

# 医学图像压缩

胡珈魁；杨晨曦；秦川

## 第一部分 设计概述

### 1.1 设计目的

随着医学成像技术的发展和一起扫描精度的提高，扫描产生的医学图像数据量越来越大，并且在临床中要求病人的诊疗影像数据保留数月甚至数年，于是医学图像存储时需要使用数据压缩技术。与此同时，远程医疗的发展，也需要实时高速医学图像的传递，高速高效的图像压缩、解压缩技术可以大大降低图像传输时间，以获得更好的医疗体验。

### 1.2 应用领域

医务数据库医学影像数据的压缩存储

远程医疗中的高速高质影像数据压缩传输与解压缩

### 1.3 主要技术特点

本项目解决方案：主要使用两个 AI 模型，分别用于医学图像质量增强和医学图像压缩。

技术特点：

- 1、部署两个 AI 模型至 ARM 核，使用 Xilinx DPU 进行 AI 计算。
- 2、将医学 DICOM 格式文件拆分为文本和图像，分别压缩存储，对文本采用 deflate 算法进行压缩，对图像采用 GANs 与基于神经网络的压缩算法

相结合的方式压缩，在码率高度压缩的情况下，仍能对图像高保真还原。

- 3、图像质量增强部分，利用循环生成网络，用非对称的 lr-hr 数据对训练超分模型。

#### 1.4 关键性能指标

##### 1、压缩率

医学图像像素深度较自然图像更大，依据扫描仪器而定，通常可达 32k 及以上，造成医学图像占用空间极大，在保证医学图像各类细节情况下，压缩像素深度。

##### 2、压缩效率

调用模型时，使用 Xilinx DPU 进行 AI 计算、使用 vitis-ai 工具对模型进行量化。

#### 1.5 主要创新点

- (1)、利用 GANs 与基于神经网络的压缩算法相结合的方式对医学图像压缩。
- (2)、利用循环生成网络，用非对称的 lr-hr 数据对训练超分模型，进行医学图像质量增强。

## 第二部分 系统组成及功能说明

### 2.1 整体介绍

本项目功能主体由两个 AI 模型组成，一个用于医学图像质量增强一个用于

医学图像压缩。整体流程：在传输 DICOM 至 Ultra96 后，ARM 核首先将 DICOM 拆分为文本与图像两部分，文本部分通过 deflate 算法进行压缩，图像压缩图像增强部分通过拉普拉斯模糊评价算法判断传入图像是否需要质量增强，若需要质量增强，则调用图像增强 xmodel，使用循环生成网络进行质量增强，若不需要则直接调用图像压缩 xmodel，采用 GANs 与神经网络压缩算法结合的方式对其像素深度进行压缩。

## 2.2 各模块介绍

以下是整个工程前端的实现，后端采用 vitis-ai 进行量化、剪枝等

```
.
├── bit_estimator.py
├── config.py          训练配置
├── convert
│   ├── onnx.py        转换为 onnx 格式，支持使用 vitis-ai onnx 进行转换
│   ├── quantize_result vitis-ai 输出的量化结果
│   │   ├── bias_corr.pth
│   │   ├── custom.json
│   │   ├── EDICConvert.py
│   │   ├── quant_info.json
│   │   └── ULTRA96.dcf
│   └── xmodel.py      vitis-ai 转换脚本
├── criterions.py
├── gdn_converted.py
├── gdn.py
├── make_dataset.py
├── model
│   └── pretrain.pth    预训练模型
├── models.py
├── README.md
├── train.py
├── utils.py
└── valid.py
```

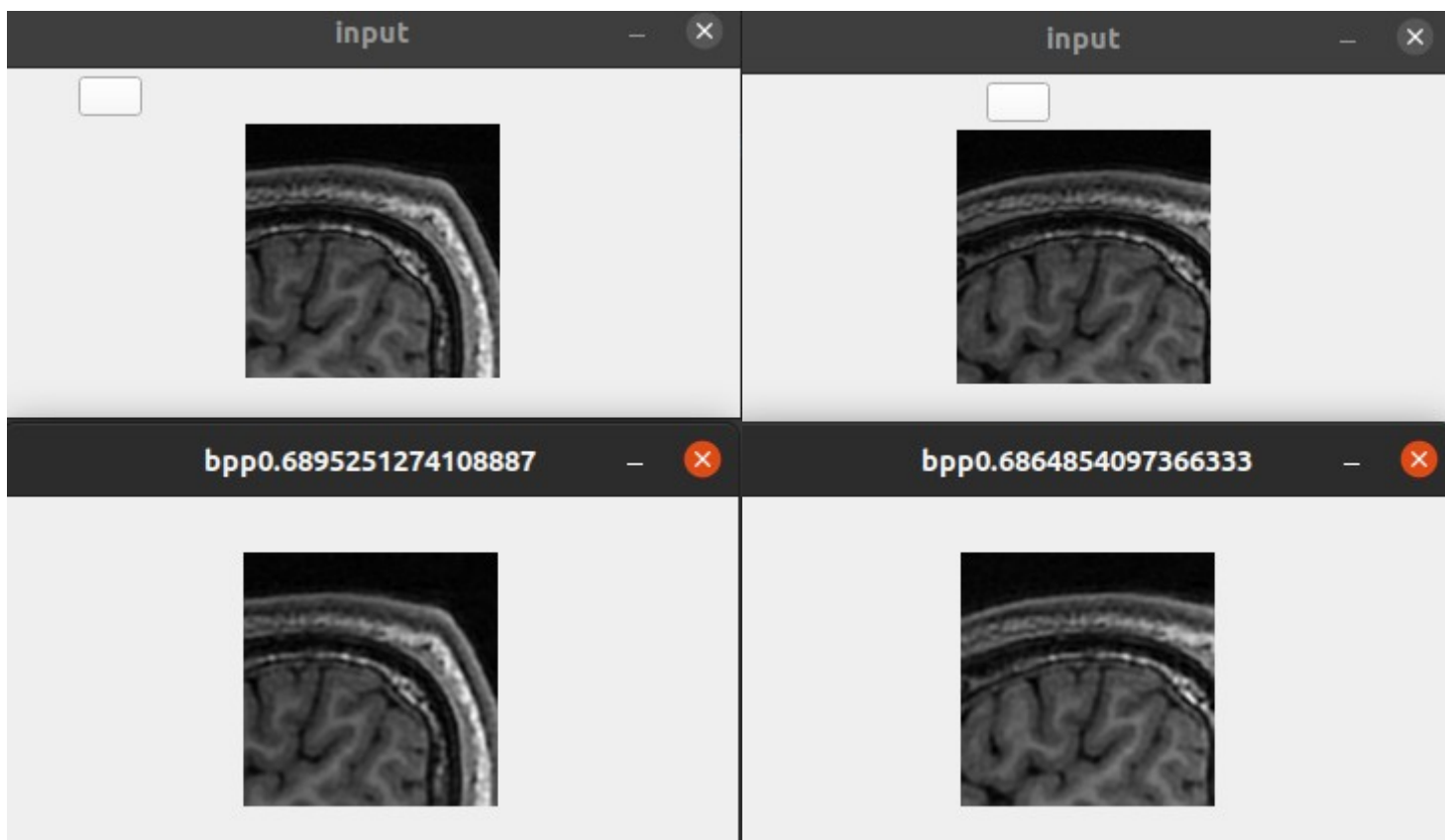
5 directories, 28 files

### 第三部分 完成情况及性能参数

DICOM 在经过 SITK 之后归一为 0-255, bpp 为 8

在部署到 DPU 上, 我们的 bpp 可以降低为 1.37, 这意味着源文件大小为 1M 的话, 压缩后文件大小为 1.37/8 M, 同时我们的图像 PSNR 到达 33, MS-SSIM 到达 0.97

在 CPU/GPU 上, 我们的 bpp 可以进一步降低为 0.7 左右, PSNR 相对较低, (由于 bpp 进一步被压缩了)



## 第四部分 总结

### 4.1 可扩展之处

该项目主要是算法实现，可以集成在如远程问诊等硬件操作

### 4.2 心得体会

vitis-ai 对 pytorch 的支持并不完善，我们需要对计算图中间过程进行大量近似操作

## 第五部分 参考文献

```
@article{liu2020unified,  
  title={A Unified End-to-End Framework for Efficient Deep Image Compression},  
  author={Liu, Jiaheng and Lu, Guo and Hu, Zhihao and Xu, Dong},  
  journal={arXiv preprint arXiv:2002.03370},  
  year={2020}  
}
```

```
@article{mentzer2020high,  
  title={High-Fidelity Generative Image Compression},  
  author={Mentzer, Fabian and Toderici, George and Tschannen, Michael and Agustsson, Eirikur},  
  journal={arXiv preprint arXiv:2006.09965},  
  year={2020}  
}
```

Maeda, Shunta. "Unpaired Image Super-Resolution using Pseudo-Supervision." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020.

## 第六部分 附录

### 1. 当 batchsize 为 1 时，batchnorm 等效于 instancenorm

设 Tensor 的形状为  $N \times H \times W \times C$  ( $N$  是 batchsize,  $H/W$  是 feature 的高/宽,  $C$  是 feature 的 channel)

故 BN 在 batch 的维度上 norm，归一化维度为[N,H,W]：

```
mu = np.mean(x,axis=0)

sigma2 = np.var(x,axis=0)

x_hat = (x-mu)/np.sqrt(sigma2+eps)

out = gamma*x_hat + beta
```

IN 是在 H，W 轴上计算的 normalization，所以归一化纬度为[H,W]：

def InstanceNorm(x, gamma, beta):

```
# x_shape:[B, C, H, W]

results = 0.

eps = 1e-5

x_mean = np.mean(x, axis=(2, 3), keepdims=True)

x_var = np.var(x, axis=(2, 3), keepdims=True)

x_normalized = (x - x_mean) / np.sqrt(x_var + eps)

results = gamma * x_normalized + beta

return results
```

故当 N=1 时，同 Batchsize=1，BN 与 IN 等效。

## 2. $x^{0.5}$ 在 $x<1$ 时可以泰勒展开，用于近似，取前五项

必要条件： $x^{0.5}$  在  $x<1$  时任意阶可导

由于  $f(z)$  可以写成  $f(z) = \frac{1}{2\pi i} \oint_l \frac{f(\zeta)}{\zeta - z} d\zeta$ ，因此具备连续可导的能力。

故  $f(x)=x^{0.5}$  任意阶可导。

充分条件：余项趋于 0

由 
$$R_n(x) = \frac{f^{(n+1)}(c)}{(n+1)!} (x-a)^{n+1}$$
，带入  $f(x)=x^{0.5}$ ，当  $x<1$  时，分母趋向于无穷，故余项趋于 0。

故  $f(x)=x^{0.5}$  在  $x<1$  时可导。

### 3. 5\*5 卷积核等效于两个 3\*3 卷积核级联

设输入大小为 (H,W)，滤波器大小为 (3,3) 或 (5,5)，输出大小为 (OH,OW)，填充为 0，步幅为 1.

第一级卷积：

$$OH=(H+0-3)/1+1=H-3+1=H-2$$

$$OW=(W+0-3)/1+1=W-3+1=W-2$$

第二级卷积：

$$OH_2=(H-2+0-3)/1+1=H-4$$

$$OW_2=(W-2+0-3)/1+1=W-4$$

当卷积核为 (5,5)：

$$OH=(H+0-5)/1+1=H-4$$

$$OW=(W+0-5)/1+1=W-4$$

上下两式相等，故等效

