西南大學
SOUTHWEST UNIVERSITY

基于深度潜在空间变换的旧照片修复

数字图像处理小组成员：

姓名	学号	分工
李洁	222020335220024	PPT 制作+汇报讲解
辛月	222020335220027	PPT 制作
王一舟	222020335220175	实验报告
杜昊	222020335220006	代码实现
严中圣	222020335220177	代码实现+实验报告

2021 年 6 月 25 日

目录

一、实验背景

照片的拍摄是为了冻结那些原本已经逝去的快乐时光。纵然时光流逝，人们仍然可以通过观看照片来唤起对过去的回忆。尽管如此，旧照片在恶劣的环境条件下保存会变质，这会导致有价值的照片内容永久受损。幸运的是，随着移动摄像头和扫描仪的普及，人们现在可以将照片数字化，并邀请一位熟练的专家进行修复。然而，手工修整通常是费时费力的，这使得成堆的旧照片无法恢复。因此，设计能够即时修复旧照片的自动算法对于那些希望恢复老照片的人来说是很有吸引力的。

在深度学习时代之前，有一些方法尝试通过自动检测局部缺陷（如划痕和瑕疵）恢复照片，并使用修复技术填充受损区域。然而，这些方法都侧重于对缺失的内容进行补齐，没有一种方法能够修复空间上均匀的缺陷，如胶片颗粒、棕色问题、褪色等，因此与现代摄影图像相比，复原后的照片仍然显得过时。随着深度学习的出现，人们可以利用卷积神经网络强大的表示能力，即学习特定任务的映射，来解决各种低级图像恢复问题。利用卷积神经网络强大的表现能力，即从大量合成图像中学习特定任务的映射。

然而，同样的框架并不适用于旧照片的复原。首先，旧照片的退化过程相当复杂，没有能够真实再现旧照片伪影的退化模型。因此，从这些合成数据中得到的模型在真实照片上的推广效果很差。第二，旧照片受到退化复合物的困扰，固有地需要不同的修复策略：在空间上均匀的非结构化缺陷，如胶片颗粒和颜色褪色，应利用邻近区域的像素进行恢复，而结构化缺陷，如划痕、灰尘斑等，则应进行修复使用全局映像上下文修复。为了避免这些问题，我们借助亚洲微软研究院与香港城市大学开源项目 Bringing-Old-Photos-Back-to-Life 将旧照片复原问题描述为一个三域转换问题，从而有效地实现了旧照片的复原效果。

二、实验目的

我们计划借助开源项目 Bringing-Old-Photos-Back-to-Life 来复原严重退化的旧照片，具体解决以下两类图像复原问题：

- 复原非结构化退化图像（如噪声、模糊、褪色和低分辨率）；
- 复原结构化退化图像（如孔洞、划痕和斑点）。

但是在老照片修复中面临许多的图像处理问题，例如填孔洞、去划痕、上色、去噪等，即包含了多种图像退化问题。而使用深度学习进行训练时往往需要制造样本对，但是真实的低质量数据包含多种退化问题，想要通过正常图像模拟出完全符合数据分布的低质量图像几乎是不可能的。所以为了回避样本制造的问题，我们将老照片修复模拟成三重态域翻译的问题，三个域分别是真实的老照片的

域 R、合成的低质照片的域 X、真实的高质照片（GT）的域 Y。我们利用三个领域的数据（即真实的旧照片、合成图像和相应的地面真实情况），并在潜在空间中进行翻译。合成图像和真实照片首先通过共享变分自动编码器（VAE）转换到同一个潜在空间。同时，训练另一个 VAE 将地面真实干净图像投影到相应的潜在空间中。然后利用合成图像对学习两个潜在空间之间的映射，从而将损坏的图像还原为干净的图像。

三、实验原理

与传统的图像复原任务相比，老照片复原更具挑战性。首先，旧照片包含了更复杂的退化，很难真实地建模，而且合成照片和真实照片之间总是存在一个域间隙。因此，网络通常不能通过单纯地从合成数据中学习来很好地概括真实照片。其次，旧照片的缺陷是多重退化的复合物，因此本质上需要不同的修复策略。非结构化缺陷，如薄膜噪声、模糊、褪色等，可以利用局部区域内的像素点，用空间均匀的滤波器进行恢复；另一方面，对于划痕、斑点等结构化缺陷，应考虑全局背景，以保证结构的一致性。下面，我们分别针对上述的泛化问题和混合退化问题提出解决方案。

3.1 基于隐空间翻译的图像复原

为了减小域间隙，我们将旧照片复原问题归结为一个图像翻译问题，我们将干净图像和旧照片视为来自不同域的图像，并希望学习它们之间的映射关系。然而，与连接两个不同域的一般图像翻译方法不同，我们跨三个域翻译图像：真实光域 R，图像遭受人工退化的合成域 X，以及包含未退化图像的相应地面真实域 Y。这种三重态域翻译在我们的任务中至关重要，因为它利用了未标记的真实照片以及大量与地面真实相关的合成数据。

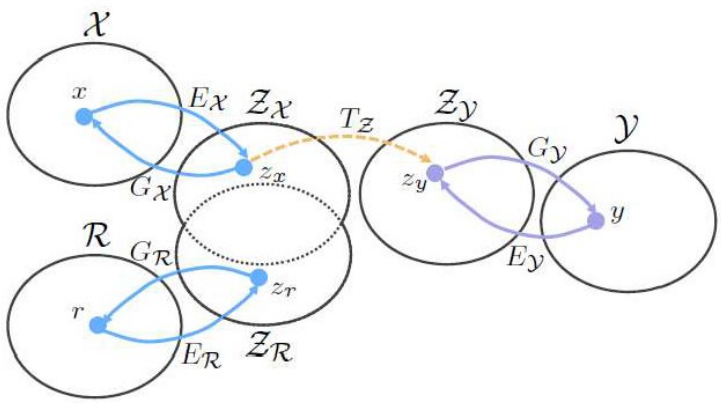

图 1 三重态域翻译方法阐释图

具体地说，利用$(x,y)_{1-N}$的合成数据对，我们学习了从损坏图像的潜在空间

Z_x到地面真实的潜在空间Z_y的翻译，通过映射 $T_z: Z_x \to Z_y$，其中Z_y可以通过生成器 $G_z: Z_y \to Y$ 进一步反转为Y。通过学习潜在空间的翻译，可以通过依次执行映射来恢复真实的旧照片 r：

$$r_{R\to y} = G_y \circ T_z \circ E_R(r) \tag{1}$$

3.1.1 VAE 潜空间中的域对齐

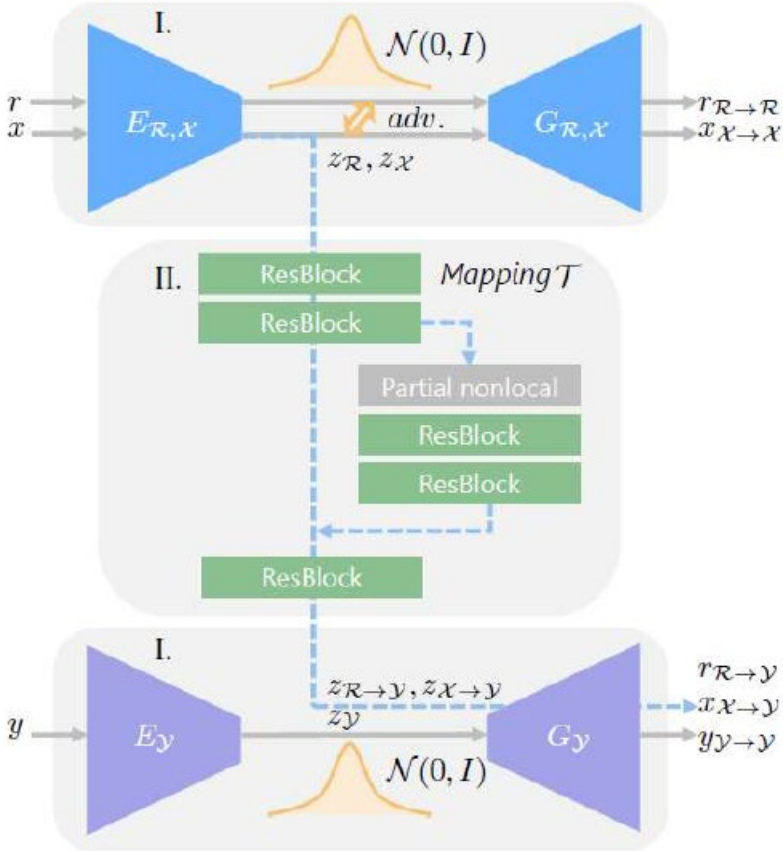

图 2 修复网络结构示意图

方法的一个关键是满足 R 和 X 被编码到同一个潜空间的假设。为此，我们建议使用变分自动编码器（VAE）来编码具有紧凑表示的图像，其域间隙由一个对抗性鉴别器进一步检查。我们使用图 2 所示的网络体系结构来实现这个概念。在第一阶段中，学习两个 VAE 作为潜在表征。旧照片{r}和合成图像{x}通过编码器 $E_{R,X}$ 和生成器 $G_{R,X}$ 共享第一个名为 VAE_1，而地面真实图像{y}通过编码器-生成器对{E_Y,G_Y}输入到第二个 VAE_2 中。VAE_1 对 r 和 x 是共享的，目的是将来自两个损坏域的图像映射到共享的潜在空间。VAE_s 假设隐码的分布具有高斯先验性，因此可以通过从潜在空间采样来重建图像。

我们使用 VAE 而不是普通的自动编码器，因为 VAE 由于 KL 正则化而具有更密集的潜在表现，这有助于为{r}和{x}产生更紧密的潜在空间，从而导致更小的畴隙。为了进一步缩小这个缩小的空间中的域间隙，我们建议使用一个对抗

性网络来检查剩余的潜在缺口。具体地说，我们训练另一个区分 Z_R 和 Z_X 的鉴别器 $D_{R,X}$，其损失函数定义为：

$$\mathcal{L}^{\text{latent}}_{\text{VAE}_1,\text{GAN}}(r,x) = \mathbb{E}_{x\sim\mathcal{X}}[D_{\mathcal{R},\mathcal{X}}(E_{\mathcal{R},\mathcal{X}}(x))^2] + \mathbb{E}_{r\sim\mathcal{R}}[(1-D_{\mathcal{R},\mathcal{X}}(E_{\mathcal{R},\mathcal{X}}(r)))^2] \tag{2}$$

同同时，VAE_1 的编码器 $E_{R,X}$ 试图用矛盾的损耗欺骗鉴别器，以确保 R 和 X 映射到同一空间。结合潜在对抗损失，VAE_1 的总目标函数为：

$$\min_{E_{\mathcal{R},\mathcal{X}},G_{\mathcal{R},\mathcal{X}}} \max_{D_{\mathcal{R},\mathcal{X}}} \mathcal{L}_{\text{VAE}_1}(r) + \mathcal{L}_{\text{VAE}_1}(x) + \mathcal{L}^{\text{latent}}_{\text{VAE}_1,\text{GAN}}(r,x) \tag{3}$$

3.1.2 通过潜在映射进行恢复

利用 VAEs 捕获的潜码，第二阶段利用合成图像对$\{x,y\}$并提出通过映射其潜在空间来学习图像恢复（图 2 中的映射网络 M）。潜在修复的好处是三方面的。首先，由于 R 和 X 在同一个潜在空间中排列，从 Z_X 到 Z_Y 的映射也可以很好地推广到 R 中的图像恢复。其次，在紧凑的低维潜在空间中的映射原则上比在高维图像空间中更容易学习。另外，由于两个 VAE 是独立训练的，因此两个流的重建不会相互干扰。在给定 Z_X 映射的潜在代码 z_y 的情况下，生成器 G_y 始终可以获得绝对干净的图像而不会退化，而如果我们学习像素级的翻译，退化很可能仍然存在。该映射网络的 loss 如下：

$$\mathcal{L}_{\mathcal{T}}(x,y) = \lambda_1\mathcal{L}_{\mathcal{T},\ell_1} + \mathcal{L}_{\mathcal{T},\text{GAN}} + \lambda_2\mathcal{L}_{\text{FM}} \tag{4}$$

3.2 多重退化恢复

在 3.1.2 的映射网络中，主要是用的是残差模块，由于感受野的限制，网络主要关注局部的特征。然而一些老照片中一些结构的损坏需要更大范围内的信息进行搜索填充，因此需要让设计的网络即支持获取局部的信息，又支持获取全局的信息。因此，我们添加了一个含有非局部模块的全局信息提取的分支。这里采用一个 mask 作为输入，来防止图片中损坏区域的像素不会被用于修复损坏区域。

对于一个 HWC 维度的特征 F，m 表示同样大小的单通道二值 mask 图像，m 中值为 1 时表示损坏区域，值为 0 表示正常区域。那么对于 F 中位置 i 与位置 j 之间的关系可以表示为$s_{i,j}$，$s_{i,j} \in R^{HW\times HW}$，是每个 pixel 之间的关系。公式如下：

$$s_{i,j} = (1-m_j)f_{i,j} / \sum_{\forall k}(1-m_k)f_{i,k} \tag{5}$$

从公式中可以看出，$s_{i,j}$表示像素 j 对于像素 i 的归一化表示，也可以看作时像素点 j 贡献的权重值，当像素点在 m 中标记为腐坏点时，贡献值为 0；其中，

$$f_{i,j} = \exp(\theta(F_i)^T \cdot \phi(F_j)) \tag{6}$$

上面的式子中F_i、F_j是维度为 C*1 的向量；θ 与 ϕ 是将向量映射到高斯分布的函数，所以最终$f_{i,j}$是一个标量。最终部分非局部的输出为：

$$O_i = \nu\left(\sum_{\forall j} s_{i,j}\mu(F_j)\right) \tag{7}$$

这是每个位置相关特征的加权平均值。不难看出，$s_{i,j}$ 就是权重值，而这里的u、v用于进一步整合特征。u、v、θ、ϕ 都是 1×1 卷积。通过这样的上述的模块，就可以起到全局感受野的作用。但是只希望对被腐坏的区域进行上述操作，其他区域不需要这样的操作，因此我们又做了区域融合，即当 mask 中对应的区域被标记为损坏区域时使用 global 的信息，否则使用局部特征信息。

$$F_{fuse} = (1-m)\odot\rho_{\text{local}}(F) + m\odot\rho_{\text{global}}(O) \tag{8}$$

其中⊙表示矩阵的 Hadamard 积，至此模型设计完毕。

四、实验内容

4.1 配置实验环境

操作系统：Ubuntu 20.04

实验环境：CUDA+Cudnn+Opencv+Nvidia GPUs

Requirements: 见文件 requirements.txt

4.2 对非结构化退化图像的复原

4.2.1 Stage1:整体修复(Overall Restoration)

在步骤一中，我们调用 Pix2PixHD 模型进行整体质量修复，该模型使用 Pytorch 构建，对传统的 GAN 模型进行了改进，不再输入随机噪声，而是输入用户所给图片，同时使用成对的数据(paired data)进行训练；其次为了解决 Pix2Pix 网络产生高维数据输出困难的问题，Pix2PixHD 采取了金字塔式的方式，先输出低分辨率的图片，再将低分辨率图片作为另一个网络的输入，然后生成分辨率更高的图片。于是模型推理后最终就可以实时获取高分辨率、高质量图像。

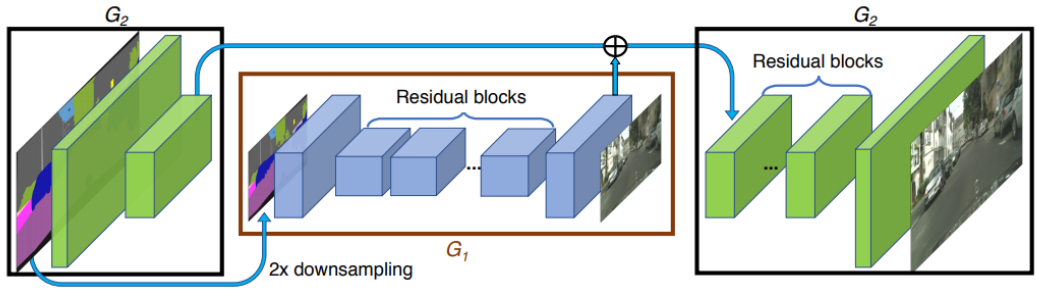

图 1 Pix2PixHD 模型网络结构

```
(.venv) root@Yan-HUAWEI:/home/god/Bringing-Old-Photos-Back-to-Life# python3 run.py
--input_folder "/home/god/Bringing-Old-Photos-Back-to-Life/test_images/old" --outpu
t_folder "/home/god/Bringing-Old-Photos-Back-to-Life/output_images/old" --GPU -1
Running Stage 1: Overall restoration
Now you are processing M.png
Now you are processing N.JPG
Now you are processing O.jpg
Now you are processing a.png
Now you are processing b.png
Now you are processing c.png
Now you are processing d.png
Now you are processing e.png
Now you are processing f.png
Now you are processing g.png
Now you are processing h.png
Finish Stage 1 ...
```

图 2 步骤 1 整体修复代码调试过程

图 3 步骤 1 整体修复示意图

4.2.2 Stage2:面部检测(Face Detection)

步骤 2 调用 dlib 库，使用 dlib 的 68 特征点人脸检测模型，具体为先检测人脸区域，其次再检测人脸器官，再通过 ResNet34 生成 128 维向量，再经过仿射变换得到人脸区域图像。

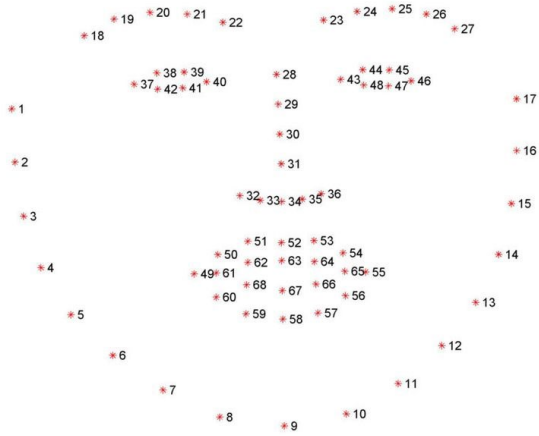

图 4　dlib 的 68 个人脸特征点示意图

```
Running Stage 2: Face Detection
Warning: There is no face in b.png
1
1
1
Warning: There is no face in f.png
1
1
1
Warning: There is no face in e.png
Warning: There is no face in d.png
1
Finish Stage 2 ...
```

图 5 步骤 2 人脸检测代码调试过程

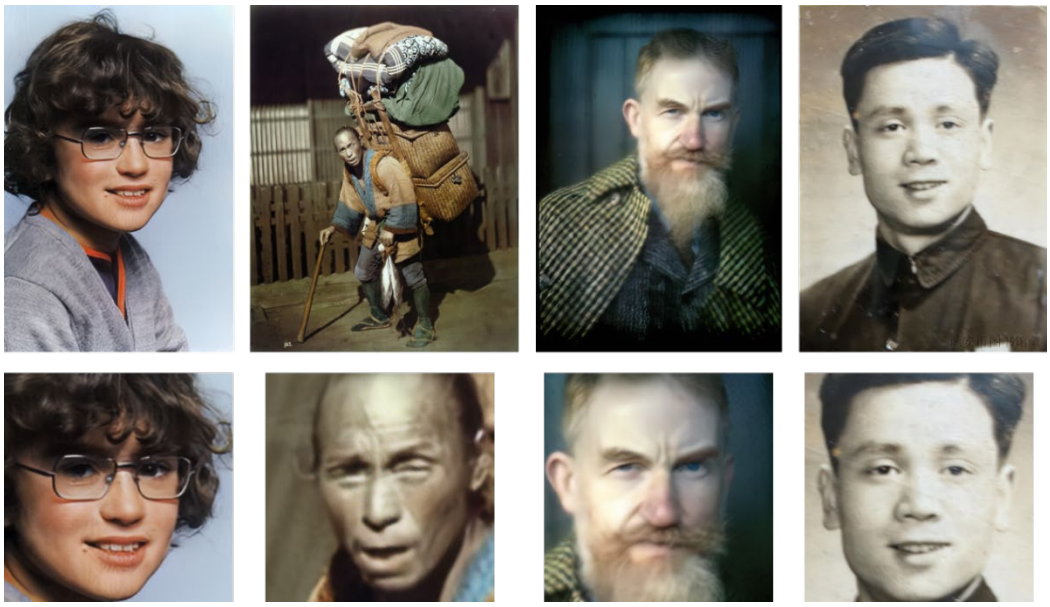

图 6 步骤 2 人脸检测效果图

4.2.3 Stage3:面部增强(Face Enhancement)

在步骤 3 中，我们调用 Pix2Pix 模型，对检测出的人脸进行增强修复，Pix2Pix 模型是一个基于条件生成对抗网络(cGAN)的图像翻译模型，其由生成器 G 和判别器 D 组成，其算法如图 7 所示。对于我们的任务，生成器 G 的目标是将语义

标签图转换成真实可见的图像，同时判别器旨在判别真实图像和转换图像。具体实现效果见图 9。

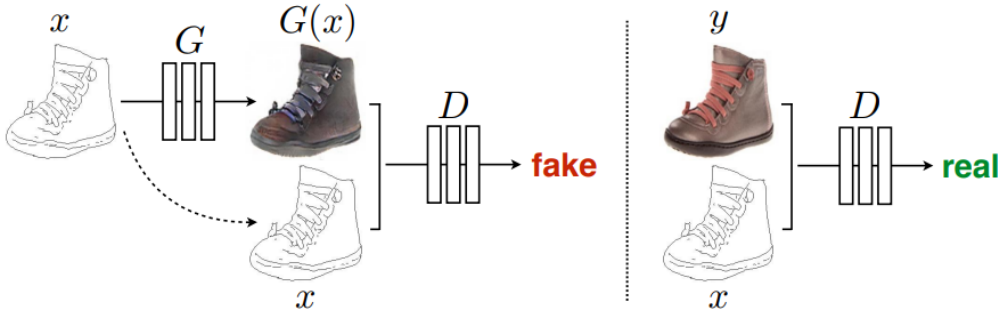

图 7 Pix2Pix 模型算法示意图

```
Running Stage 3: Face Enhancement
dataset [FaceTestDataset] of size 7 was created
The size of the latent vector size is [8,8]
Network [SPADEGenerator] was created. Total number of parameters: 92.1 million. To
see the architecture, do print(network).
hi :)
Finish Stage 3 ...
```

图 8 步骤 3 人脸增强代码调试过程

图 9 步骤 3 人脸图像增强示意图

4.2.4 Stage4:融合(Blending)

步骤 4 中，我们首先将步骤 1 修复照片的对齐人脸与步骤 3 增强修复的对齐人脸进行直方图匹配，再将直方图匹配后的人脸逆仿射变换，还原到步骤 1 修复照片的尺寸与角度，最后将逆变换后的人脸通过高斯滤波器进行平滑处理，融合到步骤 1 修复的照片，即得到最终效果。

```
Running Stage 4: Blending
Warning: There is no face in b.png
Warning: There is no face in f.png
Warning: There is no face in e.png
Warning: There is no face in d.png
Finish Stage 4 ...


All the processing is done. Please check the results.
```

图 10 步骤 4 图像融合代码调试过程

图 11 非结构化退化图像复原最终效果图

4.3 对结构化退化图像的复原

4.3.1 Stage1:整体修复(Overall Restoration)

(1) 划痕检测

对于结构化退化的图像，我们首先进行划痕检测。第一步我们调用 Unet 模型进行图像分割。Unet 建立在 FCN 的网络架构上，并扩大了这个网络框架，添加了上采样的阶段，且添加了很多特征通道，从而保证了原图像纹理的信息能够保持高分辨率，所以更加适合超大图像分割。Unet 整体结构即先编码（下采样）再解码（上采样），下采样通过 max pool 2×2 进行 1/2 下采样，下采样之间是两个 conv 卷积层，这里使用的是 valid 卷积。在通过下采样提取目标特征后，再开始上采样部分，将特征提取部分对应的通道数相同尺度融合。图像处理后通过 sigmoid()运算，将超过阈值(0.4)的像素定义为划痕点，否则定义为 0，完成图像分割，最后逐个对其像素点进行分类，将检测出的划痕点置为 1，其他为 0。

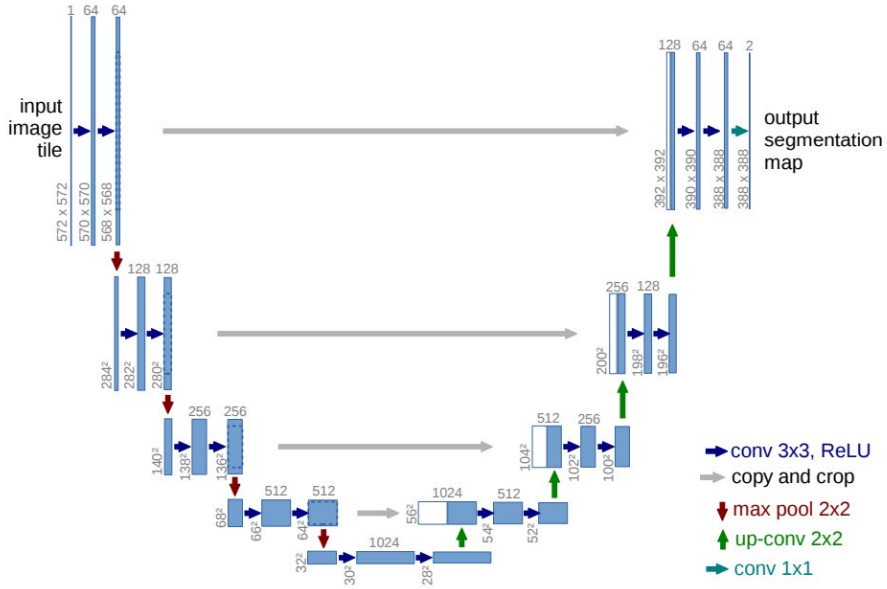

图 12 Unet 框架网络结构图

图 13 部分划痕检测效果图

(2) 整体修复

在进行划痕检测后，同非结构化退化图像处理，我们调用 Pix2PixHD 模型进行整体修复，其中对于有掩码图像(有划痕)的处理，无划痕处保留原像素，划

```
Running Stage 1: Overall restoration
initializing the dataloader
model weights loaded
directory of testing image: /home/god/Bringing-Old-Photos-Back-to-Life/test_images/
old_w_scratch
processing E.png
processing a.png
processing b.png
processing c.png
processing d.png
You are using NL + Res
Now you are processing E.png
/home/god/Bringing-Old-Photos-Back-to-Life/.venv/lib/python3.8/site-packages/torch/
nn/functional.py:3454: UserWarning: Default upsampling behavior when mode=bilinear
is changed to align_corners=False since 0.4.0. Please specify align_corners=True if
 the old behavior is desired. See the documentation of nn.Upsample for details.
  warnings.warn(
Now you are processing a.png
Now you are processing b.png
Now you are processing c.png
Now you are processing d.png
Finish Stage 1 ...
```

图 14 整体修复代码调试过程

痕处用 unit8(255)填充，再进行规整图像，输入图像与掩码数据，进行模型推理图像修复过程，整体修复完毕。

图 15 步骤 1 整体修复效果图

4.3.2 Stage2:面部检测(Face Detection)

同非结构化退化图像 4.2.2 处理，在整体修复后对人脸面部进行处理，调用 dlib 人脸检测模型进行面部检测。

```
Running Stage 2: Face Detection
1
1
1
1
Warning: There is no face in d.png
Finish Stage 2 ...
```

图 16 面部检测代码调试过程

图 17 步骤 2 面部检测效果图

4.3.3 Stage3:面部增强(Face Enhancement)

步骤 3 同 4.2.3 对截取面部图像调用 Pix2Pix 模型进行面部图像增强，得到增强后的人脸图像。

```
Running Stage 3: Face Enhancement
dataset [FaceTestDataset] of size 4 was created
The size of the latent vector size is [8,8]
Network [SPADEGenerator] was created. Total number of parameters: 92.1 million. To
see the architecture, do print(network).
hi :)
Finish Stage 3 ...
```

图 18 面部增强代码调试过程

图 19 步骤 3 面部增强效果图

4.3.4 Stage4:融合(Blending)

同 4.2.4，将图像增强后的人脸图像融合到步骤 1 整体修复后的图像中，得到最终修复效果图。

```
Running Stage 4: Blending
Warning: There is no face in d.png
Finish Stage 4 ...


All the processing is done. Please check the results.
```

图 20 图像融合代码调试过程

图 21 结构化退化图像复原最终效果图

五、实验总结

5.1 模型的评价

5.1.1 模型的优点

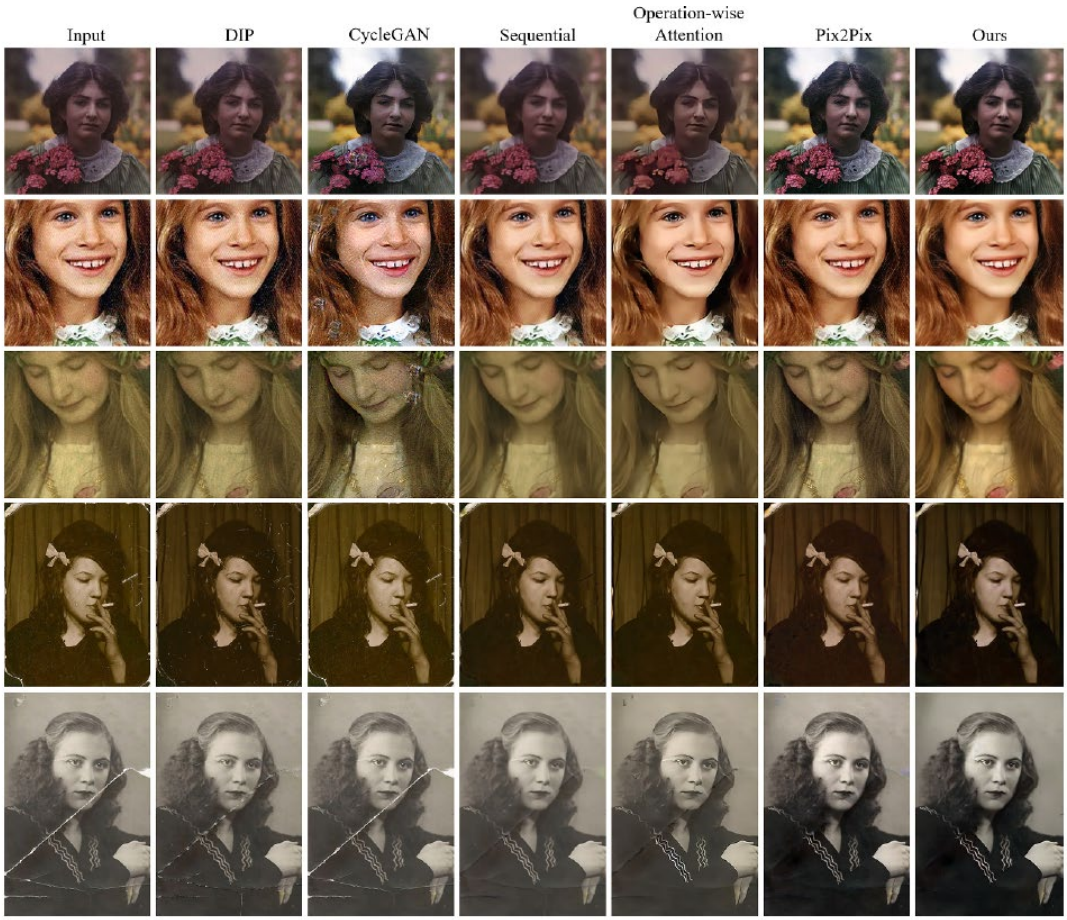

图 22 不同模型图像复原效果比对图

我们将模型与现有的方法进行了定性比较，照片上的裂痕被修复意味着我们能同时修复非结构化和结构化退化。从修复效果来看，我们的着色更自然更接近真实，并且我们的方法对重建的人脸更生动。造成这种现象的潜在原因是我们在面部增强充分利用人脸的细微细节，恢复结果明显优于其他方法。同时，我们提出了一种新的三重态域翻译网络来恢复旧照片中的混合退化。减少了旧照片与合成图像之间的域间隙，并在潜在空间学习干净图像的转换。与以往的方法相比，我们的方法不存在泛化问题。此外，我们还提出了一种部分非局部块，它利用全局上下文恢复潜在特征，从而使划痕得到较好的结构一致性修复。效果显示我们的方法在恢复严重退化的旧照片时表现出更良好的性能。

比对显示，Pix2pix 可以与我们的合成图像方法相媲美，但是在视觉上不如我们的方法。一些胶片噪声和结构性缺陷仍然存在于最终输出中。这是由于合成图像与真实照片之间存在域间隙，使得该方法无法推广。相比之下，我们的方法给出了清晰、清晰的图像，用不明显的伪影合理地填充了划痕。除了成功地解决

了数据合成中考虑的伪影，我们的方法还可以适当地增强照片的色彩。总的来说，我们的方法给人的视觉上最愉快的结果，恢复后的照片看起来像现代摄影图像。

5.1.2 模型的缺陷

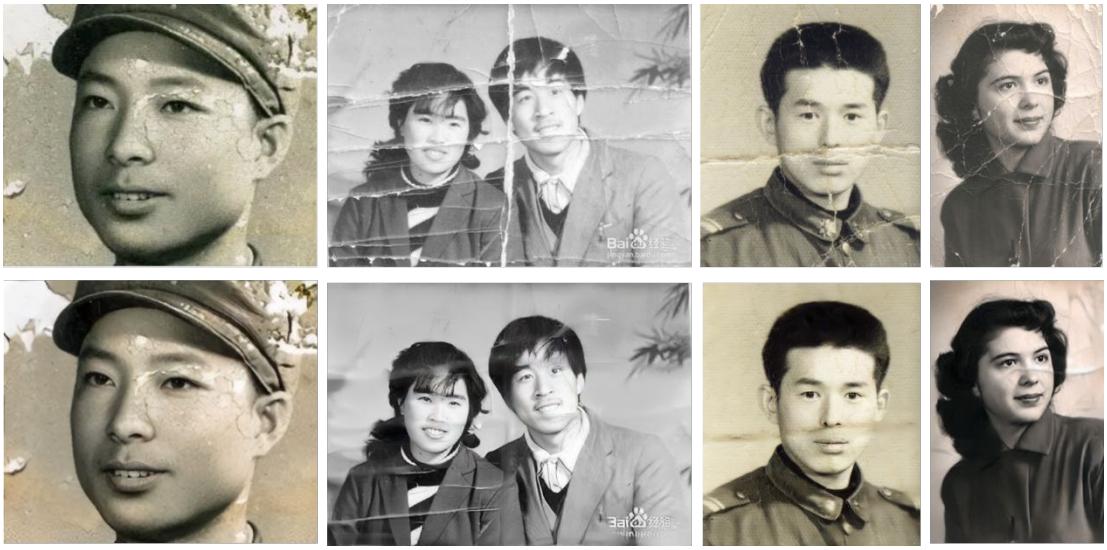

图 23 模型的缺陷：无法处理复杂的着色

在处理过程中，我们发现我们的方法无法处理复杂的着色，如图 23 所示。这是因为我们的数据集包含了很少有这样缺陷的旧照片。我们可以通过显式地考虑合成过程中的阴影效果或添加更多这样的照片作为训练数据来解决这个限制。

此外，对于高分辨率的图像我们的弱监督模型处理需要较大的算力支持，如何更好地优化模型的耗时与计算量目前还没有更好的解决方案。

5.2 实验感悟

数字图像与我们密切相关，它伴随人们的生活、学习、工作，并在军事、医学和工业等方面发挥着极大的作用。图像作为信息的重要载体，在信息传输方面有着声音、文字等信息载体不可替代的作用。近年来随着深度学习等新兴人工智能技术高速发展，计算机视觉领域也取得了蓬勃发展，如何将其更多地应用到生活实践中，则是我们需要继续探索的。

通过本次实验，我们小组利用深度潜在空间变换的三重态域图像翻译技术让老照片重新焕发了光彩，相比过去的热门模型如 Pix2Pix、CycleGAN 等我们的模型取得了显著的优势，在代码复现的过程中，我们深切感受到了深度学习领域的神奇之处，同时也对新一代计算机视觉的崛起萌发出了探索的决心与希冀。在数字图像处理发展日新月异的今天，新的技术不断萌发，所以仍需我们不断地去学习，去获取前沿的科研动态来充实自己的水平。

参考文献

[1] Wan, Ziyu, et al. "Bringing old photos back to life." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020.

[2] Isola, Phillip, et al. "Image-to-image translation with conditional adversarial networks." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017.

[3] Wang, Ting-Chun, et al. "High-resolution image synthesis and semantic manipulation with conditional gans." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018.

[4] Ronneberger, Olaf, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation." International Conference on Medical image computing and computer-assisted intervention. Springer, Cham, 2015.

[5] https://blog.csdn.net/weixin_41943311/article/details/109312044

[6] https://blog.csdn.net/weixin_41943311/article/details/109362303