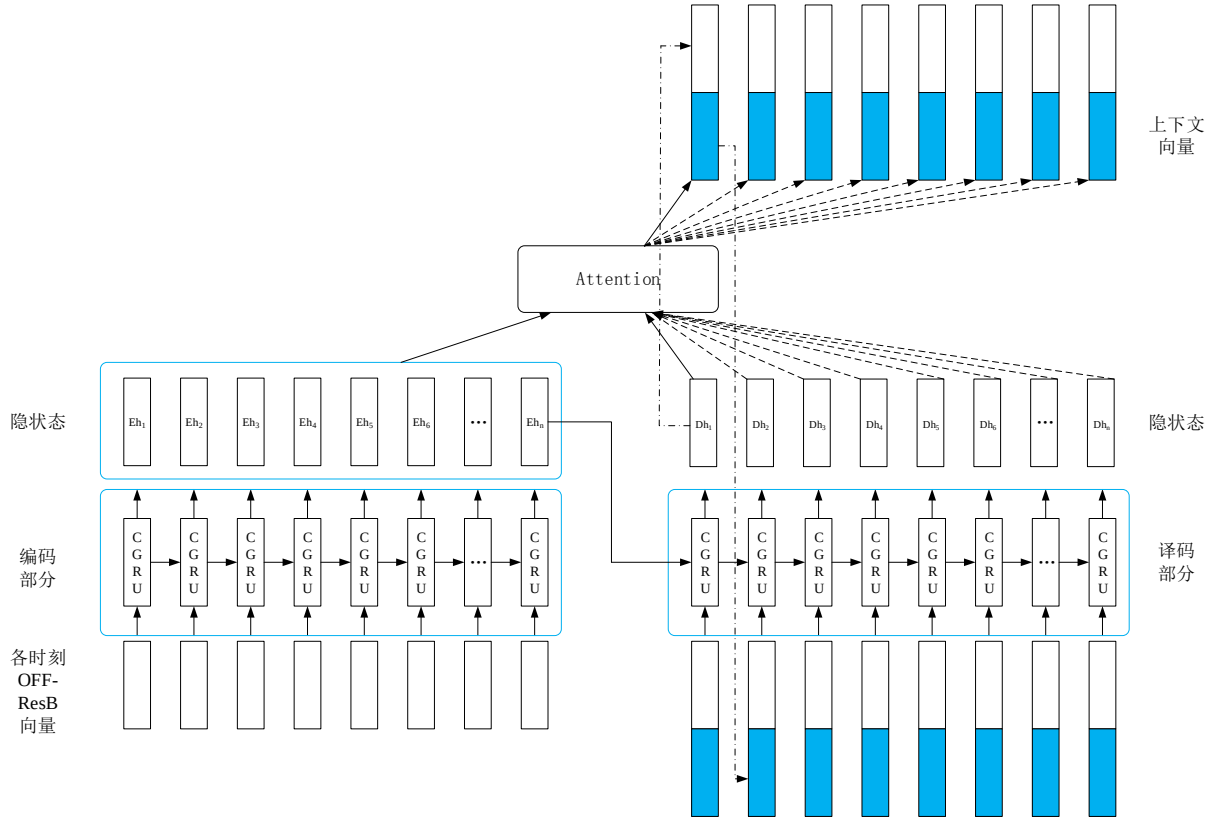


图 5-10 注意力生成网络

则对于时刻 t（$T_x<t<T_y$），整个注意力生成网络的内部关键参数的计算方法如下：

（1）编码部分的隐状态 Eh

在时刻 t，编码部分的输入为 x_t，与上一时间点的隐状态 Eh_{t-1}，通过 CGRU 单元的处理，输出该时刻的隐状态 Eh_t，表示为：$Eh_t = \mathrm{CGRUenc}\ (x_t, Eh_{t-1})$。

（2）译码部分的隐状态 Dh

在 t 时刻，将 D_t 和 t-1 时刻译码部分的隐状态 Dh_{t-1} 通过 CGRU 单元处理，得到 t 时刻的隐状态 Dh_t，表示如下：$Dh_t = \mathrm{CGRUdec}(y_{t-1}, Dh_{t-1})$。

（3）上下文向量 c

顾名思义，上下文向量是表示相邻时刻点关联信息的一个量，利用上下文向量可以从前一时刻的隐状态预测下一时刻的隐状态。数学表示如下：

$$c_i = \sum_{j=1}^{T_x} a_{ij} Eh_j \tag{5.21}$$

也就是说 c_i 是编码部分隐状态序列（$Eh_1, Eh_2, \ldots, Eh_{Tx}$）的加权平均。

（4）注意力分布权重 a

a 是编码部分各个时间点的隐状态对应的权重，反应的是各个隐状态的重要程度，因此也是注意力分布情况的表现，定义为：

$$a_{ij}=\frac{\exp(e_{ij})}{\sum_{k=1}^{T_x}\exp(e_{ik})} \tag{5.22}$$

（5）相关性得分 e

e 是译码部分的隐状态和编码部分隐状态的相关性得分，定义为 e_{ij}=score(Dh_i,Eh_j)。

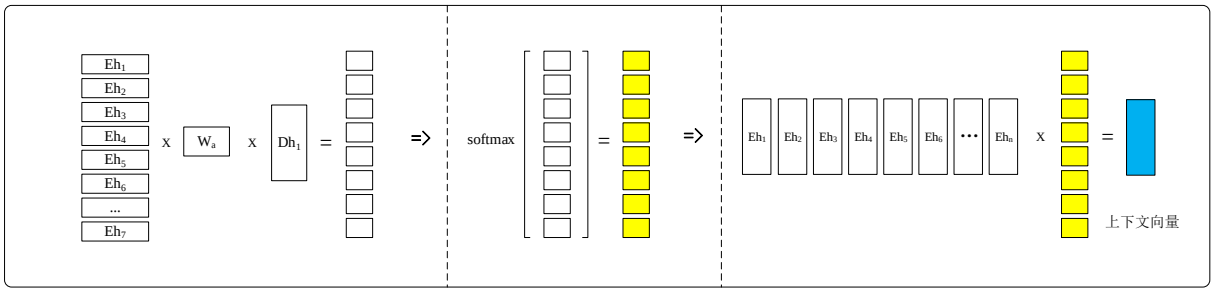
（6）译码部分的输出 y_t

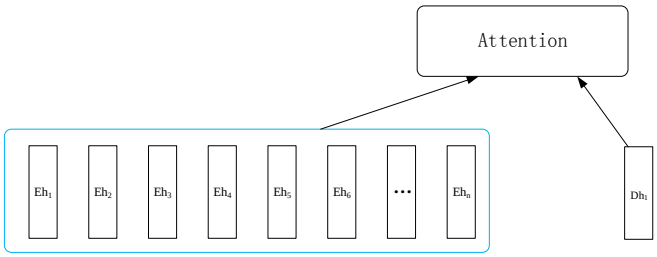
令 y_t 表示 t 时刻译码部分的输出，则有 $y_t=\tanh(W_c[Dh_t, c_t])$。其中 Dh_t 表示 t 时刻译码部分的隐状态，c_t 表示 t 时刻的上下文（context）向量,W_c 是两个向量的连接矩阵。

因为网络的计算是循环往复的，为了方便描述，采用数学上微分的思想，可以从序列中的某个时刻 t 切入；又因为网络具有记忆特性，那么在 t 时刻前的各个时间点相关参数都是已知的。因此，对于时刻 t 网络中各参数的变化过程是：

（a）首先，在编码部分可以计算出隐状态序列 $EH_{Tx}=(Eh_1, Eh_2, \ldots, Eh_{Tx})$，注意 $T_x<t$，也就是说在 t 时刻前，EH_{Tx} 已经“记忆”在网络中；

（b）然后，计算 t 时刻译码器的隐状态 Dh_t，即单个 CGRU 单元的输出；





编码部分各CGRU单元从时刻1到n的隐状态　　译码部分CGRU单元在时刻1的隐状态

图 5-11 上下文向量的计算过程

（c）接着，将 EH_{Tx} 和 Dh_t 输入到注意力模块，计算上下文向量 c_t；

（d）最后，将上下文向量 c_t 和译码器隐状态 Dh_t 进行非线性组合，得到译码器的输出 y_t，并作为下一时间点译码器的输入，如此循环计算。特别地，当 $t=T_y$ 时，得到输入特征序列在注意力机制下的最终结果。

步骤(c)中，由注意力分布权重计算上下文向量的过程中，采用上一小节中提到的第二种（general）score 函数，计算过程如图 5-11 所示。

至此，本章完成了对基于动态特征的人脸真伪检测框架各个子模块的设计。

5.5 本章小结

本章提出了一种人脸动态特征的提取方法，旨在探索连续视频帧中人脸动作模式，发现能够更有效地区分人脸真伪的特征。方法的架构分为四大部分：首先是层级特征提取模块，用于获取单帧图像中的不同层级特征；然后是光流引导特征残差模块，它将光流引导特征经过残差模块的处理作为短期动态特征，来表示相邻两帧图像中人脸的动作信息；接着是卷积门控循环单元模块，提取连续帧之间的长期依赖信息；最后是注意力生成网络，它把一段时间内的视频帧按照重要程度加权组合，得到更有效的长期动态特征，来代表该时间段内视频中人脸的动作模式作为动态特征，并以此特征来区分人脸真伪。该动态特征提取方法能够得到泛化性能更强、鲁棒性更高的人脸特征，从而提高人脸真伪检测性能。

第6章 检测方案的实验和结果分析

前文介绍了人脸静态特征和动态特征的提取方法以及基于两种特征相融合的检测方案，而本章将分别对静态特征、动态特征和融合特征进行相应的实验和分析。首先本章将对三个常用的人脸防伪数据集进行介绍，然后对深度学习平台实验方案进行设计，并完成实验数据预处理、训练参数设置以及实验结果分析。最后，本章将对静态特征、动态特征和融合特征的实验结果进行对比，验证融合特征对于提升人脸真伪检测性能的优越性。

6.1 实验设置

6.1.1 数据集

实验所用的数据集是 CASIA-MFSD[54]、Replay-Attack[55]和 SiW[56]，它们属于人脸防伪领域最常用的几个数据集，模拟了不同场景下的打印照片攻击和重放视频攻击，主要用于设计的深度学习网络的训练和测试。三种数据集说明如下：

（1）CASIA-MFSD：该数据库包含 50 个不同身份的人，每个人在 3 种不同图像分辨率和光照条件下生成了 12 个视频，总共 600 个视频，其中 20 个人的 240 个视频用于训练，另外 30 个人的 360 个视频用于测试。如下图所示，视频模拟了不同条件下的打印攻击和重放攻击。其中，左上角的 4 张图片代表低质量（Low Quality）视频，左下角是正常质量（Normal Quality）视频，右边是高质量（High Quality）视频。对于每种质量视频，从左往右依次为真实人脸、弯曲照片攻击、剪切照片攻击和重放视频攻击。

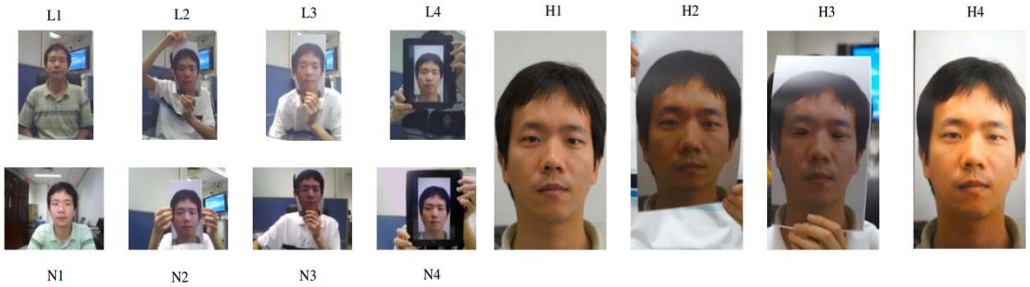


图 6-1 CASIA-MFSD 数据集示例

（2）Replay-Attack：该数据库包含 50 个不同人员，总共 1200 个真脸和假脸的视频，其中训练集、开发集和测试集的人数分别为 15，15 和 20。这些视频是在两种不同的光照条件下采集的，一种

是受控（controlled）的光照条件，一种是不利（adverse）的光照条件，如下图所示。其中，第一行是在受控光照条件下，第二行是在不利光照条件下。对于两种光照条件，从左向右分别表示真实人脸、打印照片攻击、手机重放视频攻击和平板电脑重放视频攻击。

图 6-2 Replay-Attack 数据集示例

（3）SiW：该数据库有 165 个不同的被测人员，每人有 8 个真脸视频和 20 个假脸视频，总共 4620 个视频。这些视频中人脸与摄像机的距离、光照、姿态和表情等都有变化，采集数据的设备也各有不同，模拟的欺骗攻击场景也更加复杂，因此也更具挑战性，如下图所示。其中，第一行表示的是真实人脸视频，从左往右前 3 张反应的是距离变化，后 3 张分别表示的是姿态、表情和光照的改变；第二行代表的是不同的攻击场景，前 2 张表示的是图像质量不同的两种打印攻击，后 4 张表示的是不同设备上的重放视频攻击。

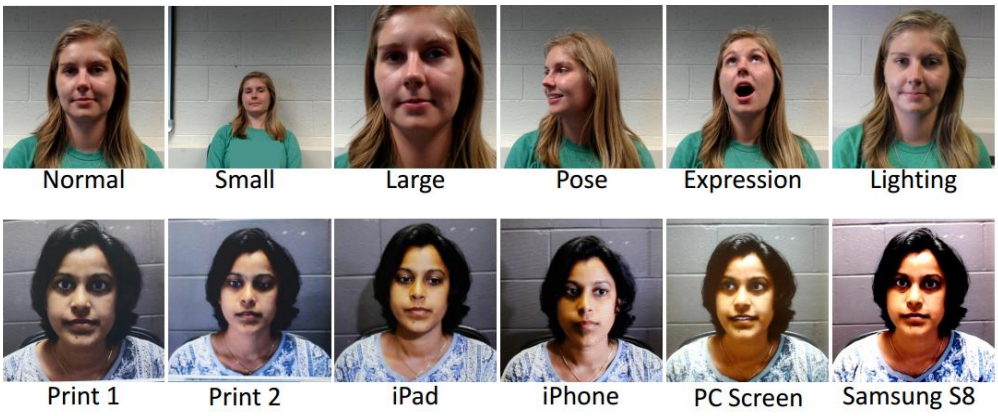


图 6-3 SiW 数据集示例

综上所述，三种数据集用不同的方式模拟了打印攻击和重放攻击。各个数据集不仅在人物数据采集的数量上不同，采集的方案和攻击的设备也有所不同。本文对三个数据集中的各要素，包括人物数量、数据形式、真假数量、人脸姿态、表情变化、光照变化、攻击方式和重放设备进行了对比，具体情况如下表所示。

表 6.1 三种数据集的对比

数据集	CASIA-MFSD	Replay-Attack	SiW
人物数量	50	50	165
数据形式	视频	视频	视频
真/假数量	150/450	200/1000	1320/3300
人脸姿态	正脸	正脸	[-90°, 90°]
表情变化	无	无	有
光照变化	无	有	有
攻击方式	1 种打印攻击 1 种重放攻击	1 种打印攻击 2 种重放攻击	2 种打印攻击 4 种重放攻击
重放设备	iPad	iPhone 3GS、iPad	iPad Pro、iPhone 7、 Galaxy S8、Asus MB168B

6.1.2 评价标准

在人脸防伪领域，为了评价检测方案的性能，通常使用以下几个评价指标：准确率、等错误率、半错误率和平均分类错误率。为了清楚地解释这几个评价指标，首先引入真正例（True Positive, 简称 TP）、假正例（False Positive，简称 FP）、真负例（True Negative，简称 TN）和假负例（False Negative，简称 FN）这四个概念。如下表所示：

表 6.2 TP、FP、TN 和 FN

		预测结果	
		正例	负例
真实情况	正例	TP	FN
	负例	FP	TN

在人脸真伪检测的场景下，正例表示的是真脸，负例代表假脸。通常会设定一个阈值 θ，当预测结果大于 θ 判定为正（Positive，即真脸），当预测结果小于 θ 判定为负（Negative，即假脸）。

准确率（Accuracy，简记为 ACC）可定义为：ACC＝（TP＋TN）/（TP＋FN＋FP＋TN），在人脸真伪分类的情况下，表示的是“将真脸预测为真脸并且将假脸预测为假脸占所有样本”的概率。

错误接受率（False Acceptance Rate，简称 FAR）可定义为：FAR＝FP/（FP＋TP），其含义是“预测为真脸实际为假脸占所有预测为真脸的样本”的概率。

错误拒绝率（False Rejection Rate，简称 FRR）定义为：FRR＝FN/（FN＋TN），也就是说“预测为假脸实际为真脸占所有预测为假脸的样本”的概率。

一般来说，FAR 和 FRR 是同一个算法系统里的两个参数，彼此相互矛盾，此消彼长。如图 5-4，FAR 随阈值增大而减小，FRR 随阈值增大而增大。

等错误率（Equal Error Rate，简称 EER）是 FAR 和 FRR 相等时的错误率，可以把 FAR 和 FRR 整合成一个参数，用来评价算法的整体性能。下图中两条曲线交点所对应的错误率即是 EER。

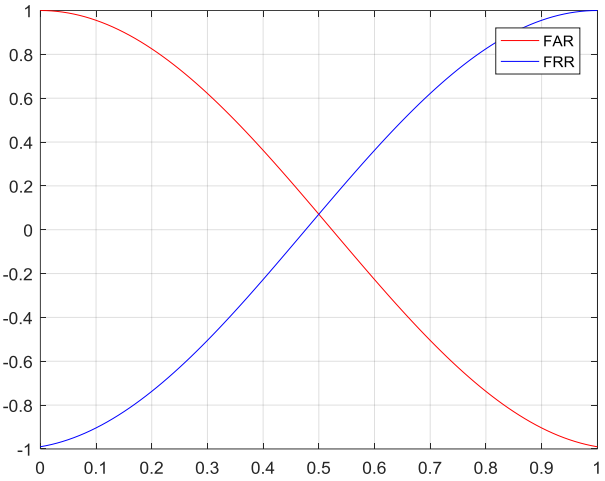


图 6-4 FAR 和 FRR 曲线

半错误率（Half Total Error Rate，简称 HTER）也是评价算法整体性能的一个参数，其数学定义为：HTER = （FAR + FRR）/2，相较于 EER，由于其计算的简洁性，在某些情况下更为常用。

攻击表示分类错误率（Attack Presentation Classification Error Rate，简称 APCER）是假正例率在人脸防伪领域的另一种专用称法，表示“预测为真脸实际为假脸占所有实际为假脸的样本”的概率，定义为：APCER = FPR = FP/（FP + TN）。

真实表示分类错误率（Bona Fide Presentation Classification Error Rate，简称 BPCER）表示“预测为假脸实际为真脸占所有实际为真脸的样本”的概率，定义为：BPCER = 1 - TPR = FN/（TP + FN）。

由上，可以得到平均分类错误率（Average Classification Error Rate，简称 ACER）的定义：ACER = （APCER + BPCER）/2。它与 EER 和 HTER 类似，用来衡量分类器的整体性能，通常与前两者结合使用，更充分地评价防伪方案的优劣性。

6.1.3 软硬件配置

表 6.3 硬件配置信息

硬件	配置
架构	X64
显卡配置	NVIDIA GeForce GTX 1070Ti
显存大小	8GB
CPU	Intel(R) Core(TM)i5-6600K CPU@3.50GHz, 4 个内核，8 个逻辑处理器
内存	32GB

整个网络是在 NVIDIA GeForce GTX 1070Ti GPU 上通过 Pytorch 深度学习框架进行训练的，表 6.3 和表 6.4 列出了实验所用的硬件和软件信息。

表 6.4 软件版本信息

操作系统	Ubuntu 16.04
深度学习框架	Pytorch 0.4.1
Python	3.6.4
numpy	1.14.3

6.2 基于静态特征检测方案的实验与结果分析

基于静态特征的检测方案实际上是在前文所述的静态特征提取方法的基础上增加了一个分类模块，分类模块同样也采用全连接层，也就是基于提取到的人脸静态特征通过分类模块对人脸真伪进行分类。实验总体上分为三个步骤，首先是数据预处理，用于生成 3D 点云图和深度图真实标记；然后是数据集内部测试，将静态特征的检测方案分别与典型的人脸防伪算法进行对比，比较训练和测试采用相同数据集时算法性能的表现差异；最后是数据集交叉测试，同样是将静态特征的检测方案分别与典型的人脸防伪算法进行对比，比较训练和测试采用不同数据集时算法性能的表现差异。

6.2.1 数据预处理

（1）3D 点云图的生成

对于上述的 3 个不同的数据集，需要通过预处理，得到真假人脸深度图的真实标记，用于深度图估计网络（DMENet）的训练。预处理过程分为两步，首先对原数据集的真脸图像用 PRNet 生成相应的 3D 点云图，再分别对真假脸的深度归一化，得到各自的深度图真实标记。由于特殊的网络结构和目标函数的设计，PRNet 能够很好地应对人脸姿态角度变化、光照改变和人脸遮挡不可见等异常情况。本节基于 PRNet 网络，加载预训练的模型参数，对输入的 2D 图像生成了相应的 3D 点云图。

（2）深度图真实标记的生成

为了得到能够很好地表征人脸信息进而区分人脸真伪的深度图，首先需要人脸深度图的真实标记，然后以此作为监督信息训练神经网络。根据真假脸深度信息的不同，分别采用两种方法生成相应的深度图真实标记。生成真脸的真实标记有两个步骤，先是通过 PRNet 得到输入 2D 图像中人脸的 3D 点云图，再是对 3D 点云图进行深度归一化，得到真脸深度图的真实标记。对于假脸的真实标记，为了方便计算采用简洁的设定，直接把输入 2D 图像各像素点的深度置为 0，得到假脸深度图的真实标记。

数据预处理后，真脸原图的变化如下图所示，从左往右依次为真脸的 2D 输入图像、PRNet 预测

的 3D 点云图和真脸的深度真实标记。

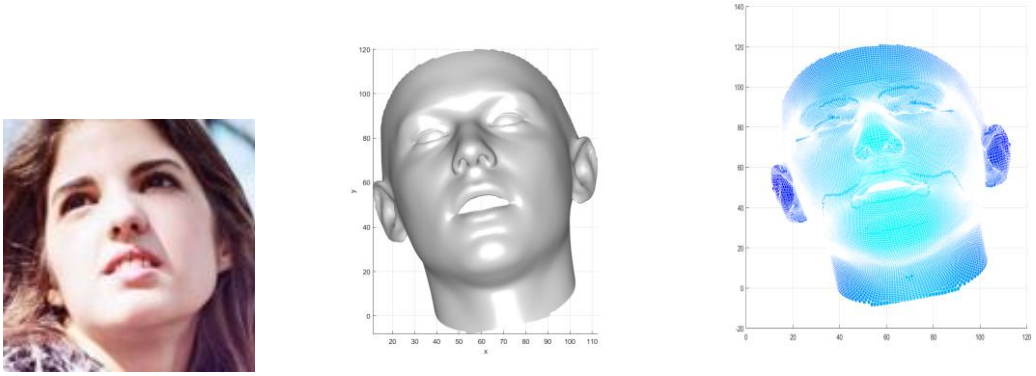


图 6-5 真脸的原图、3D 点云图和深度图的真实标记

6.2.2 实验结果与分析

（1）训练参数设置

对原始数据集 CASIA-MFSD、Replay-Attack 和 SiW 进行预处理后，得到的新数据集由原始图像与其对应的深度图真实标记所组成。以人脸深度作为监督信息训练 DMENet 得到真假人脸深度图的估计网络，网络具体的超参数设置如下表所示。网络的初始学习率设为 0.003，每过 10 轮降为十分之一，训练轮数（epoch）设置为 40，批大小（batch size）为 16，网络参数采用随机初始化，使得其满足均值为 0 标准差为 0.02 的正态分布。

表 6.5 网络训练超参数设置

学习率	轮数		
	0 ~ 10	10 ~ 20	20 ~ 30
	0.003	0.0003	0.00003
批大小	16		
初始化方法	随机初始化		

（2）数据集内部测试

本文使用较为复杂的 SiW 数据集进行内部测试，由于 SiW 数据集中人脸的光照、表情、姿态以及视频播放设备都各有不同，分别侧重不同的关注点。因此，本文根据文献[56]中制定的 3 种规则进行测试。具体规则如下：

a) 规则 1：偏重人脸在姿态和表情上的变化，用数据集中 90 个人员视频的前 60 帧训练，而测试时使用 75 个人员视频的全部帧；

b) 规则 2：偏重重放攻击的播放设备的变化，训练时用的是 3 种移动终端采集的视频，测试时用 1 种移动终端采集的视频；

c) 规则 3：侧重攻击种类的变化，用打印（重放）攻击的人员视频训练，用重放（打印）攻击

的人员视频测试。

规则的细节设置如下表所示：

表 6.6 SiW 测试规则

规则	项目	人员数量	攻击模式
1	训练	90	前 60 帧
	测试	75	全部帧
2	训练	90	3 种移动终端
	测试	75	1 种移动终端
3	训练	90	打印（重放）
	测试	75	重放（打印）

本文将 LBP 算法[7]与 FAS-BAS 算法[56]作为基于静态特征的算法的对比对象，其中 LBP 算法是前文提到的一种基于人工设计特征的算法，而 FAS-BAS 算法采用的则是 CNN 与 RNN 组合架构的算法，下表列出了三种算法在 SiW 上的攻击表示分类错误率（APCER）、真实表示分类错误率（BPCER）和平均分类错误率（ACER）三个方面的对比结果：

表 6.7 各算法模型在 SiW 上的 APCER (%)、BPCER(%)和 ACER(%)

规则	**算法模型**	**APCER (%)**	**BPCER(%)**	**ACER(%)**
1	LBP	10.69	10.69	10.69
	FAS-BAS	3.58	3.58	3.58
	静态特征算法	1.26	1.08	1.17
2	LBP	2.23±0.72	2.23±0.72	2.23±0.72
	FAS-BAS	0.57±0.69	0.57±0.69	0.57±0.69
	静态特征算法	0.16±0.27	0.28±0.27	0.22±0.27
3	LBP	19.72±5.66	19.72±5.66	19.72±5.66
	FAS-BAS	8.31±3.81	8.31±3.80	8.31±3.81
	静态特征算法	3.26±0.68	3.25±0.68	3.26±0.68

从上表可以看出：

（a）在三种规则下 LBP 算法的性能表现最差，这是因为它是以图像的局部纹理信息这一人工设计特征作为判别依据，存在通用性较差的缺点；

（b）FAS-BAS 与 LBP 算法相比表现稍好，因为它基于深度学习方法能够提取表现性更强的特征，但是它里面的 LSTM 架构容易丢图像的失空间信息，存在稳定性不强的缺点；

（c）与 LBP 算法相比，静态特征算法在三种规则下都取得了更好的分类效果，这是因为它采用深度学习的方式，能够通过深度网络自动学习人脸特征，改善了人工设计特征通用性差的问题；

（d）与 FAS-BAS 算法相比，静态特征算法同样在三种规则下都取得了更好的效果，这是因为它一方面采用 PRNet 重建人脸 3D 点云图，能够适应不同姿态、表情和光照条件的变化；另一方面采用人脸深度图作为特征，引入了明确的监督信息，增强了模型的可解释性，同时保留了图像的空间信

息，提高了检测的准确度。

（3）数据集交叉测试

在数据及交叉测试中，分别以 CASIA-MFSD、Replay-Attack 为训练集和测试集用半错误率 HTER（%）为评估标准，将静态特征算法与 LBP 和 FAS-BAS 算法进行比较，具体交叉测试结果如下表：

表 6.8 各算法模型交叉测试结果

算法模型	训练	测试	训练	测试
	CASIA-MFSD	Replay-Attack	Replay-Attack	CASIA-MFSD
LBP	47.0		39.6	
FAS-BAS	27.6		28.4	
静态特征算法	19.6		25.2	

由上表可以看出：

（a）在两种交叉测试中，LBP 算法的表现最差，FAS-BAS 表现居中，静态特征算法表现最好，这同样是因为静态特征算法以人脸深度图作为判别依据，改善了人工设计特征通用性差、LSTM 丢失空间信息的缺点;

（b）在两种交叉测试中，静态特征算法的 HTER 都较小，这说明静态特征算法对于不同数据集的泛化性能较强。

6.3 基于动态特征检测方案的实验和结果分析

基于动态特征的检测方案与基于静态特征的检测方案类似，同样是在动态特征提取方法的基础上增加了一个分类模块，分类模块同样采用全连接层。实验总体上也分为三个步骤，首先是数据预处理，对原数据集的人脸视频进行采样；然后与静态特征检测方案实验类似，依次进行数据集内部测试和数据集交叉测试，比较动态特征算法与典型人脸防伪算法的性能差异。

6.3.1 数据预处理

由前文可知，动态特征反映了人脸在时间域上的变化信息，因此输入网络的视频帧需要在时间上连续，从帧之间的前后变化中获取人脸的有效信息。而连续的前后两帧人脸的变化往往非常有限，所以为了提取出有效的动态特征，需要对原始视频进行采样。采样的帧间隔不能太小，否则帧变化过小就失去了采样的意义；但是帧间隔也不能太大，否则一方面容易导致人脸变化信息的不连贯，另一方面可能会丢失关键的动态信息。因此，为了让输入的帧能够有效地反映人脸信息在时间域上的变化，本方案规定每隔 M 帧进行采样。数据预处理的流程如下图所示：

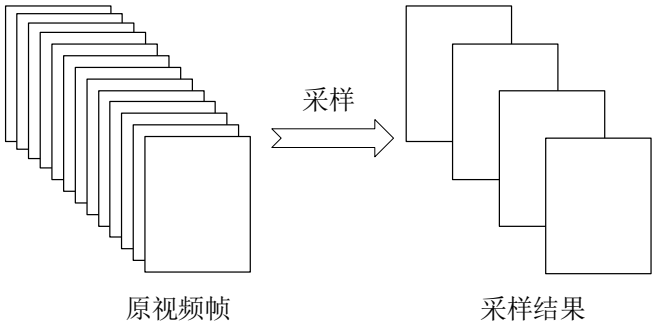


图 6-6 视频数据预处理

每次输入网络模型进行训练的帧的数量为 N。N 的取值与实验所用显卡内存有关，内存越大，N 可取的值就越大。经过反复的实验，发现 M=3，N=4 时，整体效果较好。

6.3.2 实验结果与分析

（1）训练参数设置

网络的学习率（learning rate, lr）固定为 0.01，批大小为 4，训练轮数设置为 10，网络参数采用随机初始化，使得其满足均值为 0 标准差为 0.02 的正态分布。网络的超参数设置下表所示。

表 6.9 网络训练超参数设置

学习率（lr）	0.01
批大小（batch size）	4
训练轮数（epoch）	10
初始化方法	随机初始化

（2）数据集内部测试

与静态特征算法实验类似，动态特征算法的数据集内部测试同样使用 SiW 数据，并采用三种规则进行测试，并且依然与 LBP 算法和 FAS-BAS 算法进行对比。下表是三种算法在 SiW 数据集上的攻击表示分类错误率（APCER）、真实表示分类错误率（BPCER）和平均分类错误率（ACER）三个评价标准的测试结果：

表 6.10 各算法模型在 SiW 上的 APCER (%)、BPCER(%)和 ACER(%)

规则	算法模型	APCER (%)	BPCER(%)	ACER(%)
1	LBP	10.69	10.69	10.69
	FAS-BAS	3.58	3.58	3.58
	动态特征算法	2.13	1.27	1.70
2	LBP	2.23±0.72	2.23±0.72	2.23±0.72
	FAS-BAS	0.57±0.69	0.57±0.69	0.57±0.69
	动态特征算法	0.13±0.25	0.25±0.25	0.19±0.25
3	LBP	19.72±5.66	19.72±5.66	19.72±5.66
	FAS-BAS	8.31±3.81	8.31±3.80	8.31±3.81
	动态特征算法	4.53±0.82	4.52±0.82	4.53±0.82

在动态特征实验中，仍然采用了静态特征实验中的 LBP 算法和 FAS-BAS 算法作为比较对象。从对比结果表格中可以看出，LBP 算法的性能仍然最差，FAS-BAS 次之，动态特征算法性能要优于这两种算法，这是因为它提取视频中人脸的动作模式作为特征，避免了 LBP 通用性差和 FAS-BAS 缺少空间信息的缺点。

（3）数据集交叉测试

在数据集交叉测试中，同样用半错误率 HTER（%）为评估标准，分别以 CASIA-MFSD、Replay-Attack 交替作为训练集和测试集，将 LBP 和 FAS-BAS 算法与动态特征算法进行比较，具体测试结果如下表所示：

表 6.11 各算法模型交叉测试结果

算法模型	训练	测试	训练	测试
	CASIA-MFSD	Replay-Attack	Replay-Attack	CASIA-MFSD
LBP	55.9		57.6	
FAS-BAS	27.6		28.4	
动态特征算法	18.3		24.7	

由上表中同样可以看到，动态特征算法要优于 LBP 算法和 FAS-BAS 算法，采用光流引导特征表示连续帧中人脸的短期动态变化，采用 CGRU 模块对短期动态变化进行积累得到长期动态变化，使得在获得连续帧时序关系的同时，又保留了人脸的空间信息；同时采用注意力机制根据重要程度对不同时间点的视频帧加权组合得到人脸的动态特征，对不同数据集具有较强的泛化性能。

6.4 基于融合特征检测方案的实现和结果分析

（1）训练参数设置

在进行融合特征实验时，保持静态特征和动态特征算法的内部网络参数不变，而调整融合系数 λ 的值，经过反复的试验，发现 λ 取值为 0.2 时，融合特征算法的各项错误率最低，性能最好。

（2）数据集内部测试

与静态特征算法实验类似，动态特征算法的数据集内部测试同样使用 SiW 数据，并采用三种规则进行测试，并且依然将 LBP 算法与 FAS-BAS 算法和基于动态特征的算法进行对比。下表是三种算法在 SiW 数据集上的攻击表示分类错误率（APCER）、真实表示分类错误率（BPCER）和平均分类错误率（ACER）三个评价标准的测试结果如表 6.12 所示。

由表可知，在三种规则下融合特征始终比静态和动态特征具有更好的效果，这是因为融合特征综合了两者的优势，既以深度图作为监督信息指导网络模型训练，又有 OFF-ResB 和 CGRU 模块在保留空间信息的前提下获取视频帧之间的长短期依赖关系，因此具有更好的通用性和更高的稳定性。

表 6.12 各算法模型在 SiW 上的 APCER (%)、BPCER(%)和 ACER(%)

规则	算法模型	APCER (%)	BPCER(%)	ACER(%)
1	静态特征算法	1.26	1.08	1.17
	动态特征算法	2.13	1.27	1.70
	融合特征算法	0.92	0.84	0.88
2	静态特征算法	0.16±0.27	0.28±0.27	0.22±0.27
	动态特征算法	0.13±0.25	0.25±0.25	0.19±0.25
	融合特征算法	0.12±0.23	0.24±0.23	0.18±0.23
3	静态特征算法	3.26±0.68	3.25±0.68	3.26±0.68
	动态特征算法	4.53±0.82	4.52±0.82	4.53±0.82
	融合特征算法	3.12±0.66	3.12±0.66	3.12±0.66

（3）数据集交叉测试

在数据集交叉测试中，依然用半错误率 HTER（%）为评估标准，分别以 CASIA-MFSD、Replay-Attack 交替作为训练集和测试集，将 LBP 和 FAS-BAS 算法与动态特征算法进行比较，具体测试结果如下表所示：

表 6.13 各算法模型交叉测试结果

算法模型	训练	测试	训练	测试
	CASIA-MFSD	Replay-Attack	Replay-Attack	CASIA-MFSD
静态特征算法	19.6		25.2	
动态特征算法	18.3		24.7	
融合特征算法	17.0		23.8	

由上表可知：

（a）在两种交叉测试中，动态特征算法相较于静态特征算法其 HTER 更低，这说明动态特征算法相比较静态特征算法对不同数据集的泛化性更强；

（b）在两种交叉测试下，融合特征算法相较于静态特征和动态特征算法其 HTER 都更低，这说明融合特征算法能够更加广泛地适用于不同的数据集，反映不同数据集视频中人脸的特征，具有更好的泛化性能。

6.5 本章小结

本章首先对实验所用的数据集、评价标准和软硬件配置进行了说明，然后分别对基于静态特征、动态特征和融合特征的检测方案做了实现，并且对其实验结果进行了分析和讨论。静态特征算法和动态特征算法在数据集内部测试时，它们的 ACER 相比于 LBP 算法和 FAS-BAS 算法都更低；并且在数据集交叉测试时，也有类似的结果。同时本文将两种方案进行融合并对其进行实验分析，实验结果表明融合特征算法比单独的静态特征算法或动态特征算法具有更好的性能。由此可以看出，本

文设计的基于静态特征和动态特征的算法可以有效地分辨人脸的真伪，而融合特征综合了两者的优势，能够更好地表达区分人脸真伪的信息，因此具有更广泛的应用前景。

第7章 总结与展望

7.1 工作总结

本研究课题旨在设计出移动智能终端背景下的虚假人脸检测方案，主要完成了以下几项工作：

（1）对人脸特征提取算法进行了研究，调研了基于人工设计特征和基于深度学习的两大类人脸特征提取算法的优缺点，针对移动智能终端场景的特点，选择了基于深度学习的方式。

（2）对基于静态特征的人脸防伪技术进行了研究，将人脸深度图作为检测的依据，利用 PRNet 生成的 3D 点云图作为真实标记，指导深度网络模型进行训练，得到深度图作为静态特征，用于人脸真伪的分类。同时利用三个不同的数据库对该设计方案进行了实验，分类结果表明该检测方案与 LBP 算法和 FAS-BAS 算法相比通用性和稳定性更好，能够有效地区分人脸的真伪。

（3）对基于动态特征的人脸防伪技术进行了研究，创新地用光流引导特征来表示连续帧的动态变化，采用 CGRU 模块提取视频流中的动态信息，并且采用注意力机制根据重要程度对不同时间点的视频帧加权组合，得到包含人脸动态信息的特征图。同时在三个数据库上进行了验证，根据结果分析可得该检测方案同样能够改善 LBP 算法和 FAS-BAS 算法的缺点，具有更好的人脸真伪判别性能。

（4）对人脸的静态特征和动态特征进行了融合，将融合特征作为人脸真伪辨别的依据，并且在三个数据库上进行了验证，结果表明融合特征相较于单一的静态特征或动态特征错误率更低，能够更好地用于区分人脸真伪。

综上所述，本文完成了基于融合特征的人脸防伪方案的设计，为移动智能终端的人脸登录的安全问题提供了一种有效的解决方案。但是受时间和精力所限，尚未完成该检测方案在真机上的实现和部署，需要今后进一步的工作。

7.2 工作展望

受到时间和精力两方面的限制，本文的研究工作仍然存在一些需要完善的地方，这也是后续工作的改进方向，具体可以总结如下：

（1）在融合特征的实验中，将某段连续帧的第一帧深度图作为静态特征，没有验证采用中间某帧或者最后一帧是否对结果有更好的帮助，或者是否应该选择更具有代表性的中间帧进行验证以达到更好的效果，这些都需要进一步的实验去验证。

（2）本文对三个方案的实验都是在 Pytorch 深度学习框架上进行的，由于时间和精力所限，没有将算法模型部署到手机、平板电脑等真实场景中，尚未实现真实可用的人脸真伪检测系统，这将是本文后续工作的重要一环。

（3）本文采用的数据集属于人脸防伪领域常用的几个数据集，但现有的这些数据集仍然不能充分地模拟人脸欺骗攻击在移动智能终端场景下的真实情况，这导致算法模型的实用性和可迁移性较差。在以后的研究中，还需要进一步通过对真实场景的模拟、对人脸数据的采集、对数据的分类和标注等环节规范化，从而构建更多的数据库，更好地发展和提升人脸防伪技术。

致 谢

时间过得很快，三年研究生的学习生活即将结束。回首在九龙湖畔的三年，脑海里涌现出的一个个难忘的场景，在此要感谢一路走来帮助过我的小伙伴和大伙伴们。

首先要感谢的是我的导师宋宇波，宋老师为人幽默风趣，平易近人，喜欢和我们分享一些生活中的趣事，或是火热的新闻段子，或是奇妙的最新科技，与老师交谈总能在谈笑风生中学到知识。我十分荣幸能够成为宋老师的学生，感谢宋老师三年来对我科研和学业的关心和支持，特别是在毕业论文写作阶段不厌其烦地指导，帮助我理清写作的思路，完成论文的撰写工作。

同时感谢学办孙威书记和郭玉珍老师，在我担任学办助管的时候，你们对我工作的帮助和生活上的关怀，让我感受到了很多的温暖。感谢教学线和信安学科的各位老师，在我求学的路上不吝指导，解答了我课业学习上的很多问题，为科研打下了基础。

然后要感谢师兄董启宏、张克落和师姐杨慧文，在我科研上遇到困难时，你们总是非常热心地与我一起分析问题，提供解决问题的思路，正是因为你们的帮助，我才能十分顺利地解决科研道路上遇到的种种困难，学习到了正确高效的科研方法。还要感谢我的同门武威和黄强，和你们一起做项目，一起解决问题，让我得到了成长。感谢师弟宋睿、李轩、石伟、杨俊杰、赵灵奇、张仕奇、樊明和师妹祁欣妤，你们的欢声笑语让实验室充满了快乐科研的氛围。感谢工程一班的同学们，你们的陪伴让我的科研生活增添了很多美好的记忆。

特别感谢我的父母，感谢你们二十五年来对我的养育与关怀，你们的爱是我面对生活困境继续前行的动力。在以后的工作和学习中，我也会继续承载着你们的期望，努力生活，积极向上。

最后感谢本论文的评阅老师们，祝你们一切顺利。

参考文献

[1] Alotaibi A, Mahmood A. Deep face liveness detection based on nonlinear diffusion using convolution neural network[J]. Signal, Image and Video Processing, 2017, 11(4): 713-720.

[2] Boulkenafet Z, Komulainen J, Li L, et al. OULU-NPU: A mobile face presentation attack database with real-world variations[C]//2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017). IEEE, 2017: 612-618.

[3] Chingovska I, Erdogmus N, Anjos A, et al. Face recognition systems under spoofing attacks[M]//Face Recognition Across the Imaging Spectrum. Springer, Cham, 2016: 165-194.

[4] Jourabloo A, Liu Y, Liu X. Face de-spoofing: Anti-spoofing via noise modeling[C]//Proceedings of the European Conference on Computer Vision (ECCV). 2018: 290-306.

[5] Bhattacharjee S, Mohammadi A, Marcel S. Spoofing deep face recognition with custom silicone masks[C]//2018 IEEE 9th International Conference on Biometrics Theory, Applications and Systems (BTAS). IEEE, 2018: 1-7.

[6] Li L, Feng X, Boulkenafet Z, et al. An original face anti-spoofing approach using partial convolutional neural network[C]//2016 Sixth International Conference on Image Processing Theory, Tools and Applications (IPTA). IEEE, 2016: 1-6.

[7] Boulkenafet Z, Komulainen J, Hadid A. Face spoofing detection using colour texture analysis[J]. IEEE Transactions on Information Forensics and Security, 2016, 11(8): 1818-1830.

[8] de Freitas Pereira T, Anjos A, De Martino J M, et al. Can face anti-spoofing countermeasures work in a real world scenario?[C]//2013 international conference on biometrics (ICB). IEEE, 2013: 1-8.

[9] Määttä J, Hadid A, Pietikäinen M. Face spoofing detection from single images using texture and local shape analysis[J]. IET biometrics, 2012, 1(1): 3-10.

[10] Yang J, Lei Z, Liao S, et al. Face liveness detection with component dependent descriptor[C]//2013 International Conference on Biometrics (ICB). IEEE, 2013: 1-6.

[11] Patel K, Han H, Jain A K. Secure face unlock: Spoof detection on smartphones[J]. IEEE transactions on information forensics and security, 2016, 11(10): 2268-2283.

[12] de Freitas Pereira T, Anjos A, De Martino J M, et al. LBP− TOP based countermeasure against face spoofing attacks[C]//Asian Conference on Computer Vision. Springer, Berlin, Heidelberg, 2012: 121-

132.

[13] Patel K, Han H, Jain A K. Secure face unlock: Spoof detection on smartphones[J]. IEEE transactions on information forensics and security, 2016, 11(10): 2268-2283.

[14] Pan G, Sun L, Wu Z, et al. Eyeblink-based anti-spoofing in face recognition from a generic webcamera[C]//2007 IEEE 11th International Conference on Computer Vision. IEEE, 2007: 1-8.

[15] Sun L, Pan G, Wu Z, et al. Blinking-based live face detection using conditional random fields[C]//International Conference on Biometrics. Springer, Berlin, Heidelberg, 2007: 252-260.

[16] Kollreider K, Fronthaler H, Faraj M I, et al. Real-time face detection and motion analysis with application in "liveness" assessment[J]. IEEE Transactions on Information Forensics and Security, 2007, 2(3): 548-558.

[17] Lucena O, Junior A, Moia V, et al. Transfer learning using convolutional neural networks for face anti-spoofing[C]//International Conference Image Analysis and Recognition. Springer, Cham, 2017: 27-34.

[18] Gan J, Li S, Zhai Y, et al. 3d convolutional neural network based on face anti-spoofing[C]//2017 2nd international conference on multimedia and image processing (ICMIP). IEEE, 2017: 1-5.

[19] Shao R, Lan X, Yuen P C. Deep convolutional dynamic texture learning with adaptive channel-discriminability for 3D mask face anti-spoofing[C]//2017 IEEE International Joint Conference on Biometrics (IJCB). IEEE, 2017: 748-755.

[20] Xu Z, Li S, Deng W. Learning temporal features using LSTM-CNN architecture for face anti-spoofing[C]//2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR). IEEE, 2015: 141-145.

[21] Wang Y, Nian F, Li T, et al. Robust face anti-spoofing with depth information[J]. Journal of Visual Communication and Image Representation, 2017, 49: 332-337.

[22] Boulkenafet Z, Komulainen J, Hadid A. Face antispoofing using speeded-up robust features and fisher vector encoding[J]. IEEE Signal Processing Letters, 2016, 24(2): 141-145.

[23] Nagpal C, Dubey S R. A performance evaluation of convolutional neural networks for face anti spoofing[J]. arXiv preprint arXiv:1805.04176, 2018.

[24] Yang J, Lei Z, Li S Z. Learn convolutional neural network for face anti-spoofing[J]. arXiv preprint arXiv:1408.5601, 2014.

[25] Wang Y, Sun Y, Liu Z, et al. Dynamic graph cnn for learning on point clouds[J]. arXiv preprint arXiv:1801.07829, 2018.

[26] Ronneberger O, Fischer P, Brox T U. Convolutional networks for biomedical image

segmentation[C]//Paper presented at: International Conference on Medical Image Computing and Computer-Assisted Intervention2015.

[27] Zhang X, Zhou X, Lin M, et al. Shufflenet: An extremely efficient convolutional neural network for mobile devices[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 6848-6856.

[28] Szegedy C, Ioffe S, Vanhoucke V, et al. Inception-v4, inception-resnet and the impact of residual connections on learning[C]//Thirty-First AAAI Conference on Artificial Intelligence. 2017.

[29] Shen Z, Liu Z, Li J, et al. Dsod: Learning deeply supervised object detectors from scratch[C]//Proceedings of the IEEE International Conference on Computer Vision. 2017: 1919-1927.

[30] Gers F A, Schraudolph N N, Schmidhuber J. Learning precise timing with LSTM recurrent networks[J]. Journal of machine learning research, 2002, 3(Aug): 115-143.

[31] Lin T Y, Dollár P, Girshick R, et al. Feature pyramid networks for object detection[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 2117-2125.

[32] Hu P, Ramanan D. Finding tiny faces. Computer Vision and Pattern Recognition (CVPR)[C]//IEEE Conference on. IEEE. 2017.

[33] He K, Gkioxari G, Dollár P, et al. Mask r-cnn[C]//Proceedings of the IEEE international conference on computer vision. 2017: 2961-2969.

[34] Zhang C L, Luo J H, Wei X S, et al. In defense of fully connected layers in visual representation transfer[C]//Pacific Rim Conference on Multimedia. Springer, Cham, 2017: 807-817.

[35] Atoum Y, Liu Y, Jourabloo A, et al. Face anti-spoofing using patch and depth-based CNNs[C]//2017 IEEE International Joint Conference on Biometrics (IJCB). IEEE, 2017: 319-328.

[36] Feng Y, Wu F, Shao X, et al. Joint 3d face reconstruction and dense alignment with position map regression network[C]//Proceedings of the European Conference on Computer Vision (ECCV). 2018: 534-551.

[37] Fan H, Su H, Guibas L J. A point set generation network for 3d object reconstruction from a single image[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 605-613.

[38] 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.

[39] Dou P, Shah S K, Kakadiaris I A. End-to-end 3D face reconstruction with deep neural

networks[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017: 5908-5917.

[40] Liu Y, Jourabloo A, Ren W, et al. Dense face alignment[C]//Proceedings of the IEEE International Conference on Computer Vision. 2017: 1619-1628.

[41] Jackson A S, Bulat A, Argyriou V, et al. Large pose 3D face reconstruction from a single image via direct volumetric CNN regression[C]//Proceedings of the IEEE International Conference on Computer Vision. 2017: 1031-1039.

[42] Bas A, Huber P, Smith W A P, et al. 3D morphable models as spatial transformer networks[C]//Proceedings of the IEEE International Conference on Computer Vision. 2017: 904-912.

[43] Deng J, Cheng S, Xue N, et al. Uv-gan: Adversarial facial uv map completion for pose-invariant face recognition[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 7093-7102.

[44] Moschoglou S, Ververas E, Panagakis Y, et al. Multi-attribute robust component analysis for facial UV maps[J]. IEEE Journal of Selected Topics in Signal Processing, 2018, 12(6): 1324-1337.

[45] Xue N, Deng J, Cheng S, et al. Side information for face completion: a robust pca approach[J]. IEEE transactions on pattern analysis and machine intelligence, 2019.

[46] da Silva Pinto A, Pedrini H, Schwartz W, et al. Video-based face spoofing detection through visual rhythm analysis[C]//2012 25th SIBGRAPI Conference on Graphics, Patterns and Images. IEEE, 2012: 221-228.

[47] Sun S, Kuang Z, Sheng L, et al. Optical flow guided feature: A fast and robust motion representation for video action recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 1390-1399.

[48] Horn B K P, Schunck B G. Determining optical flow[J]. Artificial intelligence, 1981, 17(1-3): 185-203.

[49] Fu J, Zheng H, Mei T. Look closer to see better: Recurrent attention convolutional neural network for fine-grained image recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 4438-4446.

[50] Chen L C, Yang Y, Wang J, et al. Attention to scale: Scale-aware semantic image segmentation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 3640-3649.

[51] Xu K, Ba J, Kiros R, et al. Show, attend and tell: Neural image caption generation with visual

attention[C]//International conference on machine learning. 2015: 2048-2057.

[52] Lin H, Shi Z, Zou Z. Fully convolutional network with task partitioning for inshore ship detection in optical remote sensing images[J]. IEEE Geoscience and Remote Sensing Letters, 2017, 14(10): 1665-1669.

[53] Hu J, Shen L, Sun G. Squeeze-and-excitation networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 7132-7141.

[54] Zhang Z, Yan J, Liu S, et al. A face antispoofing database with diverse attacks[C]//2012 5th IAPR international conference on Biometrics (ICB). IEEE, 2012: 26-31.

[55] Chingovska I, Anjos A, Marcel S. On the effectiveness of local binary patterns in face anti-spoofing[C]//2012 BIOSIG-proceedings of the international conference of biometrics special interest group (BIOSIG). IEEE, 2012: 1-7.

[56] Liu Y, Jourabloo A, Liu X. Learning deep models for face anti-spoofing: Binary or auxiliary supervision[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 389-398.

硕士阶段发表论文

[1] **魏一鸣**，宋宇波．Android 手机应用数据的提取与分析 [C]．东南大学校庆学术会议，2017.

[2] **魏一鸣**，宋宇波．基于 DenseNet 架构的人脸识别技术研究 [C]．东南大学校庆学术会议，2018.