

# 基于深度学习的自助CT识别平台

谢志鹏<sup>1</sup>, 韩心雨<sup>2</sup>, 张皓然<sup>2</sup>, 刘 荟<sup>2</sup>, 阿拉依·赛力克<sup>2</sup>

<sup>1</sup>上海理工大学机械工程学院, 上海

<sup>2</sup>上海理工大学光电信息与计算机工程学院, 上海

收稿日期: 2023年12月15日; 录用日期: 2024年3月8日; 发布日期: 2024年3月15日

## 摘 要

传统看病流程有相当一部分的时间都耗费在排队取片、找医生诊断上, 这给紧张的医疗资源和紧急的病人情况带来了极大的挑战。快速诊断并进行早期干预成为一种趋势。在此背景下, 本项目基于人工智能设计了一个能够自主识别病毒的CT平台, 使用理论较为成熟的深度残差网络进行CT识别。卷积神经网络拥有强大的自适应性、学习性、全局最优等功能, 在CT识别中表现出较好的性能。用户通过手机扫描二维码上传肺部CT图像至云端服务器, 训练好的CT病毒模型对上传的图片按照三个方面即正常、细菌性肺炎和病毒性肺炎进行预估处理, 并将预估值以图片的形式返回到用户端, 使医生在模型的辅助下能够快速地对就诊人进行诊断, 可进一步减少医生在确诊新冠肺炎疑似病例上的工作量。

## 关键词

人工智能, 深度学习, 深度残差网络, CT图像识别

# Self-Service CT Recognition Platform Based on Deep Learning

Zhipeng Xie<sup>1</sup>, Xinyu Han<sup>2</sup>, Haoran Zhang<sup>2</sup>, Hui Liu<sup>2</sup>, Sailike Aliyi<sup>2</sup>

<sup>1</sup>School of Mechanical Engineering, University of Shanghai for Science and Technology, Shanghai

<sup>2</sup>School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai

Received: Dec. 15<sup>th</sup>, 2023; accepted: Mar. 8<sup>th</sup>, 2024; published: Mar. 15<sup>th</sup>, 2024

## Abstract

A considerable part of the traditional medical treatment process is spent queuing up to get pictures and finding doctors for diagnosis, which brings great challenges to the shortage of medical

resources and urgent patient conditions. Rapid diagnosis and early intervention have become a trend. In this context, this project designs an independent CT virus recognition platform based on artificial intelligence, and uses the deep residual network which is more mature in artificial intelligence to carry out CT recognition. Convolutional neural network has strong self-adaptability, learning and global optimization, and shows good performance in CT recognition. The user uploads lung CT images to the cloud server by scanning two-dimensional code of mobile phone. The trained CT virus model estimates the uploaded images according to three aspects: normal, bacterial pneumonia and viral pneumonia, and returns the estimated values to the user in the form of pictures, so that doctors can quickly diagnose patients with the assistance of the model, which can further reduce the workload of doctors in the diagnosis of suspected cases of COVID-19.

## Keywords

Artificial Intelligence, Deep Learning, Residual Networks, CT Pattern Recognition

Copyright © 2024 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

## 1. 引言

检测作为病原学证据被公认为新冠肺炎诊断的主要参考标准。随着临床诊断数据的积累，冠状病毒肺炎的影像学大数据特征变得逐渐清晰，CT 影像诊断结果变得更加重要。在国家卫生健康委员会公布的诊疗方案第五版中表明，临床诊断无需依赖核酸检测结果，CT 影像临床诊断结果可作为新冠肺炎病例判断的标准。而新冠肺炎患者的 CT 胸片的影像特征表现为单肺或双肺多发、斑片状或节段性磨玻璃密度影等细微变化[1]。一位新冠肺炎病人的 CT 影像大概在 300 张左右，这给医生临床诊断带来巨大压力，医生对一个病例的 CT 影像肉眼分析耗时大约为 5~15 分钟。因此有必要进行 CT 图像识别的研究，自动识别 CT 图像，提示肺部可疑病灶，能够帮助医生快速筛查疑似患肺炎的病人，从而提高诊断质量和诊断效率。

目前，胸部 X 线 X 光是确诊肺炎的最主要手段，对患者的治疗和疾病的调查具有十分重要的意义。X 射线检查是一种极具挑战的工作，对医师的医学图像处理水平提出了更高的要求。计算机辅助诊断(Computer-Aided Diagnosis, CAD)是近几年发展起来的一种新的医学手段。在此基础上，结合计算机辅助诊断，对提高医师鉴别准确率，减少漏诊和误诊，提高工作效率具有重要意义。

随着图像处理，模式识别，科学计算可视化等学科的发展，国际上对肺炎图像进行了大量的研究。上海交通大学李阳等利用图像处理技术将肺炎图像的局部二值模式(Local Binary Pattern, LBP) [2]特征信息作为识别特征，然后用支持向量机(Support Vector Machine, SVM)算法进行肺炎图像识别。虽然 SVM 算法对于处理高维度、非线性问题有较大优势，但是当训练样本较大时，计算量会呈指数增长，导致模型最终的效率偏低[3]。近年来，以卷积神经网络(Convolutional Neural Network, CNN) [4]为代表的深度学习(Deep Learning)算法在计算机视觉领域的成功应用，为 CNN 在医疗图像处理领域的应用奠定了基础。刘长征等[5]利用卷积神经网络实现了肺炎影像的判别。实验结果表明，基于卷积神经网络的特征提取与手工提取的特征相比，其特征表达能力更强。然而，目前大多数人使用的卷积神经网络结构较为单一，其特征学习能力较弱，因此，其识别精度仍有待提高。目前，在图像识别领域，多采用基于深度卷积神经网络(ImageNet)的学习方法，但深度神经网络存在训练难度大等问题。因此，对已有的网络结构进行优化是非常必要的。

## 2. 原理分析

### 2.1. 人工智能与卷积神经网络

#### 2.1.1. 卷积神经网络介绍

在机器学习领域，有一个研究方向就是利用神经网络学习复杂的抽象逻辑，即神经网络。神经网络是一种以神经网络为基础，对数据进行学习的一种算法。由于受到运算容量及数据规模的限制，早期神经网络的学习深度通常只有 1 到 4 层，且网络的表示能力受到限制。伴随着计算资源的不断增长以及大数据时代的来临，GPU 的高速并行性以及海量数据的出现，使得大规模神经网络的训练变得更加容易。

2006 年，美国学者 Geoffrey Hinton 等首次提出了一种基于分层预训练的多层神经网络模型，并在 MNIST 手写体图像上获得了比支持向量机更高的准确率，开创了“第三代人工智能”时代。在论文[6]中，Geoffrey Hinton 首次提出了 Deep Learning 的概念，这也是(深层)神经网络被叫作深度学习的由来。2011 年，Xavier Glorot 等人首次提出了一种基于 ReLU 的激励函数，它是目前应用最多的一种激励功能。2012 年，Alex Krizhevsky 提出了 8 层的深层神经网络 AlexNet [7]，它采用了 ReLU 激活函数，并使用 Dropout 技术来防止过拟合，同时抛弃了逐层预训练的方式，直接在两块 NVIDIA GTX580GPU 上训练网络。在 ILSVRC-2012 图像识别竞赛中，AlexNet 名列第一，其错误率为 10.9%，较第二名下降了 10.9%。

自从 AlexNet 模型被提出以来，已经有很多的方法被发布出来，包括 VGG 系列，GoogLeNet 系列，ResNet 系列；DenseNet 等。该系列的 ResNet 模式在保证相同的性能或更好的情况下，将一个网络的层次扩展到几百个，甚至上千个层次。该方法思路简洁、通用性强、收敛速度快，是目前最有代表性的深度学习方法。

#### 2.1.2. 卷积网络模型

图像分类是一个普遍存在的问题。该神经网络以图像资料作为输入，以各分类中各样本的概率分布作为其输出。在实际应用中，一般选择具有较大可能性的分类来进行分类。图像识别是当前最早将深度学习方法运用到图像分类中的一个问题，其典型代表包括 VGG 系列，DenseNet 系列，ResNet 系列等。他们三者的差别如下图 1-3 所示。

**VGG 模型：**VGG 模型是一种简单的网络，它采用相同的卷积核大小(3 × 3)，并采用最大的池化大小(2 × 2)。另外，VGG 算法的运算代价很大，因为它所用的参数太多，对内存的要求也很高。这些参数中的绝大部分都来源于首个完全连通层。

**DenseNet 模型：**又称为稠密卷积网络(DenseNet)。它引入了具有相同特征图大小的任意两个层之间的直接连接。DenseNet 可以自然地扩展到数百层，同时没有优化的困难。实验测试中，随着参数的增加，Densenet 趋向于在准确性上取得一致的提高，而没有任何性能下降或过拟合的迹象。此外，较之于 VGG 模型，Densenet 需要更少的参数和更少的计算，且具有最先进的性能。

**Resnet (深度残差网络)：**AlexNet、VGG、GoogLeNet 等网络模型的出现将神经网络的发展带入了几十层的阶段，研究人员发现网络的层数越深，越有可能获得更好的泛化能力。但是当模型加深以后，网络变得越来越难训练，这主要是由于梯度弥散和梯度爆炸现象造成的。为了解决深层神经网络的梯度弥散和梯度爆炸现象，2015 年，微软亚洲研究院何凯明等人发表了基于 Skip Connection 的深度残差网络(Residual Neural Network, 简称 ResNet)算法[8]，并提出了 18 层、34 层、50 层、101 层、152 层的 ResNet-18、ResNet-34、ResNet-50、ResNet-101 和 ResNet-152 等模型，ResNet 在 ILSVRC 2015 挑战赛 ImageNet 数据集上的分类、检测等任务上面均获得了最好性能，截止目前，ResNet 仍是受欢迎的主流模型。本论文以 ResNet-34 模式为研究对象。

ConNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input(224 x 224 RGB image)					
conv3-64	conv3-64 LRN	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 conv1-256	conv3-256 conv3-256 conv3-256	conv3-256 conv3-256 conv3-256 conv3-256
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 conv1-512	conv3-512 conv3-512 conv3-512	conv3-512 conv3-512 conv3-512 conv3-512
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 conv1-512	conv3-512 conv3-512 conv3-512	conv3-512 conv3-512 conv3-512 conv3-512
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

Figure 1. VGG model structure diagram  
图 1. VGG 模型结构图

Layers	Output size	DenseNet-121	DenseNet-169	DenseNet-201	DenseNet-264
Convolution	112×112	7×7 conv, stride 2			
Polling	56×56	3×3 max pool, stride 2			
Dense Block (1)	56×56	$\begin{bmatrix} 1 \times 1 \text{conv} \\ 3 \times 3 \text{conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{conv} \\ 3 \times 3 \text{conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{conv} \\ 3 \times 3 \text{conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{conv} \\ 3 \times 3 \text{conv} \end{bmatrix} \times 6$
Transition Layer (1)	56×56	1×1 conv			
	28×28	2×2 average pool, stride 2			
Dense Block (2)	28×28	$\begin{bmatrix} 1 \times 1 \text{conv} \\ 3 \times 3 \text{conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{conv} \\ 3 \times 3 \text{conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{conv} \\ 3 \times 3 \text{conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{conv} \\ 3 \times 3 \text{conv} \end{bmatrix} \times 12$
Transition Layer (2)	28×28	1×1 conv			
	14×14	2×2 average pool, stride 2			
Dense Block (3)	14×14	$\begin{bmatrix} 1 \times 1 \text{conv} \\ 3 \times 3 \text{conv} \end{bmatrix} \times 24$	$\begin{bmatrix} 1 \times 1 \text{conv} \\ 3 \times 3 \text{conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{conv} \\ 3 \times 3 \text{conv} \end{bmatrix} \times 48$	$\begin{bmatrix} 1 \times 1 \text{conv} \\ 3 \times 3 \text{conv} \end{bmatrix} \times 64$
Transition Layer (3)	14×14	1×1 conv			
	7×7	2×2 average pool, stride 2			
Dense Block (4)	7×7	$\begin{bmatrix} 1 \times 1 \text{conv} \\ 3 \times 3 \text{conv} \end{bmatrix} \times 16$	$\begin{bmatrix} 1 \times 1 \text{conv} \\ 3 \times 3 \text{conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{conv} \\ 3 \times 3 \text{conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{conv} \\ 3 \times 3 \text{conv} \end{bmatrix} \times 48$
Classification Layer	1×1	7×7 global average pool			
		1000D fully-connected, softmax			

Figure 2. DenseNet model structure diagram  
图 2. DenseNet 模型结构图

layer name	Outupt size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7,64, stride 2				
		3×3 max pool ,stride 2				
conv2_x	56×56	$\begin{bmatrix} 3 \times 3,64 \\ 3 \times 3,64 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3,64 \\ 3 \times 3,64 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1,64 \\ 3 \times 3,64 \\ 1 \times 1,256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1,64 \\ 3 \times 3,64 \\ 1 \times 1,256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1,64 \\ 3 \times 3,64 \\ 1 \times 1,256 \end{bmatrix} \times 3$
conv3_x	28×28	$\begin{bmatrix} 3 \times 3,128 \\ 3 \times 3,128 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3,128 \\ 3 \times 3,128 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1,128 \\ 3 \times 3,128 \\ 1 \times 1,512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1,128 \\ 3 \times 3,128 \\ 1 \times 1,512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1,128 \\ 3 \times 3,128 \\ 1 \times 1,512 \end{bmatrix} \times 8$
conv4_x	14×14	$\begin{bmatrix} 3 \times 3,256 \\ 3 \times 3,256 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3,256 \\ 3 \times 3,256 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1,256 \\ 3 \times 3,256 \\ 1 \times 1,1024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1,256 \\ 3 \times 3,256 \\ 1 \times 1,1024 \end{bmatrix} \times 23$	$\begin{bmatrix} 1 \times 1,256 \\ 3 \times 3,256 \\ 1 \times 1,1024 \end{bmatrix} \times 36$
conv5_x	7×7	$\begin{bmatrix} 3 \times 3,512 \\ 3 \times 3,512 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3,512 \\ 3 \times 3,512 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1,512 \\ 3 \times 3,512 \\ 1 \times 1,2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1,512 \\ 3 \times 3,512 \\ 1 \times 1,2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1,512 \\ 3 \times 3,512 \\ 1 \times 1,2048 \end{bmatrix} \times 3$
	1×1	average pool,1000-d fc,softmax				
FLOPs		1.8×10 <sup>9</sup>	3.6×10 <sup>9</sup>	3.8×10 <sup>9</sup>	7.6×10 <sup>9</sup>	11.3×10 <sup>9</sup>

Figure 3. Structure diagram of the Resnet model

图 3. Resnet 模型结构图

### 2.1.3. ResNet 网络原理

ResNet 短路连接卷积层的输入、输出，并插入 Skip Connection 形成残差网络，实现残差连接，如图 4 所示，输入  $x$  经过两层卷积，获得特征转换后的输出  $F(x)$ ，再将输出  $F(x)$ 与输入  $x$  进行对应的加法运算，获得最后的输出  $H(x)$ ：

$$H(x) = x + F(x) \tag{1}$$

Residual mapping 指的是“差”，也就是输出  $H(x)$ -输入  $x$ ，所以残差指的就是  $F(x)$ 部分。 $F(x) = H(x) - x$  为残差映射，其优化比  $H(x)$ 更简单，所以使用残差网络相当于学习的目标不是最优解  $H(x)$ 和全等映射  $x$ ，而是他们的差。

为了使输入  $x$  与输出  $F(x)$ 能够进行相加运算，需要输入  $x$  的 shape 与输出  $F(x)$ 的 shape 完全一致。当输入、输出的 shape 不一致时，通常情况下在 Skip Connection 上对输入  $x$  进行卷积操作，使其 shape 与输出  $F(x)$ 的 shape 大小一致，如下图 4 中 identity( $x$ )函数所示，其中 identity( $x$ )绝大多数是  $1 \times 1$  的卷积运算，主要在 height 和 width 不变的情况下，调节输入的通道数。

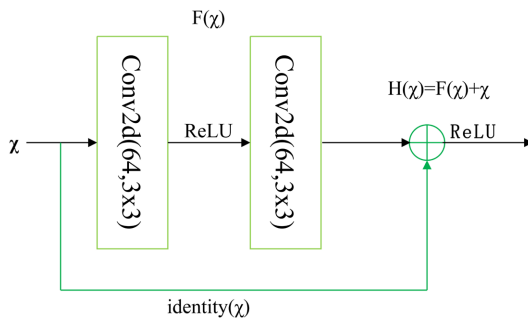
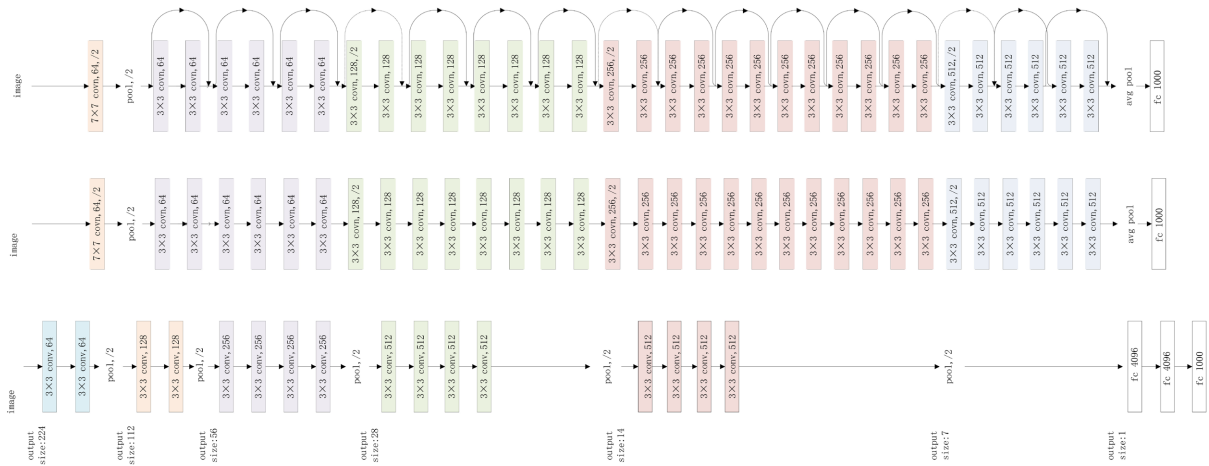


Figure 4. Residuals module

图 4. 残差模块

图 5 对比了 34 层的深度残差网络、34 层的普通深度网络以及 19 层的 VGG 网络结构。可以看到，ResNet 通过短路连接插入形成残差网络，达到了较深的网络层数，解决了深层神经网络的梯度弥散和梯度爆炸现象。



**Figure 5.** Comparison of network structures  
**图 5.** 网络结构比较

## 2.2. 基于深度学习的肺部 CT 识别

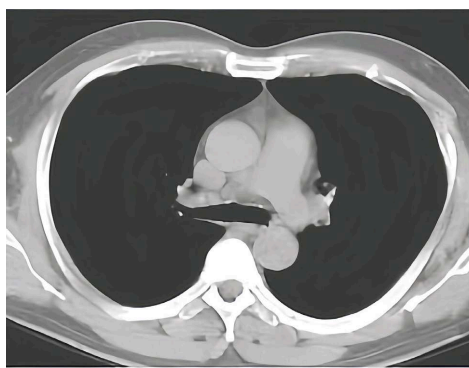
影像辨识功能：影像辨识模式的作用，在于侦测使用者所上传之肺部 CT 影像，并判定其罹患肺炎的机率，协助医师初筛可疑肺炎病人。

### 2.2.1. 数据集来源

本项目采用 Kaggle 开放源码数据库，共 6000 例 CT 图像，共分为 3 类，即正常、病毒性肺炎和细菌性肺炎，每一类大约 2000 例。

### 2.2.2. 数据集处理

本文参照 ResNet 模式，在 Windows 10 x64，版本的 tensorflow2.0 的基础上建立了一个以 ResNet 为基础的图像识别网络。tensorflow 是由谷歌人工智能团队谷歌大脑(Google Brain)开发和维护的面向科学研究和谷歌产品开发的规模深度学习应用框架。首先 CT 扫描结果是 DICOM 格式，神经网络则是 3 通道 JPEG 压缩，并且每一张照片的长度和宽度都是固定的，最后得到的 CT 样本见图 6。



**Figure 6.** CT sample after processing  
**图 6.** 处理后的 CT 样品

### 2.2.3. 模型搭建及训练

本项目以面向深度学习的 python 语言为基础，借助 tensorflow 构建好神经网络框架，并配置 GPU 环境，项目所采用的 GPU 为 NVIDIA RTX3070，模型训练超参数配置如表 1 所示。

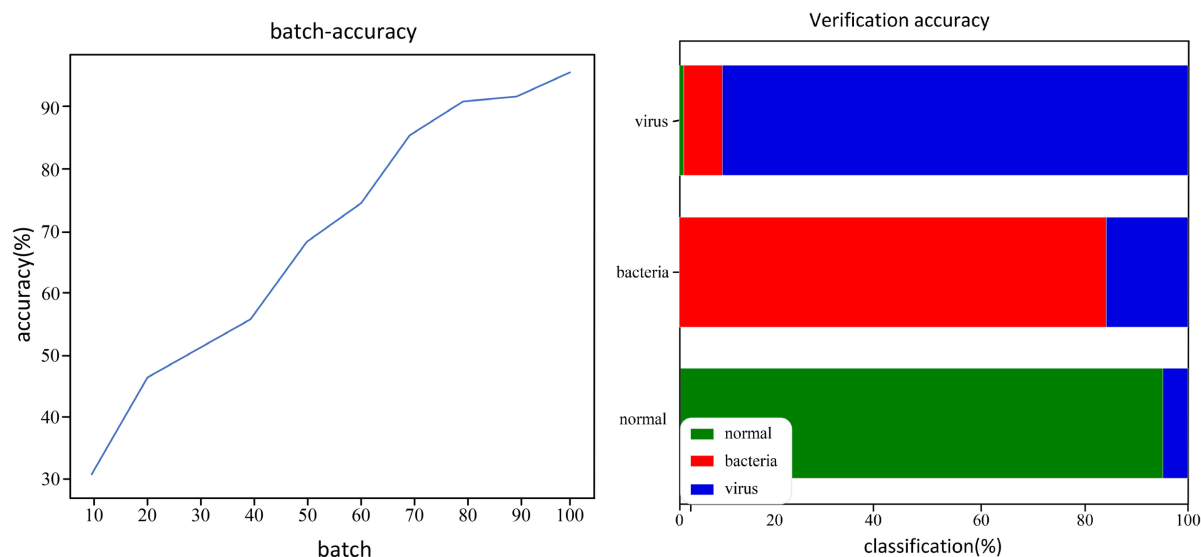
**Table 1.** Model training hyperparameter configuration  
**表 1.** 模型训练超参数配置

参数	设定值
批量大小(batch size)	64
学习率(learning rate)	0.01
迭代次数(epochs)	1000
优化器(optimizer)	Adagrad
损失函数(loss)	交叉熵
dropout	0.3
中止迭代次数	4次不下降

### 3. CT 图像识别及检测预估

#### 3.1. CT 图像识别准确率

本项目选择 kaggle 库中自带的检验集作为 CT 影像试验的检验样本。CT 检查的一个重要指标就是要将患者与正常人区别开来，一旦确诊为肺炎，就需要区分是病毒性肺炎还是细菌性肺炎。实际验证的结果表明训练的模型分辨肺炎和正常的准确率达到 99.98%，满足实际应用需求(图 7)。



**Figure 7.** Image recognition model accuracy

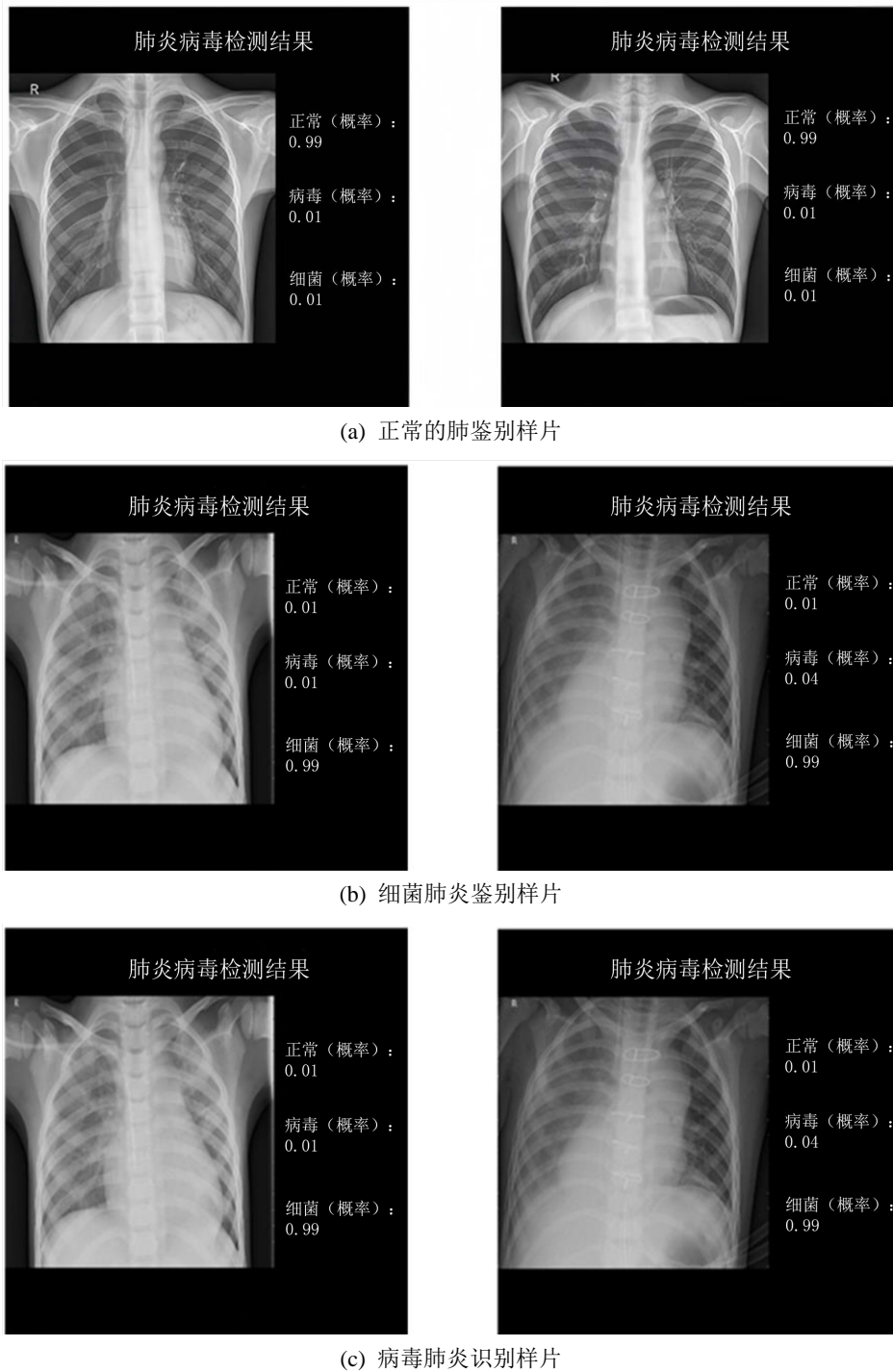
**图 7.** 图像识别模型准确率

#### 3.2. CT 图像检测预估

对 CT 影像进行辨识后，应将其信息反馈给使用者。以 CT 图像为载体，同时包含正常、病毒及细菌三种形态的鉴别预估价，并在使用者界面上进行显示，其详细样本见图 8。

从图 8 可以看出，CT 图像识别样片被划分为三个类别，分别是(a)正常的肺鉴别样片、(b)细菌肺炎鉴别样片、(c)病毒性肺炎鉴别样片，每种都有两张样片供分析。从(a)中可以看出，这两张样本都是正常的肺部 CT 影像，它们的鉴别预估值都是 0.99，而细菌(几率)和病毒(几率)都是 0.01，在有一定误差的情况下，它们的检测结果与现实相符。从(b)中可以看出，这两张样本都是细菌肺炎的 CT 图像，它们的鉴别

预估值都是 0.01，细菌的概率都是 0.99，病毒的概率是 0.01，在有一定误差的情况下，它们的检测结果与现实相符。从(c)中可以看出，这两张样本都是病毒肺炎的 CT 图像，它们的鉴别预估值都是以 0.01 为标准(概率)、0.01、0.38 的细菌(概率)、以及病毒(概率)都是 0.99，在有一定误差的情况下，它们的检测结果与现实是一致的。



**Figure 8.** CT identification detection prediction  
**图 8.** CT 识别检测预估



## 4. 结论

本文采用人工智能领域较为成熟的深度残差网络,利用具有推理、规划和学习能力的深度学习算法,构建一套基于深度学习的CT病毒自动辨识平台。具有很强的适应性,学习性,全局最优等能力,在CT识别中表现出较好的性能。将用户的初始CT图像进行处理,变换成固定长度、宽度的3通道JPEG图像,再用训练好的深度残差网络模型对处理后的CT样片进行图像识别分类,判断样片的实际情况,将患者与正常人区别开来,并且还能区分是病毒性肺炎还是细菌性肺炎。这一整个流程能够在20秒左右就能出具智能分析结果,实际验证的结果表明训练的模型分辨肺炎和正常的准确率达到99.98%,满足实际应用需求。

## 参考文献

- [1] 司延栋. 火眼金睛, 识别病毒性肺炎的CT影像学特征[J]. 健康生活, 2023(10): 10-11.
- [2] Zhao, G., Ahonen, T., Matas, J., *et al.* (2011) Rotation-Invariant Image and Video Description with Local Binary Pattern Features. *IEEE Transactions on Image Processing*, **21**, 1465-1477. <https://doi.org/10.1109/TIP.2011.2175739>
- [3] 吕洪艳, 刘芳. 组合核函数SVM在特定领域文本分类中的应用[J]. 计算机系统应用, 2016, 25(5): 124-128.
- [4] Hershey, S., Chaudhuri, S., Ellis, D.P.W., *et al.* (2017) CNN Architectures for Large-Scale Audio Classification. 2017 *IEEE International Conference on Acoustic, Speech and Signal Processing*, New Orleans, 5-9 March 2017, 131-135. <https://doi.org/10.1109/ICASSP.2017.7952132>
- [5] 刘长征, 相文波. 基于改进卷积神经网络的肺炎影像判别[J]. 计算机测量与控制, 2017, 25(4): 185-188.
- [6] Hinton, G.E. and Salakhutdinov, R.R. (2006) Reducing the Dimensionality of Data with Neural Networks. *Science*, **313**, 504-50. <https://doi.org/10.1126/science.1127647>
- [7] Hinton, G.E., Srivastava, N., Krizhevsky, A., Sutskever, I. and Salakhutdinov, R.R. (2012) Improving Neural Networks by Preventing Co-Adaptation of Feature Detectors. arXiv:1207.0580v1.
- [8] He, K.M., Zhang, X.Y., Ren, S.Q. and Sun, J. (2016) Deep Residual Learning for Image Recognition. 2016 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, 27-30 June 2016, 770-778. <https://doi.org/10.1109/CVPR.2016.90>