AGTGAN：用于古文字照片生成的非配对图像转换

Hongxiang Huang*
杨代辉*
戴刚*
eehxhuang@mail.scut.edu.cn
华南理工大学
中国广州

韩震
慕尼黑路德维希-马克西米利安大学
慕尼黑
德国慕尼黑

王宇毅
瑞士联邦理工学院
苏黎世，瑞士
中车研究院
中国株洲

林健文
香港理工学院
大学
中国香港

杨帆
华南理工大学
中国广州

黄双平†
华南理工大学
琶洲实验室
中国广州

刘永革
安阳师范学院
中国安阳

何梦超 达摩院，阿里巴巴
集团 中国杭州

arXiv:2303.07012v1 [cs.CV] 13 Mar 2023

摘要

古文字研究对于考古学和语文学具有重要价值。古文字照片是重要的资料形式，但人工进行古文字照片识别极其耗时，且依赖专业知识。因此，人们迫切需要自动分类方法。然而，由于缺少标注数据，目前的分类性能受到限制。数据生成是解决数据稀缺问题的一种成本低廉且有效的方法。尽管如此，古代摄影文字丰富多样的字形和复杂的背景纹理使生成任务难度很大，导致现有方法的结果不尽如人意。为此，本文提出一种名为 AGTGAN 的无监督生成对抗网络。该方法通过显式建模全局和局部字形风格，并结合笔画感知纹理迁移与配套的对抗学习机制，能够生成字形多样、纹理逼真的字符。我们在古文字照片数据集（如 OBC306 和 CSDD）上评估了该方法。与其他最先进的方法相比，我们的方法在多项指标上表现更优，生成样本的多样性和真实性也显著更高。利用生成的图像，在规模最大的甲骨文字体数据集上进行的实验表明，我们的方法可显著提升分类准确率，最高达 16.34%。源代码见 https://github.com/Hellomystery/AGTGAN。

计算机分类系统概念

• **计算方法 → 图像处理**；**计算机视觉**。

关键词

古文字生成，图像到图像转换，生成对抗网络

ACM参考文献格式：

黄洪翔、杨代辉、戴刚、韩震、王雨怡、林健文、杨帆、黄双平、刘永革和何梦超。2022年。AGTGAN：用于古文字照片生成的非配对图像转换。载于第*30*届*ACM*国际多媒体会议（*MM '22*），*2022*年*10*月*10*日至*14*日，葡萄牙里斯本。ACM，纽约，纽约州，美国，12页。https://doi.org/10.1145/3503161.3548338

1 引言

即使时光流逝，人类探索文明起源的好奇心也永不消退。通过祖先留下的印记，也就是古文字，现代人能够了解古代文明。众所周知的古文字有多种，例如甲骨文、楔形文字、玛雅文字、印度河文字和古埃及象形文字。识读这些文字的内容，是推动古代文明和文学领域学术研究的重要前提。

传统上，古文字研究一直是古文字学专家的专属领域。通常，他们通过在古代文献照片上手动模拟古代字符，将古文字照片（PC）转换为模拟字符（SC）。模拟字符通常被收录在古文字字典中 [4, 11, 55]，，作为研究古文字的主要工具。图 1 展示了一些甲骨文和楔形文字的模拟字符与古文字照片。然而，专家依靠经验和直觉识别古文字照片，导致效率低下且存在很大歧义。利用深度学习方法进行自动古文字影像识别，在加速古文字研究进程方面具有巨大潜力，例如释读 [4, 11, 55], 和复原。

*两位作者对本研究的贡献相同。†通讯作者。

允许免费制作本作品全部或部分内容的数字或纸质副本，供个人或课堂使用，但不得为营利或商业利益而制作或分发副本，且副本首页须保留此声明和完整引文。对于本作品中由 ACM 以外的其他方拥有版权的组件，必须尊重其版权。注明出处后可摘录。若要以其他方式复制、转载、发布至服务器或分发给列表中的用户，须事先获得具体许可和/或支付费用。请通过 permissions@acm.org 申请许可。*MM '22，2022*年*10*月*10*日至*14*日，葡萄牙里斯本 © 2022 Association for Computing Machinery。ACM ISBN 978-1-4503-9203-7/22/10...$15.00
https://doi.org/10.1145/3503161.3548338

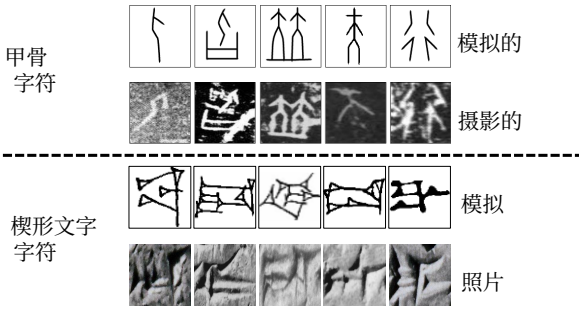


图1：模拟字符和照片字符示例。第一行和第二行分别为模拟甲骨字符和甲骨照片字符。第三行和第四行分别为模拟楔形文字字符和楔形文字照片字符。

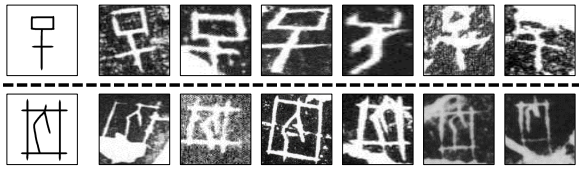


图2：具有多样字形和复杂背景纹理的 POC 示例。

然而，深度学习方法需要大量高质量标注数据，才能训练出准确的分类器 [36, 54, 63, 64]。此外，收集和标注古文字照片数据既耗时，又依赖专业知识。因此，由于标注数据稀缺，现有古文字照片分类器的性能仍不尽如人意。

合成新数据是解决数据稀缺问题的一种方法 [15, 46]。生成对抗网络（GAN） [13] 为图像生成开辟了新道路，并带来了许多令人惊叹的成果 [5, 8, 25, 34, 35]。然而，生成古代摄影文字并非易事。以图 2 所示的甲骨文照片字符（POC）为例。由于拍摄角度和书写者的差异，POC 会呈现出各种全局和局部的形状变化，例如倾斜或旋转、笔画相对位置的变化等。在纹理方面，由于存在不规则噪声，POC 印有复杂的背景。因此，多样的字形和复杂的背景纹理是字符生成中需要考虑的重要方面，这也增加了任务的难度。

主流的字符生成方法认为，字符图像包含内容和风格信息，而风格可分为形状风格和纹理风格 [2, 9]。一些方法 [53, 65] 基于解耦表示并行学习提取内容和风格信息，然后将提取的信息耦合起来，生成特定风格的字符。这些方法主要将字符的风格视为其形状风格。

然而，古代摄影文字复杂的纹理增加了内容与风格解耦的难度，导致使用这些方法时生成质量下降。此外，一些两阶段方法 [2, 27] 将字符图像生成分为形状建模阶段和笔画渲染阶段。尽管如此，这些方法仍需要大量配对数据或类别标签进行训练，这对于古文字而言成本高昂且不切实际。此外，现有主流字符生成方法有一个共同的缺点：它们对形状风格的建模不够明确，因此在缺乏强监督的情况下，难以学习多样化的全局和局部形状模式。总而言之，生成

古代摄影文字的生成需要考虑复杂的形状风格和纹理风格。此外，最好减少对非配对数据和其他标注信息的需求，并通过无监督学习实现这一点。

为此，我们提出了一种新颖的无监督生成模型，称为字形变换与纹理迁移生成对抗网络（AGTGAN）。该模型学习从模拟字符到古文字照片的复杂映射，从而合成多样且真实的摄影字符图像。模型由字形变换生成对抗网络（GTG）和纹理迁移生成对抗网络（TTG）级联而成，并支持端到端训练。本文的贡献有四点：

（1）所提出的新型摄影字符生成模型，即 AGTGAN，是首个用于丰富带标注摄影字符数据的方法。该方法整合了 GTG 和 TTG，分别用于生成字形变化和执行纹理迁移。通过新颖的关联对抗训练机制，整个网络可进行端到端训练。（2）我们提出了一种字形形状生成器，将仿射变换与薄板样条（TPS）变换相结合，以显式建模古代摄影字符的全局和局部形状风格。此外，我们引入噪声注入来提高形状变换的随机性，并提出信号与噪声平衡正则化，引导模型生成多样且有意义的字形。（3）我们为 TTG 引入了一种新的笔画感知一致性损失，以解决纹理迁移过程中生成的古文字照片存在的模糊问题。（4）定量和定性评估结果表明，我们生成的样本具有良好的多样性和出色的真实性。在最大的摄影甲骨字符（POC）数据集 OBC306 [19], 上使用生成样本进行的实验表明，我们的方法使 POC 字形分类准确率绝对提升了 16.34%。

2 相关工作

字符生成长期以来一直被视为一项重要挑战，而生成古代摄影文字却没有得到应有的关注。目前，古代文字生成仅限于某一类古代字符，且生成样本未能呈现古代字符原有的字形和纹理特征 [43]。在本节中，我们

首先回顾古代摄影文字生成相关领域的工作，然后介绍字符生成中常用的一种框架。

2.1 手写文本生成

我们首先回顾手写文本生成的方法，因为古文字本质上是手写或手刻文字。手写文本生成任务与古文字照片生成类似，旨在模仿人类风格的自然手写体。早期的两步法 [32, 50]先生成独立字母，再将它们串接起来，组成完整单词。这些方法依赖手工制定规则，只能生成变化有限的手写体。近年来，深度生成模型能够直接生成完整单词图像。人们使用循环神经网络和生成对抗网络生成多种风格的手写体 [8, 14, 25]。[28] 和 [3] 则在少样本设置下，通过提取风格特征，并以预定义的固定长度文本内容为条件来生成文本图像。此外，[37]是一种通过变换形状来增加手写文本图像数量的对抗式增强方法。上述方法的设计目标仅是生成拉丁字符，而且其中大多数需要昂贵的文本标注来提升生成质量。

2.2 字体生成

字体生成通常被视为文本图像的风格迁移任务，许多研究采用图像到图像翻译方法来处理。例如，zi2zi [49] 通过条件生成对抗网络实现字体风格迁移。在 zi2zi 的基础上，DCFont [26] 引入了风格分类器，以获得更好的风格表示。TET-GAN [57] 通过风格化子网络和去风格化子网络，学习解耦并重新组合文本图像的内容特征与风格特征，但它只能迁移纹理风格。如果仅依赖单个自编码器对字体风格进行建模，就难以同时刻画复杂的形状和纹理风格。Samaneh 等人 [2] 提出了两阶段字体生成框架 MC-GAN，将字体风格划分为形状风格和纹理风格，并使用两个不同的网络依次转换形状和迁移纹理。然而，该方法仅适用于 26 个英文字母，导致其泛化能力有限。

对于复杂字符（如汉字）的字体生成，字形形状变换始终是研究者关注的重点。[10, 22, 27, 47, 51–53, 59, 65] 针对单色中文字体生成任务提出了相关方法，将字体风格简化为形状风格。EMD [65] 基于解耦思想，使用不同的编码器提取字体的内容向量和形状风格向量。SA-VAE [47] 证明，汉字的领域知识（如部首和笔画信息）有助于提升输出图像质量。CalliGAN [52] 和 StokeGAN [59] 分别为条件生成对抗网络和 CycleGAN [66], 添加额外的字符部件编码，以利用先验知识保持结构信息。RD-GAN [22]提出了部首提取模块，将部首提取为先验知识，从而提升判别器的性能，并生成固定风格的未见字符。与以往方法不同，ChiroGAN [10] 和 SCFont [27] 采用了不同的字体生成范式：先提取骨架，再绘制笔画。然而，DG-Font [53] 和 ZiGAN [51],指出，上述方法 [10, 22, 27, 47, 52, 59] 需要昂贵的监督信息，例如笔画信息或配对数据。DG-Font引入了特征变形跳跃连接，通过可变形卷积 [6] 实现，以提升网络生成笔画或部首形状变形的能力。ZiGAN从非配对数据中学习额外的结构知识，以增强对字符内容的粗粒度理解。受 [2]和 [65],启发，AGIS-Net [9]将解耦框架的解码器分为两部分：形状风格重建分支和纹理风格重建分支，从而能够同时实现复杂字符的形状变换和纹理迁移。

然而，现有的字体生成方法可能并不适用于古代摄影文字的生成，原因如下。一方面，古代摄影文字的域内字形比一般字体生成任务中的字形更加多样。这是因为古代摄影文字由不同时期的不同人刻写，并从不同角度拍摄。相反，在一般字体生成任务中，每种字体的形状风格通常是一致的，因为每种字体中的字符都由同一位书写者书写。另一方面，尽管古代摄影汉字的纹理特征（例如颜色、亮度和颗粒）相对稳定，但内容（字形）与纹理风格难以解耦，因为背景噪声的纹理与前景笔画的纹理非常相似，容易造成混淆和误解。

2.3 图像到图像转换

近年来，图像到图像转换取得了令人瞩目的成果 [23, 30, 40, 41, 66]。[23] 首次提出条件生成对抗网络，无需手动调整映射函数，即可根据参考输入获得所需输出。然而，该方法需要配对训练数据。为解决配对数据不可用的问题，[66] 在损失函数中使用了循环一致性正则项。这两项工作启发了许多后续方法 [17, 20, 31]。然而，这些方法只能处理一对一转换。为建模一对多映射，[30] 和 [21] 提出了一种解耦表示框架，可将源内容转换为给定风格。在 CUT [40], 中，研究者指出，[20, 21, 30, 31, 66] 所采用的循环一致性损失假设两个域之间的关系是双射，这一假设往往过于严格。另一种解决方案是引入对比学习，通过 PatchNCE 损失实现非配对图像到图像转换，从而最大化输入图像与输出图像中对应图像块之间的互信息。尽管如此，上述图像到图像转换方法能够进行纹理迁移，但受形状变化转换 [12] 的限制。

为解决这一问题，一些方法尝试通过直接生成图像像素来建模形状风格 [12, 17]。[12] 提出了一种采用空洞卷积的判别器 [58], 用于训练能够感知形状的生成器，从而实现全局形状变换。[17] 在比较学习策略中采用不同的嵌入网络来实现局部几何变换。[60] 提出了一种空间变换网络（STN） [24] ，并采用薄板样条

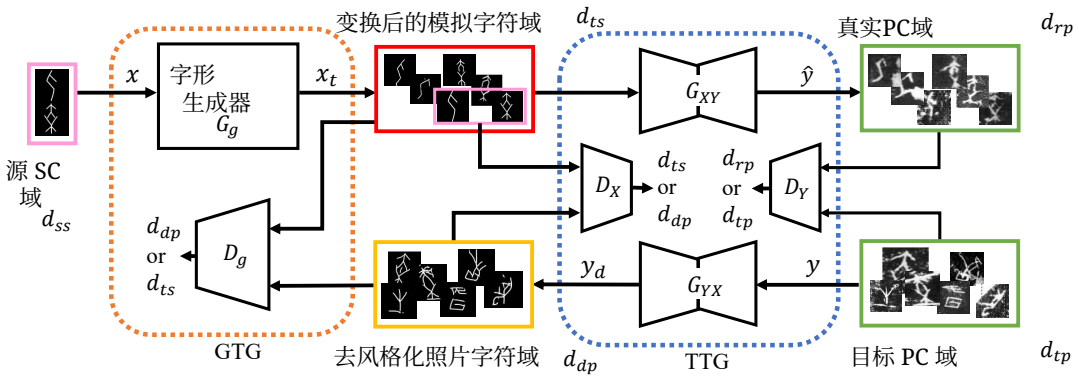


图 3：方法概述。给定源模拟字符，经过一对多字形形状变换生成变换后的模拟字符域。随后，通过纹理迁移生成真实影像字符。反向而言，给定目标影像字符，会生成去风格化照片字符域，以引导字形形状变换。

（TPS）变换对场景文本图像进行显式变换，并使用 CycleGAN 转移图像的纹理风格。然而，[60] 中连接 STN 和 CycleGAN 的判别器无法在纹理特征的干扰下辨别形状差异。这些方法只考虑了全局或局部形变。然而，生成古文字的字形不仅需要全局变换，也需要局部变换，因为字形的细节（如笔画和部首）各不相同。

3 提出的方法

我们的目标是学习从源模拟字符（SC）域 d_{ss}到目标照片字符（PC）域 d_{tp}的一对多映射。该映射应确保所有多模态 PC 输出均保留输入 SC 的字形类别，同时呈现丰富多样的字形形状和纹理风格。

我们不采用从 d_{ss}到 d_{tp}的单步映射，而是将其划分为两个阶段：字形形状映射和纹理映射。由此得到另外两个中间域：变换后的模拟字符域 d_{ts}和去风格化照片字符域 d_{dp}。此外，我们将最终生成的 PC 域定义为逼真的 PC 域 d_{rp}。图3展示了所有这些域及整体模型，其中 x、x_t、y_d、$\hat{y}$和 y分别表示从 d_{ss}、d_{ts}、d_{dp}、d_{rp}和 d_{tp}中采样的字符。

如图3所示，我们提出的模型称为 AGTGAN，由分别用于字形形状变换和纹理迁移的 GTG 与 TTG 组成。此外，我们引入关联对抗训练机制，以协同提升 GTG 和 TTG 的性能，使整个网络能够进行端到端训练。

3.1 字形变换生成对抗网络

字形变换生成对抗网络（GTG）由经过精心设计的字形形状生成器 G_g和卷积神经网络判别器 D_g组成，旨在生成多样且与 PC 字形形状相似的字形。

3.1.1 字形形状生成器。 根据我们的观察，直接预测字形形状来显式建模

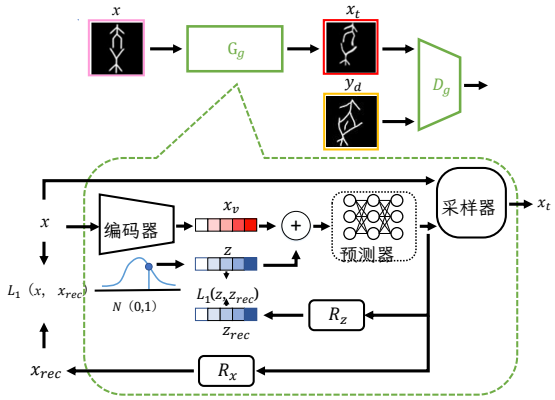


图 4：字形变换生成对抗网络的结构。

无需强监督，即可同时实现全局和局部形状变化。因此，我们使用空间变换网络（STN），通过预测的变形网格（例如仿射变换矩阵）对图像像素进行重采样，从而显式实现形状变化。如图 4 中绿色虚线框所示，我们的 G_g由一个 STN 组件 [24] 和两个重建网络 R_z 与 R_x组成，分别用于重建噪声和输入图像。STN 组件包括一个编码器 和一个预测器，二者配合用于估计仿射变换参数和 TPS 变换参数；此外还包括一个采样器，用于通过使用估计出的参数对输入 x 进行重采样来生成 x_t。不同于仅使用仿射变换或 TPS 变换 [24, 44], 的方法，我们的 G_g 结合了仿射变换和 TPS 变换，以便在全局和局部层面精细地模拟 PC 字形形状。具体而言，仿射变换会增强整体字形形状，产生全局形状变化，例如旋转、平移和缩放等；而 TPS 则会增强局部形状变化，例如笔画长度和扭曲。我们在附录中比较了同时使用 TPS 和仿射变换与其他情况的效果。

为实现字形变化，我们向 SC 特征中注入高斯噪声 z，从而获得多样化的输出。然而，如 [67], 中所述，生成对抗网络容易忽略添加的噪声，因而产生彼此相似的输出。为此，[67] 从输出中重建噪声向量，以保留噪声的影响。然而，他们忽略了一个事实：当噪声的影响远超输入特征时，噪声会降低生成图像的真实性。为避免这种情况，我们设计了两个重建网络 R_z 和 R_x，在训练期间分别根据相同的估计参数恢复注入的噪声 z 和输入 x。借助我们设计的 SNR 损失（参见"信号与噪声重建损失"一节），R_z 和 R_x使信号与噪声的影响在训练期间相互竞争，并最终达到平衡状态。有关 G_g 架构的更多详情，请参阅附录。

3.1.2 信号与噪声重建损失。如前所述，信号与噪声重建（SNR）损失旨在平衡输入信号和噪声的影响。信号重建损失确保转换后的SC保留原始字形结构，即生成的SC与输入应属于相同的字符类别；而噪声重建损失则保证输出多样性。

SNR损失包含三项。前两项分别对应输入信号重建误差和噪声重建误差，如下所示：

$$\mathrm{L}_1\left(x, x_{rec}\right)+\mathrm{L}_1\left(z, z_{rec}\right), \tag{1}$$

其中，x_{rec}和 z_{rec}分别表示重建后的信号和噪声，L_1 表示 L_1-距离。值得注意的是，如果上述两项未得到适当正则化，损失函数可能会被其中一项主导。这可能导致输出单调（多样性不足）或完全随机（毫无意义）。为了解决这一问题，我们引入一个称为重建误差比（RER）的平衡项，以平衡信号和噪声的影响，如下所示：

$$\mathrm{RER}=\log\left(\frac{\mathrm{L}_1\left(z, z_{rec}\right)}{\mathrm{L}_1\left(x, x_{rec}\right)}\right). \tag{2}$$

加入此 RER 项后，信噪比损失定义如下：

$$\mathrm{L}_{\mathrm{snr}}\left(G_g\right)=\mathrm{L}_1\left(x, x_{rec}\right)+\mathrm{L}_1\left(z, z_{rec}\right)+\alpha\cdot\mathrm{RER}, \tag{3}$$

其中，α是动态系数。我们进一步通过超参数 $M>1$对 RER 加以约束。在训练过程中，如果 RER > log M，则设定 $\alpha=1$。此时，噪声重建效果远差于信号重建效果。这意味着信号占据主导地位，因此我们使用正平衡项来惩罚较大的比率。相反，如果 RER < $-logM$，则设定 $\alpha=-1$。此时，信号重建效果远差于噪声重建效果。这意味着噪声占据主导地位，因此我们使用负平衡项来惩罚较小的比率。如果 RER 落在理想范围内，即 [–log M, log M], 则设定 $\alpha=0$，也就是不施加任何额外的惩罚。此时，信号和噪声的影响将保持平衡。

3.1.3 多样性损失。多样性与两个变换参数 $P(E(x)\ z_1)$ 和 $P(E(x)\ z_2)$ 之间的差异呈正相关。这两个参数分别由注入噪声 z_1 和 z_2 的两个混合信号向量估计得出，其中 z_1 和 z_2 是从同一高斯分布中随机抽取的。多样性损失定义如下：

$$\mathrm{L}_{\mathrm{div}}\left(E, P\right)=-\mathrm{L}_1\left(P\left(E(x), z_1\right), P\left(E(x), z_2\right)\right), \tag{4}$$

其中，E 和 P 分别表示 STN 中的编码器和预测器。

3.2 纹理迁移生成对抗网络

我们采用一种循环结构生成对抗网络 [66],，称为纹理迁移生成对抗网络（TTG），为 SC 添加纹理风格，即 x和 x_t。TTG 由两个生成器 G_{XY}和 G_{YX}，以及两个判别器 D_Y和 D_X组成（见图3中的蓝色虚线框）。为了增强 TTG 对印刷字符生成的适应性，我们提出了一种笔画感知循环一致性损失，以防止 PC 在纹理迁移过程中丢失笔画或变得模糊。

3.2.1 笔画感知循环一致性损失。与其他图像相比，字符图像中的字符通常占据较小区域，却包含几乎所有字形信息。这使得现有的循环结构网络不适用于字符生成，因为它们对待几乎所有像素的方式都相同。因此，输出会变得模糊，甚至出现字符不完整的问题。为了解决这一问题，我们提出一种笔画感知循环一致性损失，引导 TTG 更加关注前景字符，具体如下：

$$\begin{aligned}\mathrm{L}_{\mathrm{sacyc}}\left(G_{XY}, G_{YX}\right)=&\\ &\mathbb{E}_{x_t}\left[W\odot\mathrm{L}_1\left(G_{YX}\left(G_{XY}\left(\right.\right.\right.\right.\\ &\left.\left.\left.\left.x_t\right)\right), x_t\right)\right]\\ &+\mathbb{E}_y\left[\mathrm{L}_1\left(G_{XY}\left(G_{YX}\left(y\right)\right), y\right)\right],\end{aligned} \tag{5}$$

其中，$\odot$ 表示逐元素乘积，W 表示从 SC 中提取的权重矩阵，如下所示：

$$W_{ij}=\begin{cases}\frac{C}{S_{fg}}\cdot S_{bg}, x_t^{ij}\in foreground\\ 1, x_t^{ij}\in background\end{cases}, \tag{6}$$

其中，S_{fg} 和 S_{bg} 分别表示前景区域和背景区域的面积（以像素为单位）。常数参数 $C\geq 1$ 用于权衡前景清晰度与背景真实性。值得注意的是，笔画感知信息无需额外标注，通过图像二值化即可轻松获取。

3.3 关联对抗训练

我们提出一种关联对抗训练方法，通过两个过渡域 d_{ts} 和 d_{dp} 连接 GTG 与 TTG，以交换字形形状信息，并将其与纹理风格融合，如图3所示。通过这种方式，两个生成对抗网络联合训练，从而相互适应、相互促进。

为了实现从源 SC 域 d_{ss}到目标 PC 域 d_{tp}的字形形状映射，第一个生成对抗网络中的 G_g 尝试探索多样的字形，而 D_g 则尝试判别字形究竟来自变换后的模拟字符域d_{ts}，还是去风格化照片字符域 d_{dp}。d_{dp}由具有循环结构的第二个生成对抗网络生成，作为两个生成对抗网络之间的桥接域，并引导输入 SC 的字形变形。通过对抗学习，GTG 使 d_{ts}的分布接近 d_{dp}的分布。

我们使用过渡域 d_{dp}，而非目标 PC 域 d_{tp}，来引导字形变换，因为两者之间存在

d_{tp}和 d_{ss}之间存在一些纹理风格差异，这会干扰 GTG 的字形变换学习。通过训练 TTG 逐步去风格化后，可以得到 d_{dp}，其中保留了丰富的全局和局部字形形状模式。随着 d_{dp}的纹理风格逐渐减弱，GTG 会专注于字形形状的差异，并精准学习字形形状特征。

GTG 的损失函数由最小二乘生成对抗损失 [38],SNR 损失和多样性损失组合而成，如下所示：

$$\begin{aligned} \mathrm{L}_{\mathrm{G_g}} =& \mathbb{E}_x \left[D_g\left(G_g(x)\right)-1\right]^2 \\ &+\mathrm{L}_{\mathrm{snr}}\left(G_g\right)+\mathrm{L}_{\mathrm{div}}\left(E,P\right), \end{aligned} \tag{7}$$

$$\begin{aligned} \mathrm{L}_{\mathrm{D_g}} =& \frac{1}{2}\mathbb{E}_x \left[D_g\left(G_g\left(x\right)\right)\right]^2 \\ &+\frac{1}{2}\mathbb{E}_{y_d}\left[D_g\left(y_d\right)-1\right]^2. \end{aligned} \tag{8}$$

为实现从源 SC 领域 d_{ss} 到目标 PC 域 d_{tp}的风格映射， G_{XY} 以 d_{ts}中的 SC 作为输入，并生成真实的 PC，以欺骗相应的 D_Y。 D_Y试图区分 PC 样本所属的域，即真实PC域 d_{rp}或目标 PC 域 d_{tp}。

我们将 d_{ts} 而非 d_{ss} 用作 TTG 的输入域，因为在学习迁移纹理风格时， d_{ss}与 d_{tp}之间的字形差异很容易使 TTG 混淆。在 GTG 训练过程中， d_{ts}样本的字形会逐渐接近 d_{dp}的字形。这样，TTG 就能专注于纹理风格差异，并更准确地捕捉纹理特征。GTG 通过 d_{ts}向 TTG 传递字形形状信息，从而促使 G_{XY} 生成具有逼真字形和纹理风格的 PC。

TTG 的损失函数由最小二乘生成对抗损失和笔画感知循环一致性损失组合而成，如下所示：

$$\begin{aligned} \mathrm{L}_{\mathrm{G_{XY}},\mathrm{G_{YX}}} =& \mathbb{E}_{x_t}\left[D_Y\left(G_{XY}(x_t)\right)-1\right]^2 \\ &+\mathbb{E}_y\left[D_X\left(G_{YX}(y)\right)-1\right]^2 \\ &+\lambda \mathrm{L}_{\mathrm{sacyc}}\left(G_{XY},G_{YX}\right), \end{aligned} \tag{9}$$

$$\begin{aligned} \mathrm{L}_{\mathrm{D_Y},\mathrm{D_X}} =& \frac{1}{2}\mathbb{E}_{x_t}\left[D_Y\left(G_{XY}\left(x_t\right)\right)\right]^2 \\ &+\frac{1}{2}\mathbb{E}_y\left[D_Y\left(y\right)-1\right]^2 \\ &+\frac{1}{2}\mathbb{E}_y\left[D_X\left(G_{YX}\left(y\right)\right)\right]^2 \\ &+\frac{1}{2}\mathbb{E}_x\left[D_X\left(x\right)-1\right]^2, \end{aligned} \tag{10}$$

其中，λ是一个超参数，用于控制笔画感知循环一致性损失的相对重要性。

通过为 GTG 和 TTG 提出的关联对抗训练，两者可以作为统一流程（即 AGTGAN）和谐地共同训练。

4 实验

4.1 数据集

模拟字符5519 [33] 包含来自 5,491 个类别的 44,868 个干净的模拟甲骨文字（SOC）实例，几乎涵盖了所有已发现的类别。

OBC306 [19] 包含来自306个字符类别的309,511个样本，可从开源的OBI数据库获取。值得注意的是，SOC5519中的大多数字符类别在OBC306中都不可用因为OBC306中的类别数量远少于SOC5519中的类别数量。

HCCC [56] 是一个图像集，由现存楔形文字泥板手抄本手动模拟并整理而成。该数据集包含 50 个类别的 4,416 个样本，这些类别在多个语料库中最常出现。

CSDD [7] 为 81 张泥板图像上的符号提供边界框标注和类别标签。我们根据这些边界框标注，对 233 个类别的图像进行了分割，共得到 2,576 张摄影楔形文字字符（PCC）图像。

4.2 训练细节

对于所有实验，我们设置 $\lambda = 10$、 $C = 2$和 $M = 6$。所有模拟字符图像和摄影图像均调整为 64×64。我们将 GTG 的初始学习率设为 0.0001，将 TTG 的初始学习率设为 0.001，并使用 Adam 优化器 [29]，以批量大小 64 进行优化。前 15,000 次迭代中，学习率保持不变；随后在接下来的 15,000 次迭代中，将学习率线性衰减至零。

4.3 定量评估指标与用户研究

FID. 我们使用 Fréchet 初始特征距离（FID） [18]，通过计算生成分布与真实分布之间基于 Inception 网络 [48]提取特征的距离，来评估生成图像的真实性。FID 越低，生成图像的质量越好。

NDB and JSD. 我们使用统计差异箱数（NDB）和 Jensen-Shannon 散度（JSD） [42] 来评估生成图像的真实性。与 FID 相比，NDB 和 JSD 指标直接作用于图像像素，不依赖学习到的表征。因此，这些指标对图像中的像素级差异更加敏感。我们将 NDB 的箱数设为 50。

LPIPS. 我们使用学习感知图像块相似度（LPIPS）指标 [62] 来衡量生成图像的多样性。我们为每个类别生成 1,000 个样本，并计算样本对之间的 LPIPS 距离。所有类别的 LPIPS 距离平均值作为最终的 LPIPS 值。

用户研究。 我们基于成对比较开展了一项用户研究。给定 AGTGAN 和其他模型生成的 POC 图像组，每位受试者都需要以真实的 POC 图像为参照，回答“哪个 POC 图像组更逼真？”

4.4 与最先进方法的比较

我们从视觉质量、定量指标、用户研究和分类性能等方面，将所提出的 AGTGAN 与最先进的无监督图像到图像模型进行比较，包括 CycleGAN [66], DRIT++ [30]、NICE-GAN [20], DCLGAN [17], 和 DG-Font [53], 。

4.4.1 **生成结果。**图5展示了随机选取的字符类别所生成的一些照片级古代字符。可以看出，我们的方法生成的字形具有显著的多样性：笔画和部首呈现出丰富的局部变化，整个字符的尺寸和倾斜程度也各不相同，

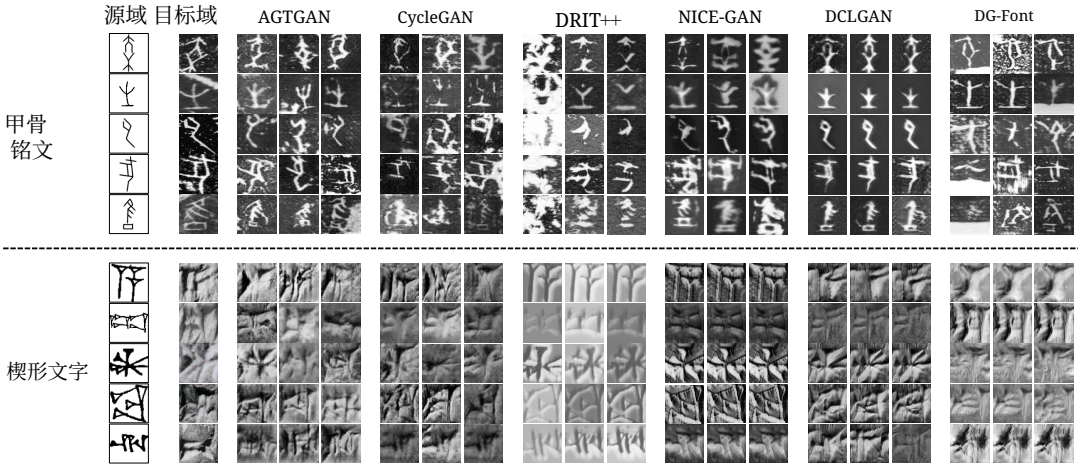


图5：生成的POC和PCC图像。第一列为源SC图像，第二列为目标PC图像，其余各列由不同方法生成。

表1：生成的POC定量评估。

方法	FID↓	NDB↓	JSD↓	LPIPS↑
DCLGAN	176.46	41	0.178	0.234
DG-Font	168.93	39	0.123	0.224
NICE-GAN	140.22	41	0.104	**0.325**
CycleGAN	130.76	35	0.135	0.252
DRIT++	117.03	37	0.083	0.299
AGTGAN	**99.48**	**26**	**0.066**	0.284

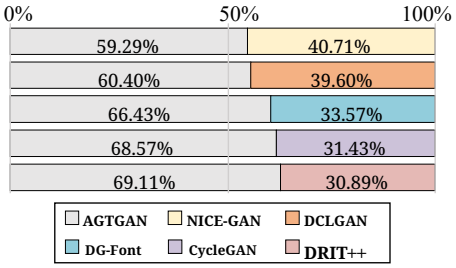


图 6：通过人类评估生成的 POC 的真实性。数字表示对比样本对的偏好百分比。

同时保留了原始字形标签。在纹理风格方面，我们生成的 POC 呈现出更自然的背景噪声，而生成的 PCC 则具有更好的立体效果。此外，我们的方法生成的甲骨文和楔形文字均不存在模糊问题。

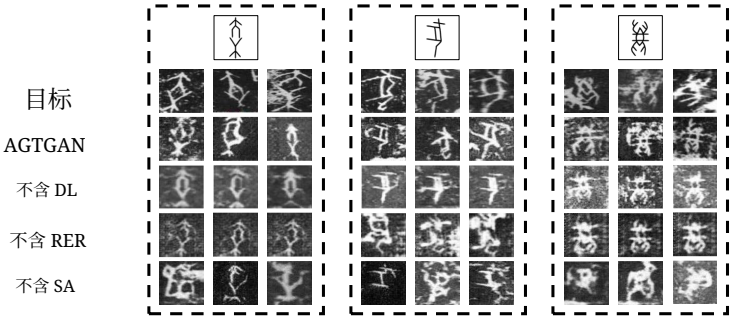
相比之下，CycleGAN 会生成一些不合理的字形，例如缺少某些笔画或多出错误的笔画。这主要是因为迁移后的背景纹理会干扰字符笔画，导致生成的字形出现模糊。DG-Font、DRIT++ 和 NICE-GAN 生成的结果更差，例如字形难以辨认，图像带有伪影或雾化效果。一个可能的

表 2：不同方法所达到的平均类别准确率。“仅源数据”指在不使用生成的增量图像的情况下训练识别器。

方法	TOP-1(%)		
	OBC306	FS	ZS
仅源域	69.02	5.36	0
DG-Font	75.11	38.87	27.45
CycleGAN	78.07	54.46	48.28
DCLGAN	80.77	58.33	54.90
DRIT++	81.04	57.14	51.72
NICE-GAN	83.62	68.75	79.31
AGTGAN（我们的方法）	**85.36**	**81.25**	**93.10**

原因是它们依赖内容和风格的解耦潜向量，而复杂字形和纹理的摄影字符难以获得这类向量。尽管 DCLGAN 能生成清晰的字形，但生成样本的背景纹理十分单调，与目标纹理风格相差甚远。这是因为所采用的对比学习策略容易捕捉字形的语义信息，却忽略了大部分背景细节。此外，许多生成样本（例如“DRIT++”、“NICE-GAN”和“DCLGAN”下方第二行和第三行的样本）仍与源字符相似，字形变化较少，这表明基线方法难以学习目标中的字形多样性。尽管 DG-Font 使用可变形卷积来学习形状风格，但只能产生轻微的局部变形。值得一提的是，我们在附录中展示了本方法在零样本生成方面的巨大潜力。

表 A2 总结了定量结果，得出的结论与上述视觉分析类似。与所有方法相比，我们的方法取得了最佳的 FID、NDB 和 JSD 分数。尽管 DRIT++ 和 NICE-GAN 的 LPIPS 分数高于我们的方法，但这种高多样性源于混乱的纹理和



LPIPS↑	FID↓	NDB↓	JSD↓
0.284	**99.48**	**26**	**0.066**
0.262	111.50	32	0.074
0.297	105.77	33	0.089
0.301	106.61	31	0.069

图7：AGTGAN各组成部分的消融研究。我们展示了源SOCs（第1行）、目标POC（第2行）、AGTGAN生成的样本（第3行），以及移除DL（第4行）、移除RER（第5行）和移除SA（第6行）后AGTGAN生成的样本。

不完整的字形。值得注意的是，HCCC 和 CSDD 数据集的分类标准并不一致，因此在应用 FID 指标时，无法将参照 HCCC 生成的样本与 CSDD 的真实数据进行评估。应用于生成楔形文字的 LPIPS、NDB 和 JSD 结果也表明，我们的方法优于其他对比方法，详情见附录。

图6中的用户研究结果表明，与其他方法相比，更多人认为我们的AGTGAN生成的POC更接近目标POC。

4.4.2 分类性能。 我们进一步通过在 OBC306 上开展分类实验，评估生成样本的质量。按照与 [19]，相同的方案，我们随机选取四分之一的样本用于测试，同时确保每个类别至少有一个测试样本，其余样本则构成真实的 POC 训练集。更多设置详情请参见附录。我们还测量了少数类的分类性能，以进一步研究生成的 POC 的效果。包含 1 至 10 个样本的类别构成少样本（FS）子集，而不包含任何真实训练样本的类别则构成零样本（ZS）子集。

表2汇总了 TOP-1 平均类别准确率。TOP-3 和 TOP-5 准确率列于附录中。表2显示，使用 AGTGAN 生成的样本后，OBC306、FS 和 ZS 上均可取得最高的 POC 分类准确率。与表2中的“仅源数据”相比，使用我们生成的样本至少提升了 16.34%。使用 AGTGAN 生成的训练数据后，FS 上的分类准确率为 81.25%，ZS 上则为 93.10%。结果表明，我们的方法能显著提高少数类的识别准确率。

4.5 消融研究

图7总结了用于评估 AGTGAN 不同部分影响的消融研究。

多样性损失（w/oDL）：我们的模型在不使用多样性损失的情况下，生成的 POC 图像中的字形图案单调，且几乎与 SOC 字形对齐。因此，生成的 POC 的多样性并不理想。这也可以从其在所有对比方法中最低的 LPIPS 值（即 0.262）看出。

重建误差比（无 RER）：从图7中“无 RER”这一行可以看到两种截然不同的生成字形。一种几乎没有变换，例如第一个和第三个字形类别；另一种则存在较大扭曲，超出了合理的字形空间，例如第二个字形类别。这是因为如果不施加 RER 来保持平衡，优化过程中信号与噪声之间的平衡可能会被打破。这可能导致噪声变得很小，进而被网络滤除；也可能导致噪声变得很大并占据信号的主导地位。

笔画感知循环一致性损失（无 SA）：图7中“无 SA”这一行显示，生成样本存在笔画缺失，或背景纹理被错误地添加到前景字符上。这会大幅降低生成质量。生成样本的 LPIPS 达到 0.301 的高值，这主要是由混乱字形而非合理变化导致的。

5 结论

我们提出了一种新颖的字符生成模型 AGTGAN。据我们所知，这是首个能够自动生成丰富且逼真的古代摄影文字的方法。因此，这些生成图像在一定程度上解决了古代摄影文字自动分类任务因缺乏高质量标注数据而面临的最关键问题。

未来工作的一个自然方向是将我们提出的方法扩展到更通用、更复杂的书写形式，例如手写文本和公式。此外，创意字体生成任务（例如字体设计和书法仿写）也是该方法的其他潜在应用。

6 致谢

本研究部分得到国家自然科学基金委员会（编号：62176093、61673182、61936003）、广州市重点领域研发计划（编号：202206030001）、广东省基础与应用基础研究基金（编号：2021A1515012282）、GD-NSF（编号：2017A030312006）以及阿里巴巴创新研究项目的资助。

参考文献

[1] Yannis M. Assael、Thea Sommerschield 和 Jonathan Prag。2019。利用深度学习修复古代文本：希腊铭文学案例研究。发表于 EMNLP-IJCNLP。6367–6374。[2] Samaneh Azadi、Matthew Fisher、Vladimir G Kim、Zhaowen Wang、Eli Shechtman 和 Trevor Darrell。2018。用于小样本字体风格迁移的多内容生成对抗网络。发表于 CVPR。7564–7573。[3] Ankan Kumar Bhunia、Salman Khan、Hisham Cholakkal、Rao Muhammad Anwer、Fahad Shahbaz Khan 和 Mubarak Shah。2021。手写文字变换器。发表于 IEEE/CVF国际计算机视觉大会论文集。1086–1094。[4] EA Wallis Budge。2012。《亡灵书》象形文字词汇。Courier Corporation。[5] Jiezhang Cao、Langyuan Mo、Yifan Zhang、Kui Jia、Chunhua Shen 和 Mingkui Tan。2019。多边际 Wasserstein生成对抗网络。神经信息处理系统进展 32（2019）。[6] Jifeng Dai、Haozhi Qi、Yuwen Xiong、Yi Li、Guodong Zhang、Han Hu 和 Yichen Wei。2017。可变形卷积网络。发表于 IEEE 国际计算机视觉大会论文集。764–773。[7] Tobias Dencker、Pablo Klinkisch、Stefan M Maul 和 Björn Ommer。2020。利用转写对齐进行弱监督的楔形文字符号检测深度学习。Plos one 15，12（2020），e0243039。[8] Sharon Fogel、Hadar Averbuch-Elor、Sarel Cohen、Shai Mazor 和 Roee Litman。2020。ScrabbleGAN：半监督可变长度手写文本生成。发表于 CVPR。4324–4333。[9] Yue Gao、Yuan Guo、Zhouhui Lian、Yingmin Tang 和 Jianguo Xiao。2019。通过单阶段小样本学习合成艺术字形图像。ACM TOG 38，6（2019），1–12。[10] Yiming Gao 和 Jiangqin Wu。2020。基于GAN的无配对汉字图像转换：通过骨架变换与笔画渲染实现。发表于 AAAI，第34卷。646–653。[11] William Gates。1978。《玛雅字形纲要词典：附对照表及其关系分析》，并收录作者发表于《玛雅学会季刊》的“字形研究”一文。Courier Corporation。[12] Aaron Gokaslan、Vivek Ramanujan、Daniel Ritchie、Kwang In Kim 和 James Tompkin。2018。改进无监督图像到图像迁移中的形状变形。发表于 ECCV。649–665。[13] Ian J. Goodfellow、Jean Pouget-Abadie、Mehdi Mirza、Bing Xu、David Warde-Farley、Sherjil Ozair、Aaron Courville 和 Yoshua Bengio。2014。生成对抗网络。发表于 NeurIPS，第27卷。[14] Alex Graves。2013。利用循环神经网络生成序列。arXiv预印本 arXiv:1308.0850（2013）。[15] Ting Guo、Xingquan Zhu、Yang Wang 和 Fang Chen。2019。用于深度不平衡学习的判别式样本生成。发表于 IJCAI。[16] Arwa Hamed Salih Hamdany、Raid Rafi Omar Al-Nima 和 Lubab H Albak。2021。利用人工神经网络转换楔形文字符号。Telkomnika 19，2（2021），438–443。[17] Junlin Han、Mehrdad Shoeiby、Lars Petersson 和 Mohammad Ali Armin。2021。用于无监督图像到图像迁移的双重对比学习。发表于 CVPR。746–755。[18] Martin Heusel、Hubert Ramsauer、Thomas Unterthiner、Bernhard Nessler 和 Sepp Hochreiter。2017。采用双时间尺度更新规则训练的生成对抗网络收敛于局部纳什均衡。发表于 NeurIPS，第30卷。[19] 黄双平、Haobin Wang、刘永革、Xiaosong Shi 和 Lianwen Jin。2019。OBC306：大规模甲骨文字符识别数据集。发表于 ICDAR。681–688。[20] Siyu Huang、Haoyi Xiong、Zhi-Qi Cheng、Qingzhong Wang、Xingran Zhou、Bihan Wen、Jun Huan 和 Dejing Dou。2020。利用外观感知姿态风格化器生成的人物图像。发表于 IJCAI。623–629。[21] Xun Huang、Ming-Yu Liu、Serge Belongie 和 Jan Kautz。2018。多模态无监督图像到图像迁移。发表于 ECCV。172–189。[22] Yaoxiong Huang、何梦超、Lianwen Jin 和 Yongpan Wang。2020。RD-GAN：通过部首分解与渲染实现小样本/零样本汉字风格迁移。发表于 ECCV。156–172。[23] Phillip Isola、Jun-Yan Zhu、Tinghui Zhou 和 Alexei A Efros。2017。利用条件生成对抗网络进行图像到图像转换。发表于 CVPR。1125–1134。[24] Max Jaderberg、Karen Simonyan、Andrew Zisserman 等。2015。空间变换网络。发表于 NeurIPS，第28卷。2017–2025。[25] Bo Ji 和 Tianyi Chen。2019。用于手写文本的生成对抗网络。arXiv预印本 arXiv:1907.11845（2019）。[26] Yue Jiang、Zhouhui Lian、Yingmin Tang 和 Jianguo Xiao。2017。DCFont：端到端深度中文字体生成系统。发表于 SIGGRAPH Asia 2017 技术简报。1–4。[27] Yue Jiang、Zhouhui Lian、Yingmin Tang 和 Jianguo Xiao。2019。SCFont：通过深度堆叠网络实现结构引导的中文字体生成。发表于 AAAI 人工智能大会论文集，第33卷。4015–4022。[28] Lei Kang、Pau Riba、Yaxing Wang、Marçal Rusiñol、Alicia Fornés 和 Mauricio Villegas。2020。GANwriting：内容条件生成风格化手写文本图像。发表于欧洲计算机视觉会议。Springer，273–289。[29] Diederik P Kingma 和 Jimmy Ba。2014。Adam：一种随机优化方法。arXiv 预印本 arXiv:1412.6980（2014）。[30] Hsin-Ying Lee、Hung-Yu Tseng、Qi Mao、Jia-Bin Huang、Yu-Ding Lu、Maneesh Singh 和 Ming-Hsuan Yang。2020。Drit++：通过解耦表示实现多样化的图像到图像转换。国际计算机视觉杂志 128，10（2020），2402–2417。[31] Jianxin Lin、Yingce Xia、Yijun Wang、Tao Qin 和 Zhibo Chen。2019。基于多路径一致性正则化的图像到图像转换。载于 IJCAI。2980–2986。[32] Zhouchen Lin 和 Liang Wan。2007。风格保持的英文手写体合成。模式识别 40，7（2007），2097–2109。[33] Guoying Liu 和 Feng Gao。2018。基于深度卷积神经网络的甲骨文识别。《计算机学报》13（2018），1442–1450。[34] Zhiqiang Liu、Chengkai Huang 和 Yanxia Liu。2021。通过对抗协作改进知识蒸馏。arXiv 预印本 arXiv:2111.14356（2021）。[35] Zhuoman Liu、Wei Jia、Ming Yang、Peiyao Luo、Yong Guo 和 Mingkui Tan。2021。通过自一致生成网络实现深度视图合成。《IEEE多媒体汇刊》（2021）。[36] Zhiqiang Liu、Yanxia Liu 和 Chengkai Huang。2021。半在线知识蒸馏。载于英国机器视觉会议。英国机器视觉协会出版社，33。[37] Canjie Luo、Yuanzhi Zhu、Lianwen Jin 和 Yongpan Wang。2020。学习增强：面向文本识别的联合数据增强与网络优化。载于 IEEE/CVF计算机视觉与模式识别会议论文集。13746–13755。[38] Xudong Mao、Qing Li、Haoran Xie、Raymond YK Lau、Zhen Wang 和 Stephen Paul Smolley。2017。最小二乘生成对抗网络。载于 ICCV。2794–2802。[39] Michail Panagopoulos、Constantin Papaodysseus、Panayiotis Rousopoulos、Dimitra Dafi 和 Stephen Tracy。2008。古希腊铭文的自动书写者识别。IEEE模式分析与机器智能汇刊 31，8（2008），1404–1414。[40] Taesung Park、Alexei A Efros、Richard Zhang 和 Jun-Yan Zhu。2020。用于非配对图像到图像转换的对比学习。载于欧洲计算机视觉会议。施普林格，319–345。[41] Elad Richardson、Yuval Alaluf、Or Patashnik、Yotam Nitzan、Yaniv Azar、Stav Shapiro 和 Daniel Cohen-Or。2021。以风格编码：用于图像到图像转换的 StyleGAN 编码器。载于 CVPR。2287–2296。[42] Eitan Richardson 和 Yair Weiss。2018。论生成对抗网络与高斯混合模型。arXiv 预印本 arXiv:1805.12462（2018）。[43] Eugen Rusakov、Kai Brandenbusch、Denis Fisseler、Turna Somel、Gernot A Fink、Frank Weichert 和 Gerfrid GW Müller。2019。利用循环一致对抗网络生成楔形文字符号。载于第五届历史文档成像与处理国际研讨会论文集。19–24。[44] Baoguang Shi、Xinggang Wang、Pengyuan Lyu、Cong Yao 和 Xiang Bai。2016。利用自动校正实现鲁棒的场景文本识别。载于 CVPR。4168–4176。[45] Thea Sommerschield。2020。Ralegh Radford Rome奖：利用机器学习修复古代文本：希腊与拉丁铭文学案例研究。《罗马英国学院论文集》88（2020），387–388。[46] Sungho Suh、Paul Lukowicz 和 Yong Oh Lee。2022。用于不平衡数据分类的判别性特征生成。模式识别 122（2022），108302。[47] Danyang Sun、Tongzheng Ren、Chongxun Li、Hang Su 和 Jun Zhu。2017。通过阅读少量示例学习书写风格化汉字。arXiv 预印本 arXiv:1712.06424（2017）。[48] Christian Szegedy、Sergey Ioffe、Vincent Vanhoucke 和 Alexander A. Alemi。2017。Inception-v4、Inception-ResNet 以及残差连接对学习的影响。载于 AAAI。4278–4284。[49] Yuchen Tian。2017。zi2zi：利用条件生成对抗网络精通中国书法。载于 https://kaonashi-tyc.github.io/2017/04/06/zi2zi.html。[50] Jue Wang、Chenyu Wu、Ying-Qing Xu、Heung-Yeung Shum 和 Liang Ji。2002。基于学习的草书手写体合成。载于第八届手写识别前沿国际研讨会论文集。157–162。[51] Qi Wen、Shuang Li、Bingfeng Han 和 Yi Yuan。2021。ZiGAN：通过小样本风格迁移方法实现精细化中国书法字体生成。载于第二十九届ACM多媒体国际会议论文集。621–629。[52] Shan-Jean Wu、Chih-Yuan Yang 和 Jane Yung-jen Hsu。2020。CalliGAN：具备风格与结构感知能力的中国书法字符生成器。arXiv 预印本 arXiv:2005.12500 （ 2020 ）。[53] Yangchen Xie 、 Xinyuan Chen 、 Li Sun 和 Yue Lu。2021。DG-Font：用于无监督字体生成的可变形生成网络。载于 CVPR。5130–5140。[54] Haoming Xu、Runhao Zeng、Qingyao Wu、Mingkui Tan 和 Chuang Gan。2020。用于视听事件定位的跨模态关系感知网络。载于第二十八届ACM多媒体国际会议论文集。3893–3901。[55] Zhongshu Xu。2021。《甲骨文字典》。四川辞书出版社。[56] Kenji Yamauchi、Hajime Yamamoto 和 Wakaha Mori。2018。构建手写楔形文字字符图像集。载于语言资源与评估会议。[57] Shuai Yang、Jiaying Liu、Wenjing Wang 和 Zongming Guo。2019。TET-GAN：通过风格化与去风格化实现文本效果迁移。载于 AAAI人工智能会议论文集，第33卷。1238–1245。

[58] Fisher Yu 和 Vladlen Koltun。2015。通过空洞卷积实现多尺度上下文聚合。*arXiv* 预印本 *arXiv:1511.07122*（2015）。[59] Jinshan Zeng、Qi Chen、Yunxin Liu、Mingwen Wang 和 Yuan Yao。2021。Strokegan：通过笔画编码减少中文字体生成中的模式崩溃。载于 *AAAI*论文集，第3卷。[60] Fangneng Zhan、Hongyuan Zhu 和 Shijian Lu。2019。用于图像合成的空间融合生成对抗网络。载于 *CVPR*。3653–3662。[61] Chongsheng Zhang、Ruixing Zong、Shuang Cao、Yi Men 和 Bofeng Mo。2020。人工智能驱动的甲骨文识别与残片拼合。载于 *IJCAI*。5309–5311。[62] Richard Zhang、Phillip Isola、Alexei A Efros、Eli Shechtman 和 Oliver Wang。2018。深层特征作为感知度量具有非凡效能。载于 *CVPR*。586–595。[63] Yifan Zhang、Hanbo Chen、Ying Wei、Peilin Zhao、Jiezhang Cao、Xinjuan Fan、Xiaoying Lou、Hailing Liu、Jinlong Hou、Xiao Han 等。2019。从整张幻灯片成像到显微成像：用于组织病理学癌症图像分类的深度显微成像适应网络。载于医学图像计算与计算机辅助干预国际会议。360–368。[64] Yifan Zhang、Ying Wei、Qingyao Wu、Peilin Zhao、Shuaicheng Niu、Junzhou Huang 和 Mingkui Tan。2020。用于医学图像诊断的协同无监督域自适应。*IEEE*图像处理汇刊 29（2020），7834–7844。[65] Yexun Zhang、Ya Zhang 和 Wenbin Cai。2018。分离风格与内容以实现广义风格迁移。载于*CVPR*。8447–8455。[66] Jun-Yan Zhu、Taesung Park、Phillip Isola 和 Alexei A Efros。2017。使用循环一致对抗网络进行非配对图像到图像转换。载于*ICCV*。2223–2232。[67] Jun-Yan Zhu、Richard Zhang、Deepak Pathak、Trevor Darrell、Alexei A Efros、Oliver Wang 和 Eli Shechtman。2017。通过强制双循环一致性实现多模态图像到图像转换。载于*NeurIPS*。465–476。

A 附录

表 A1：GTG组件的架构。“conv”表示卷积层，后接通道数。“convT”表示转置卷积层，后接通道数。“BN”和“MP”分别表示批归一化层和最大池化层。加上用于TPS网格变换的 N^2 个坐标，以及仿射变换的4个参数（包括旋转、缩放和平移），共得到 $2N^2+4$ 个需要学习的参数。

编码器 &预测器	R_x & R_z
输入 $1\times64\times64$	输入 $1\times(2N^2+4)$
64通道卷积层,批归一化,ReLU,最大池化	全连接层-($2N^2$+4),批归一化,ReLU
卷积-128 ,BN,ReLU,MP	1024维全连接层 ,BN,ReLU
64通道卷积层,批归一化,ReLU,最大池化	1024维全连接层,批归一化,ReLU
16通道卷积层 ,BN,ReLU,MP	1024维全连接层
重塑为 1×1024	重塑为 $64\times4\times4$
1024维全连接层 ,BN,ReLU	转置卷积-64 ,BN,ReLU,MP
1024维全连接层,批归一化,ReLU	转置卷积-128,批归一化,ReLU,最大池化
1024维全连接层 ,BN,ReLU	转置卷积-64 ,BN,ReLU,MP
全连接层-($2N^2$+4)	转置卷积-1，批归一化，ReLU，最大池化
输出 $1\times(2N^2+4)$	输出 $1\times64\times64$



(i) (ii)

图 A1：（i）我们的 TPS + 仿射变换和（ii）仅使用 TPS 变换。

表 A2：生成的楔形文字照片字符的定量评估。

方法	NDB↓	JSD↓	LPIPS↑
DCLGAN [17]	16	0.342	0.220
DRIT++ [30]	12	0.446	0.214
NICE-GAN [20]	12	0.310	0.223
CycleGAN [66]	11	0.174	0.231
DG-Font [53]	10	0.156	0.188
AGTGAN（我们的方法）	**4**	**0.074**	**0.253**

A.1 架构细节

字形形状生成器 G_g 包含一个空间变换网络 (STN) [24] 组件和两个重建网络， R_x 和 R_z。其中，STN 用于预测薄板样条 (TPS) 网格坐标和仿射变换参数。 R_x 和 R_z 分别用于还原输入字形和噪声。

STN 的编码器、预测器 以及 R_x & R_z 的详细信息见表 A1。第一列中的四个卷积层构成编码器，而重塑操作之后的另外四个全连接层构成预测器。我们使用编码器 & 预测器的逆向结构构建信号重建网络 R_x，而噪声重建网络 R_z 的结构与R_x中的全连接层部分相同。

A.2 仿射变换与 TPS 变换组合的效果

仿射变换可以产生全局变形，例如字符的旋转、平移和缩放；而 TPS 变换则倾向于造成局部笔画修改。图 A1 展示了仿射变换与 TPS 变换的组合能够同时产生全局和局部形状变形，而单独使用 TPS 变换则缺乏全局形状变形。

A.3 未见字符类别的生成 POC

收集成对的古文字数据成本高昂，甚至不切实际。得益于无监督学习，我们的方法能够生成训练期间未见过的样本，这对于解决数据稀缺问题具有重要意义。图 A2 展示了我们的 AGTGAN 与其他模型的对比结果。我们随机选取了 OBC306 中未出现的九种字符类别进行比较。可以看到，即使没有真实的 POC 样本可用于训练，我们的 AGTGAN 仍能生成逼真且多样的甲骨文照片字符（POC）。其他模型生成的 POC 则或多或少存在缺陷，例如字形变化不够丰富、笔画模糊、雾效、字形混乱等。

A.4 生成 PCC 的定量评估

表 A2 总结了生成的楔形文字照片字符（PCC）的定量结果。在 NDB 和 JSD 方面，与其他方法相比，我们的方法表现更优，且与次优方法相比优势显著。此外，在衡量生成样本多样性的 LPIPS 指标上，我们的方法也取得了最佳结果。

A.5 分类设置详情

分类性能通过测试集所有类别的平均类别准确率来衡量，而非所有样本的准确率。否则，整体准确率将受到多数类的主导。使用平均类别准确率能够平等地反映每个字符类别的准确率，包括少数类。对于训练样本少于 750 个的类别，我们使用对比方法生成新的 POC，以确保真实 POC 与生成的 POC 合并后，每个类别至少有 750 个样本。对于样本数超过 750 的类别，则不使用额外的生成图像来训练分类器。值得注意的是，750 是 OBC306 各类别的平均样本数。根据 [19], ，我们选择性能最佳的 Inception-v4 作为甲骨文照片字符识别器的骨干网络。



图 A2：未见类别的生成的POC图像。

表 A3：不同方法获得的类别平均准确率。“仅源域”表示训练识别器时不使用任何生成图像。

方法	TOP-3(%)			TOP-5(%)		
	OBC306	FS	ZS	OBC306	FS	ZS
仅源域	78.78	8.93	0	80.78	12.50	0
DG-Font	84.53	50.85	37.25	87.42	58.40	47.06
CycleGAN	88.98	67.86	72.41	91.94	76.79	82.76
DRIT++	90.24	71.43	75.86	92.44	75.00	82.76
DCLGAN	91.31	77.47	74.51	93.21	81.30	78.43
NICE-GAN	92.77	82.14	96.55	94.17	83.93	96.55
AGTGAN（我们的方法）	**93.13**	**85.71**	**96.55**	**94.61**	**87.50**	**96.55**

A.6 TOP-K 分类性能

TOP-3 和 TOP-5 平均类别准确率列于表 A3 中。在训练分类器时加入生成样本后，我们取得了最佳的 TOP-3 和 TOP-5 性能，平均类别准确率分别达到 93.13% 和 94.61%。这些结果显著优于所有其他模型和“仅源域”，证明了我们方法的优越性方法。此外，我们可以看到，在零样本设置下，AGTGAN 和 NICE-GAN 的 TOP-3 和 TOP-5 准确率均达到 96.55%。这是因为 OBC306 的 29 个零样本类别中，每个类别只有一个测试样本，而这两种方法增强了其中 28 个类别的预测性能，使其 TOP-3 和 TOP-5 分类准确率均达到 100%。因此，零样本准确率可按如下方式计算：

$$\frac{28 \times 100\%}{29} = 96.55\%.$$