中图分类号：　　单位代号：10280
密　　级：　　学　　号：22721473

上海大学

专业硕士学位论文

SHANGHAI UNIVERSITY
PROFESSIONAL MASTER'S DISSERTATION

题目	基于字形调制的鲁棒文档水印技术研究

作　　者：何承桦
学科专业：电子信息
导　　师：张新鹏
完成日期：2025 年 5 月

姓　　名：何承桦　　学号：22721473

论文题目：基于字形调制的鲁棒文档水印技术研究

上海大学

本论文经答辩委员会全体委员审查，确认符合上海大学硕士学位论文质量要求。

答辩委员会签名：

主　　席：

委　　员：

导　　师：

答辩日期：　　年　　月　　日

姓　　名：何承桦　　学号：22721473

论文题目：基于字形调制的鲁棒文档水印技术研究

上海大学学位论文原创性声明

本人郑重声明：所呈交的学位论文是本人在导师指导下，独立进行研究工作所取得的成果。除了文中特别加以标注和致谢的内容外，论文中不包含其他人已发表或撰写过的研究成果。参与同一工作的其他研究者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。

学位论文作者签名：

日 期：　　年　　月　　日

上海大学学位论文使用授权说明

本人完全了解上海大学有关保留、使用学位论文的规定，即：学校有权保留论文及送交论文复印件，允许论文被查阅和借阅；学校可以公布论文的全部或部分内容。

（**保密论文在解密后应遵守此规定**）

学位论文作者签名：　　**导师签名：**

日 期：　　年　　月　　日　　日 期：　　年　　月　　日

上海大学工程硕士学位论文

基于字形调制的鲁棒文档水印技术研究

作　　者：何承桦

学科专业：电子信息

导　　师：张新鹏

上海大学通信与信息工程学院

二 O 二五年五月

A Dissertation Submitted to Shanghai University for the Degree of Master in Engineering

Robust Text Document Watermarking Technology Based on Glyph Modification

Candidate： **Chenghua He**

Major： **Electronic Information**

Supervisor： **Xinpeng Zhang**

School of Communication and Information Engineering

Shanghai University

May, 2025

摘 要

随着信息数字化进程的持续推进，电子文档在开放网络环境中的使用与流通也变得更加频繁，如何应对电子文档的非法传播、篡改和贩卖问题成为倍受关注的热点。文档水印通过在文档中嵌入秘密信息来证明该文档的归属，具有保护和追溯文档版权的功能。过往的文档水印方法多数基于对排版、背景格式或语义的调制，难以兼顾水印的隐蔽性、鲁棒性与嵌入容量。相比之下，基于字形调制的水印方法因具有高隐蔽性和大嵌入容量，成为近年来的研究重点。然而，该类方法难以在不同语言中应用，并且在文本密集或排版紧凑场景下的鲁棒性仍有待提升。为此，本文围绕基于字形调制的文档水印技术展开研究，旨在提升其在不同场景下的实用性与鲁棒性，主要工作如下：

1) 为了在不同语言和样式的字符轮廓中施加可检测的扰动，本论文设计出基于自适应字形扰动的字体变体生成算法。该算法首先将字体中字符的轮廓坐标渲染为灰度图像并计算其形心位置，通过形心位置自适应地选择扰动方向。随后，按指定方向对字符轮廓实施微小的扰动操作，包括坐标平移、旋转和尺度扰动，并通过多轮迭代的方式优化扰动效果。实验表明，该算法能够在不同语言类型的字体中引入可检测扰动，且经过扰动生成的字体变体具有与原始字体接近的视觉质量。

2) 为了提升文档水印方法在密集文本场景中的适用性，本文利用字体及其变体共同作为水印载体，设计出基于语义投影分割的文档水印嵌入和提取算法。在嵌入端，按照水印序列依次将文档原始字体替换为对应的变体以进行水印嵌入；在提取端，通过语义辅助的投影分割方法获取每个字符的灰度图像，将图像的形心数据与字符的原始形心数据进行对比以实现水印信息判别。实验表明，该水印算法能够获得更高的水印容量，在文档字符间隔较小时仍具有较高识别率，引入的视觉失真也更小。

3) 在上述算法框架的基础上，为进一步提升字形水印面对不同信道失真的鲁棒性，本文提出了基于孪生神经网络的字形文档水印算法。在训练阶段，将字符的原始和变体图像分别输入孪生网络的两个分支，通过对比学习来建模二者在

高维特征空间中的差异；在提取阶段，将待测字符图像与对应原始图像输入训练完毕的网络，输出该字符图像的水印判别结果。对比实验表明，该水印算法在应对格式压缩、图像失真和跨操作系统渲染等复杂场景下的水印提取准确率高于现有方法，验证了该算法的鲁棒性。

关键词：文本水印，字体变体，字形修改，多媒体取证

ABSTRACT

With the continuous advancement of digitization, the dissemation of electronic documents in network environments is becoming increasingly frequent. The issue of illegal distribution, tampering, and resale of electronic documents has become a critical challenge that demands urgent attention. Document watermarking techniques embed hidden information into documents to prove their ownership, providing effective means for copyright protection and traceability. Traditional watermarking methods primarily rely on modifying text layouts, document backgrounds, or semantic features, which meet difficulties in the balance among imperceptibility, robustness, and embedding capacity. In contrast, glyph-based watermarking methods take characters as the unit of message embedding, offering better imperceptibility and watermark capacity than former methods. However, such methods face challenges in achieving multilingual adaptability and robustness when texts are arranged in compact formats within documents. To this end, this dissertation conducts a systematic study on glyph-based document watermarking techniques, aiming to enhance the robustness and practicality across diverse usage scenarios. The main contributions are as follows:

1) This dissertation proposes a font variant generation algorithm based on adaptive glyph perturbation, which aims at introducing detectable perturbations into character contours of different languages and font styles. The algorithm first renders the coordinates of character outline into grayscale images and calculates their centroids. It then adaptively selects the perturbation direction based on the centroid position, and applies slight geometric perturbations such as translation, scaling, and rotation to the character outlines. At last, the algorithm ensures sufficient centroid displacement in character through applying perturbations iteratively on the character outlines. Experimental results show that the proposed method introduces distinguishable perturbations into font variants across multiple font types without degrading their visual quality.

2) To improve the applicability of text watermarking methods in dense text scenarios, this dissertation developes a semantic projection-based document watermarking algorithm using both original fonts and their variants as watermark carriers. In watermark embedding phase, each character is replaced with its corresponding variant according to current watermark bit. In extraction phase, a semantic-assisted projection segmentation scheme is used to extract individual character from the document images. The centroid of each extracted character is then compared with the reference centroid from the original font to determine the embedded watermark bit. Experiments demonstrate that the proposed algorithm achieves higher watermark capacity and maintains high recognition accuracy even in documents with narrow character spacing, and the visual quality of watermarked documents are also ensured.

3) To further enhance the robustness against transmission distortions, this dissertation introduces a glyph-based document watermarking algorithm using a Siamese neural network. In the training phase, original and variant character images are fed into branches of the Siamese network to learn their differences in latent space through contrastive learning. For watermark extraction, images of the watermarked character and its corresponding original character are made pairs and sent into the network to obtain the watermark. Experiments show that the proposed algorithm significantly improves the extraction accuracy under various distortions, such as format compression, image degradation, and cross-platform rendering, verifying the robustness of the proposed algorithm.

Keywords: Text watermarking, Font variant, Glyph modification, Multimedia forensics

目　录

第一章 绪论

本章主要介绍文档水印技术的研究背景，研究现状以及关键问题。其中 1.1 节介绍了文档的版权保护问题和文档水印技术的应用场景；1.2 节介绍了文档水印技术的国内外研究现状及挑战；1.3 节介绍了本文的主要研究内容和各章节安排；1.4 节为本章小结。

1.1 研究背景与意义

数字文档，或称电子文档，是指人们在社会活动中创造的，储存在各种电子设备中的文字、图片材料。它们通过计算机系统进行操作，并可以在通信网络上传输。随着互联网技术的快速发展和无纸化办公的日益普及，数字文档的传播变得更加便捷和广泛，然而，这种易传播的特性也导致了数字文档的盗版、非法复制、篡改和窃取等安全问题的频繁发生。对许多涉及机密或敏感信息的文档（如政府公文、法律合同、企业内部报告等）而言，一旦被不当获取或篡改，不仅会使文档持有者遭受经济或声誉损失，还可能引发更为严重的社会影响。起初，人们开发出数字版权管理等技术方案，用于对文档内容进行加密、授权和访问控制。然而，此类方案通常依赖于严苛的访问控制策略或加密机制，一旦文档被合法读取并以明文形式呈现，便难以有效追踪其后续传播途径或篡改痕迹。如何在不影响文档正常阅读和使用的情况下，对其内容进行识别与防伪，成为数字文档版权保护中亟待解决的重要问题。

由此，人们开始关注数字水印技术，将数字水印用于保护包含电子文档在内的多种媒体内容[1]。数字水印技术通过在媒体中嵌入可见或不可见的信息，即水印，来为媒体附加不同的安全功能，例如版权保护[2-4]、信息溯源[5]和泄密追踪等。数字水印技术可按照媒体种类的不同进行分类，其中针对于电子文档这一媒体的一类数字水印技术便称为文档水印技术。相比于其他媒体，文档由于占用内存小、内容可复制、易于修改等特点，在内容保护问题上面临着独特的挑战：既要兼顾语义和格式排版的完整性，又需要在跨软件与跨平台的使用环境中保持方法的鲁

棒性。尽管上述挑战增加了文档水印技术的应用难度，但日益迫切的文档保护需求仍推动着研究人员不断投身至该领域的研究中。

文档水印技术的核心思想通过对文档视觉、排版或语义等属性施加显式或隐式的改动，将可追踪、可验证的信息嵌入文档内容之中。一旦文档在跨媒体传输、截屏分享或格式压缩过程中被他人非法获取或篡改，文档水印便可作为“数字签名”进行溯源和完整性校验。文档水印有以下两方面特性：一方面，面对日益复杂的传播渠道与日渐严苛的信息合规要求，水印技术能够在不破坏文档正常阅读和使用的前提下，对其进行防伪标识和后续追溯；另一方面，水印为取证和责任追究提供了关键依据，文档持有者即便在文档扩散后仍可进行有效监控。这些特性使得文档水印的应用场景十分广泛，从电子书到学术出版物，再到政府公文和法律合同，都有文档水印的用武之地。

综上所述，文档水印作为数字水印的一大重要分支，在如今的数字版权保护领域扮演着日益重要的角色。在过往的文档水印研究中，许多研究人员已经针对不同文档水印应用场景提出了可行的算法与实施方案，但要使水印方法同时具有高隐蔽性与大水印容量，并能够在不同的信道失真后实现精确、完整的水印提取，仍是当前文档水印领域研究中亟待攻克的关键课题。

1.2 国内外研究现状

学者和研究人员对文档水印技术的关注可以追溯到数十年前。早在 1990 年，Komatsu 等人[6]就提出了作用于文档图像的数字水印概念。随后，更多的学者开始关注文档水印技术，研究在不显著改变文档外观或语义的前提下嵌入可溯源和可验证的隐藏信息的水印技术方法[7-12]。如图 1.1 所示，根据水印载体元素的不同[13]，文档水印可大致分为基于排版和背景格式调制[14-18]、基于语义调制[9, 19-21]以及基于字形调制[22-24]三种类型。在三种类型的文档水印中，基于排版和背景格式调制以及基于字形调制的文档水印仅涉及对文档外观的修改，不修改文档的语义内容；相反地，基于语义调制的方法仅涉及对语义的修改，而不需要修改文档的排版或字体等样式。

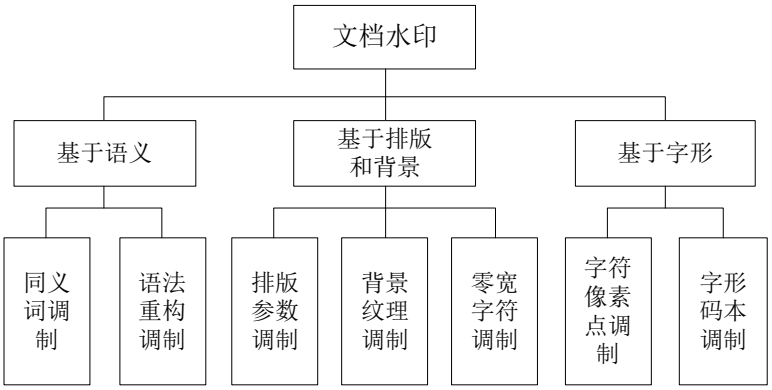


图 1.1 文档水印分类

1.2.1 排版和背景格式调制

基于排版和背景格式调制的文档水印是早期的文档水印方法。此类方法通过调整文档的物理布局或背景特征实现信息嵌入。Brassil 等人[14]首先提出通过微调排版来嵌入水印，其核心思想是利用排版参数的微小偏移编码二进制信息，例如将某行文字整体左、右平移数个像素来承载水印。随后，Bender 等人[15]进一步提出基于句间间距和段落间距调整的水印算法，减少了水印对文档视觉效果的影响。然而，此类基于排版格式的方法嵌入容量普遍较低，且对于图像噪声的抵抗性较差。

除了排版格式，也有学者使用文字和文档的背景来承载水印。Wu 等人[16]提出一种基于文字背景纹理调制的方案，通过修改字符边缘区域的文档底纹来嵌入水印。该方法修改的像素量小，因此有较强的隐蔽性，但同样难以在噪声和压缩攻击后恢复。此外，也有基于零宽字符的格式文档水印方法[17, 18]，通过在文档中插入宽度为零的字符来嵌入水印。尽管此类方法能取得较好的嵌入容量和隐蔽性，但由于零宽字符在文档图像中无法检出，且容易因文本编辑操作而丢失，此类方法难以满足如今网络空间中的文档水印应用需求。

1.2.2 语义调制

由于上述修改排版和格式的种种弊端，学者开始关注基于语义调制的文档水印方法。该类方法不需要改动文档的任何样式，只通过修改文档中的特定文本来实施水印嵌入。语义调制主要包括同义词替换[9-13, 19, 20]和语法重构[7, 8, 21]两类。同

义词替换是一类基础的语义调制方法。Topkara 等人[9]提出一种适用于英语的语义水印，通过替换文本中的同义词来进行水印嵌入，并通过维护一个优先级列表来保证检出率和水印容量。

另一类语义调制方法是语法重构。Atallah 等人[7]提出基于自然语言的文本水印算法，通过语法树和句子改写来承载水印信息，并结合了大素数密钥来增强水印序列的鲁棒性。近年来，随着自然语言处理技术的进步，基于语义调制的文档水印也获得了一定的发展[25]。Abdelnabi 等人[26]利用含对抗训练的编码-解码器生成含水印文本，通过替换原始文本来嵌入水印。虽然此类方法通过借助神经网络提高了水印文本的内容完整性，但由于大语言模型的深度参与，此类方法常受制于计算资源的规模和模型版权的要求等因素。

总之，对比排版和背景格式调制的文档水印方法，语义调制方法虽然保证了文档的视觉整体性和嵌入容量，但水印的实施方式会受到语言种类和风格的严重限制，导致水印的应用场景较少。

1.2.3 字形调制

区别于前两种水印方法，基于字形调制的文档水印方法（简称“字形水印”）不涉及语义和除字体外其他格式参数的修改，因而在隐蔽性和适用范围上有先天性的优势，这也让该类方法成为当前文档水印研究的热点。此类方法通过修改字符的几何形态实现水印嵌入。首先，Mei 等人[22]将文档黑白图像视为保护对象，提出了一种文档水印方法，通过修改字符的连通边界嵌入水印和标记，在提取时，通过识别标记来定位水印位置并检出。类似的方法还有张等人[23]，赵等人[24]。这些字形调制方法虽然改动了字符的形状，但其改动主要在文档图像层面实施，不涉及字符本身的几何特性。

近年来，Xiao 等人[27]提出了一种英文字形水印方法，他们根据中文字符的笔画特征人工设计了字形码本，在码本中，同一个字符拥有多个外观类似的不同字形，用于代表不同的水印比特。在嵌入端，他们用码本中的字形替换原始字符进行水印嵌入；而在提取端，他们利用卷积神经网络识别字形并提取水印。尽管方法中的字形修改阶段需要人工介入，该方法仍为后续的字形水印研究提供了开

创性的框架思路。Qi 等人[28]进一步提出了与字形码本方法类似的中英字形水印框架，他们通过对中、英文和数字字符的某一笔画施加左右方向的扰动生成矢量字库。在嵌入阶段，使用矢量字库中的字形代替原始字符；在提取阶段，他们通过一种字符模板匹配方法定位水印并提取。

经过上述字形水印方法的启发，Yang 等人[29]设计了一种多语言通用的字体水印方法，该方法能够自动地生成用于水印嵌入的字形码本，避免了人工扰动字形的繁琐工序。在水印的嵌入和提取阶段，他们使用一种相关质心定位比较算法，在两个相邻的文档字符中嵌入一个比特的水印并提取。该方法改进了过往字形水印需要人工设计字形码本的低效率过程，提高了字形水印的语言通用性。然而，该方法在水印提取阶段采用了图像投影法进行字符分割，导致在文档字符间隔较小时水印误码率明显上升，影响了水印的鲁棒性。随后，Yang 等人[30]将神经网络引入了字形水印框架中，使用一个以编码器-解码器为基础结构的神经网络生成含水印字形图像，然后，通过优化字符坐标来拟合水印字形以达到在文档中嵌入水印的目的。该方法能够适用多种字体，且借助了神经网络自动为字形施加了扰动，避免了对字形码本的启发式设计。但该方法在字符密集且字体较小的场景的提取准确率同样较低。此外，由于字形码本的设计过程由神经网络主导，该方法生成的水印字体常含有可察觉的垂直方向扰动，在一定程度上减弱了水印算法的隐蔽性。

回顾过往的字形水印方法，利用字形码本替换原始字符以嵌入水印，并利用图像处理方法进行水印提取的流程成为了具有通用性的水印框架。但根据所使用的字形码本以及图像处理方法的不同，水印方法的具体性能依然有很大差异，并且在字符密集场景下的鲁棒性普遍不够理想。然而，随着电子文档版权保护的需求日益增加，实现高鲁棒性的文档水印方法具有越来越大的实用价值。因此，对现有的字形水印方法框架进行改进，在保持高隐蔽性的前提下增加水印在字符密集场景下的鲁棒性具有重要的研究意义。

1.3 本文研究内容及结构安排

1.3.1 研究内容

现有的字形文档水印方法主要使用字形码本，或称字体变体作为水印载体，使用传统的图像处理方法对文档中的水印进行提取。虽然此类方法具有较强的视觉隐蔽性，但现有方法仍无法在具有高隐蔽性的同时兼顾鲁棒性和语言通用性，尤其是在文档字符排列紧凑的场景下。因此，本文针对现有方法中的两个重点部分：字体变体生成和字形水印嵌入与提取，展开了三项研究，具体工作如下：

针对现有字体变体生成方法难以在变体中施加可检测几何扰动的问题，本文设计了基于自适应字形扰动的字体变体生成算法。字体变体通过微调原始字体的字符轮廓生成，但该过程目前没有可视化手段，这影响了字体变体的生成效率。因此，本文提出一种字体的可视化字形扰动方法，利用字符的字形轮廓坐标，对字符进行归一化渲染生成字符图像，并在归一化图像的几何约束下进行坐标的扰动。以该扰动方法作为基础，本文设计出一种字体变体生成算法，以每个字符的初始形心所在位置为依据，自适应地选择一个方向对字符坐标施加扰动。随后，利用归一化图像检测形心位置的偏移，并通过迭代扰动的方式确保偏移足够充分，即产生可检测的扰动。本方法生成的字体变体能在保持较高视觉质量的同时，为变体字符引入可被检测的形心偏移。并且，本方法能够适用于不同语言和不同风格的多种字体，具有较强的通用性。

现有的字形水印方法在提取水印前需借助传统的图像投影法进行字符分割，这显著降低了密集排版场景下的水印识别率。为此，本文设计出一种基于语义投影分割的字形文档水印算法。在嵌入之前，利用前述的字体变体生成方法，预先为文档中的原始字体生成其变体，并为原始字体构建字符的形心字典。在嵌入阶段，选定文档中的部分字符作为待嵌入位置，当水印位为“0”时，保持该字符的字体不变；而当水印位为“1”时，将待嵌入位置的字符字体替换为其变体。在提取阶段，首先使用光学字符识别技术对字符进行初步分割同时记录语义信息，随后对初步分割得到的图像进行水平、垂直投影分割得到每个字符的最小外接矩形，并利用每行的矩形宽度众数修复初步分割中的错误字符，得到分割完毕

的字符图像。最后，利用字符的语义信息查询其形心字典，判定每个字符的形心是否偏移以提取水印比特序列。本方法能显著减少排版紧凑场景下的字符分割错误，增强了方法的鲁棒性。

当水印文档图像遭受传输信道失真后，现有的字形水印方法容易出现水印误判或丢失的问题，为提升字形水印面对传输信道失真的鲁棒性，本文设计出基于孪生神经网络的字形文档水印算法。构建一个包含两个特征提取子网络的孪生网络结构，两个子网络将在训练时共享权重的变化。在训练阶段，将原始字体与字体变体的字符图像作为一个图像对，经过噪声增强层后输入到孪生网络的两个分支中。通过对比学习，特征提取子网络将专注于原始字体与其变体之间的高维特征差异。随后，将待测的含水印字符图像与其原始字符图像配对，送入训练完毕的孪生网络中进行水印判定。由于噪声增强层中对于字符图像可能出现的失真进行了模拟，本方法能更好地从传输信道失真后的图像中提取出水印，拥有更强的场景通用性。

1.3.2 结构安排

本学位论文共分为六章，各章内容及其逻辑关系如图 1.2 所示。

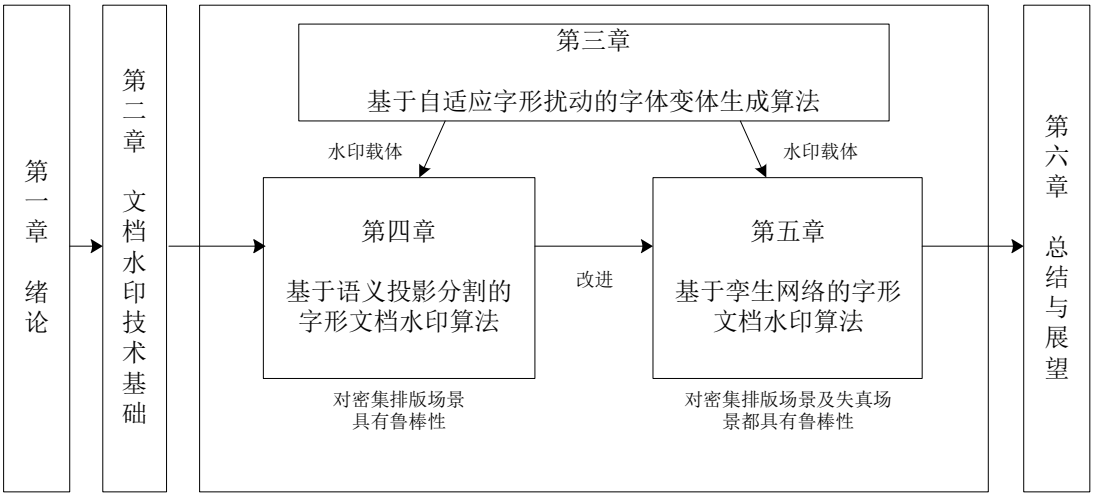


图 1.2 论文各章内容及其逻辑关系

第一章 绪论：本章主要介绍了文档水印的研究背景、研究意义和国内外的研究现状。另外，本章还总结了本论文的研究内容创新点与结构安排。

第二章 文档水印技术基础：本章首先介绍了文档水印的基本概念与分类，随后说明了文档水印的嵌入和提取策略，以及性能评估指标，最后介绍了字形水印技术的相关概念与基础算法。

第三章 基于自适应字形扰动的字体变体生成算法：本章介绍了所提出的字体变体生成方法。首先提出了一种以可视化方式对字体字形施加扰动的方法，然后以该方法为基础，设计了一种字体变体生成方法，通过对字体进行迭代式的形心扰动，从而引入可检测的形心偏移。

第四章 基于语义投影分割的字形文档水印算法：本章设计了一种字形文档水印方法，利用原始字体及其变体承载水印，随后使用语义辅助的投影分割法获取水印文档中的字符图像，并根据字符图像的形心偏移提取水印比特。

第五章 基于孪生网络的字形文档水印算法：本章改进了上述字形文档水印方法，通过孪生神经网络完成对原始字体及其变体特征的对比学习，进而使用该网络对字符图像中的水印进行判别。

第六章 总结与展望：本章总结了本文的研究工作，并对文档水印领域的未来研究进行了展望。

1.4 本章小结

本章介绍了文档水印技术的研究背景及研究意义，并分类介绍了国内外的文档水印研究成果。随后，通过分析现有方法的不足引入了本文的研究内容，并在最后介绍了本文的结构安排。

第二章 文档水印技术基础

本章节主要介绍了文档水印的基础概念、理论与方法。2.1 节介绍文档水印的基本概念与分类；2.2 节介绍文档水印的基本嵌入与提取流程；2.3 节介绍文档水印技术的性能评价指标；2.4 节介绍字形文档水印相关的字体处理与字符分割算法；2.5 节进行本章小结。

2.1 文档水印基本概念与分类

文档水印是数字水印技术在文档这一载体上的具体应用，其目的是让文档具备信息追踪与认证的功能，使文档即使在经过复制、篡改和非法传播后仍能够被溯源和取证。除文档之外，数字水印技术的应用载体还包括图像、音频和视频等，它们分别形成了各自的细分领域。与其他载体不同，文档的内容信息大部分由离散化的符号（文字、数字）组成，这导致文档的内容表达与视觉呈现对格式化和可读性有更高的要求。因此，相比于在其它载体上应用的水印技术，文档水印技术需要更加谨慎地选择嵌入方式，以确保水印既难以被文档的使用者察觉，又能够在各种水印攻击中被保留。

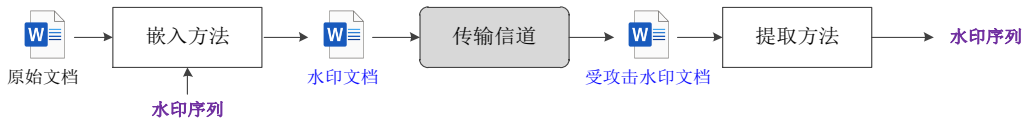


图 2.1 文档水印常见流程示意图

文档水印技术的基本思路是选择文档中的特定元素作为载体，通过适度的扰动或调制，嵌入用户不易察觉但能被有效提取的水印信息。在实际应用时，文档水印需要经过嵌入、传输和提取三个阶段，其流程如图 2.1 所示。在嵌入阶段，嵌入者首先需要准备待嵌入的秘密信息内容，然后将其编码，成为适合嵌入到文档中的水印比特流。待嵌入的信息可以是文档的作者标识、用户标识、版权声明信息或序列号等，也可以是这些内容的集合。随后，嵌入者根据选定的嵌入方法，将水印比特流按适当的方式插入到文档内容的特定位置上，例如文档的字符形状、排版、背景格式或语义结构等。此时文档水印的嵌入阶段结束。为了保证水

印的隐蔽性，嵌入完毕的水印文档通常应该具备与原始文档较为相似的格式外观与语义内容，同时水印信息在用户正常使用过程中不应该被察觉或引起怀疑。

在提取阶段，水印文档的接收者通过预先规定好的嵌入规则，从文档中的相应位置处提取水印比特流。随后，接收者需要将比特流进行解码，以获取水印嵌入者希望传达的秘密信息。在理想情况下，文档的接收者能够获取完整的水印比特流，从而百分之百地恢复出秘密信息，而在实际水印应用中，由于水印嵌入和文档传播过程引入的各种噪声的存在，水印信息通常会有部分损失，导致接收者恢复出的信息包含一定的失真。根据文档水印在视觉上的隐蔽程度[31, 32]，以及在提取过程中对原始文档的依赖与否[33, 33]，文档水印可进一步分为显式/隐式文档水印，以及盲/非盲文档水印两种分类形式。

2.1.1 显式/隐式文档水印

根据水印嵌入者是否有意希望文档使用者观察到水印的存在，文档水印可以被划分为显式水印和隐式水印两类。显式文档水印指，水印嵌入者希望文档使用者在正常阅读时能够明确观察到水印信息。该类水印在日常生活中受到了广泛应用，常见的表现形式有文字底纹、背景图案、印章和品牌标识等，用于在文档中表明版权的归属、秘密级别或文件状态等信息。这类水印的优点在于实施方式简单，通常不需要复杂的水印嵌入算法，并且能起到明显的提醒或声明作用。然而，正因为显式水印需要确保文档使用者能明显察觉，该类水印通常不适用于对隐蔽性要求较高的水印使用场景。

与显式水印不同，隐式文档水印的嵌入者通常希望水印不被文档使用者察觉和观测，即在保持隐蔽的前提下进行信息的传输。这类水印一般通过微调文档的各种内容参数，例如字体形状、间距、背景、排版结构或语义等方式实现，用于版权保护和用户溯源等安全敏感场景。相比显式水印，隐式水印通常具有更高的隐蔽性，使文档的使用者难以察觉水印的存在；但相对的，隐式水印的实施需要更加复杂的技术流程，嵌入者亦需要关注视觉隐蔽性与水印鲁棒性的平衡。总的来说，显式和隐式水印各有不同的使用目的和应用场景，并且都起到了文档保护的功能，但目前研究的热点及难点通常集中在隐式水印方面。在信息安全领域，研究人员提及的“水印”一词通常指隐式水印，而在日常生活中常被提及的“水

印”则一般指显式水印。在本文的研究中，主要关注的是隐式文档水印。除非特别指明，则“水印”指隐式水印。

2.1.2 盲/非盲文档水印

根据水印提取时原始文档参与与否，文档水印算法可分为非盲水印与盲水印两类。非盲水印指，水印的提取需要借助原始文档的信息作为参考。得益于原始文档的信息，该类水印通常具有更高的提取准确率和更强的鲁棒性。然而，该类水印的不足在于需要存储原始文档信息，在实际部署过程中带来了额外的存储和管理成本，降低了水印的实用性。盲水印算法则在提取时无需原始文档的参与，仅凭水印文档本身即可提取出嵌入的水印。这类水印虽然对算法的隐蔽性和鲁棒性提出了更高的要求，但在实际场景中具有更强的实用性，尤其适用于当今网络空间中文档受到多次传播或跨平台使用的情况。

在文档水印发展的早期阶段，非盲水印受到了广泛的研究，原因是当时的许多算法还不够成熟，需要依赖原始文档的辅助信息以提升算法的识别率或隐蔽性。近年来，文档水印领域已逐渐将研究焦点转向盲水印，以追求水印方法在实际应用中的广泛适用性与鲁棒性。本文主要关注的是盲文档水印系统。

2.2 文档水印嵌入和提取策略

区别于前文对文档水印总体框架与分类的介绍，本小节主要聚焦于水印在嵌入和提取过程中使用到的具体技术策略。在嵌入过程中的技术策略决定了水印信息如何映射并调制到文档中，而提取过程中的技术策略则决定了如何在各种干扰下从文档中恢复有效的水印信息。这些嵌入与提取方法的自身特性会直接关系到水印系统在隐蔽性、鲁棒性与水印容量等方面的综合表现。

2.2.1 水印嵌入策略

水印嵌入是指将原始信息编码后，以难以察觉的方式嵌入到文档的内容或结构中。该过程包含三个关键环节，分别是选择原始信息编码方式、确定水印嵌入位置和选择调制方式。水印原始信息一般需要通过编码转换为适合调制的比特

流。文档水印中，原始信息是可读的自然信息，例如用户名称代号、文档编号和时间戳等，这些信息通常不适合以明文形式展示在文档中，因此需要编码为二进制格式的秘密信息，以水印形式包含在文档中用于检测。在该编码过程中，可根据水印嵌入容量和鲁棒性的平衡决定是否引入差错控制编码。差错控制编码通过增加冗余信息以提升水印信息在文档传输过程中的容错能力，但会相应降低水印容量。常见的差错控制编码有 BCH 码、汉明码等。此外，对于水印原始信息不确定的情况，通常使用随机二进制比特序列代表编码后的秘密信息。

水印嵌入位置的选择对于水印的鲁棒性提取与隐蔽性提升至关重要。在文档水印算法中，优良的嵌入位置应同时满足不易被用户察觉，不易被攻击者定位，以及在格式变化下具有稳定性等条件。不同于图像或视频水印的连续像素嵌入，文档水印的嵌入位置通常是离散的语义文本或结构，如字符、词语、段落间隔、空白符和字体样式等。若将完整的水印不加分隔地叠加于这些载体之上，文档的语义或结构就会遭到严重的破坏，影响水印的隐蔽性和场景实用性。因此，研究人员通常将水印信息按一定规律分散布置于文档的多个位置，提升水印的隐蔽性与抗攻击能力。常见的分布方式有等差间隔、随机间隔和使用密钥的映射间隔等。

水印的调制方式决定用何种方式施加扰动。区别于水印嵌入位置，水印调制方式指，在确定的嵌入位置采用怎样的文档修改方式以表示水印信息。根据修改对象不同，水印可调制于排版属性（如行距、字距）、语义结构（如语序调整、同义替换）、可见图层（如背景颜色、底纹样式）或字体（如字形微调、笔画扰动）等。在实际嵌入过程中，研究人员需要特别关注调制方式对文档视觉效果的影响，通过主客观评价来确保文档视觉与语义上的整体性。在本研究中，主要考虑的编码策略为随机编码，嵌入位置为文档的正文区域，嵌入方式为字体调制。

2.2.2 水印提取策略

水印提取的任务是在各种可能的干扰条件下，从已嵌入水印的文档中恢复原始水印信息。提取策略的设计不仅要求具有高识别准确率，还需具备一定的容错能力，以应对常见的文档和图像层面的攻击。如 2.1 节所述，现有的文档水印方法通常以盲水印为主，以适应当今网络环境下的文档水印要求。相应地，文档水印的提取需要采用盲提取方法。盲提取由于不依赖原始文档，更加适合大规模分

发和部署，但在鲁棒性与通用性设计上更具挑战性。在盲水印中，文档水印提取过程通常包括以下步骤：首先，根据与嵌入端一致的密钥或伪随机序列定位嵌入区域；其次，从目标文档中提取相关特征，并转换为候选水印比特；最后，通过阈值判定、多轮投票等策略对提取结果进行水印判别和校正，并通过解码恢复。其中，提取特征的方法由嵌入策略决定，通常是水印嵌入方法中相应步骤的逆过程；阈值判定和投票的操作则是为了提升水印在经过噪声和攻击之后的提取准确率；解码是信息编码的逆过程，通常在阈值判定和投票之后进行，用于将水印比特恢复为原始信息。

在上述步骤中的特征提取方法主要包含基于文本特征的统计分析、几何特征匹配和深度学习提取等。基于文本特征的统计分析方法通过分析文档中各个字符的位置、字距和行距等统计特征的变化，利用预设的扰动模型来判别水印信息。该方法适用于排版属性和语义结构调制的水印提取；基于几何特征匹配的方法则通过识别文档中特定属性的端点、位置和形状的差异度来判别水印，适用于可见图层和字形调制的水印提取；基于深度学习的提取方法利用大量样本数据训练神经网络，使网络具有识别文档中细微的几何或语义变化的能力。这类方法适用于多种水印嵌入技术，但在数据准备和模型训练阶段需要消耗的时间与计算资源相比其他方法更大。总之，水印提取的步骤与方法选择通常与水印嵌入位置和方式密切相关，并且不同的提取策略也会带来提取精度、鲁棒性与资源消耗的差异。在本研究中，对应于嵌入阶段使用的字形调制方法，在提取阶段主要使用的方法是几何特征匹配与深度学习提取。

2.3 文档水印性能评估指标

在包含文档水印在内的数字水印技术研究中，研究人员通常从隐蔽性、鲁棒性和嵌入容量三个方面对算法的整体性能[35-37]进行评估。对于每个方面，已有若干定性或定量指标被提出并受到认可，这些指标为文档水印方法的性能评估提供了依据。

2.3.1 隐蔽性评价

水印的隐蔽性指，嵌入水印后的文档内容是否能保持原有视觉效果和语义表达，水印信息对于文档读者而言是否不可察觉。隐蔽性高的水印方法应该在嵌入信息的同时尽可能保持文档的外观和语义可读性不受影响。在文档水印中，对隐蔽性的评估主要包含主观和客观评价两种方法[39]，其中客观评价方法根据水印类型会使用不同的评价指标，如表 2.1 所示。

表 2.1 文档水印隐蔽性评价方法

评价方法 \ 水印类型	基于视觉修改	基于语义修改
主观评价方法	人工观察测试	
客观评价方法	PSNR、SSIM	ΔPPL、BERTScore

隐蔽性的主观评价通常通过人工观察测试进行。选择一组非相关课题研究者的测试人员，数量通常不少于 20 位。随后，安排测试人员在预先准备的实验环境中阅读测试文档，并让测试人员根据自己的主观感受对文档的外观或语义质量按照规定分值评分。分值可由测试制定者自行规定，通常是 1 分到 5 分，共 5 个分值，从低到高分别代表质量差、较差、一般、较好、好。最后，将该组所有测试人员给出的评分进行平均处理，得到的平均分即为人工观察测试的最终结果。

上述人工观察测试的流程适用于评价各种类型的文档水印，对于不同类型的水印嵌入方式，其测试得分的含义也不同。当文档水印的嵌入方式为基于排版格式、背景格式或字形调制时，测试得分代表文档的视觉质量好坏；当嵌入方式为基于语义调制时，得分则代表文档的语义可读性好坏。此外，该类型的测试结果受实验环境影响较大，为保证测试结果能最大程度地代表真实文档读者的主观感受，研究人员需要使准备的实验环境尽量接近真实的文档使用场景。

隐蔽性的客观评价主要通过文档图像和语义层面的定量计算指标进行。对于引起文档外观改变的水印方法，研究人员通常将文档视为图像，通过图像的视觉评价指标评判隐蔽性。在计算机视觉领域，适用于文档图像的常见评价指标有峰值信噪比（Peak Signal-to-Noise Ratio, PSNR）和结构相似度（Structure Similarity, SSIM）两种[40-42]。PSNR 的概念最初来自音频领域，表示声音信号的最大功率与

噪声功率的比值。在文档图像中，参考图像的概念则等同于信号，而待测图像与参考图像的差异则等同于噪声。此时 PSNR 的含义变为衡量参考图像与噪声之间的功率差异量级，单位为 dB，其计算公式如（2.1）所示：

$$\mathrm{PSNR} = 10 \cdot \log_{10}(\frac{MAX_x^2}{\mathrm{MSE}}) \tag{2.1}$$

其中，x 表示参考图像，MAX_x 表示参考图像像素的最大值，在灰度图像中通常为 255，MSE 表示均方根误差（Mean-Square Error, MSE），其计算公式如（2.2）：

$$\mathrm{MSE} = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[x(i,j) - y(i,j)]^2 \tag{2.2}$$

其中，x 和 y 分别表示参考图像和待测图像，m 和 n 分别表示参考图像或待测图像的宽和高。通过计算 PSNR 可以量化表示嵌入水印过程中引入的噪声大小。PSNR 越高，代表水印引入的噪声越小。SSIM 则更侧重于图像结构相似性的衡量，由三个部分的相似性组成，分别是亮度（luminance）、对比度（contrast）和结构（structure）。三个部分的计算公式如（2.3）：

$$\begin{cases} l(x,y) = \dfrac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \\ c(x,y) = \dfrac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2} \\ s(x,y) = \dfrac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3} \end{cases} \tag{2.3}$$

其中，x 和 y 分别表示参考图像和待测图像，μ_x，σ_x 和 μ_y，σ_y 分别表示 x 和 y 在滑动窗口内的均值和方差，σ_{xy} 表示 x 和 y 在滑动窗口内的协方差，滑动窗口的大小根据图像尺寸进行设置。C_1、C_2、C_3 为常数，其中 $C_1 = (0.01L)^2$，$C_2 = (0.03L)^2$，$C_3 = C_2/2$，L 为像素值的范围大小，在灰度图像中为 255。将三个部分的相似度相乘即构成了 SSIM，具体计算公式如（2.4）：

$$\begin{aligned} \mathrm{SSIM}(x,y) &= [l(x,y)\cdot c(x,y)\cdot s(x,y)] \\ &= \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2+\mu_y^2+C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \end{aligned} \tag{2.4}$$

值得注意的是，图像 PSNR 和 SSIM 关注的差异主要来自像素层面。对于颜色种类单一、黑白像素分布具有一定规律的文档图像而言，该类指标并不能准确反映其视觉差异程度。因此，文档水印方面的研究通常倾向于使用主观视觉评价方法进行视觉差异的衡量。

对于修改语义的文档水印方法而言，水印文档与原始文档的差异体现在词汇、句式或者篇章结构上。研究人员通常使用自然语言质量评估指标来量化此类方法的隐蔽性。常用的客观评价指标[43, 44]包括困惑度（Perplexity, PPL）与 BERTScore（Bidirectional Encoder Representations from Transformers Score），它们分别从语言流畅度与内容一致性两个方面衡量水印对文档语义的影响。PPL 是衡量语言模型对文本预测能力的指标，它反映了文本在某一语言模型下的自然性。在文档水印方法中，通过计算困惑度的差值来表征嵌入水印前后的语言流畅度差异。具体地说，将原始文档与水印文档分别输入到同一预训练语言模型中，对比两者的困惑度差值ΔPPL。若差值接近零或者小于零，则说明水印的嵌入几乎不影响语言流畅度；若差值较大，则说明水印带来了可被检测的语义异常。

BERTScore 是一种基于深度语言模型的语义相似度指标，它利用 BERT 等预训练语言模型获取句子的上下文向量，再通过 token 级别的余弦相似度计算一对文本之间的语义保真度。原始文档与水印文档的 BERTScore 数值分布在-1 到 1 之间，越接近 1 表示两者在语义空间中的距离越小，语义的隐蔽性越高。在实际评估中，研究人员通常联合使用ΔPPL 与 BERTScore，以同时度量语义文档水印的语言自然性与语义一致性。若某种语义水印方法在保持低ΔPPL 的同时具有较高 BERTScore，则说明该方法在隐蔽性方面具有较好的表现。由于基于字形扰动的方法只修改文档的外观，不涉及语义修改，因此在本文的研究中使用的隐蔽性评价方法主要为主观评价中的人工观察测试方法和客观评价中的 PSNR、SSIM 指标测试方法。

2.3.2 鲁棒性评价

鲁棒性是衡量水印在文档经历干扰或攻击后能否被正确提取的能力。对电子文档而言，常见的干扰包括文档格式转换、有损压缩、截屏、打印扫描和跨操作系统渲染等。为了定量分析鲁棒性，文档水印算法通常采用比特错误率（Bit Error

Rate, BER）和归一化相关系数（Normalized Correlation Coefficient, NCC）作为评价指标[45-47]，通过在算法实施过程中模拟各类型的干扰和攻击进行鲁棒性测试。BER 是鲁棒性最通用的测试指标，通过计算原始水印二进制序列与从受攻击的文档中提取出的二进制序列的差异比特个数得出，通常用百分比表示。其计算公式如（2.5）：

$$\mathrm{BER}=\frac{1}{L}\sum_{i=1}^{L}\mathbb{I}(w_i\neq\hat{w}_i)\times 100\% \tag{2.5}$$

其中，L 表示原始水印的长度，$\mathbb{I}(x)$ 表示指示函数，若 x 为真，则 $\mathbb{I}(x)$=1，否则 $\mathbb{I}(x)$=0；w_i 表示原始水印序列的第 i 个比特，$\hat{w}_i$ 则表示从受攻击文档提取出的水印序列第 i 个比特。根据文档受到攻击的强弱程度，BER 可以在 0~50%范围之间波动，0%代表能准确提取所有水印，50%则代表几乎无法识别出水印的存在。需要提及的是，当使用 BER 作为性能评估指标时，可以使用其互补指标比特准确率（Accruracy, ACC）达到相同的评判效果，ACC 的计算公式如式（2.6）：

$$\mathrm{ACC}=1-\mathrm{BER} \tag{2.6}$$

NCC 用于描述原始水印与提取结果之间的相似度。与 BER 不同，NCC 不仅考虑了单个水印比特的差异，也考虑了水印序列的连续性。NCC 的计算公式如（2.7）：

$$\mathrm{NCC}=\frac{\mathrm{cov}(x,y)}{\sigma_x\sigma_y} \tag{2.7}$$

其中，$\mathrm{cov}(x,y)$ 表示原始水印 x 与提取结果 y 之间的协方差，σ_x 和 σ_y 分别表示原始水印与提取结果的方差。NCC 没有单位，取值范围为-1 到 1，当数值接近 1 时，表示提取结果信号与原始水印几乎一致。在本文的文档水印研究中，由于水印的嵌入以离散的字符为载体，因此本文选用 ACC 作为鲁棒性评价的指标。

2.3.3 嵌入容量

嵌入容量指水印技术在载体中可承载的有效信息量，是衡量文档水印技术实用性的核心指标之一。对于一种特定的水印算法，其嵌入容量的上限通常在算法设计阶段就能得到初步的确定，并在后期依据实际隐蔽性和鲁棒性约束对该上限进行微调。根据观测尺度不同，嵌入容量指标[48, 49]可分为总嵌入容量 C_{doc}、每页

嵌入容量C_{page}和每字符嵌入容量C_{char}三种类型，对应的单位分别为每文档比特数(Bits Per Document, bpd)、每页比特数(Bits Per Page, bpp)和每字符比特数(Bits Per Character, bpc)。C_{doc}适合衡量在文档层面嵌入水印的方法，例如基于背景或排版格式调制的技术等；C_{page}和C_{char}则用于从更加微观的角度衡量嵌入容量，适用于基于语义或字形调制的文档水印技术方法，三种指标的换算关系如式（2.8）：

$$C_{doc} = C_{page} \times N_{page} = C_{char} \times N_{char} \tag{2.8}$$

其中，N_{page}和N_{char}分别指文档的页数和字符数。借助式（2.8），不同容量的指标可以在横向比较时进行归一化处理，以消除文档长度与排版差异带来的影响。水印嵌入容量同隐蔽性和鲁棒性之间存在相互制约的关系。若注重水印容量的提升，水印方法往往需要在更多文档元素中施加扰动或者增加单个扰动的幅度，这容易留存视觉或语义上的痕迹，从而削弱隐蔽性和鲁棒性；若利用编码上的冗余来增加鲁棒性，又会引起水印的嵌入容量下降。因此，研究人员在设计文档水印方法时，必须注重嵌入容量与隐蔽性和鲁棒性的平衡以提升方法的实用性。

不同类型的水印方法拥有的嵌入容量基准也不同。以C_{char}为参照，对于基于背景或排版调制的方法，其嵌入容量通常在 10^{-3} ~ 10^{-2} bpc 之间；基于语义调制的方法因为可通过段落与句子结构嵌入水印，一般具有 10^{-2} ~ 10^{-1} bpc 的嵌入容量；而基于字形调制的方法因为载体单元更小，其嵌入容量通常能稳定达到 0.5 bpc 及以上。由于本文关注的文档水印方法主要为基于字形调制的方法，在本文的后续章节中选择C_{char}作为嵌入容量指标，以便在提出的方法和实验中更明确地描述水印的载荷性能。

2.4 文档水印字体处理与字符分割基本算法

本节主要介绍处理文档字体过程中使用的基本算法，包括构成字体轮廓的贝塞尔曲线方程、字体修改算法与文档图像的字符分割算法。

2.4.1 贝塞尔曲线

贝塞尔曲线是一种参数化的多项式曲线[50]，常被应用于计算机矢量图形的绘制[51]。一条贝塞尔曲线由数个控制点构成，根据控制点数量的不同，贝塞尔曲线可细分为线性曲线、二次贝塞尔曲线和高次贝塞尔曲线。n 阶贝塞尔曲线的一般化方程如式（2.9）：

$$B(t)=\sum_{i=0}^{n}\binom{n}{i}P_i(1-t)^{n-i}t^i,\ \ t\in[0,1] \tag{2.9}$$

其中，$B(t)$ 指贝塞尔曲线方程，t 为曲线的位置标度；n 为曲线的次数或阶数；$\binom{n}{i}$ 为二项式系数；P_i 代表曲线的控制点，取 $i=0$ 和 $i=n$ 时，P_0 和 P_n 分别为曲线的起点和终点。当式（2.9）中 n 取 2 时的曲线称为二次贝塞尔曲线，字体文件常采用该曲线绘制字形轮廓。此时式（2.9）可展开为式（2.10）：

$$B(t)=(1-t^2)P_0+2t(1-t)P_1+t^2P_2,\ \ t\in[0,1] \tag{2.10}$$

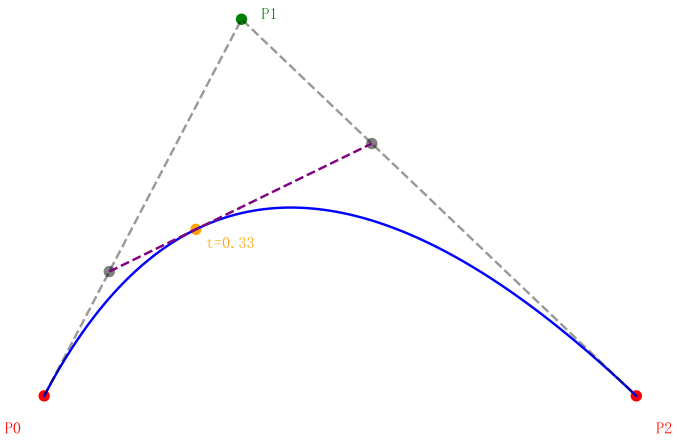


图 2.2 二次贝塞尔曲线示意图

该曲线以 P_0 为起点，P_2 为终点，控制点 P_1 决定曲线偏离直线段 P_0P_2 的幅度。t 可视为贝塞尔曲线的行进速度，如图 2.2 所示。当 $t=0$ 时，曲线落在起点 P_0 处；当 $t=1$ 时，曲线落在终点 P_2 处。随着 t 从 0 逐渐增加到 1，曲线会从起点逐渐延伸到终点，在两端之间形成一条光滑路径。通过应用二次贝塞尔曲线，字体只需要使用三个点的坐标即可记录字符轮廓中的一条光滑曲线，并可以通过使 t 连续变

化来快速计算曲线每个点的像素位置。这不仅节省了字体文件的存储空间，也在同时方便了计算机系统对字符轮廓的绘制。

2.4.2 字体修改流程

表 2.2 TTF 表组成

标签	表全称	表内容描述
cmap	character to glyph mapping	记录字符 unicode 码到字形序号的映射关系
glyf	glyph data	存储字形轮廓数据，包括轮廓数量、端点、控制点坐标和标志位等
head	font header	记录字体全局信息，包括坐标范围、时间戳、版本号等
hhea	horizontal header	定义水平排版的全局参数
hmtx	horizontal metrics	存储各字形的水平度量值
loca	index to location	存储 glyf 表中各字形的偏移地址
maxp	maximum profile	存储字体的最大坐标点数、最大轮廓数等上限参数
name	naming	存储字体的名称、家族名、样式、版权信息等
post	PostScript compatibility	存储与 PostScript 相关的兼容性信息

字体是字符形状的集合，不同的字体通过不同的几何参数呈现出独特的书写外观风格。在计算机系统中，最常用的字体文件格式为 TrueType Font（TTF），其文件使用二进制存储。TTF 内部通常包含多个表结构，如表 2.2 所示。它们共同描述了字符轮廓、度量信息、命名元数据和渲染指令等内容。在这些表结构中，与字符外形联系最紧密的是 glyf 表，该表以二次贝塞尔曲线的坐标格式存储了所有字符的外观轮廓。

通过修改字体文件的表中内容，研究人员可以对字符的外观进行坐标层面的编辑，实现字体的再设计或样式微调的目的。该修改流程如图 2.4 所示，包括以下步骤：首先解析原始字体 TTF，定位 cmap、loca 和 glyf 表等关键表的位置，并提取字符的贝塞尔曲线坐标序列；随后对需要修改的字符施加扰动，例如笔画缩放、平移；扰动完毕后，按照原索引顺序进行字符表的数据重组；最后将其按规范重新封装为二进制格式，并存储为新的字体文件。该修改流程除了用于文档水印领域，也可广泛适用于调整字体风格、增加字体的通用性和兼容性等其他领域的字体应用场景。

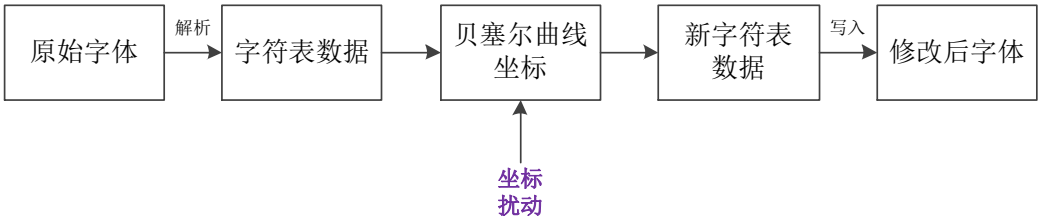


图 2.4 字体修改流程示意图

2.4.3 字符分割算法

字符分割是将图像中的字符进行定位并分离的算法，在字符识别和自然语言处理等领域有着广泛的应用。常用的字符分割方法可分为两类：基于传统图像处理的方法[52, 53]与基于深度学习的方法[54, 55]。基于传统图像处理的字符分割算法主要使用投影法进行。投影法（Projection）[56]是计算机视觉领域的一种经典算法，其核心思想是通过在图像水平或垂直方向上进行像素密度的统计以确定字符的边界。具体做法如图 2.5，首先计算水平投影的像素直方图，根据直方图中黑色像素的波谷定位文本行区域的开始和结束高度；然后分割出文本行区域，对每个文本行计算垂直方向投影的像素直方图，定位直方图中的波谷位置；最后根据两次直方图的定位进行图像切割，获取单个字符的图像。投影法的特点有计算速度快，能实现对文档、车牌等规范字符的准确分割等，并且可额外通过形态学操作以避免误分割。由于投影法的像素计算过程容易受到噪声影响，通常需要先进行图像层面的预处理工作，例如将彩色或灰度图像通过自适应阈值化转换为二值图像，以及对倾斜图像应用霍夫变换来进行角度纠正等。

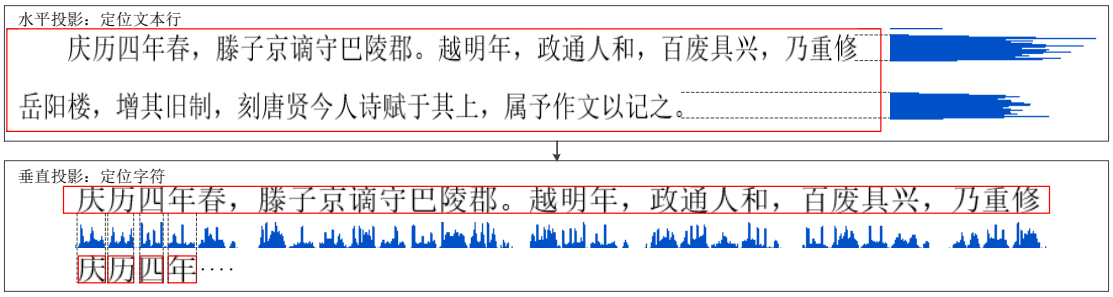


图 2.5 字符投影分割算法示意图

神经网络对分类和识别任务有较强的适用性。得益于深度学习技术的飞速发展，如今基于神经网络的字符分割方法在处理复杂图像时能取得更好的鲁棒性。此类方法的核心是借助目标检测模型如 Fast Region-based Convolutional Network（Fast R-CNN）[57]、You Only Look Once version 7（YOLOv7）[58]或 Dilated Convolutional Network with Bidirectional Long Short-Term Memory（DB-Net）[59]等实现字符区域的大致检测，再对该区域使用非极大值抑制算法分割得到独立字符。在实际应用中，研究人员常结合目标检测模型与字符识别模型，以实现字符图像的端到端处理任务。由于需要处理的字符图像以文档图像为主，因此在本文的研究中使用的字符分割算法是基于传统图像处理的投影法。

2.5 本章小结

本章介绍了文档水印算法中涉及到的概念与基础技术，为本文后续章节的算法设计与实验提供了必要的理论支持。首先，介绍了文档水印的基本概念与分类方式，并说明了本文的研究主要集中在隐式盲水印类别；随后，介绍了文档水印技术的一般嵌入与提取流程，以及评估文档水印算法性能的指标。最后，介绍了与后续章节相关的字体处理与字符分割技术算法，以便读者能更清晰地理解后续章节的算法实施过程。

第三章 基于自适应字形扰动的字体变体生成算法

本章节主要介绍基于自适应字形扰动的字体变体生成算法。在 3.1 节中，介绍字体变体生成算法的研究背景与关键问题；在 3.2 节中，提出了一种基于可视化字符图像的字形扰动方法；在 3.3 节介绍了如何自适应地引入字形扰动以生成字体变体，进而增强所生成字体变体的视觉质量，同时提升字体变体的信息承载能力；在 3.4 节中对实验结果进行了分析讨论；在 3.5 节中进行了本章小结。

3.1 引言

字体是承载文本信息的核心媒介，其设计不仅影响信息传达的准确性和效率，还决定着文本的视觉美感和可读性[60,61]。字体变体[62]生成是字体设计领域的一项分支技术，指在不改变字符语义的前提下，对字符的形状进行细微调整或变化，从而生成与原始字符风格相近但略有差异的新字体[63]。该技术在计算机视觉和信息隐藏领域都展现出较高的应用价值：在计算机视觉领域，生成风格一致的字体变体能够丰富视觉表现，为版面设计提供多样性而不损害可读性[64]；在信息隐藏领域，微小的字形差异可用于在文本中隐含附加信息，实现信息的无形承载。基于字体变体具有的以上特性，越来越多的研究人员开始关注用于信息隐藏特别是文档水印的字体变体生成算法。

适用于文档水印的高质量字体变体应该兼顾以下三个方面的特性：视觉一致性[65]、多语言通用性[66]和扰动可检测性[67]。在文档水印领域，研究人员尝试了多种不同的方法来生成高质量的字体变体。首先，Qi 等人提出[28]了一种基于人工修改的字体变体生成方法，通过手动调整数个字符的拓扑结构以生成字体变体，并在电子文档中利用字体变体承载水印信息。该方法生成的字体变体虽在打印扫描场景下展现出较高的特征保留率，但扰动位置需要手工选择的特性导致该方法在水印容量和通用性上受到较大的限制；随后，Xiao 等人[27]提出了一种通过字体生成模型来产生变体的方法，该方法引入了生成对抗网络构建字体变体，利用深度神经网络模型自动保持笔画连贯性。然而该字体变体生成方法需要超过 10,000 个字形样本训练模型，且对于不同字体需要独立训练模型参数，这些模型

的特性造成了该方法的实际应用受限。同时，此方法生成的字体变体与原始字体间的差异仅能在字符宽度大于 200 像素时被成功识别，导致该字体变体无法在普通文本尺寸的文档中承载信息；最近，Yang 等人[29]提出了一种通过调整字符质心所在平面来生成字体变体的方法，他们将质心位移作为编码对象，通过有规律地扰动字符笔画造成质心的平移，并使用邻近字符的相对质心作为了秘密信息的载体。该方法的字体变体生成流程能够适用于包含英文、中文在内的多种语言，并且在跨媒体场景下仍能够保留字体变体的形状特征。但由于该方法采用了固定单向扰动的策略，导致非对称字符如“P”、“L”在扰动过程中产生可感知的图像密度失衡，从而影响字体变体的视觉质量。

为了在不影响字体变体视觉质量的前提下更好地引入可检测的字形扰动，本文提出了一种基于自适应字形扰动的字体变体生成方法，该方法能够针对不同语言字体生成与原始字体相似的字形变体，并在每个字符中以形心的偏移表示扰动。具体而言，本文首先提出了一种字形的可视化扰动算法，该算法通过字形的原始轮廓坐标生成归一化字形图像，在图像的几何分布约束下进行坐标的平移、尺度扰动和旋转等修改操作。以上述操作为基础，本文进一步提出基于自适应形心移动的字形扰动算法，通过自适应选择更小扰动的方向进行坐标修改以生成高质量的字体变体。由于利用字符图像的几何特征约束了坐标的修改方式，相比已有方法，本方案生成的字体变体在保持扰动可检测性的同时拥有与原始字体更接近的视觉效果。此外，得益于形心移动方法对形状的自适应性，本方案的字体变体生成流程可适用于中文、英文和数字等不同语言种类与字体风格，增强了此方案在混合语言场景下的适应性。

3.2 可视化字形扰动算法

3.2.1 字形坐标归一化绘图

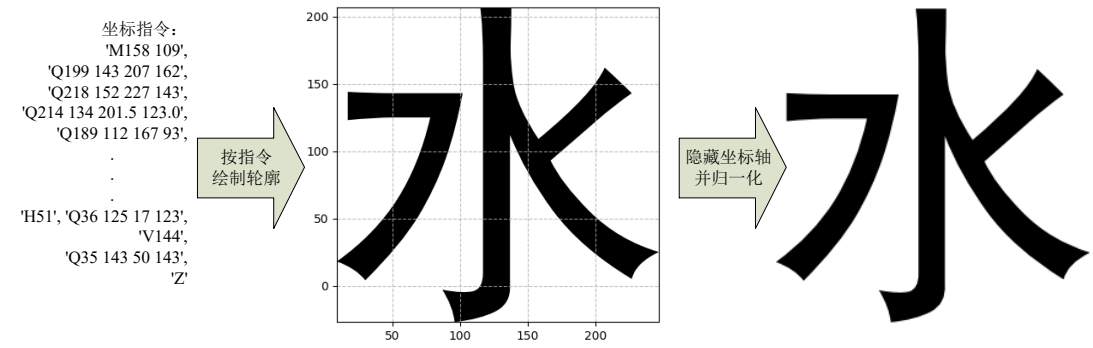
字形扰动指细微地修改字符的形状参数，其本质是对字符轮廓坐标的修改。在多数字体文件中，字符轮廓信息以贝塞尔曲线坐标的形式保存在字形表中，这些坐标定义了字符的外形曲线。字体设计时，不同的字体文件常使用不同的坐标

体系和计量单位，这导致字符坐标的修改尺度无法在多种字体文件中通用。同时，坐标形式的数据虽然适合压缩存储，但在修改过程中不便于观测其数值的变化，影响扰动的可控性。因此，在进行具体的坐标扰动操作之前，必须将字形坐标通过归一化绘图的方式实现参考尺度的统一，使坐标扰动操作具有视觉一致性。通过归一化绘图形成的字形图像不仅可以用于主观评估扰动的视觉效果，也可以作为扰动操作的约束条件以提升扰动的质量。

表 3.1 坐标指令表

指令	指令全称	含义说明
M	Move To	移动至指定坐标点并作为新轮廓的起点
L	Line To	从当前点绘制一条直线到指定点
H	Horizontal Line To	在水平方向绘制一条线段
V	Vertical Line To	在垂直方向绘制一条线段
Q	Quadratic Bezier	绘制一条二次贝塞尔曲线
T	Smooth Quadratic Bezier	绘制与前一段平滑连接的二次贝塞尔曲线
C	Cubic Bezier	绘制一条三次贝塞尔曲线
S	Smooth Cubic Bezier	绘制与前一段平滑连接的三次贝塞尔曲线
Z	Close Path	闭合当前轮廓路径，将当前点与起点连接

在字形表 glyf 中，每个字符对应一组包含移动、绘线、曲线等操作的坐标指令序列，表 3.1 解释了坐标指令中的各指令含义。以黑体字符“水”为例，图 3.1 (a)所示为字符“水”的坐标指令序列集合，其中每一行都包含了一个指令和数个坐标值。如图中的坐标指令“M 158 109”表示将绘图起点定位至(158, 109)坐标；“Q 199 143 207 162”则定义了一条以(199, 143)为控制点、(207, 162)为终点的二次贝塞尔曲线；指令“H 51”代表水平移动至纵坐标为 51，横坐标不变；指令“V 144”代表垂直移动至横坐标为 144，纵坐标不变；坐标指令“Z”代表结束。



(a) 字符坐标指令序列　　(b) 含坐标轴的字符图像　　(c) 归一化字符图像

图 3.1 坐标指令归一化绘图流程示意图

基于上述坐标指令，本文设计了一种如图 3.1 所示的字形坐标归一化绘图方法。首先，建立一个二维直角坐标系 xOy，并以右为 x 的正方向，上为 y 的正方向；随后，将坐标指令按照其曲线类型和坐标依次绘制在该坐标系上并将轮廓内部涂黑，形成如图 3.1 (b)所示含有坐标轴的字形图像；然后，将坐标轴从图像中擦除，余下字形部分，并将其归一化映射到 24×24 的像素空间，形成如图 3.1 (c)所示的字形图像。对于原始轮廓点 (x,y)，归一化映射方程如式（3.1）：

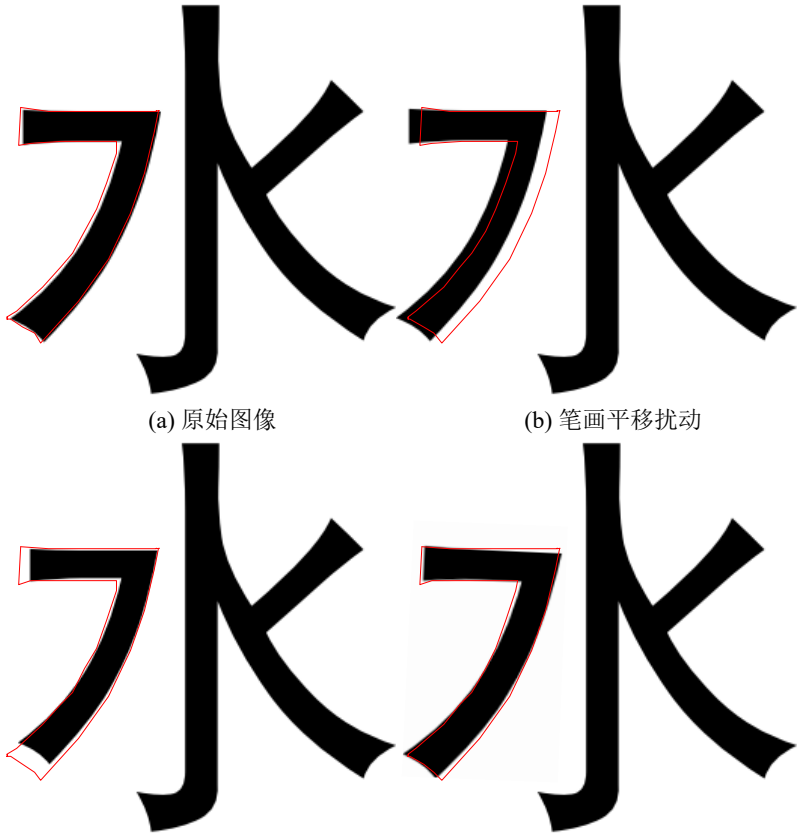
$$(x',y')=\left(\frac{x-x_{\min}}{W},\frac{y-y_{\min}}{H}\right)\times S \tag{3.1}$$

其中，(x',y') 表示映射后的坐标轮廓点，$x_{\min}$ 和 $y_{\min}$ 分别为原始字形坐标中 x 和 y 的最小值；W 和 H 分别表示字形擦除坐标轴后的长和宽，S 为目标分辨率，此处为 24 像素（pixel, px）。通过上述归一化绘图方法，可以将不同字体的字形坐标图像压缩到同一尺度基准，便于进行后续的坐标修改操作。此外，当归一化图像的目标分辨率为 24 px 时，字符图像与文档字符中字号为 16 pt 的字符拥有相同显示大小，该特性可用于实现对扰动的实时模拟和观测，进而增加扰动过程的可视化程度。

3.2.2 轮廓坐标扰动算法

字形扰动的核心在于对轮廓控制点的坐标进行微调。借助上述的坐标归一化绘图方法，字形在扰动的过程中可被实时显示，而字符图像的像素结构也可为扰动操作进一步提供优化。如图 3.2 所示，字形扰动可分为三种类别：基于笔画位置的平移扰动、基于笔画粗细的尺度扰动和基于笔画旋转的角度扰动。这三种扰

动方式可以作为字形扰动的基本操作单元，联合使用这三种扰动方法，可以对字形施加隐蔽且可提取的扰动特征。



(a) 原始图像　(b) 笔画平移扰动

(c) 笔画尺度扰动　(d) 笔画角度扰动

图 3.2 字形扰动的三种方法

基于笔画位置的平移扰动指的是在不改变笔画整体形状的前提下，对局部笔画的空间位置进行微小移动。这类扰动通常选择离交叉点较远、结构独立性较强的笔画作为操作对象，以避免破坏字符笔画之间的连接关系。在单独平移某一笔画时，平移的方向可以根据局部空间结构的余量进行选择。例如在横向或纵向留白较大的区域，优先向留白方向移动，从而在提高扰动强度的同时抑制视觉感知风险；也可对字符的多个邻近笔画施加整体平移，实现在扰动的同时保持字符的局部结构不变。平移扰动的方位通常分为上下扰动和左右扰动两种，也可将两种方位按照不同比例结合以实现多个方位的灵活平移。使用平移扰动时，需要遵循两项原则：一是需要注意笔画平移时的边界位置，避免平移之后的笔画接触到其他笔画或超出当前字符边界；二是位移幅度需控制在人眼最小可觉差（Just Noticeable Distortion, JND）[68, 69]范围内，避免影响字符的整体外观。在实践中，平移的幅

度通常不超过字符高度的 1/12。对于原始坐标点(x,y)，将其平移Δt后的坐标点(x',y')可通过式（3.2）计算得出：

$$(x',y')=(x+\Delta t,y+\Delta t) \tag{3.2}$$

基于笔画粗细的尺度扰动通过微调控制点之间的相对距离，实现在局部范围内对笔画宽度的扩展与收缩。其实质是对笔画的轮廓控制点沿轮廓的法线方向进行平移。加粗笔画时，平移的方向为法线正方向；减细笔画时，平移方向为法线反方向。该类扰动方法除了需要遵循平移扰动方法的规则之外，还需要额外注意减细幅度与笔画原始宽度的关系。若减细幅度过大，笔画的原始形状就可能发生扭曲，破坏字符的语义识别性。在实践中对于原始坐标点(x,y)，尺度扰动后的坐标点(x',y')可通过式（3.3）计算得到：

$$\begin{cases}(x',y')=(x+\Delta x,y+\Delta y)\\ \Delta x=\Delta t\cos\alpha\\ \Delta y=\Delta t\sin\alpha\end{cases} \tag{3.3}$$

其中，Δt表示平移的幅度，Δx和Δy分别表示Δt沿法线方向的x和y分量，α表示法线与x坐标轴的夹角。

基于笔画旋转的角度扰动是指对笔画的轮廓控制点施加轻微旋转变换，从而改变笔画的倾斜角度。旋转中心通常选择笔画的端点或形心所在处，实现绕某点旋转或原地旋转的扰动功能。对于原始坐标点(x,y)，旋转后的坐标点(x',y')可通过式（3.4）计算得到：

$$\begin{bmatrix}x'\\ y'\end{bmatrix}=\begin{bmatrix}\cos\theta & -\sin\theta\\ \sin\theta & \cos\theta\end{bmatrix}\begin{bmatrix}x-x_0\\ y-y_0\end{bmatrix}+\begin{bmatrix}x_0\\ y_0\end{bmatrix} \tag{3.4}$$

其中θ为旋转角度，(x_0,y_0)为旋转中心。实际的旋转角度通常限定在±0.5°到±5°之间，避免造成不同笔画之间的角度突变，影响笔画整体流畅性。

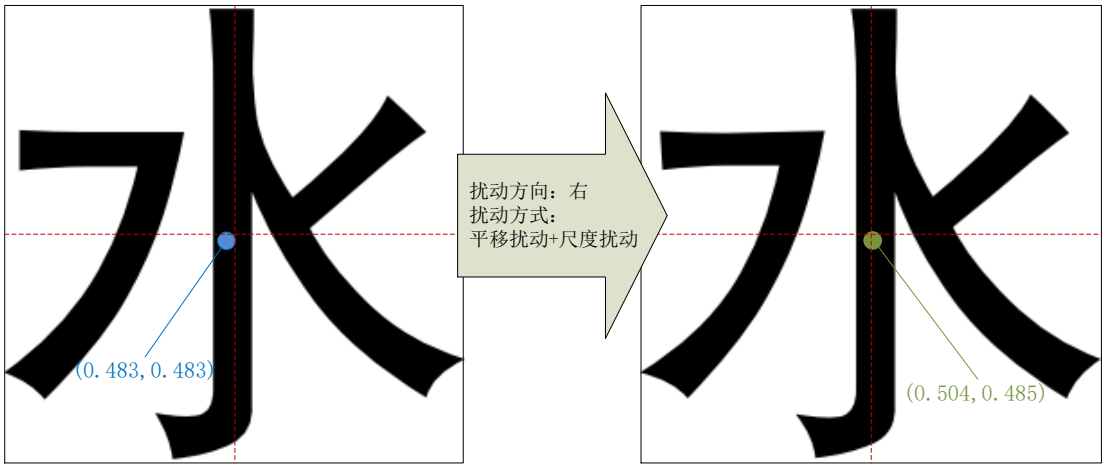
上述三种基本扰动方法可以在字符的某一个笔画上单独应用，也可以在一个笔画上通过多种扰动方法的叠加，形成更为复杂的字符形变特征。在多种扰动方式结合时，不仅可以增加生成字体变体的结构多样性，还能够有效分散单一扰动时可能集中产生的视觉痕迹，从而在整体上提升扰动后的字符视觉自然性和结构稳定性。

3.3 字体变体生成算法

3.3.1 自适应形心移动

如何在保持字符视觉一致性的同时引入可被检测的扰动，是字体变体生成任务的核心问题。由于不同字符在设计结构上存在天然的不对称性，尤其在拉丁字母、符号字符以及部分结构简单的汉字中，字符的几何形心往往偏离图像的中心位置。若通过固定方向和幅度的扰动策略对字形进行修改，容易导致非对称字符在扰动后出现笔画与空白区域的不均衡性加剧的问题。这类因为字形修改不当而产生的视觉失衡现象会影响字体变体的观感，影响字体变体的实用性。因此，针对字符自身的笔画分布特征，设计一种自适应确定扰动方向和幅度的字形扰动机制，对于实现字体变体的视觉一致性和扰动可检测性的平衡具有重要意义。

基于上述分析，本文提出了一种自适应形心移动算法，根据字符图像的初始形心位置动态选择字形扰动方向，并利用字形的局部结构特征灵活控制扰动的幅度大小。图 3.3 展示了使用自适应形心移动算法后的字形扰动效果，可见使用该算法可以在最大限度保持字符外观稳定的前提下引入可检测的变化。



(a) 原始字符图像　　(b) 扰动完毕的字符图像

图 3.3 自适应形心移动算法示意图

自适应形心移动算法根据字符图像的初始形心坐标确定扰动方向。通过前文所述的归一化绘图过程，字符轮廓可以被映射到长宽固定的标准图像空间中形成标准灰度字符图像，其色彩深度为 8 位。该图中每个像素的灰度值可以视为灰度权重，因此可以使用如式（3.5）所示的灰度加权法计算图像的形心：

$$C=(c_x,c_y)=\left(\frac{\sum_{i=1}^{W}\sum_{j=1}^{H}g(i,j)\cdot i}{\sum_{i=1}^{W}\sum_{j=1}^{H}g(i,j)},\frac{\sum_{i=1}^{W}\sum_{j=1}^{H}g(i,j)\cdot j}{\sum_{i=1}^{W}\sum_{j=1}^{H}g(i,j)}\right) \tag{3.5}$$

其中，C代表形心，c_x表示形心的x轴坐标数值，c_y表示形心的y轴坐标数值；W表示图像的像素宽度，i表示当前像素的x轴坐标数值；H表示图像的像素高度，j表示当前像素的y轴坐标数值；$g(i,j)$表示当前像素的灰度值。

在获得字符的形心位置后，依据形心在水平方向上的偏移量，自适应地确定扰动方向r。具体规则如下：若c_x位于字符图像左半区，即形心的轴坐标数值不大于图像宽度的二分之一，则扰动方向规定为1，意为向右施加字形扰动，以平衡整个字符图像的几何分布；反之，则扰动方向规定为-1，意为向左施加字形扰动。该判定规则可表示为式（3.6）：

$$r=\begin{cases}1, & 若c_x\le W/2\\ -1, & 若c_x>W/2\end{cases} \tag{3.6}$$

其中，c_x表示形心的x轴坐标数值，W表示图像的像素宽度。扰动方向决策完成后，根据字符图像原始的坐标尺度动态确定扰动的位移量P。位移量通过扰动幅度因子Δt来动态调节，具体计算方式如式（3.7）所示：

$$P=\Delta t\cdot W\cdot \mathrm{sign}(r) \tag{3.7}$$

其中，Δt默认取$1/16$；W表示图像的像素宽度；$\mathrm{sign}(x)$表示符号函数，当x大于0时取值为1，当x等于0时取值为0，当x小于0时取值为-1，此处用于指示扰动操作的方向。根据确定好的扰动方向和扰动幅度，对字符轮廓控制点坐标进行包含平移扰动和尺度扰动的综合扰动操作，其具体规则为：

1) 平移扰动：当扰动方向为1时，对字符的外轮廓坐标以右为扰动方向进行幅度为Δt的平移扰动操作，对内轮廓坐标以左为扰动方向进行幅度相同的平移扰动操作；当扰动方向为-1时，对字符的外轮廓坐标以左为扰动方向进行幅度为Δt的平移扰动操作，对内轮廓坐标以右为扰动方向进行幅度相同的平移扰动操作。
2) 尺度扰动：当扰动方向为1时，对形心在左半平面的轮廓进行幅度为Δt的尺度缩小扰动操作，对形心在右半平面的轮廓进行幅度为Δt的尺度加

大扰动操作；当扰动方向为–1时，对形心在左半平面的轮廓进行幅度为 Δt 的尺度加大扰动操作，对形心在右半平面的轮廓进行幅度为 Δt 的尺度缩小扰动操作。

按照上述规则实施扰动，可以在保持笔画原始结构不被破坏的前提下使字符的形心逐步向预期方向偏移。而且，由于引入自适应的扰动机制，该算法有效避免了在处理非对称字符时，已有算法的固定方向扰动策略引起的视觉失衡问题。需要注意的是，虽然在扰动规则制定过程中固定了扰动幅度 Δt，但根据字形轮廓的坐标分布不同，相同的 Δt 会产生不同的形心偏移量。因此，必须在扰动完毕后通过归一化绘图的方式重新渲染字符图像，并计算形心偏移量以观测形心扰动结果是否达到预期。

3.3.2 字体变体生成

自适应形心移动算法用于在坐标层面对字形进行精细扰动以实现形心坐标的位移。坐标修改完成后，需要进一步依靠字体变体生成技术，将修改后的字形坐标正确应用于字体文件中，生成可显示在文档中的字体变体。本研究提出一种字体变体生成算法，该算法结合了 3.3.1 中提出的自适应形心移动策略，利用循环修改字符坐标的方式引入形心位移，并通过可扩展标记语言（Extensible Markup Language, XML）作为中间格式实现字符数据的读取和写入。对于原始字体 F，生成字体变体 F' 的算法实施步骤如下：

1) 读取 F 中的 name 和 glyf 表数据，并将表数据以 XML 格式分别保存为 N 和 G；
2) 将表 G 中的当前待处理字符记为 u，获取该字符的一组轮廓坐标，记为 $g_0(u)$，并对 $g_0(u)$ 使用坐标归一化绘图方法生成字符图像 $I_0(u)$。其中下标 0 表示当前字符的迭代修改次数为 0；
3) 计算 $I_0(u)$ 的形心坐标，记为 $C_0(u)$。若 $C_0(u)$ 位于 $I_0(u)$ 的左半平面，将扰动方向 r 设置为 1；若 $C_0(u)$ 位于 $I_0(u)$ 的右半平面，则将 r 设置为 –1；

4) 根据扰动方向 r，对坐标 $g_0(u)$ 应用自适应形心移动算法进行单次坐标扰动。将经过此次扰动的坐标记为 $g_1(u)$。下标 1 表示当前字符 u 经过了 1 次迭代修改；

5) 重新对 $g_1(u)$ 使用坐标归一化绘图方法生成字符图像 $I_1(u)$，并计算其形心坐标 $C_1(u)$；

6) 计算形心位移量 $S=\left|C_1(u)-C_0(u)\right|$，若 $S\geq T$，视为对字符 u 的扰动已经完成，进入步骤 7；若 $S<T$，视为扰动未完成，重复步骤 4~6 以进行迭代式扰动，直到 $S\geq T$ 或者达到规定扰动次数上限。T 指阈值，本文的实验中取字符图像宽度的 1/32。设置扰动次数上限的目的是防止坐标扰动出现特殊情况时整个流程陷入死循环。在本文的实验中，根据经验将其设置为 3，即最多进行三次扰动，若三次扰动后字符的形心位移量 S 仍未达到阈值则直接进入下一字符的扰动环节；

7) 将处理完毕的坐标数据导入表 G，并从表 G 中获取下一个待处理字符，重复步骤 2~7 以实现对该待处理字符的扰动；若表 G 中的所有字符都已处理完毕，进入步骤 8；

8) 在表 N 中按需修改字体名称等信息，并将表 G 和表 N 导入原始字体 F 中以形成字体变体 F'。

上述步骤也可用图 3.4 所示的流程图表示，流程中使用的字体表文件包括 glyf 表和 name 表。其中，glyf 表是字符形状表，该表记录了字符轮廓的坐标控制点和指令序列，是生成字体变体的核心表；name 表是字体名称表，用于区分含扰动的字体变体和无扰动的原始字体。除上述两张表外，字体文件中还包含多个表结构，但在该扰动过程中其他表通常不进行修改。

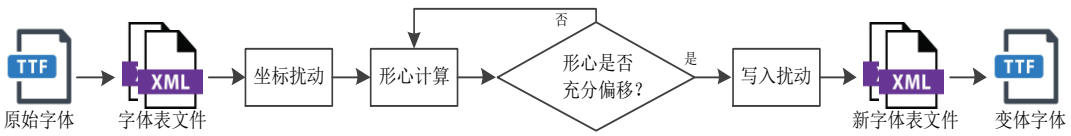


图 3.4 字体变体生成算法流程示意图

3.4 实验结果与分析

3.4.1 实验设置

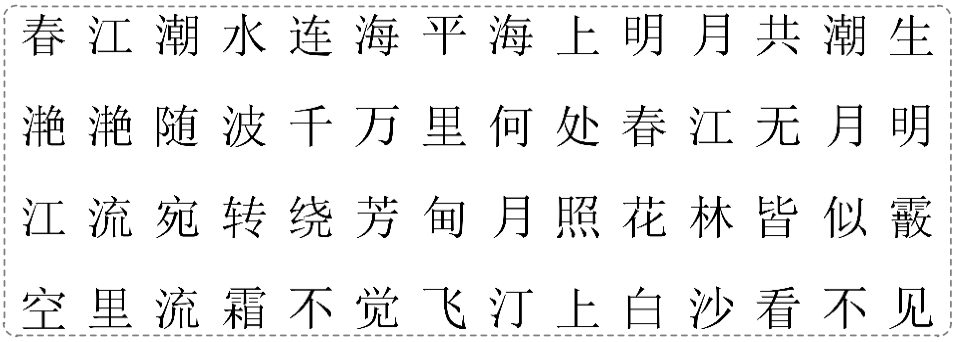
本节主要通过实验讨论基于自适应字形扰动的字体变体生成算法的性能，包括生成的字体变体在视觉质量、扰动可检测性和字体通用性三个方面。由于字体变体本身以字体文件形式存在，这不利于在实验中对字体变体的各项性能进行评估。所以在本实验中，使用字符图像作为字体变体的性能评估对象。字符图像通过归一化绘图方法形成，除非另有说明，字符图像的大小采用小四字号，对应的国际通用字符大小单位为 12 pt。在文档中显示时，字符的字号大小 f 与像素长宽 w 有如式（3.8）的对应关系：

$$w = S \cdot 4f / 3 \tag{3.8}$$

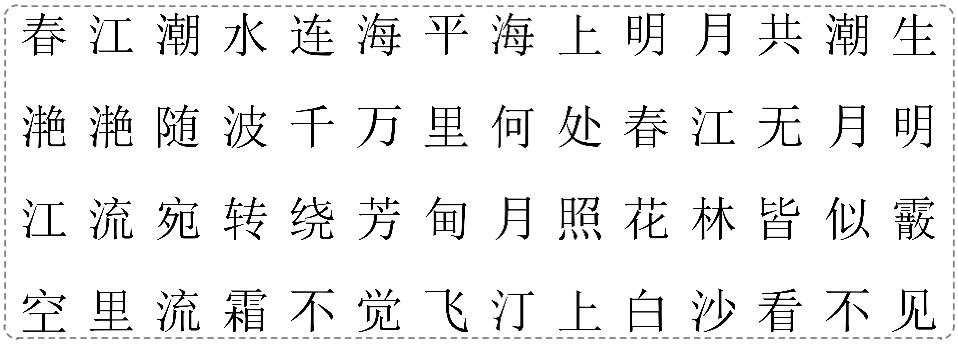
其中 S 为文档缩放比率。当 $S = 100\%$ 时，12 pt 的字号对应的像素长宽为 16×16 px。本节选用了现有性能最好的字体变体生成方法[29]作为基线方法，在实验中利用此方法生成了字体变体，用于对比验证本章提出方法的性能。实验中选择的字体变体参数为 $N = 2$，$r = 8$，与论文提供的默认参数保持一致。

3.4.2 视觉质量评估

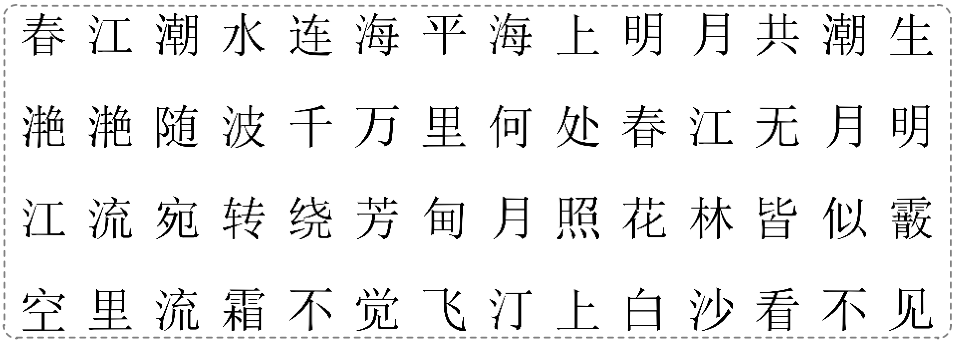
在本小节实验中，分别使用主观和客观评价的方式测试字体变体的视觉质量。主观评价实验邀请了 20 位测试人员参与实验，要求测试人员对字体变体的图像进行视觉质量评分。实验中选用的原始字体为宋体，包含三组字体大小不同的字符图像，分别为 12 pt、14 pt 和 16 pt；每组图像中的字体变体有三种不同的扰动幅度，分别为 $\Delta t = 1/24$、$\Delta t = 1/16$ 和 $\Delta t = 1/12$，对应生成的变体名称为变体 1，变体 2，变体 3。对于每种变体，使用相同的中文内容生成图像。图 3.5 展示了字体大小为 12 pt 的部分变体图像，用于展示主观评价实验的真实效果。为了更加明确地展示字体变体的轮廓细节，此处的图像分辨率调整为 240%，而在实验中的图像分辨率为 100%。主观评价实验的结果如表 3.2 所示，其中评分 5 分为满分，最低分为 1 分。



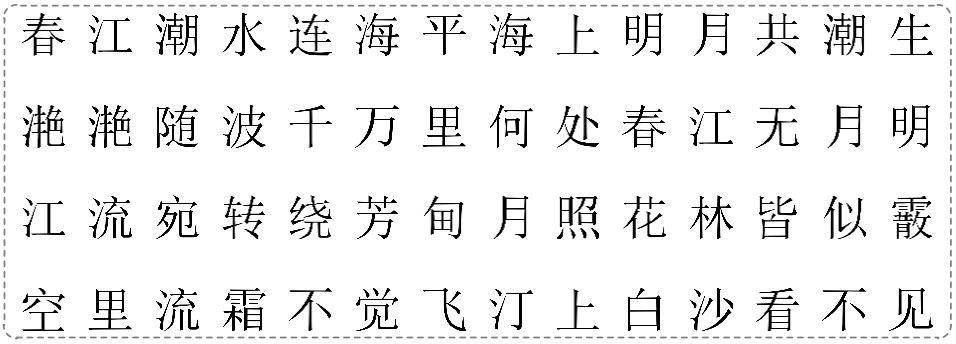
（a）原始图像



（b）变体 1



（c）变体 2



（d）变体 3

图 3.5 视觉质量主观评估所用字符图像示例：（a）变体 1，$\Delta t = 1/24$，（b）变体 2，$\Delta t = 1/16$，（c）变体 3，$\Delta t = 1/12$

从主观评价结果中可知，当$\Delta t = 1/24$或$\Delta t = 1/16$时，字体变体与原始字体的文档图像在人眼视觉效果上较为接近，即字体变体的扰动未对字符的视觉效果造成显著影响。而当扰动幅度继续加大时，主观评分的下降较为明显，表明字体变体的视觉效果受到了一定程度的影响。基于上述结果，本文选择将$\Delta t = 1/16$作为字体的默认扰动幅度，以实现扰动可检测性和视觉质量的平衡。

视觉质量的客观评估使用图像层面的指标 PSNR 和 SSIM 进行衡量。实验使用的原始字体为宋体，字符大小为 12 pt。生成的字体变体有 3 种扰动幅度，每个扰动幅度中，随机选取了 200 个变体字符的图像，并计算 PSNR 和 SSIM 的平均值，其结果如表 3.3 所示。从结果中可以得出，在$\Delta t = 1/12$时，本方法的指标与基线方法几乎相同；而在$\Delta t = 1/24$和$\Delta t = 1/16$时，本文方法生成字体变体的视觉质量指标均优于基线方法，表示生成的字体变体与原始字体的视觉质量更加接近。

综合视觉质量的主观评价与客观评价实验结果，可以看出本文提出的字体变体生成方法能够产生与原始字体视觉效果接近的字体变体。该字体变体相比过往的变体生成方法有更高的视觉质量，因而在作为秘密信息的载体使用时有更强的隐蔽性。

表 3.2 字体变体的视觉质量主观评分

字体大小	原始字体	变体 1	变体 2	变体 3
12 pt	5	4.85	4.70	4.10
14 pt	5	4.70	4.60	3.95
16 pt	5	4.75	4.55	3.65

注：5 分表示“视觉质量好”，4 分表示“视觉质量较好”，3 分表示“视觉质量一般”，2 分表示“视觉质量较差”，1 分表示“视觉质量差”。

需要提及的是，对于内容为字符的文档图像而言，其扰动后的客观评价指标 PSNR 和 SSIM 的绝对值通常低于内容为风景、事物或人物的自然图像。这是因为相比于其他图像，字符图像的笔画在变动时会引起大量像素的数值变化，且像素的变化幅度通常较为显著，如图 3.6 所示。在该示意图中，对字符施加$\Delta t = 1/24$

的扰动后，其像素在几乎所有图像区域都产生了变化，进而引起了 PSNR 和 SSIM 的降低。但在人眼视觉系统中，该字符扰动后与原始图像的差异很小。

表 3.3 字体变体的视觉质量客观评分

评价指标	变体 1	变体 2	变体 3	基线方法[29]
PSNR (dB)	21.53	20.37	19.50	19.96
SSIM	0.9598	0.9359	0.8795	0.8713

(a) (b) (c)

图 3.6 字符原始灰度图像与变体图像放大后的像素级差异示例：(a) 放大后的字符原始图像，(b) 放大后的字符变体图像，(c) 两幅图像的像素差异分布，取值范围为 0~1，本示例中差异分布的数值范围为 0~0.65

3.4.3 扰动可检测性评估

在本章提出的算法中，字体变体生成的最终目的是让字体变体携带可检测的几何扰动，从而用于携带秘密信息，因此需要设计实验以检测变体的字符上是否产生了有效的扰动。基于上述分析，本节实验中定义了如式（3.9）的扰动成功率 R，用于衡量对原始字体的扰动是否有效：

$$R = \frac{C_s}{C_0} \times 100\% \tag{3.9}$$

其中 R 代表扰动成功率，即在设定迭代次数内完成扰动的字符比例；C_s 代表在规定迭代次数内扰动完毕的字符；C_0 指需要扰动的所有字符。在本章使用的基于形心偏移的扰动算法中，使用式(3.10)来判断单个字符是否扰动完毕：

$$C_i = \begin{cases} 1, & 若S_i \geq T \\ 0, & 若S_i < T \end{cases} \tag{3.10}$$

其中 C_i 代表当前字符；S_i 代表字符形心相对于字符宽度的偏移量，以百分比表示；T 代表阈值，本文中设置为字符图像宽度的 1/32，对应百分比为 3.125%。

在本实验中，首先以宋体为原始字体生成字体变体，按简体中文常用字集 GB2312 一级汉字的字符编码生成字符图像，该集合为简体中文中使用率最高的汉字字符集合，共包含 3755 个字符。图 3.7 展示了对宋体施加幅度为 $\Delta t = 1/12$ 的扰动后，字符形心的偏移量曲线，图中的大多数字符形心都产生了 4%左右幅度的偏移。该曲线图表明本章提出的扰动规则能适用于字体中的几乎所有字符，并自动完成对字符轮廓的扰动，引入可检测的形心偏移。

表 3.4 不同扰动幅度下的扰动成功率

评价指标	变体 1	变体 2	变体 3	基线方法[29]
扰动成功率 R (%)	95.23	96.48	98.24	96.14

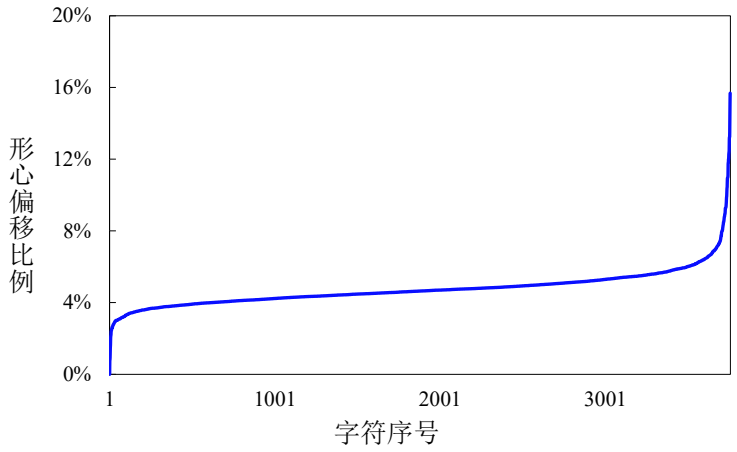


图 3.7 字符扰动后的形心偏移 S_i（升序排列）

为了验证本章方法能对不同幅度的扰动生效，实验测试了在不同扰动幅度的变体中的扰动成功率。利用宋体产生三种幅度不同的字体变体，记为变体 1、变体 2 和变体 3，对应的扰动幅度分别为 $\Delta t = 1/24$、$\Delta t = 1/16$ 和 $\Delta t = 1/12$，并选用基线方法[29]生成的字体变体作为对照。统计上述变体在字形扰动过程中的扰动成功率，得到的结果如表 3.4 所示。该结果表明，使用本章的字体变体生成方法

可以对原始字体中的绝大多数字符施加可检测的形心扰动，即便对于最小的扰动幅度$\Delta t = 1/24$，变体中也有超过 95%的字符含有可检测的形心偏移。而从与基线方法的对比数据可看出，本章提出的字体变体生成方法在$\Delta t = 1/16$和$\Delta t = 1/12$时的扰动效果都优于基线方法，说明本方法生成的字体变体能够稳定在字符中施加可检测的扰动。

3.4.4 字体通用性评估

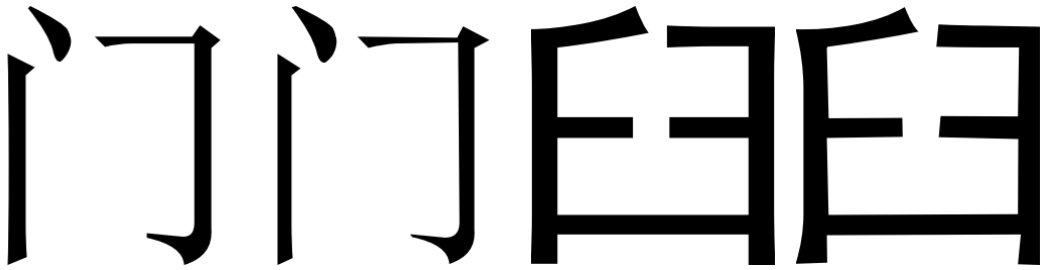
字体通用性反映了字体变体生成算法在不同语言和不同字体风格的字符集下的适应能力。本实验通过测试能否在不同语言和字体风格的字符集上的扰动成功率以评估所提出方法的字体通用性。实验中涵盖了三种语言类型，分别为中文、英文和数字，每种语言类型需要使用不同的字符集。其中，中文字符集为简体中文常用字集 GB2312 一级汉字，共 3755 个字符，字体分别为黑体、楷体与微软雅黑；英文字符集为 Basic Latin，共 95 个字符，字体为新罗马字体(Times New Roman, TNR)；数字的字符集为阿拉伯字符集 Arabic Basic，共 128 个字符，字体同为新罗马字体。对上述字符集所在的字体应用统一的自适应扰动流程生成字体变体，并分别统计不同扰动幅度下的扰动成功率R，其结果如表 3.5 所示。

实验结果表明，在不同扰动幅度下，本章方法在英文和数字字符中可以成功对字符施加扰动，仅在极少数字符中扰动效果未达预期；在简体中文字符集中，对除宋体之外的三种最常用字体的扰动都能达到 94%以上的扰动成功率。整体来看，本文提出的自适应字形扰动算法在不同语言和风格下均表现出良好的适应性，能够广泛支持不同语言类型文档的字体变体生成需求。

在实验中可注意到，采用相同扰动参数时，中文字符的扰动成功率会低于英文与数字字符。该现象主要因为中文字符的整体数量庞大且笔画分布多样，因而更容易出现扰动未能引起充分形心偏移的特殊字符。图 3.8 展示了扰动效果未能达到预期的字符图像示例，分别为宋体字符“门”和黑体字符“臼”，其中左侧为原始字符图像，右侧为扰动字符图像。但受益于算法的自适应选择逻辑，此类扰动不成功的字符占比极小，可以使用人工复核的方式对扰动进行微调，以保证扰动的完备性。

表 3.5 不同语言类型字体的扰动成功率 R (%)

扰动幅度	汉字-黑体	汉字-楷体	汉字-微软雅黑	英文字符-TNR	数字字符-TNR
$\Delta t = 1/24$	96.98	95.47	94.72	98.94	96.88
$\Delta t = 1/16$	97.20	96.25	95.71	100	98.44
$\Delta t = 1/12$	97.47	96.40	96.03	100	99.22



(a) 宋体“门”形心偏移：0.746%　　(b) 黑体“臼”形心偏移：0.563%

图 3.8 扰动效果未达到阈值的字符图像示例

3.5 本章小结

字体变体可用于在文档中承载秘密信息，但现有的字体变体生成方法难以在不影响视觉质量的前提下保证扰动可检测性。本章提出了一种基于形心偏移的字体变体生成方案，可生成出与原始字体视觉差异较小，且字符含有可检测几何扰动的字体变体。该方案首先将字体中的字符坐标通过归一化绘图进行可视化，再根据字符图像的原始形心对字符坐标进行有规律的扰动，从而引入形心的偏移。最后，利用自适应迭代扰动和可视化结合的字体修改方法，确保字体中的每个字符形心偏移量达到设定阈值。实验结果表明，本章提出的字体变体生成算法可以在保持变体高视觉质量的前提下，为变体中的绝大多数字符引入可检测的形心扰动。此外，得益于坐标修改方法对形状的自适应能力，本算法可适用于中文、英文和数字等不同语言种类与字体风格的应用场景，相比其他单语言字体生成方法具有更强的通用性。

第四章 基于语义投影分割的字形文档水印算法

本章主要介绍了基于语义投影分割的字形文档水印算法。其中，4.1 节主要介绍了字形文档水印算法的研究背景和关键问题；4.2 节主要介绍了该水印算法的整体框架结构；4.3 节和 4.4 节分别介绍了水印的嵌入和提取算法；4.5 节为算法的实验设计与结果分析；4.6 节为本章小结。

4.1 引言

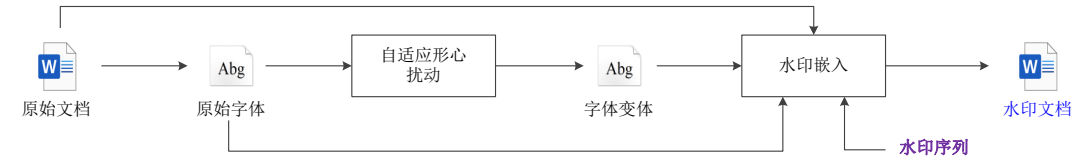
文档水印是信息安全领域中用于保护电子文档的一类技术方法。其技术关键在于如何选择合理的水印载体与嵌入方式，使得嵌入的信息既不对文档造成显著的视觉影响，又能保持较高的提取准确率与鲁棒性。在过往的文档水印研究中，研究人员多关注于段落结构、间距排版或背景纹理等文档宏观层面的调制方式，但此类方法难以在大嵌入容量的前提下保证水印的隐蔽性和鲁棒性。相比之下，基于字形调制的文档水印方法因其作用于字符几何层面的特点，能取得更高的嵌入容量和更低的视觉扰动，此类方法也因此成为目前的主流文档水印方法[70]。高性能的字形文档水印需要解决两个核心问题：一方面，水印算法需要实现字符级别的水印嵌入，使得每个字符既能稳定地携带信息，又不破坏文档整体的视觉协调性；另一方面，水印需要在经过跨媒介传输或截屏等信道干扰后，仍能准确地从文档图像的字符中提取出来。尤其是在文档中字符间距紧凑、字号较小的场景下，过往的字形水印提取方法往往难以正确地分割文档字符，导致水印提取的准确率明显下降。

针对上述问题，本文提出一种基于语义投影分割的字形文档水印算法，该算法利用自适应扰动生成的字体变体作为水印的承载单元，并设计了一种结合文本语义结构与文档像素信息的字符分割策略用于水印信息提取。借助自适应扰动的字体变体，本水印算法减少了嵌入过程中的字符改动数量，进而提升了算法隐蔽性；同时，由于语义信息的引入，本算法在分割字符并提取水印的过程中能取得更高的准确率，尤其是在字符之间的间隔较小时。相比其他字形文档水印方法，

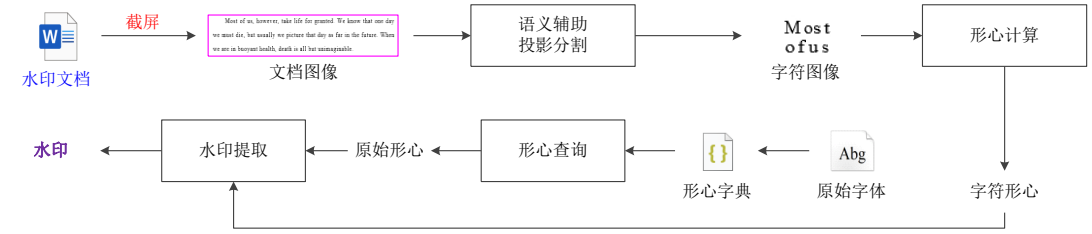
本算法不仅能保持在多种文档使用场景中的格式鲁棒性，同时水印容量和隐蔽性能也得到了提升。

4.2 算法整体框架

基于语义投影分割的字形文档水印算法包含两个阶段，分别为水印嵌入阶段和水印提取阶段，整体算法框架如图 4.1 所示。在水印嵌入阶段，算法以网络流通中的常见文本文档作为原始文档输入，选定部分段落中的字符作为嵌入位置，并根据待嵌入比特值决定该字符使用原始字体还是变体字体。若比特为“0”，则保留原始字体；若为“1”，则将目标字符替换为对应的字体变体。随后，将字体替换完毕的文档重新保存，形成包含水印的文档作为输出。



(a) 水印嵌入阶段



(b) 水印提取阶段

图 4.1 水印算法整体框架示意图

在水印提取阶段，算法首先对水印文档图像进行字符级别的切分。为此，本研究提出了一种结合语义信息与视觉投影的语义辅助投影分割方法：先通过光学字符识别技术（Optical Character Recognition, OCR）[72]识别文本行与字符边界框，在此基础上使用投影分割算法对字符块进行结构细化，提取边缘完整的字符图像。随后，计算每个字符图像的水平形心位置，并与预先构建的字体形心字典进行比较。当字符的形心偏移量超过设定阈值时，判定该字符来自字体变体，水印比特值为“1”；否则判定为原始字体，水印比特为“0”。最终，按照嵌入顺序还原出完整的二进制水印信息。由于在嵌入阶段选用了字体变体与原始字体共同

作为水印载体，本方法显著减少了对文档外观的影响，有效保证了水印的隐蔽性；而借助所提出的语义辅助投影分割模块，本方法在面对字符间距较窄或字体较小的实际应用场景时，仍能稳定地完成字符分割任务，并且有效减少了因为字符分割错误引起的水印差错传播现象，进而提升了水印的提取准确率。

4.3 水印嵌入算法

鉴于字形文档水印不修改文档排版格式且不变动文本语义等诸多有益特性，本章设计了一种基于字体变体的字形水印嵌入方法。该方法以字符的字体类型作为调制单元，利用字体变体与原始字体之间在字符形心层面的微小可测差异完成字符级别的信息承载。嵌入过程可简述为三个步骤：嵌入位置确定、水印比特值编码和字体替换，过程示意如图 4.2。

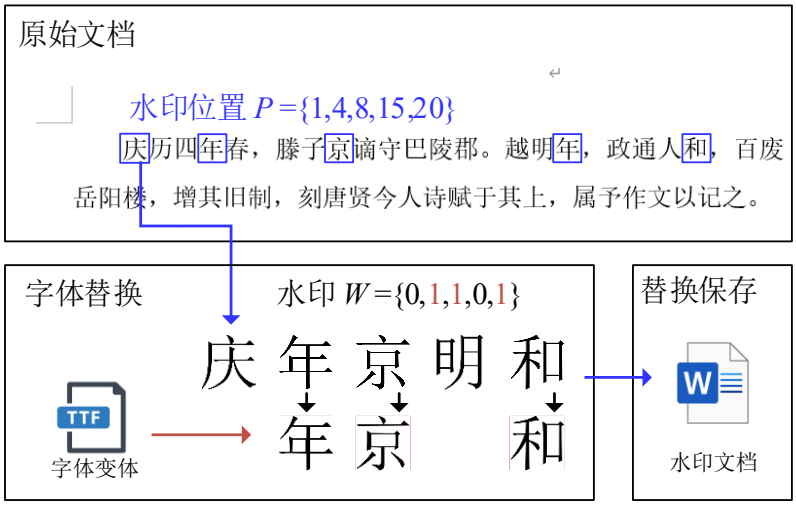


图 4.2 水印嵌入算法示意图

给定一个待处理的文本文档 D，其中的文本部分记为 $T=\{t_1,t_2,...,t_n\}$；给定一串待嵌入的水印比特序列 $W=\{w_1,w_2,...,w_m\}$，其中 $w_i\in\{0,1\}$。算法需要从 n 个字符位置中选择 m 个用于水印嵌入。为了避免对文档视觉的扰动过于集中，采用伪随机选择策略确定嵌入位置，由用户指定的密钥 K 作为随机种子，对文档中字符进行均匀采样，从而得到嵌入位置集合 $P=\{p_1,p_2,...,p_m\}$，其中 $p_i\in[1,n]$。对于每一个嵌入位置 p_i，根据第 i 位的水印比特 w_i 决定字符 t_{p_i} 的选用字体：

1) 若 $w_i=0$，则 t_{p_i} 保持使用其原始字体；

2) 若 $w_i=1$，则 t_{p_i} 使用由原始字体预先生成的字体变体。

此过程可视为将水印比特值编码到了字体选用中。随后，将含有混合字体的文档导出为水印文档 D_w，此时视为水印嵌入完成。需要注意的是，除字体外，D_w 的格式应与 D 保持完全一致。同时，为了确保嵌入过程的通用性，算法预先剔除了文本部分 T 的特殊符号和零宽字符等特殊字符。

4.4 水印提取算法

在本文的研究中，主要针对的数字传输信道为文档——图像，故水印提取算法将截屏后的文档图像作为输入，并在考虑图像失真的情况下完成字符定位与比特提取操作。在该传输信道中，可能包含的失真有分辨率变化、文档渲染差异和图像压缩等。针对上述失真，本文提出了一种水印提取算法，该算法使用语义辅助投影分割法进行字符分割，随后使用形心字典记录的字体信息进行水印判定。其中，语义辅助投影分割法将 OCR 识别与传统投影分析方法进行了结合，用于提升水印提取过程中的字符定位准确性，进而增加水印判别精度；形心字典以键值对形式储存了原始字体中的字符的形心信息，可在水印判别过程中使用以优化提取过程的形心判别速度，同时避免水印序列的差错传播。水印提取算法的整体流程图如 4.3 所示。

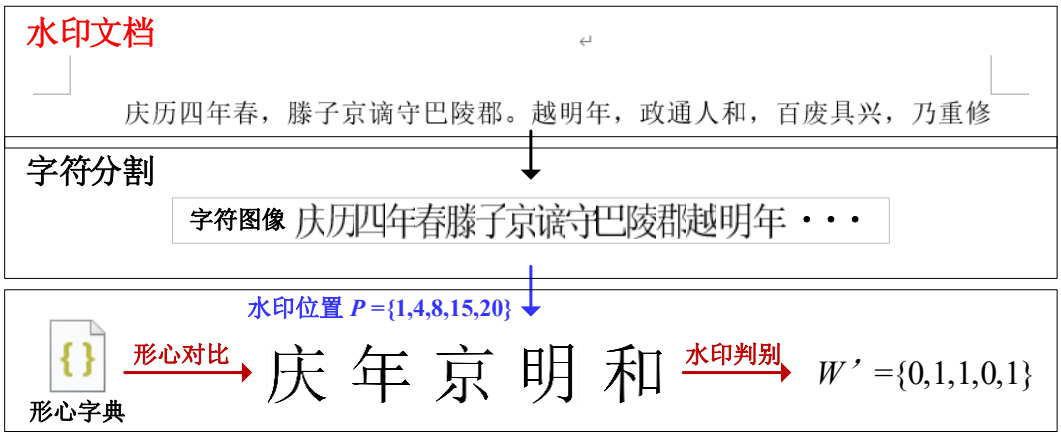


图 4.3 水印提取算法示意图

对于输入的文档图像，首先采用 OCR 技术对图像中的文本行和字符进行初步定位。虽然 OCR 过程提供的字符框可以粗略地实现字符分割的功能，但由于

OCR 性能受限于模型训练及图像质量，其定位过程中容易产生字符粘连或切割误差。受启发于传统图像投影分割方法，本研究提出如图 4.4 所示的结合 OCR 与投影分割法的语义辅助投影分割法，其实施步骤如下：对于如图 4.4 第一列的黑色边框文本图像，首先对图像使用 OCR 技术进行初步分割，以获得单个字符的粗分割候选框，得到如图 4.4 中，第二列粉色边框所示的字符图像。该图像中通常只包含单个字符，但其边界会随每次 OCR 过程发生抖动，不利于后续的水印判别。随后，对每个字符的粗分割图像依次进行垂直方向和水平方向的投影密度计算，并根据投影结果微调图像的边界，排除字符周围的冗余空白区域，只保留字符的最小外接矩形框，其结果如图 4.4 中第三列蓝色边框的图像所示。最后，利用文本行中图像矩形框宽度的众数值，对该行字符图像的分割结果进行筛选，将宽度异常的矩形框重新分割，以修复分割时出现的过分割和欠分割问题。

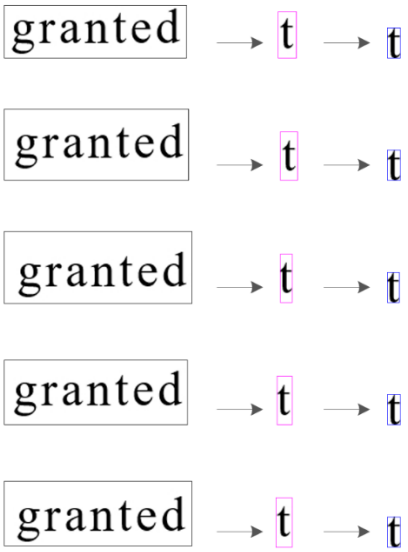


图 4.4 语义投影分割算法效果示意图

通过语义投影分割算法获取字符图像后，使用式（4.1）所示的灰度加权方式计算图像在 x 轴方向的形心相对位置 c_x：

$$c_x = \frac{\sum_{x,y} x \cdot g(x,y)}{W \cdot \sum_{x,y} g(x,y)} \tag{4.1}$$

其中 W 为字符图像的宽度，$g(x,y)$ 为字符图像在位置 (x,y) 的像素灰度值。将计算得出的 c_x 与预先建立的字体形心字典进行比对，计算形心偏移量。字典中保存

了每个字符在原始字体中归一化后的形心参考值。若该字符的形心偏移量 S 超过设定阈值 θ，即满足式（4.2）：

$$S = |c_x - c_0| > \theta \tag{4.2}$$

则判定该字符使用的是字体变体，嵌入比特为“1”；否则视为原始字体，嵌入比特为“0”。其中，c_0 表示字符的原始 x 轴形心；θ 根据字体渲染分辨率经验设定，通常为归一化图像宽度的 $1/32$。最后，根据嵌入时的水印字符位置，依次利用字典进行水印比特判定即可恢复出完整水印比特序列。

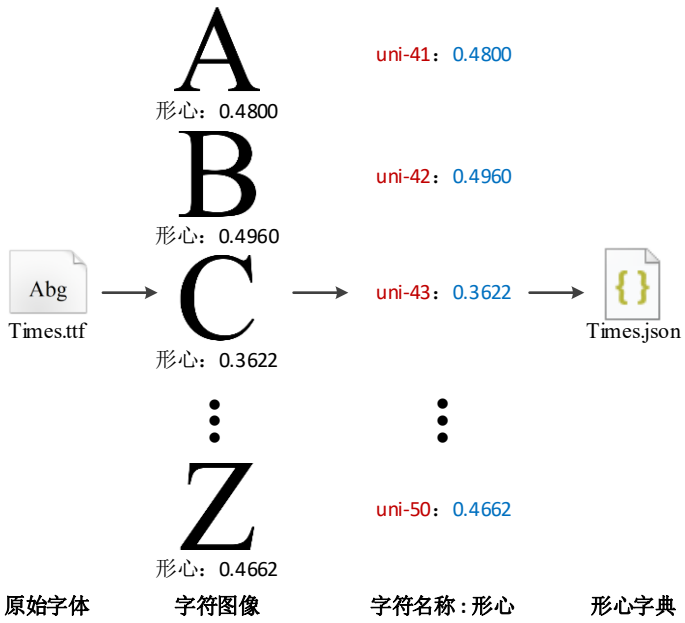


图 4.5 字体形心字典示意图

在水印提取过程中，形心字典的构建与使用起到了重要的辅助作用。该字典使用键值对（Key-Value Pair）的形式存储字符图像的形心比例数据，其示例如图 4.5，通过将每一个支持嵌入的字符原始坐标数据渲染为归一化字符图像，并计算其形心水平坐标与图像宽度的比值形成。为提升跨平台的适用性，字典构建时使用标准字号与固定归一化图像尺寸进行渲染，使其在不同显示环境下仍具有良好的几何稳定性。

4.5 实验结果与分析

4.5.1 实验设置

针对提出的字形文档水印算法，本节从三个角度对水印算法性能进行系统实验：视觉质量、鲁棒性与水印容量。其中，视觉质量用于衡量水印算法的隐蔽性，提取准确性与字符分割准确性用于综合衡量水印算法的鲁棒性。实验中，采用 Microsoft Word 2019 对文档内容进行编辑、字体替换与渲染，以模拟真实的文档生成与传播流程。根据不同的语言内容，分别选取对应语言的通用字体与排版格式构建实验文档。其中，英文实验所用文本选自英文小说《Three Days to See》，原始字体设置为 Times New Roman，字号为 12 pt；中文实验所用文本节选自古诗词，原始字体设置为宋体，字号同为 12 pt。水印信息的嵌入通过在字符上替换为对应的字体变体实现，其中比特“0”表示原始字体，“1”表示字体变体。变体字体通过前述的自适应扰动方法生成，扰动强度设置为 $\Delta t = 1/16$。

作为实验对照的基线方法是现有性能最好的字形文档水印方法[29]，其字体变体生成参数为 $N = 2$，$r = 8$，与论文[29]提供的默认参数保持一致。对于其他种类的文档水印方法，如基于排版和背景格式调制、基于语义调制等，由于这些方法不以单个文档字符作为水印载体[73]，其水印容量都较为有限，不适用于如今的文档水印应用场景，因此本文仅在 4.5.4 小节中对这些方法进行了水印容量的比较。

4.5.2 隐蔽性评估

为了验证所提出水印算法在文档层面具备良好的视觉隐蔽性，本节重点从视觉感知效果与整页图像的客观指标两个角度展开实验分析。不同于第三章在字符层面对字符变体的视觉质量进行评估，本章的隐蔽性实验侧重于考察水印嵌入后在段落级或页面级别是否对整体文档观感造成可察觉的影响。

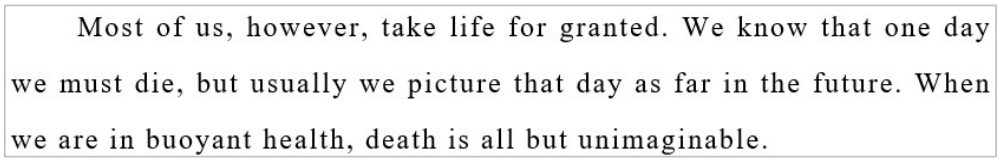
实验对比了使用本章方法与基线方法[29]嵌入水印后的视觉效果，选用的中文文档共包含 733 个中文字符，英文文档共包含 701 个英文字符，由于图像规格限制，此处只展示了文档经过裁剪后的一部分截图。嵌入水印后的视觉展示效果分别如图 4.6 与 4.7，其中(a)为原始文档截图；(b)为采用本文方法嵌入水印后的文

档截图；(c)为基线方法[29]嵌入水印后的文档截图。从图像对比可以观察到，本文方法产生的水印字体在视觉外观上与原字体的宽高度一致，并且在字符轮廓上未引入可见伪影或明显结构变化，能与原始字体保持整体视觉的一致性。而基线方法在嵌入水印时会造成个别字符出现笔画缺失或错位的现象，影响文本页面的整体排版美观和可读性。

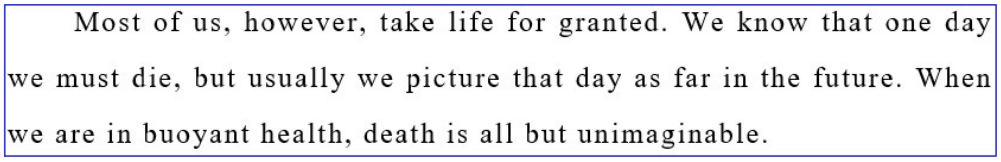
为进一步量化水印嵌入所带来的视觉扰动，本文对原始文档与水印文档之间的整页截图图像进行了客观指标的计算，结果如表 4.1 所示。从该结果可以看出，本章方法在隐蔽性的客观实验指标上的表现优于基线方法。对于文档图像的 PSNR 和 SSIM 绝对值较低的现象，本文在 3.4.2 节已进行了相关分析与说明。通过主观与客观评估的综合实验结果可以看出，本章方法在水印嵌入文档后不仅没有在人眼感知上引入明显的差异，在像素级别的性能指标上也优于现有的字形文档水印方法。

表 4.1 水印算法的视觉质量客观评分

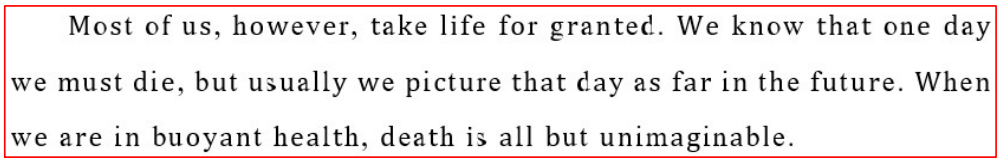
评价指标	本章方法-中文	基线方法-中文	本章方法-英文	基线方法-英文
PSNR (dB)	**27.70**	25.94	**33.14**	30.39
SSIM	**0.9641**	0.9357	**0.9957**	0.9940



(a) 原始文档



(b) 本章方法



(c) 基线方法

图 4.6 不同水印算法视觉效果对比（英文）

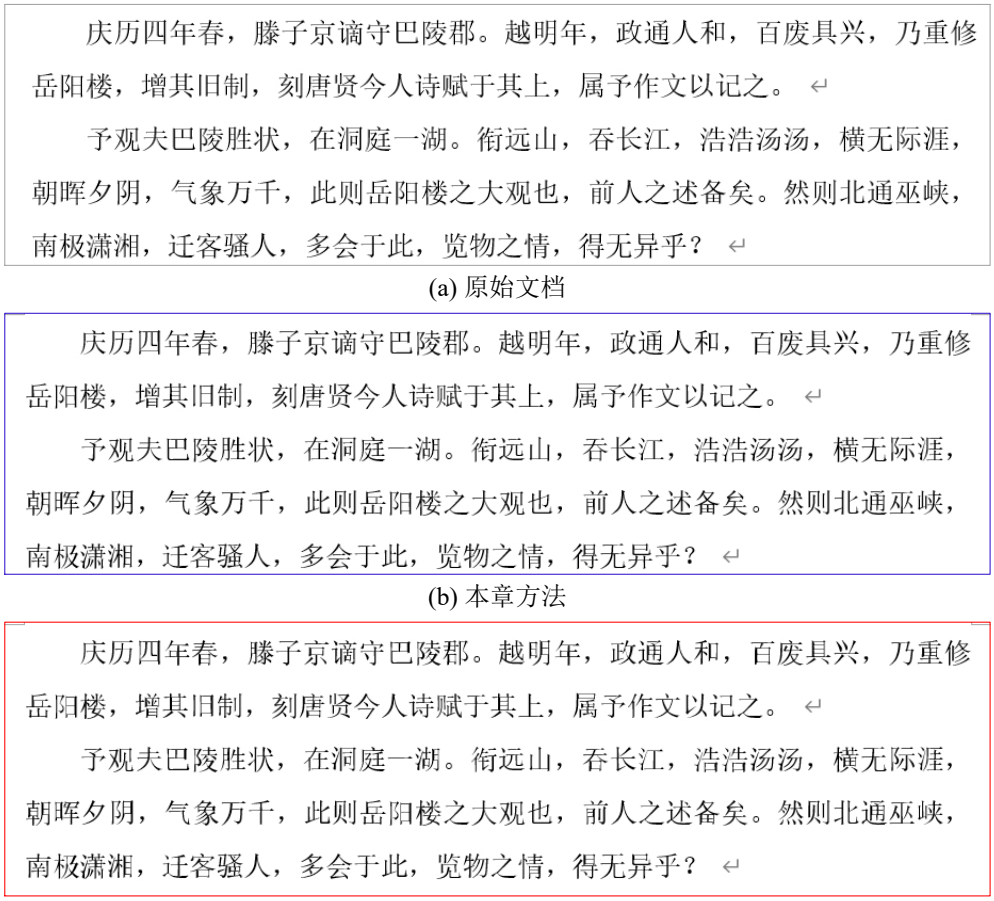


图 4.7 不同水印算法视觉效果对比（中文）

4.5.3 鲁棒性评估

4.5.3.1 提取准确性评估

在本小节实验中，采用水印提取准确率作为所提出算法的鲁棒性衡量指标。对于电子文档而言，可能遭受的信道失真与攻击有文档截图、字体缩放、图像压缩和跨平台渲染等，本小节将围绕上述内容进行鲁棒性测试。

文档截图是文档应用场景中最容易实现的信息窃取操作之一。图 4.8 和 4.9 分别展示了在中英文文档中经过文档截图后的水印提取准确率。其中，英文文档选用 Times New Roman 字体，将字号从 10 pt 至 20 pt 分别设定为实验变量，保持文本内容、和扰动强度 $\Delta t = 1/16$ 不变。中文文档采用的字体为黑体，字号范围同样设定为 10 pt 至 20 pt。实验截图使用常见社交软件腾讯 QQ 的截图工具保存为 PNG 图像格式。图中实验结果显示，本章提出的水印算法在中文文档中的水印提取准确率保持在 92%以上，在英文文档中的准确率也高于 88%，并且在两种

语言的应用中都高于基线方法。水印算法在中文文档中的性能更好的原因是相比于英文字形，中文字形的轮廓复杂度更高，水印施加在字形轮廓上的扰动更容易被系统检测并识别。

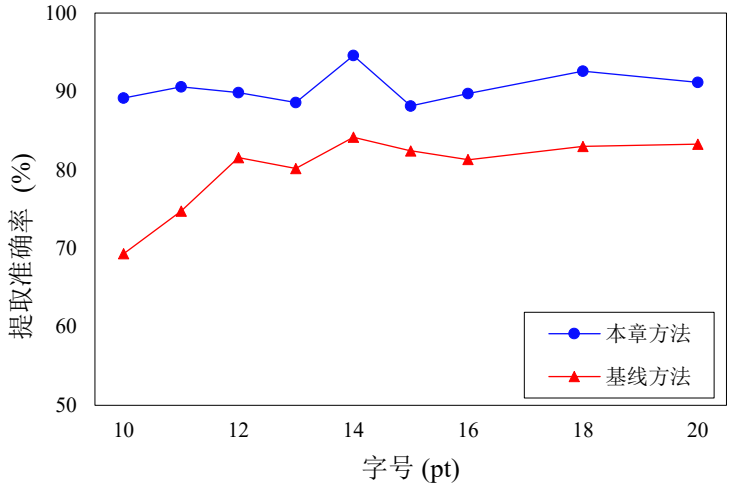


图 4.8 不同字号的英文文档水印提取准确率

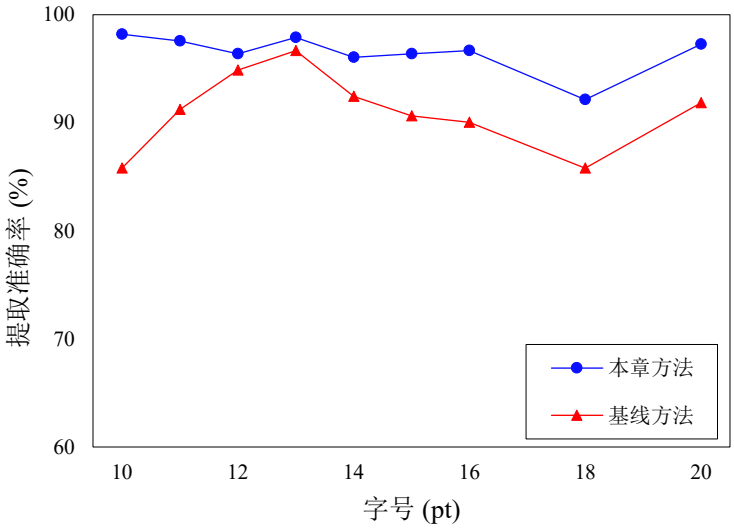


图 4.9 不同字号的中文文档水印提取准确率

实验还发现，在一些特殊字号的文档中施加水印时提取准确率略有波动，如 13 pt 和 15 pt。这主要由于此类字号是非标准字号，若受到了非官方的字形轮廓改动，此类字号的字符在文档中渲染时容易引发轮廓像素位置的偏移，从而影响扰动特征的保留与形心判定过程。尽管存在该问题，所提出算法仍能在多种字号以及不同语言条件下维持较高提取精度，体现出算法的跨语言、跨字号鲁棒性。

4.5.3.2 字符分割准确性评估

字符分割的准确与否将直接决定字形文档水印的提取效果，该指标可通过在提取过程中出现的分割错误数量进行衡量：出现的错误数量越多，字符分割表现越差。字符分割过程中出现的错误可主要分为如图 4.10 所示的三类：

a) 过分割：对于字符宽度显著大于其他字符的单个字符，其可能被识别为多个字符，从而在分割后成为两个不完整的字符图像；

b) 欠分割：对于两个间距偏小的邻近字符，其可能被识别为一个字符，从而在分割后的一个字符图像中包含两个字符；

c) 语义识别错误：对于 OCR 过程，字符可能被错误识别为外形相近的另一字符，从而造成语义识别错误。

上述三类错误可能在分割时同时或单独出现，其概率取决于字符分割方法的完备性以及文档图像中字形的失真程度。在本实验中，统计了在不同语言和字号文档的水印提取过程中的字符分割错误数量并统一以百分比表示，结果如表 4.2。从该结果可看出，基线方法在不同字号的英文文档中均有较小概率出现分割错误，而在中文文档中出现错误的概率显著增大。原因是在字号相同的情况下，中文文档因为字符间本无空格，排列更加紧凑等原因，其出现分割错误的可能性将高于英文文档。相比之下，本文方法由于引入基于语义辅助的投影分割策略，有效缓解了标准格式文档中字间距较小导致的过分割和欠分割问题。

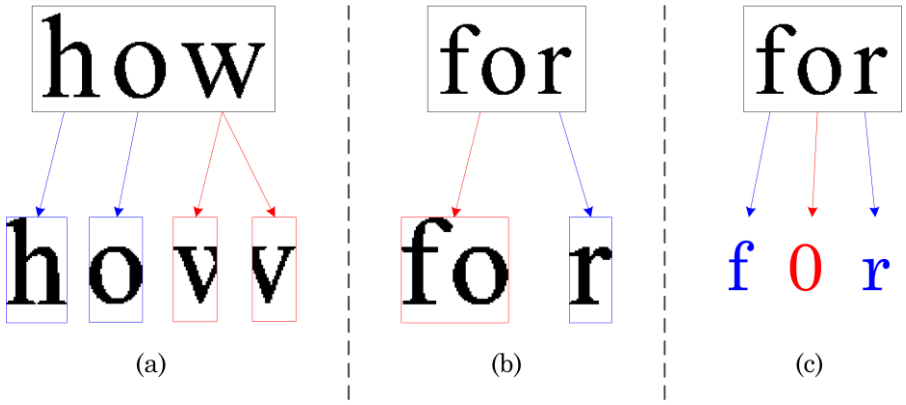


图 4.10 字符分割过程可能出现的错误示例：(a) 过分割，(b) 欠分割，(c) 语义识别错误

需要提及的是，在进行提取准确率评估部分的实验时，对上述分割过程的错误进行了手动修正以避免分割错误对结果的影响。实际应用基线方法时，此类分割错误不仅会影响当前字符的水印判别，还会因为错误传播机制[74]对后续水印判

别造成影响。而在本章提出方法中，由于在字形分割过程中使用了语义信息进行辅助验证，单个字符的水印判别错误将不会影响后续水印，这显著增强了算法的水印识别率。

表 4.2 不同字号下出现的英文字符分割错误率 (%)

字号 (pt)	10	11	12	13	14	15	16	18	20
本章方法-英文	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
基线方法-英文	0	0.71	1.72	0.86	1.14	2.28	1.85	2.85	1.43
本章方法-中文	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
基线方法-中文	3.37	12.92	10.96	2.81	4.21	3.09	4.78	5.38	6.46

4.5.4 水印容量

水印容量用于衡量水印的信息承载能力强弱。相比基于背景或语义调制的水印方法，本文采用以字符为最小嵌入单元的策略，通过字体变体进行比特调制，使每个字符理论上可嵌入 1 比特水印信息，从而显著提高了单位文档的容量上限。具体的理论水印容量对比如表 4.3 所示，数据单位为每字符可嵌入比特数 bpc。数据表明，基于背景格式调制和基于语义调制的文档水印方法的水印容量均不超过 0.1 bpc，即承载 1 比特水印平均需要 10 个以上的字符，此水印容量显然不能满足如今的文档水印要求。基于字形调制的水印方法由于将载体尺度缩小到了字符级别，其水印容量相对较大。在此基础上，本章提出方法可达到高于现有字形水印方法的嵌入容量，因此有更强的信息承载能力。

表 4.3 不同文档水印方法的理论水印容量对比

文档水印方法	背景调制水印方法[16]	语义调制水印方法[9]	字形调制水印方法[29]	本章方法
理论水印容量 (bpc)	0.003	0.073	0.500	**1.000**

在真实文档中，由于文档排版和语言风格等因素的限制，水印的实际容量通常小于等于理论值。研究人员使用嵌入比率描述水印的实际容量与理论容量之间

的关系。鉴于本章方法的理论水印容量为 1 bpc，该方法的实际水印容量与嵌入比率之间的关系为如图 4.11 所示的直线。在实际应用时，文档中通常包含多种不适合嵌入水印的字符格式，例如标题、页眉页脚和标点符号等，这些字符的占比通常不超过 20%。因此推荐将嵌入比率设置为 80%，以保留部分冗余容量，避开此类不适合嵌入水印的字符。该嵌入比率设置是经验参数，可在不影响水印性能的前提下，视文档内容与格式的差异进行上下微调。

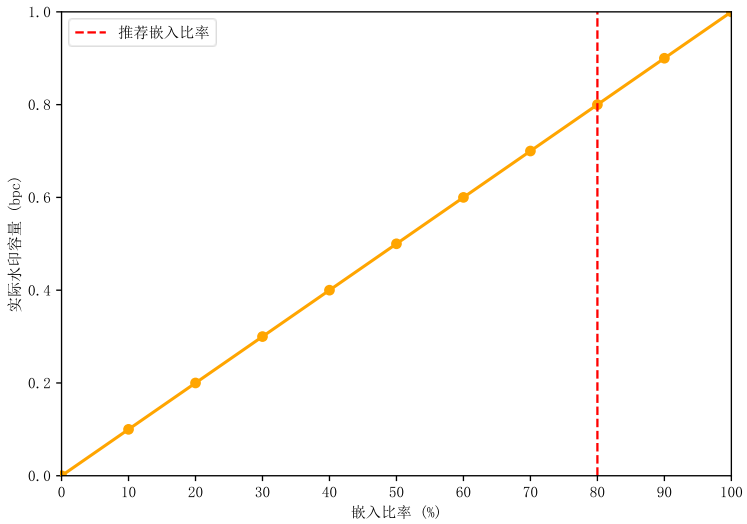


图 4.11 算法单位水印容量与嵌入比率关系示意图

4.6 本章小结

本章提出了一种基于字形调制的文档水印算法方案，旨在以字体变体作为水印载体在文档中嵌入秘密信息。该方案将字符形心位置作为水印的判别依据，在嵌入阶段通过对文档字符的字体差异化替换完成水印比特的嵌入；在提取时，首先对文档使用结合 OCR 识别与投影修正的字符分割算法，以实现对字符轮廓的稳定分离，再对分离完毕的字符图像通过形心位置判定水印。实验结果表明，本章提出算法在多种语言和字体环境下都能有效地在文档中嵌入和提取信息，在水印隐蔽性和嵌入容量方面的性能也优于现有方法。此外，所提出算法在文档字符间距较小的场景下仍能表现出较好的水印提取效果，这得益于字符分割过程使用的的语义辅助投影分割方法。该方法不仅借助 OCR 的字形候选框避免了字符的过分割和欠分割问题，还利用识别到的语义信息减少了差错传播的出现概率。

第五章 基于孪生网络的字形文档水印算法

本章主要介绍了基于孪生神经网络的字形文档水印算法。其中，5.1 节主要介绍了神经网络字形水印算法的概况与研究背景；5.2 节主要介绍了所提出网络的框架与形成方法；5.3 节介绍了基于该神经网络的字形水印嵌入与提取流程；5.4 节为所提出算法的实验设计与结果分析；5.5 节进行本章小结。

5.1 引言

在字形文档水印研究中，现有方法大多使用字符或字体本身的属性特征作为水印提取的判别依据，在字符图像受到扭曲、模糊、压缩或跨平台渲染的条件下，这些方案的水印判别准确率将明显下降。得益于神经网络对于分类任务的强大适应能力，近年来的研究人员开始将深度神经网络引入字形文档水印中，通过学习字符图像中的高维特征以进行水印的提取和判断。Xiao 等人[27]设计了一种针对英文的字形水印方法，在嵌入阶段通过人工设计承载水印的字体码本，并在提取阶段利用卷积神经网络分类器识别水印。由于需要为每个字符单独训练分类器，此方法只适用于英文和数字等字符数量较少的字体和语言。Yao 等人[75]提出了一种基于字体风格迁移的字形水印方法，该方法借助字体风格迁移的原理生成含有水印的字体，并通过计算字体图像与原始图像的欧氏距离来判定字体是否含有水印。此方法生成的水印字体拥有更好的视觉隐蔽性，但由于水印判决阶段仍使用字符图像的几何特征作为依据，此方法在面对各种信道失真时的鲁棒性不强。此外，此方法需要针对文档中使用的每一种字体分别进行网络训练，这降低了方法的语言和字体通用性，并增加了方法的应用成本。

因此，如何设计一种通用的水印框架以在不同的语言和字体中嵌入水印信息，并能够在经过各种信道传输失真后提取水印，成为字形文档水印领域的一个重要课题。孪生神经网络作为一种对比学习的网络结构，在刻画样本之间的相似性方面具有天然的优势，已经广泛应用于人脸识别、签名验证与医学图像对比等领域[76-81]。受到该结构的启发，本文提出一种利用孪生神经网络架构对字体扰动

特征进行判别的文档水印算法。该方法在嵌入阶段利用自适应字形扰动生成字形变体以承载水印，并训练网络以自动学习字形扰动过程中引起的字符形态差异；在提取阶段，使用训练完毕的网络判别字符图像的水印比特。由于在水印提取阶段不再依赖传统的图像处理方法进行判别，本方法在文档面对不同信道失真时有更好的鲁棒性，对于不同水印字体的识别通用性也更强。

5.2 字形判别孪生网络

5.2.1 网络结构

本文所设计用于字形判别的双分支孪生神经网络结构如图 5.1 所示，该网络通过对比输入的字符图像对之间的高维特征距离，实现对字符图像是否来自水印字体变体的判别。

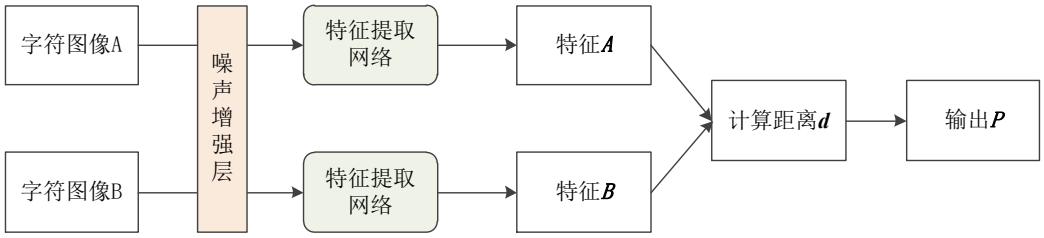


图 5.1 用于水印判别的双分支孪生神经网络结构示意图

孪生网络的核心结构是两个共享权重的特征提取子网络。对于输入的一对字符图像 I_1 和 I_2，将其分别传入两个特征提取子网络，并映射为相同维度的特征向量 $f(I_1)$ 和 $f(I_2)$。随后，通过式（5.1）计算二者的欧氏距离，作为判断相似度的依据：

$$d = \left\| f(I_1) - f(I_2) \right\|_2 \tag{5.1}$$

其中，$\left\| X \right\|_2$ 表示计算 n 维向量 $X = (x_1, x_1, ..., x_n)$ 的 L2 范数，其计算方式如（5.2）所示：

$$\left\| X \right\|_2 = \sqrt{x_1^2 + x_2^2 + \cdots + x_n^2} \tag{5.2}$$

对于两个图像的特征向量，其欧氏距离越小，表示两个图像越相似；距离越大则表示两个图像的差异越显著。在网络训练时，通过采用如式（5.3）所示的对比损失函数使其能够区分原始字体与字体变体：

$$L = y \cdot d^2 + (1-y) \cdot \max(0, m-d)^2 \tag{5.3}$$

其中，$y \in \{0,1\}$ 表示图像标签，当输入图像对来自相同类别时，例如两者都来自原始字体或是两者都来自字体变体时，$y=1$，否则 $y=0$。m 为边界超参数，用于规定不同类别特征之间的最小间隔，默认设置为 1。该损失函数鼓励相同类别样本在特征空间中距离接近，不同类别的样本之间保持至少 m 的间隔。这一训练策略不仅提升了网络对水印扰动的感知能力，同时自身具备良好的泛化性，可应用于多种字体样式与字符形状的比较任务。

5.2.2 特征提取子网络

在孪生网络中，特征提取子网络的结构可根据任务种类和计算资源进行灵活调整，但必须保证两个子网络的结构和权重变化完全相同，保证同样类型的输入能够得到相似的输出。在本章方法中，采用的特征提取子网络包含三个卷积层，其结构如图 5.2 所示，所用的网络参数如表 5.1 所示。在孪生网络的训练过程中，使用的两个特征提取子网络的权重参数同步变化，最终实现对字符图像风格特征的准确刻画。

表 5.1 特征提取子网络结构

层次	卷积核大小	步长 (Stride)	填充 (Padding)	输入通道数	输出通道数	输出特征图尺寸	激活函数
卷积层 1	3 × 3	1	1	1	16	24 × 24	ReLU
最大池化层	2 × 2	2	0	16	16	12 × 12	-
卷积层 2	3 × 3	1	1	16	32	12 × 12	ReLU
最大池化层	2 × 2	2	0	32	32	6 × 6	-
卷积层 3	3 × 3	1	1	32	64	6 × 6	ReLU
平化+全连接层	-	-	-	-	-	1 × 128	Sigmoid

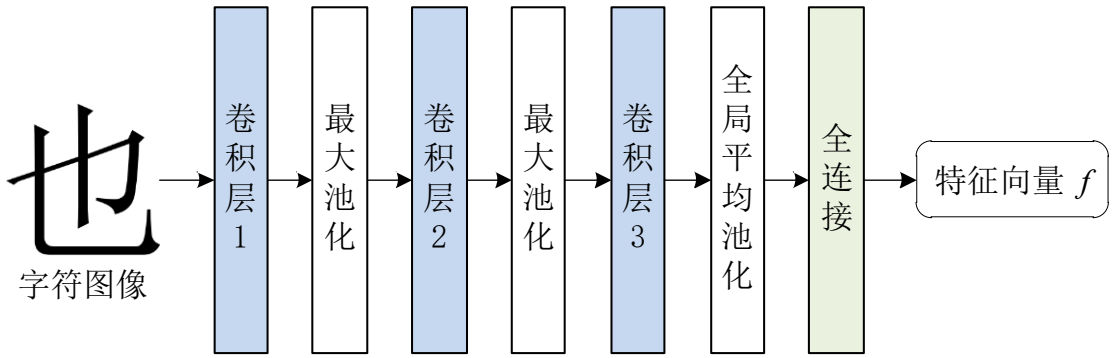


图 5.2 特征提取子网络示意图

5.2.3 噪声增强层

在真实应用场景中，截屏、压缩、分辨率缩放以及不同渲染引擎都会给字符图像带来轻微噪声或模糊失真。如果训练阶段仅使用不受干扰的字形样本，孪生网络在推理阶段面对这些噪声时往往提取不到稳定特征，导致水印判别精度下降。为提升模型的域外泛化能力与鲁棒性，本文在孪生网络输入端加入噪声增强层，其核心思想是在保持字符可辨识的前提下，为每一次前向传播随机生成轻微噪声并叠加到输入图像，使网络在训练过程中额外接受失真的信息，从而学习到更加稳健的表征。

图 5.3 噪声增强层引入的失真示意图

在本章算法中，噪声增强层对字符图像 I 进行的扰动可通过式（5.4）表示：

$$\bar{I} = I \cdot T \cdot R \cdot S \cdot C \tag{5.4}$$

其中，$\bar{I}$ 表示经过噪声层处理后的图像，T、R、S、C 分别表示平移、旋转、缩放、JPEG 压缩四种噪声攻击，其效果如图 5.3 所示。在训练过程中，上述四种噪声攻击会以一定限度内的随机形式施加到图像中，其具体含义和数值如表 5.2 所示，以此模拟信道传输过程中的失真情况。

表 5.2 噪声增强层的具体攻击参数描述

操作	描述
平移 T	将字符图像整体向水平或垂直方向施加幅度为 $\Delta \in [-0.3W, 0.3W]$ 的像素位移，W 为图像原始宽度。
旋转 R	将字符图像绕中心点施加角度为 $\theta \in [-30°, 30°]$ 的旋转变换。
缩放 S	对字符图像进行比例为 $p \in [0.7, 1.3]$ 的尺度变换。
JPEG 压缩 C	对字符图像进行一次质量因子为 $q \in [50, 90]$ 的 JPEG 编码-解码过程。

5.3 孪生网络水印算法

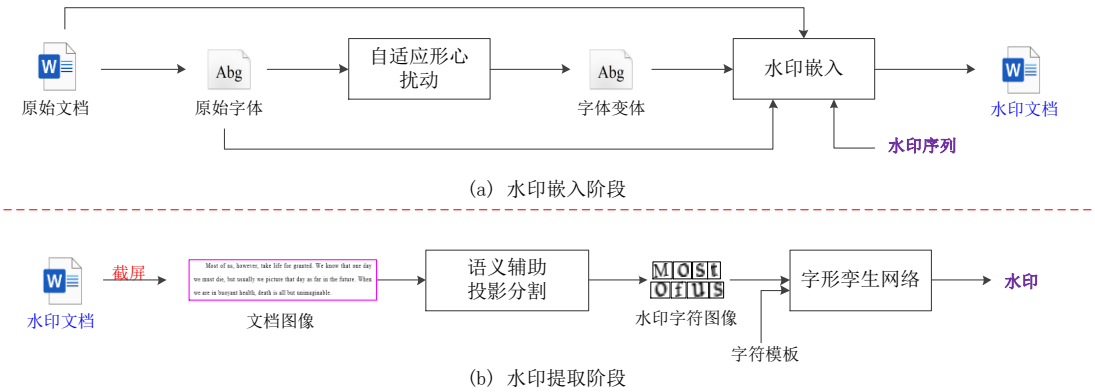


图 5.4 基于孪生神经网络的字形文档水印算法流程图

利用训练完毕的孪生神经网络，本文设计了如图 5.4 所示的字形文档水印嵌入与提取算法。在水印嵌入阶段，根据要嵌入的比特序列确定每个字符对应的字体版本。具体来说，若某一比特为“0”，则采用原始字体渲染该字符；若为“1”，则采用预生成的字体变体进行渲染。字体变体由前文提出的自适应形心扰动算法生成。

在水印提取阶段，结合文档图像的字符识别与神经网络判别机制进行水印提取。对于含水印的文档图像，使用语义投影分割算法对图像中的每个字符区域进行精确定位和裁剪。对于每个字符图像 I_i，根据原始字体文件中对应字符的轮廓数据生成归一化字符图像 $I_0(c_i)$，并将其作为标准模板图像。将 I_i 与 $I_0(c_i)$ 组合成

图像对，输入至已训练的孪生神经网络 f 中进行特征判别，并计算字符图像与模板图像之间的欧氏距离 d_i，该过程可由式（5.5）表示：

$$d_i = \left\| f(I_i) - f(I_0(c_i)) \right\|_2 \tag{5.5}$$

根据距离 d_i 与设定阈值 θ 之间的关系依次判定水印，若距离大于设定阈值，则判定该字符含有扰动，对应的水印比特为 1；反之则为 0。该过程可符号化表示为式（5.6）：

$$\mathrm{w}'=[w_1', w_2', ..., w_n'], \quad w_i' = \begin{cases} 1, & d_i \geq \theta \\ 0, & d_i < \theta \end{cases} \tag{5.6}$$

其中，w' 表示整个水印序列，w_i' 表示当前待判定的水印比特。

5.4 实验结果与分析

5.4.1 实验设置

本章实验中，对所提出的字形文档水印算法分别进行了三个方面的评估：水印提取准确性、鲁棒性和对不同字体的通用性。实验涵盖的攻击有缩放、不同软件截屏、JPEG 压缩和跨平台渲染。选用的字体为宋体和 Times New Roman，包含两种字体的原始字体和各自产生的字体变体。实验采用的训练集为两种字体通过归一化绘图方法生成的字符图像，包含 3000 个常用汉字字符和 62 个英文字符的图像。由于训练孪生网络需要构造正负样本对，本实验中构造的正样本对为原始字符图像和对应的变体字符图像，共 3062 对；负样本对通过随机配对不同字符图像形成，其规模与正样本对相同。采用的测试集包含两类，一类为将水印嵌入文档字符后的截图图像，一类为由字体文件中的轮廓数据直接生成的归一化字符图像。训练的迭代次数设置为 50，批大小为 16，优化器为适应性矩估计 (Adaptive Moment Estimation, Adam)，其超参数 $\beta_1 = 0.5$，$\beta_2 = 0.999$，学习率为 1×10^{-4}。用于训练的硬件为 NVIDIA RTX 3090 GPU × 1，软件框架为 Python 3.10+Pytorch 2.0.0+Cuda 11.6。在本章实验中，用于对比的基线方法是现有性能

指标最好的字形文档水印方法[82]，该方法以字符的形心位移作为水印嵌入和提取的依据，具有较高的水印提取准确率。

5.4.2 水印提取准确性评估

为了验证所提出的孪生网络在文档水印嵌入与提取过程中的判别能力，本小节测试了该网络在常见格式文档下，对不同字体和不同扰动强度的水印提取准确率 ACC。实验选用的英文文档共包含 311 个英文字符，中文文档包含 323 个常用汉字字符，文档统一采用标准分辨率，字体大小为 16 pt。提取准确率结果如表 5.3 所示，在常见格式文档条件下，对于不同的扰动强度 Δt，本章使用孪生网络的水印方法能够在中英文文档中都保持 90%以上的提取准确率，并且均高于基线方法。

表 5.3 不同字体与扰动强度下的提取准确率

字体类型	扰动强度 Δt	基线方法提取准确率 (%)	本章方法提取准确率 (%)
宋体	1/24	87.31	92.26
	1/16	95.36	95.97
	1/12	96.59	97.52
Times New Roman	1/24	83.60	90.03
	1/16	90.68	92.28
	1/12	91.63	93.57

5.4.3 鲁棒性评估

为了验证所提出的孪生网络水印算法在真实应用场景中的鲁棒性表现，本节设计了一系列模拟常见数字文档处理过程的失真操作，包括不同软件截屏、图像缩放、JPEG 压缩和跨平台渲染等。该实验旨在评估在不同强度的失真下，水印提取的准确性是否能够保持稳定。实验中仍然选择 ACC 作为鲁棒性的衡量指标，使用的文档字体为宋体。

实验结果如图 5.5 所示，结果表明，本章方法对于不同种类的攻击均能取得高于基线方法的提取准确率，在无攻击的系统截屏场景中准确率接近 100%；对于不同软件和跨平台渲染所带来的截屏失真，以及缩放和压缩引起的图像失真，本章方法仍能取得高于 90%的提取准确率，表明本章方法具有更强的鲁棒性。本章方法能够取得上述优势的原因在于噪声增强层的构建。由于增加了噪声层，孪生网络能够在学习原始字体与字体变体差异的同时避免被数字信道传输产生的噪声误导，从而在水印判别时表现出更强的鲁棒性。相比之下，基线方法使用传统图像处理方法识别字符图像的几何特征，在应对数字信道不规则失真时的提取准确率将明显下降。

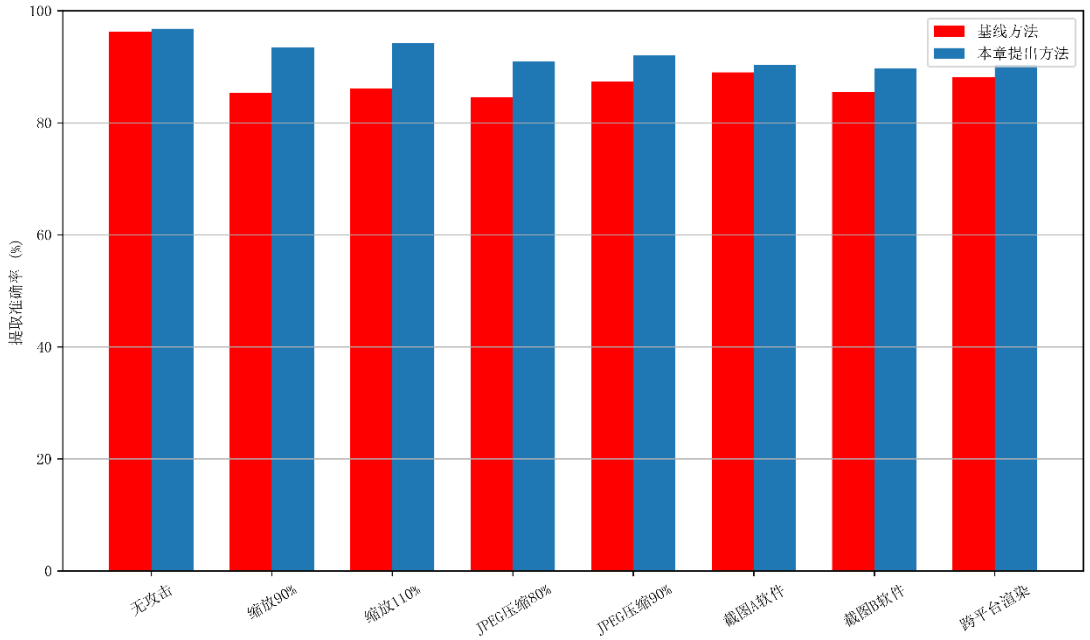


图 5.5 不同攻击下的提取准确率示意图

5.4.4 字体泛化能力评估

为了进一步评估所提出的孪生网络对于不同字体风格的适应能力，本节对方法的跨字体泛化性能[83]进行实验分析。与本章前述的实验不同，此处测试数据使用了训练过程中未出现过的新字体，以衡量网络对字体外观差异的容忍度与判别能力。

在本节实验中，保持训练集的字体和字符数量不变，而在测试阶段使用四种不同风格的新字体，分别为黑体、微软雅黑、楷体和 Arial，每种字体包含三种扰

动强度。对于每种扰动强度的字体，分别生成两类图像，一类为原始字体字符图像，一类为扰动后的字体变体图像，将两类图像按照字符构造图像对并输入至训练完毕的孪生网络进行分类预测。在中文字体中，生成的字符图像包含 500 个常用汉字字符；在英文字体中，字符图像包含 62 个大小写英文字符，字号大小为 16 pt。实验使用提取准确率 ACC 作为评估指标，结果如表 5.4 所示。

表 5.4 未经训练的字体图像提取准确率

字体类型	扰动强度 Δt	提取准确率 (%)
黑体	1/24	81.46
	1/16	85.32
	1/12	86.52
微软雅黑	1/24	79.84
	1/16	84.66
	1/12	83.50
楷体	1/24	70.64
	1/16	81.38
	1/12	85.40
Arial	1/24	83.87
	1/16	82.26
	1/12	87.10

从表中结果可看出，对于网络学习过程中未涉及的新字体，即便在 $\Delta t = 1/24$ 的扰动幅度较小的情况，孪生网络也可在多数字体上达到 80%的提取准确率，在表现最差的字体楷体上也能达到 70%的提取准确率；对于默认的字形扰动幅度 $\Delta t = 1/16$，网络识别字符水印的概率将高于 81%。上述结果表明，所提出的孪生网络水印方法具有较强的字体泛化性能，能够识别未经训练字符图像中的水印。该结果得益于孪生网络的结构模式。孪生网络在进行水印识别任务时，不仅会接受水印字符图像作为输入，还会同时接受不含水印的字符图像，通过对比两张图像在结构中的差异从而精确地判定是否存在水印。从结果中还可观察到，若继续增加扰动幅度至 $\Delta t = 1/12$，多数字体的提取准确率可进一步提升，仅在微软雅黑字体上出现例外。因此，在对提取准确率要求较高的场景下，可以考虑通过略微

牺牲字形视觉质量以换取更高的判别精度。对于增加扰动后，微软雅黑字体的提取准确率没有明显增效的现象，可能是因为网络误将微软雅黑字体中施加的扰动判定为了噪声，导致了水印比特的误判。

5.5 本章小结

本章介绍了一种使用孪生神经网络进行水印提取的字形文档水印算法。在本章中，首先介绍了现有的基于神经网络的字形水印方法，并引入了本章的算法设计。接着介绍了所设计的字形判别孪生网络，该网络使用两个共享权重的特征提取子网络识别原始字体与字体变体之间的差异，并通过引入噪声增强层进一步提高了水印识别率。然后介绍了一种使用该孪生网络的文档水印算法，该算法结合第三章提出的字体变体生成方法进行水印嵌入，并使用孪生网络进行水印判别。最后本章展示了上述算法的实验结果并进行分析。实验证明，相较于现有的字形文档水印方法，本章所提出的孪生网络水印算法在面对复杂图像失真场景下具备较强的水印提取能力。此外，实验还探索了针对未经训练的新字体水印的提取效果，证明了本章提出方法的字体通用性与泛化能力。

第六章 总结与展望

6.1 全文总结

电子文档以文字为主要内容，可凭借极小的内存消耗储存丰富的信息，此特性使得电子文档在网络空间中复制与分发的成本相对较低，进而引发了一系列文档的版权保护问题。为此，研究人员以文档中的不同元素为调制载体，设计了基于排版与背景格式、基于语义和基于字形的文档水印方法。相较于前两者，基于字形的方法以字符为最小水印单元，因而拥有更好的隐蔽性与水印容量，但在现有研究中，此类方法还无法较好地处理字符间距较小的文本密集场景，存在提取准确性不足的问题。此外，在面对信道传输失真时，现有方法的鲁棒性也仍有提升的空间。

为解决上述问题，本文针对字形文档水印技术展开了研究，旨在提升字形水印的性能。对于字形水印技术中的两个关键环节——字体变体生成和字符水印提取，本文分别对其进行了改进，同时重点关注了水印在经历传输信道失真场景中的鲁棒性提升问题。本文的具体工作内容有如下三部分：

1) 设计了一种字体变体生成方法，以应对现有方法无法稳定在字形结构中保留可检测差异的问题。该字体变体生成方法首先利用每个字符的轮廓坐标生成归一化字符图像，根据图像的形心位置自适应地判定该字符的扰动方向：若形心位置偏左，则字符的扰动方向为右；反之，则扰动方向为左。随后，对字符的轮廓坐标施加平移和尺度扰动，并在扰动后以生成字符图像的方式观测形心的偏移。最后通过迭代此扰动与观测的过程，保证字符形心的充分偏移。实验结果表明，本文提出的字体变体生成方法能在保持字体视觉质量的同时，为变体中的字符稳定施加可检测的形心偏移。此外，该方法适用于中文、英文和数字等多种字符类型，展示出良好的通用性。

2) 为了应对密集文本场景中字符误分割导致的低水印识别率问题，设计了一种基于语义投影分割的字形文档水印算法。在嵌入阶段，使用原始字体与字体变体共同承载水印，其中原始字体代表比特位“0”，字体变体代表比特位“1”。字体变体中的字符以形心偏移的方式与原始字体进行区别。在提取阶段，采用

OCR 与投影分割结合的语义投影分割策略，进行字符分割。借助 OCR 技术对文档图像中的字符进行初步分割，并记录每个水印位置的字符语义。初步分割后的字符图像通常包含不规则边界，影响后续水印判别，因此使用水平和垂直投影分割以裁剪多余边界，保留字符的最小外接矩形。接着，根据外接矩形的宽度众数值，修正字符的分割错误。完成字符分割后，利用图像的形心偏移判别水印。实验结果表明，提出方法在保持水印高隐蔽性的同时，相比现有方法提升了水印识别率，尤其是在面对字符排版紧凑场景时。并且，本方法显著降低了提取过程中水印比特出现差错传播的概率，进一步提升了算法的稳定性。

3) 对于字形文档水印方法面对传输信道失真时鲁棒性不足的问题，本文对上述框架进行了改进，提出基于孪生神经网络的字形文档水印算法。该孪生网络由两个结构相同、权重共享的特征提取子网络构成。在训练时，将原始字体字符图像与其对应变体图像作为输入对送入子网络，通过欧氏距离的对比损失函数对子网络进行训练。在水印提取时，将分割完毕的待测字符图像与对应的原始字符图像送入孪生网络，通过计算图像的特征差异判别水印。实验结果表明，提出的孪生网络字形水印算法拥有较高的提取准确率与字体泛化能力，并且在面对截图损失、格式压缩和图像缩放等失真时有更高的鲁棒性。

6.2 未来展望

本文针对文档在真实传输场景下的水印性能提升问题，提出了鲁棒的字形文档水印技术方案。该方案虽然在一定程度上增加了文档水印技术的实用性，但在多变的应用环境中，文档水印技术仍存在很多亟待解决的技术问题。具体而言，未来的研究工作可以从以下几个方面展开：

1) 本文提出的字体变体生成方案通过修改原始字体的字符轮廓进行，具有较强的可检测性。然而，和现有的字体变体生成方法类似，本方案需要对字符的每个轮廓坐标都进行修改，若对字符的修改幅度加以限制，则字体变体的可检测性将难以保证。因此，如何通过更小的坐标修改幅度实现扰动的可检测性是一个仍未解决但富有现实意义的关键问题。一个可能的路径是通过类似字体风格迁移的方法实现字体变体的生成任务。

2) 本文提出的两种字形文档水印算法都需要使用变体替换文档中的字体，从而在文档中引入可检测的字形扰动。若攻击者察觉到字体格式的变动，并通过 OCR 识别文档内容并将其转录到新文档中，本文方法将面临更大的挑战。因此，研究能够抵御 OCR 转录攻击的字形文档水印技术具有重要的实用价值。对于上述挑战，研究人员需要从 OCR 技术的原理出发，通过在文档中施加特殊的对应干扰机制以降低此类攻击的威胁。

3) 本文提出的孪生网络水印算法增强了字形文档水印算法的鲁棒性，但此方法依赖于对电子传输信道噪声的模拟。在某些使用场景中，文档还可能经历非电子的传输信道，例如打印后扫描、打印后拍照等等，此类场景会在水印文档中引入更多不规则的失真，影响水印的正确提取。因此，未来可以考虑针对此类场景进行字形文档水印算法的设计，以进一步提升算法的鲁棒性。

参考文献

[1] 吴汉舟, 张杰, 李越等. 人工智能模型水印研究进展[J]. 中国图象图形学报, 2023, 28(6): 1792-1810.

[2] Brassil J T, Low S, Maxemchuk N F. Copyright protection for the electronic distribution of text documents[J]. Proceedings of the IEEE, 1999, 87(7): 1181-1196

[3] Brin S, Davis J, Garcia-Molina H. Copy detection mechanisms for digital documents[C]//Proceedings of the 1995 ACM SIGMOD International Conference on Management of Data, San Jose, California, USA, May 22-25, 1995. New York: ACM, 1995: 398-409.

[4] Eskicioglu A M, Delp E J. An overview of multimedia content protection in consumer electronics devices[J]. Signal Processing: Image Communication, 2001, 16(7): 681-699.

[5] Wang G, Ma Z, Liu C, et al. Must: Robust image watermarking for multi-source tracing[C]//Proceedings of the AAAI Conference on Artificial Intelligence, February 20-27, 2024, Vancouver, Canada. Washington: AAAI Press, 2024, 38(6): 5364-5371.

[6] Komatsu N, Tominaga H. A proposal on digital watermark in document image communication and its application to realizing a signature[J]. Electronics and Communications in Japan (Part I: Communications), 1990, 73(5): 22-33.

[7] Atallah M J, Raskin V, Crogan M, et al. Natural language watermarking: Design, analysis, and a proof-of-concept implementation[C]//Information Hiding: 4th International Workshop, IH 2001, Pittsburgh, PA, USA, April 25-27, 2001 Proceedings 4. Berlin: Springer, 2001: 185-200.

[8] Meral H M, Sankur B, Özsoy A S, et al. Natural language watermarking via morphosyntactic alterations[J]. Computer Speech & Language, 2009, 23(1): 107-125.

[9] Topkara U, Topkara M, Atallah M J. The hiding virtues of ambiguity: quantifiably resilient watermarking of natural language text through synonym substitutions[C]//Proceedings of the 8th Workshop on Multimedia and Security, September 26-27, 2006, Geneva, Switzerland. New York: ACM, 2006: 164-174.

[10] Lalai H N, Ramakrishnan A A, Shah R S, et al. From intentions to techniques: A comprehensive taxonomy and challenges in text watermarking for large language models[J]. arXiv preprint arXiv:2406.11106, 2024.

[11] 郁艳华, 张敏瑞. 一种基于图像平滑性的脆弱性文档水印算法[J]. 西北师范大学学报: 自然科学版, 2011, 47(4): 48-51.

[12] 赵莉楠, 吴晟, 蔡灿民. 基于 VBA 文本框的 Word 文档水印的信息隐藏方法[J]. 科技广场, 2007 (3): 134-136.

[13] Taleby Ahvanooey M, Li Q, Shim H J, et al. A comparative analysis of information hiding techniques for copyright protection of text documents[J]. Security and Communication Networks, 2018, 2018(1): 5325040.

[14] BRASSIL J, LOW S, MAXEMCHUK N. Electronic marking and identification techniques to discourage document copying[C]//Proceedings of IEEE INFOCOM'95, April 2-6, 1995, Boston, MA, USA. Piscataway: IEEE, 1995: 1278-1287

[15] Bender W, Gruhl D, Morimoto N, et al. Techniques for data hiding[J]. IBM Systems Journal, 1996, 35(3.4): 313-336.

[16] WU M, LIU B. Data hiding in binary image for authentication and annotation[J]. IEEE transactions on multimedia, 2004, 6(4): 528- 538.

[17] Taleby Ahvanooey M, Dana Mazraeh H, Tabasi S H. An innovative technique for web text watermarking (AITW)[J]. Information Security Journal: A Global Perspective, 2016, 25(4-6): 191-196.

[18] Alotaibi R A, Elrefaei L A. Improved capacity Arabic text watermarking methods based on open word space[J]. Journal of King Saud University-Computer and Information Sciences, 2018, 30(2): 236-248.

[19] Kim M Y, Zaiane O R, Goebel R. Natural language watermarking based on syntactic displacement and morphological division[C]//2010 IEEE 34th Annual Computer Software and Applications Conference Workshops, July 19-23, 2010, Seoul, Korea. Piscataway: IEEE, 2010: 164-169.

[20] Halvani O, Steinebach M, Wolf P, et al. Natural language watermarking for German texts[C]//Proceedings of the First ACM Workshop on Information Hiding and Multimedia Security, June 17-19, 2013, Montpellier, France. New York: ACM, 2013: 193-202.

[21] Huo M, Somayajula S A, Liang Y, et al. Token-specific watermarking with enhanced detectability and semantic coherence for large language models[J]. arXiv preprint arXiv:2402.18059, 2024.

[22] Mei Q G, Wong E K, Memon N D. Data hiding in binary text documents[C]//Security and Watermarking of Multimedia Contents III, January 22-25, 2001, San Jose, USA. Washington: SPIE, 2001, 4314: 369-375.

[23] 张驰. 二值文本图像数字水印技术研究[D]. 重庆: 重庆大学, 2007.

[24] 赵星阳, 孙继银, 李琳琳. 基于字符阶梯边沿调整的文本水印算法[J]. 计算机应用, 2008, 28(12): 3175-3178, 3182.

[25] Liu A, Pan L, Lu Y, et al. A survey of text watermarking in the era of large language models[J]. ACM Computing Surveys, 2024, 57(2): 1-36.

[26] Abdelnabi S, Fritz M. Adversarial watermarking transformer: Towards tracing text povenance with data hiding[C]//2021 IEEE Symposium on Security and Privacy (SP), May 23-27, 2021, San Francisco, CA, USA. Piscataway: IEEE, 2021: 121-140.

[27] Xiao C, Zhang C, Zheng C. Fontcode: Embedding information in text documents using glyph perturbation[J]. ACM Transactions on Graphics (TOG), 2018, 37(2): 1-16.

[28] Qi W, Guo W, Zhang T, et al. Robust authentication for paper-based text documents based on text watermarking technology[J]. Mathematical Biosciences and Engineering, 2019, 16(4): 2233-2249.

[29] Yang X, Zhang W, Fang H, et al. Language universal font watermarking with multiple cross-media robustness[J]. Signal Processing, 2023, 203: 108791.

[30] Yang X, Zhang J, Fang H, et al. AutoStegaFont: Synthesizing vector fonts for hiding information in documents[C]//Proceedings of the AAAI Conference on Artificial Intelligence, February 7-14, 2023, Washington, USA. Washington: AAAI Press, 2023, 37(3): 3198-3205.

[31] Kankanhalli M S, Ramakrishnan K R. Adaptive visible watermarking of images[C]//Proceedings IEEE International Conference on Multimedia Computing and Systems, June 7 - 11, 1999, Washington, USA. Piscataway: IEEE, 1999, 1: 568-573.

[32] Park J H, Jeong S E, Kim C S. Robust and fragile watermarking techniques for documents using bi-directional diagonal profiles[C]//Information and Communications Security: Third International Conference, ICICS, Proceedings 3, November 13-16, 2001, Xian, China. Berlin: Springer, 2001: 483-494.

[33] Shen Y, Tang C, Fan Z, et al. Blind watermarking scheme for medical and non-medical images copyright protection using the QZ algorithm[J]. Expert Systems with Applications, 2024, 241: 122547.

[34] Wong P H W, Au O C, Yeung Y M. Novel blind multiple watermarking technique for images[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2003, 13(8): 813-830.

[35] 王宏霞, 何晨, 丁科. 基于混沌映射的鲁棒性公开水印[J]. 软件学报, 2004, 15(8): 1245-1251.

[36] Jiao S, Qiu Y, Su Q, et al. Enhancing watermarking robustness and invisibility with growth optimizer and improved LU decomposition[J]. Optik, 2025, 329: 172353.

[37] An B, Ding M, Rabbani T, et al. Waves: Benchmarking the robustness of image watermarks[J]. arXiv preprint arXiv:2401.08573, 2024.

[38] 刘昊, 孙堡垒, 郭云彪. 文本数字水印技术研究综述[J]. 东南大学学报 (自然科学版), 2007, 1(1): 226.

[39] 方涵. 屏摄鲁棒水印方法研究[D]. 合肥: 中国科学技术大学, 2021.

[40] Hore A, Ziou D. Image quality metrics: PSNR vs. SSIM[C]//2010 20th International Conference on Pattern Recognition, August 23-26, 2010, Istanbul, Turkey. Piscataway: IEEE, 2010: 2366-2369.

[41] Sara U, Akter M, Uddin M S. Image quality assessment through FSIM, SSIM, MSE and PSNR—a comparative study[J]. Journal of Computer and Communications, 2019, 7(3): 8-18.

[42] Setiadi D R I M. PSNR vs SSIM: Imperceptibility quality assessment for image steganography[J]. Multimedia Tools and Applications, 2021, 80(6): 8423-8444.

[43] Horgan J. From complexity to perplexity[J]. Scientific American, 1995, 272(6): 104-109.

[44] Zhang T, Kishore V, Wu F, et al. Bertscore: Evaluating text generation with bert[J]. arXiv preprint arXiv:1904.09675, 2019.

[45] Ali I. Bit-error-rate (BER) simulation using MATLAB[J]. International Journal of Engineering Research and Applications, 2013, 3(1): 706-711.

[46] Jeruchim M. Techniques for estimating the bit error rate in the simulation of digital communication systems[J]. IEEE Journal on Selected Areas in Communications, 1984, 2(1): 153-170.

[47] Asuero A G, Sayago A, González A G. The correlation coefficient: An overview[J]. Critical Reviews in Analytical Chemistry, 2006, 36(1): 41-59.

[48] Fu Z, Chai X, Tang Z, et al. Adaptive embedding combining LBE and IBBE for high-capacity reversible data hiding in encrypted images[J]. Signal Processing, 2024, 216: 109299.

[49] Mahto D K, Singh A K, Singh K N, et al. Robust copyright protection technique with high-embedding capacity for color images[J]. ACM Transactions on Multimedia Computing, Communications and Applications, 2024, 20(11): 1-12.

[50] Baydas S, Karakas B. Defining a curve as a Bezier curve[J]. Journal of Taibah University for Science, 2019, 13(1): 522-528.

[51] Ferraiolo J, Jun F, Jackson D. Scalable vector graphics (SVG) 1.0 specification[M]. Bloomington: Iuniverse, 2000.

[52] Casey R G, Lecolinet E. A survey of methods and strategies in character segmentation[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1996, 18(7): 690-706.

[53] Lee S W, Lee D J, Park H S. A new methodology for gray-scale character segmentation and recognition[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1996, 18(10): 1045-1050.

[54] Gonçalves G R, da Silva S P G, Menotti D, et al. Benchmark for license plate character segmentation[J]. Journal of Electronic Imaging, 2016, 25(5): 053034-053034.

[55] Fujitake M. Dtrocr: Decoder-only transformer for optical character recognition[C]//Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, January 4-8, 2024, Waikoloa, Hawaii. Piscataway: IEEE, 2024: 8025-8035.

[56] R. P. dos Santos, G. S. Clemente, T. I. Ren, et al. Text line segmentation based on morphology and histogram projection[C]//2009 10th International Conference on Document Analysis and Recognition. July 26-29, 2009, Barcelona, Spain. Piscataway: IEEE, 2009: 651-655.

[57] Girshick R. Fast r-cnn[C]//Proceedings of the IEEE International Conference on Computer Vision, December 7-13, 2015, Santiago, Chile. Piscataway: IEEE, 2015: 1440-1448.

[58] Terven J, Córdova-Esparza D M, Romero-González J A. A comprehensive review of yolo architectures in computer vision: From yolov1 to yolov8 and yolo-nas[J]. Machine Learning and Knowledge Extraction, 2023, 5(4): 1680-1716.

[59] Khan N, Haq I U, Khan S U, et al. DB-Net: A novel dilated CNN based multi-step forecasting model for power consumption in integrated local energy systems[J]. International Journal of Electrical Power & Energy Systems, 2021, 133: 107023.

[60] Doyle J R, Bottomley P A. Font appropriateness and brand choice[J]. Journal of Business Research, 2004, 57(8): 873-880.

[61] Karow P, Karow P. Font technology[M]. Springer Berlin Heidelberg, 1994.

[62] Kahan S, Pavlidis T, Baird H S. On the recognition of printed characters of any font and size[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1987 (2): 274-288.

[63] Sheppard S M, Nobles S L, Palma A, et al. One Font Doesn't Fit All: The Influence of Digital Text Personalization on Comprehension in Child and Adolescent Readers[J]. Education Sciences, 2023, 13(9): 864.

[64] Hayashi H, Abe K, Uchida S. GlyphGAN: Style-consistent font generation based on generative adversarial networks[J]. Knowledge-Based Systems, 2019, 186: 104927.

[65] 杨曦. 鲁棒文本水印的智能化方法研究[D]. 合肥: 中国科学技术大学, 2024.

[66] Knuth D E. The concept of a meta-font[J]. Visible Language, 1982, 16(1): 3-27.

[67] Lian Z, Xiao J. Automatic Shape Morphing for Chinese Characters[M]//SIGGRAPH Asia 2012 Technical Briefs. 2012: 1-4.

[68] Stern M K, Johnson J H. Just noticeable difference[J]. The Corsini encyclopedia of Psychology, 2010: 1-2.

[69] Lin W, Ghinea G. Progress and opportunities in modelling just-noticeable difference (JND) for multimedia[J]. IEEE Transactions on Multimedia, 2021, 24: 3706-3721.

[70] 孙杉. 基于自动生成字库的中文鲁棒文档水印方法[D]. 合肥: 中国科学技术大学, 2021.

[71] 孙杉, 张卫明, 方涵等. 中文水印字库的自动生成方法[J]. 中国图象图形学报, 2022, 27(1): 262-276.

[72] Eikvil L. Optical character recognition[J]. citeseer. ist. psu. edu/142042. html, 1993, 26: 1-4.

[73] 王晨, 姚晔, 李黎. 抗打印扫描和屏幕拍摄的字形扰动研究进展[J]. 应用科学学报, 2023, 41(2): 240-251.

[74] Abdelmoez W, Nassar D M, Shereshevsky M, et al. Error propagation in software architectures[C]//10th International Symposium on Software Metrics, 2004. Proceedings, September 11-17, 2004, Chicago, Illinois. Piscataway: IEEE, 2004: 384-393.

[75] Yao Y, Wang C, Wang H, et al. Embedding secret message in Chinese characters via glyph perturbation and style transfer[J]. IEEE Transactions on Information Forensics and Security, 2024, 19: 4406-4419.

[76] Bertinetto L, Valmadre J, Henriques J F, et al. Fully-convolutional siamese networks for object tracking[C]//Computer Vision–ECCV 2016 Workshops: October 8-10 and 15-16, 2016, Amsterdam, the Netherlands, Proceedings, part II 14. Cham: Springer International Publishing, 2016: 850-865.

[77] Valero-Mas J J, Gallego A J, Rico-Juan J R. An overview of ensemble and feature learning in few-shot image classification using siamese networks[J]. Multimedia Tools and Applications, 2024, 83(7): 19929-19952.

[78] Li Y, Weng L, Xia M, et al. Multi-scale fusion siamese network based on three-branch attention mechanism for high-resolution remote sensing image change detection[J]. Remote Sensing, 2024, 16(10): 1665.

[79] Kumar C R, Saranya N, Priyadharshini M, et al. Face recognition using CNN and siamese network[J]. Measurement: Sensors, 2023, 27: 100800.

[80] Ding Z, Zhou D, Li H, et al. Siamese networks and multi-scale local extrema scheme for multimodal brain medical image Fusion[J]. Biomedical Signal Processing and Control, 2021, 68: 102697.

[81] Sharma N, Gupta S, Mohamed H G, et al. Siamese convolutional neural network-based twin structure model for independent offline signature verification[J]. Sustainability, 2022, 14(18): 11484.

[82] He C, Wu D, Zhang X, et al. Watermarking Text Documents With Watermarked Fonts[C]//Proceedings of the 2024 ACM Workshop on Information Hiding and Multimedia Security, Jun 24-26, 2024, Baiona, Spain. New York: ACM, 2024: 187-197.

[83] He H, Chen X, Wang C, et al. Diff-font: Diffusion model for robust one-shot font generation[J]. International Journal of Computer Vision, 2024, 132(11): 5372-5386.

攻读硕士学位期间取得的研究成果

一、论文

[1] **He C**, Wu D, Zhang X, et al. Watermarking Text Documents With Watermarked Fonts[C]//Proceedings of the 2024 ACM Workshop on Information Hiding and Multimedia Security, Jun 24-26, 2024, Baiona, Spain. New York: ACM, 2024: 187-197. (CCF C，信息隐藏与多媒体安全领域著名国际学术会议)

二、专利

[1] **何承桦**, 吴汉舟, 张新鹏．基于字形自适应扰动与神经网络判别器的文档水印方法：202510301555.8[P]．2025-3-14．

致　谢

在论文完成之际，我想由衷感谢在硕士研究生阶段关心与帮助过我的人。

首先，感谢我的导师张新鹏教授。张老师总能在研究方向上持有独到的眼光和前瞻性的思维，也为我在学习生活和为人处世方面树立了榜样，令我受益匪浅。此外，正是因为张老师创造并带领了人工智能安全团队，我才能够加入这里，获得便利的实验资源与优越的学习环境。

其次，感谢同样为我的科研工作提供极大帮助的吴汉舟老师。吴老师不仅以身作则督促着我刻苦钻研、奋力进取，还在我的研究思路规划、论文撰写和项目工作的过程中对我进行了悉心的指导。

随后，感谢实验室的史景辉、朱伟南、杨之光、林丽娜、黄超越、赵宇辰等同门和其他朋友们。尽管研究方向不尽相同，但大家在科研工作和日常生活中互相建议、互相鼓励、共同进步，创造了一个轻松和谐的研究氛围。

然后，感谢我的父母一如既往地为我提供物质和精神上的支持，并尊重我作出的各个决定。感谢我的女朋友的一路陪伴与关心。

最后，感谢各位评审专家和老师，感谢您在百忙之中抽出时间为这篇论文进行审阅并提出宝贵意见，我将虚心接受并认真参考！

何承桦

上海大学

2025 年 05 月 17 日