PreP-OCR: A Complete Pipeline for Document Image Restoration and Enhanced OCR Accuracy

Shuhao Guan[1], Moule Lin[2], Cheng Xu[1], Xinyi Liu[1]
Jinman Zhao[3], Jiexin Fan[2], Qi Xu[4], Derek Greene[1]
[1]University College Dublin, [2]Trinity College Dublin
[3]University of Toronto, [4]Shanghai University
shuhao.guan@ucdconnect.ie, derek.greene@ucd.ie

Abstract

This paper introduces PreP-OCR, a two-stage pipeline that combines document image restoration with semantic-aware post-OCR correction to improve text extraction from degraded historical documents. Our key innovation lies in jointly optimizing image clarity and linguistic consistency. First, we generate synthetic image pairs with randomized text fonts, layouts, and degradations. An image restoration model is trained on this synthetic data, using multi-directional patch extraction and fusion to process large images. Second, a ByT5 post-corrector, fine-tuned on synthetic historical text training pairs, addresses any remaining OCR errors. Detailed experiments on 13,831 pages of real historical documents in English, French, and Spanish show that PreP-OCR pipeline reduces character error rates by 63.9–70.3% compared to OCR on raw images. Our pipeline demonstrates the potential of integrating image restoration with linguistic error correction for digitizing historical archives. https://github.com/NikoGuan/PreP-OCR

1 Introduction

In the era of massive document digitization, ensuring accurate text extraction from degraded images has become increasingly important (Shen et al., 2021). Many historical documents, scanned books, and archival materials suffer from various forms of degradation – such as blur, noise, ink bleeding, and other artifacts – due to aging and suboptimal scanning conditions (Pardo et al., 2024). These degradations not only affect the visual quality of the images, but can also severely impact the resulting performance of Optical Character Recognition (OCR) systems, leading to high error rates in extracted text (Hegghammer, 2022).

To address these challenges, this paper introduces PreP-OCR, a novel synthetic-data-driven two-stage pipeline that first restores degraded images for OCR-based text extraction and then enhances the extracted text through post-processing.

To effectively train the image restoration model, we employ a comprehensive synthetic data generation method that simulates realistic document degradation. First, we render clean text images with diverse typography, then we apply degradation operations in a randomized order with stochastic parameters (see Section 4.2), yielding a richly varied dataset, allowing models to learn a robust mapping between the original degraded inputs and their clean counterparts. Additionally, we propose a multi-directional patch extraction and fusion strategy to efficiently process larger images and further enhance overall image quality (see Section 4.3). Figure 6 shows examples of the process.

Following image restoration, in the next step of our proposed pipeline the restored images are fed into an OCR system. Although restoration significantly reduces structural ambiguities, it may not fully eliminate OCR errors. To correct any residual recognition mistakes, we incorporate a ByT5 post-OCR correction module that semantically recovers errors, even in cases where images are severely degraded and challenging to fully restore (see Section 4.4). Consequently, the restoration stage primarily resolves ambiguities in character shapes, yielding more legible images that are easier for OCR systems to recognize, while the post-correction stage mitigates systematic OCR errors through sequence-to-sequence translation.

In Sections 4.1 and 5.1, we describe the collection of numerous degraded historical book images. These images were scanned using various OCR systems, and we then constructed evaluation datasets with their corresponding ground truth texts. In Sections 5.2–5.3, we use the data to assess text reconstruction quality in different patch regions and evaluate the effectiveness of our fusion strategy. Finally, in Sections 5.4–5.5, we test the PreP-OCR pipeline on English, French, and Spanish datasets.

2 Related Work

Extensive research has demonstrated that image pre-processing can significantly improve the performance of deep learning models (Vidal and Amigo, 2012; Salvi et al., 2021). However, pre-processing within the context of OCR remains relatively underexplored, with existing methods primarily focusing on contrast enhancement and color adjustment (Gupta et al., 2007; Harraj and Raissouni, 2015; Bui et al., 2017).

Recent studies in image deblurring have introduced more advanced restoration techniques that could also benefit OCR. Early image restoration methods were primarily based on CNNs (Dong et al., 2015a,b; Zhang et al., 2017; Cho et al., 2021). Subsequent research introduced more elaborate architectures, such as residual blocks (Kim et al., 2016; Zhang et al., 2021), generative adversarial networks (GANs) (Pathak et al., 2016; Gulrajani et al., 2017; Wang et al., 2018; Kupyn et al., 2019), and attention mechanisms (Zhang et al., 2018; Yu et al., 2018). Transformers (Vaswani, 2017), which model long-range dependencies, have advanced NLP and computer vision and are now widely used in image restoration (Chen et al., 2021; Liang et al., 2021; Zamir et al., 2022).

Diffusion models have emerged as a powerful alternative for generative image tasks, optimizing a parameterized Markov chain to approximate the target distribution more accurately than many other generative frameworks. Examples in restoration include DiffIR (Xia et al., 2023) and ResShift (Yue et al., 2024), both of which are diffusion-based approaches. Several studies have also used diffusion models together with textual information to recover the appearance of ancient stele inscriptions (Zhu et al., 2024; Yang et al., 2025). In our work, we harness image-restoration models to pre-process degraded images prior to applying OCR.

The post-OCR task aims to correct errors in OCR outputs, with early methods relying on dictionary lookups or spelling checkers (Furrer and Volk, 2011; Bassil and Alwani, 2012; Estrella and Paliza, 2014; Kettunen, 2016). More recent approaches treat post-OCR correction as a sequence-to-sequence task, leveraging neural machine translation (NMT) models, such as BERT (Devlin et al., 2019), BART (Lewis, 2019) and T5 (Raffel et al., 2020) (Amrhein and Clematide, 2018; Nguyen et al., 2020; Soper et al., 2021; Maheshwari et al., 2022). Several comparative studies have shown that byte-level models, such as ByT5 (Xue et al., 2022), often achieve the best performance for post-OCR tasks (Maheshwari et al., 2022; Löfgren and Dannélls, 2024; Guan et al., 2024; Guan and Greene, 2024b).

Both image restoration and post-OCR correction require paired training data, and the availability of abundant, high-quality data is critical for success (Rijhwani et al., 2020; Mazumder et al., 2024). Consequently, researchers have explored a variety of strategies for generating synthetic data as a form of data augmentation (Hamdi et al., 2023; Shorten and Khoshgoftaar, 2019). For image deblurring and text-recognition, common techniques involve injecting noise into clean images to mimic real-world degradation (Yuan et al., 2007; Krishna et al., 2018; Rim et al., 2022; Li et al., 2023; Hamdi et al., 2023), or using methods such as StableDiffusion (Rombach et al., 2022) to create paired image edits (Brooks et al., 2023). In the post-OCR domain, synthetic training pairs are often produced by inserting controlled errors into clean text (D'hondt et al., 2017; Grundkiewicz et al., 2019; Ignat et al., 2022; Jasonarson et al., 2023; Guan and Greene, 2024a; Guan et al., 2024).

3 Problem Formulation

Our task addresses two sequential objectives: (1) restoring degraded images to enhance legibility, and (2) recovering accurate textual content from these images. We formalize these goals as follows.

Image restoration objective. Let $I_d, I \in \mathbb{R}^{H \times W}$ denote the degraded input and its sharp ground-truth image, respectively. A restoration model $\mathcal{R}$ aims to produce a restored image $\hat{I} = \mathcal{R}(I_d)$, where the objective is to maximize the Peak-Signal-to-Noise Ratio (PSNR) (Hore and Ziou, 2010) between $\hat{I}$ and I, such that:

$$\mathcal{R}^* = \arg\max_{\mathcal{R}} \text{PSNR}(\mathcal{R}(I_d), I),$$

Text recovery objective. Let T represent the ground-truth text sequence of image I_d. The restored image $\hat{I}$ is first processed by an OCR model $\mathcal{O}$, yielding predicted text $T' = \mathcal{O}(\hat{I})$. This predicted text T' is then refined by a post-processing module $\mathcal{P}$, resulting in $\hat{T} = \mathcal{P}(T')$. The objective here is to minimize the Character Error Rate (CER) between $\hat{T}$ and T:

$$\mathcal{P}^* = \arg\min_{\mathcal{P}} \text{CER}(\mathcal{P}(\mathcal{O}(\hat{I})), T),$$

These dual objectives are addressed in our two-stage pipeline. First, the restoration model $\mathcal{R}$ is optimized using synthetic paired data to restore the book images, directly enhancing character legibility (see Section 4.2). Second, the post-processor $\mathcal{P}$ is trained on synthetic training pairs simulating OCR errors to correct residual recognition mistakes (see Section 4.4). The image restoration stage reduces structural ambiguities in character shapes, while the text correction stage addresses systematic OCR errors through sequence-to-sequence translation. This cascaded approach ensures both pixel-level fidelity in $\hat{I}$ and semantic-level accuracy in the final text output $\hat{T}$.

4 PreP Pipeline

4.1 Real Evaluation Data Collection

To evaluate the performance of a model trained solely on synthetic data when applied to real-world data, we constructed a new corpus as follows. We curated a collection of 30 English books (9,606 pages), 5 Spanish books (2,404 pages), and 5 French books (1,821 pages) from the 15th to 19th centuries. Ground truth (GT) texts were sourced from clean digital books available on Project Gutenberg[1], while a set of corresponding scanned PDF files containing degraded text images was obtained from Open Library[2]. We intentionally selected older books exhibiting visible damage, as shown in the images in Figure 1. Text alignment between the OCR outputs and GT was performed using the RETAS framework (Yalniz and Manmatha, 2011), which employs dynamic programming for robust sequence matching.

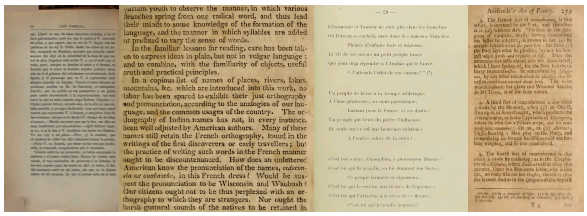

Figure 1: Example images of digitized pages from historical books, which are often affected by degraded text, aging pages, and low capture resolution.

For subsequent experiments, we pre-process the images through denoising before employing OCR. Comparative CER analysis will be conducted across three pipelines: raw images (direct OCR on original scanned pages), Pre-process (OCR after image restoration), and our proposed approach PreP-process (image restoration combined with OCR and post-correction).

4.2 Synthetic Data for Restoration

In image-to-image restoration tasks, paired data consisting of a degraded input and its corresponding clean reference is crucial for effective training. However, obtaining such paired data from real-world documents is extremely challenging because authentic clean images and their degraded counterparts are rarely available. To overcome this limitation, we employ a synthetic data generation approach that enables us to simulate realistic degradation from scratch.

Our synthetic data pipeline begins by generating a clean base image from textual content. To maximize OCR accuracy, we ignore color information and work with grayscale images. First, we collect various fonts for different languages and render multi-line text with a range of stylistic variations, including random indentation, character shifts, rotation, and bending. Additionally, the text is randomly tilted, and both line and character spacing are varied to mimic the natural irregularities found in printed documents. The generated base image serves as the clean ground truth.

To simulate real-world degradations, next we apply a series of controlled noise and distortion operations. Specifically, the pipeline adds random noise, performs resolution reduction, applies Gaussian blurring, and overlays additional artifacts such as random black or white patches of varying sizes, white or black lines (simulating scratches or folds), background textures, and stain overlays. The process also includes random morphological operations (dilation and erosion) to further simulate text imperfections. It is worth noting that these operations are applied in random order, producing diverse results depending on the sequence.

Since noise levels can vary in real-world digitized documents, we predefine four noise levels (level-1 to level-4). Higher levels introduce a wider range of noise parameters, potentially resulting in more degraded images. Additionally, 10% of the noisy images are binarized using Otsu's algorithm (Yousefi, 2011). We also stitch together images with different noise levels and fonts, as in real data, different regions on a given page can exhibit varying degrees of degradation and typographic styles.

The detailed parameters for generating the base

[1] https://www.gutenberg.org
[2] https://openlibrary.org

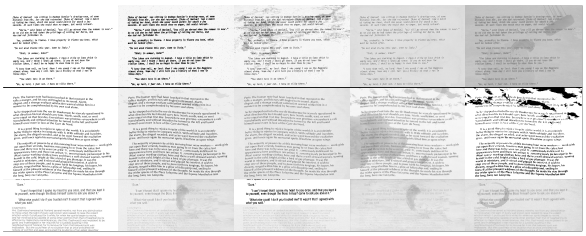

Figure 2: Example of three sets of synthetic image data. The leftmost image is the base image, while the image to its right is the corresponding degraded image.

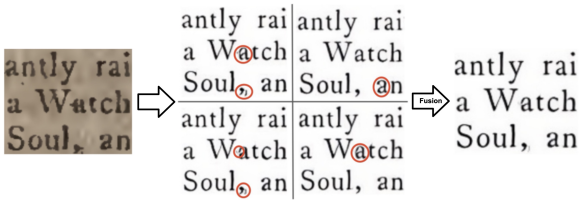

Figure 3: The left panel shows a real degraded patch. The four sub-panels in the center depict restored outputs under different scanning directions, where the red circles highlight localized artifacts or noise. On the right is the final fused result, in which these artifacts are effectively suppressed.

image and simulating noise levels are provided in Appendix B. Example images generated using this process are shown in Figure 2.

By pairing each original base image with its synthetically degraded versions, we create a large and diverse dataset. This synthetic data facilitates the robust training of our restoration model, allowing it to learn the complex mapping from degraded to clean images. As demonstrated later in Section 5, this can ultimately improve generalization performance in real-world document restoration tasks.

4.3 Patch Extraction and Fusion

When processing large images, we first partition them into multiple regions. To address stochastic noise and local inconsistencies, we adopt a multi-directional patch extraction strategy. Specifically, for each degraded image, we scan it four times: top-left to bottom-right, top-right to bottom-left, bottom-left to top-right, and bottom-right to top-left. Since image dimensions may not align perfectly with the patch stride, we pad only the edge opposite the scanning direction to ensure a fully integer-aligned pass over the entire image.

In each pass, 256×256 patches are extracted at a stride of 128 pixels. Scanning from different directions yields slightly different patches, meaning even the same region in the original image may appear with different neighboring contexts in a patch—leading to varied predictions. After restoring each patch, we discard the outer 64-pixel border and retain only the central 128×128 region, minimizing boundary artifacts. An example of the multi-direction patch extraction process is provided in Appendix A.

Each pixel in the final restored image is fused by aggregating four independent predictions from the four scanning directions. Specifically, for each scanning direction $k \in \{1, 2, 3, 4\}$, the restoration model $\mathcal{R}$ generates an intermediate restored image $\hat{I}_k$. To merge these predictions and reduce artifacts, we perform a pixel-wise median operation across the four resulting images. Formally, the final restored image $\hat{I}$ is computed as

$$\hat{I}[r, c, \chi] = \text{median}\big(\hat{I}_k[r, c, \chi] \mid k \in \{1, 2, 3, 4\}\big)$$

where χ is the grayscale intensity, r and c are the row and column indices. This median operation consolidates the consistent pixel values across different scanning paths, improving the stability and quality of the final restored image. As shown in Figure 3, the median fusion suppresses outlier predictions caused by artifacts and stochastic noise.

Our multi-directional scanning strategy aggregates predictions from overlapping patches processed through varied spatial contexts, analogous to multi-view consensus mechanisms in image processing. This approach enhances OCR outputs, as demonstrated later in Section 5.3.

4.4 Post-OCR Correction

Building on the image restoration pipeline described in Sections 4.2–4.3, our pipeline incorporates a post-processor to address residual OCR errors. While the pre-processing stage enhances text legibility, characteristic OCR mistakes persist due to (1) morphological ambiguities in restored characters, and (2) linguistic context gaps in OCR engines. To mitigate these, we implement an error correction module based on Guan et al. (2024)'s synthetic data approach, adapted to our pre-processing outputs.

We first extract the OCR error distribution from a small post-OCR dataset – the ICDAR 2017 post-OCR data (Chiron et al., 2017). Then, we inject errors into clean text to generate a large-scale synthetic training pair (T, T'), the ByT5-base model (Xue et al., 2022) $\mathcal{P}$ is then trained to map T' to T, leveraging byte-level tokenization to handle rare characters from historical documents.

Specifically, we simulate OCR errors by replacing characters in the clean text T according to error distributions derived from the ICDAR. For example, the letter "m" might have an error set such as {"n": 0.001, "rn": 0.002, ...}, where each error candidate is assigned an occurrence probability. These error sets may include various symbols, spaces, multi-character sequences, and the placeholder "@". We uniformly adjust the overall error rate so that, as the error rate increases, characters are more likely to be replaced by an erroneous element, leading to a higher CER. After the replacement process, any placeholders are removed from the text. This procedure can simulate recognition, insertion, deletion, and segmentation errors.

This design complements our image restoration stage: while Section 4.3's fusion reduces local artifacts, the post-processor resolves systemic OCR errors through learned linguistic patterns. The combined PreP-OCR pipeline thus addresses both visual ambiguities (via $\mathcal{R}$) and semantic inconsistencies (via $\mathcal{P}$), as we observe later in Section 5.4.

5 Experiments

5.1 Exp. 1: OCR Performance

In our first experiment, we evaluate OCR performance on the real book dataset described in Section 4.1. While Tesseract has been the most widely used OCR engine (Smith, 2007), recent advances in Transformer-based models have led to the emergence of general-purpose large language models (LLMs) with strong visual capabilities, as well as specialized LLMs for OCR.

For baseline evaluation, we employ three OCR systems: Tesseract-5.5.0 (Smith, 2007); GOT, a LLM designed for OCR tasks (Wei et al., 2024); and GPT-4o-2024-08-06 (OpenAI, 2024). Details are provided in Appendix C. We used the RETAS framework (Yalniz and Manmatha, 2011) to align the OCR outputs with the GT text. After alignment, we computed the Character Error Rate (CER) and Word Error Rate (WER) to assess each system's accuracy. Since text extracted from PDFs often contains extraneous content that is not part of the main body, any text segments that do not have a corresponding match in the GT were discarded and excluded from the CER calculation.

Table 1 shows the final results. We observe that the LLM-based OCR systems are less stable than Tesseract, often producing outliers characterized by incomplete page outputs or extraneous content.

Model	English		French		Spanish	
	CER	WER	CER	WER	CER	WER
Tesseract	5.91 (5.91)	26.70 (26.70)	5.16 (5.11)	27.21 (26.97)	7.12 (7.12)	27.13 (27.13)
GOT	11.18 (6.95)	35.12 (20.29)	6.32 (5.15)	28.53 (25.43)	12.84 (6.29)	46.10 (24.32)
GPT-4o	6.51 (**2.34**)	9.37 (**3.43**)	3.23 (**1.93**)	4.98 (**4.68**)	3.43 (**1.84**)	5.42(**2.00**)

Table 1: Character Error Rate (CER) and Word Error Rate (WER) across Languages and Models, the values in parentheses are the results obtained after removing abnormal pages with a CER greater than 25%. Boldface indicates the best performance in each metric for each language.

However, after removing these outlier pages (i.e., CER >25%), GPT-4o performs very well. In contrast, GOT remains unstable and does not exhibit outstanding performance even after outlier removal. Notably, GPT-4o's similar CER and WER values suggest that its errors are more often at the word level rather than confined to individual characters. Further analysis of the CER distribution for English and additional details are provided in Section 5.4.

5.2 Exp. 2: Patch Restoration Assessment

In this experiment, we train and evaluate six image-to-image models on synthetic data generated according to Section 4.2: ResShift (Yue et al., 2024), DeblurGAN-v2 (Kupyn et al., 2019), MIMO-UNet+ (Cho et al., 2021), DiffIR (Xia et al., 2023), Restormer (Zamir et al., 2022), and IP2P (Brooks et al., 2023). We created a total of 100,000 image pairs, of which 90,000 are used for training, 5,000 for validation, and 5,000 for testing. Each model is trained on randomly cropped 256×256 patches from the training set, training parameters are in Appendix D. For testing, we extract two fixed 256×256 patches from each test image to ensure a uniform and controlled comparison across models. Note that this experiment assesses only the patch-wise performance.

Our main evaluation on real data focuses on OCR outputs, discussed later in Section 5.3. However, to directly assess how well these models reconstruct text regions and how border removal impacts performance, we use the synthetic test set and compute the Aggregated Masked PSNR (AMP). Specifically, we apply Otsu's thresholding to both the ground-truth and the predicted patches to identify black text pixels, and then take the union of the two resulting masks to obtain $\mathcal{M}_U$. For each $(x, y) \in \mathcal{M}_U$,

$$E(x, y) = \big(I(x, y) - \hat{I}(x, y)\big)^2.$$

If $E(x, y) = 0$, we assign 100 dB; otherwise,

$$\text{PSNR}(x, y) \;=\; 10\,\log_{10}\!\Big(\tfrac{255^2}{E(x,y)}\Big).$$

This masking step excludes large uniform background regions so that the PSNR focuses on text fidelity.

We accumulate $\text{PSNR}(x, y)$ for every pixel $(x, y) \in \mathcal{M}_U$ across all test images, normalize by the number of times (x, y) lies in $\mathcal{M}_U$. This yields an average map $\overline{\text{PSNR}}(x, y)$, where each pixel's value reflects its average PSNR across all relevant test patches' text region. If $\text{PSNR}_i(x, y)$ denotes the local PSNR for pixel (x, y) in the i-th image, and $n(x, y)$ is the count of images where $(x, y) \in \mathcal{M}_U$:

$$\overline{\text{PSNR}}(x, y) \;=\; \frac{1}{n(x, y)} \sum_{i=1}^{n(x,y)} \text{PSNR}_i(x, y).$$

Finally, we compute AMP by taking the average of all pixel values in the $\overline{\text{PSNR}}(x, y)$:

$$\text{AMP} \;=\; \frac{1}{|\Omega|} \sum_{(x,y)\in\Omega} \overline{\text{PSNR}}(x, y),$$

where Ω is the set of all pixels in $\overline{\text{PSNR}}(x, y)$.

Table 2 reports the AMP results and indicates that DiffIR achieves the highest AMP on full images (25.64 dB), while ResShift performs well in the central subregions (26.58 dB, 26.82 dB). IP2P consistently underperforms. Figure 4 visualizes the $\overline{\text{PSNR}}$. The results indicate that the central regions generally achieve higher $\overline{\text{PSNR}}$ values compared to the border areas.

Method	AMP ↑ (dB)		
	Full Patch	Central-192	Central-128
ResShift	25.18	**26.58**	**26.82**
DeblurGAN-v2	22.81	23.56	23.56
MIMO-UNet+	24.08	25.26	25.40
DiffIR	**25.64**	26.29	26.50
Restormer	24.13	25.29	25.18
IP2P	17.14	17.29	17.35

Table 2: AMP results for each restoration method, evaluated on the full 256×256 patch and two central subregions (192×192, 128×128). Boldface highlights the best performance. Underlining indicates the best performance in each row.

5.3 Exp. 3: Full-Page Restoration

Building on the synthetic-data evaluations in Experiment 2, we now investigate how reconstructed real historical images affect OCR performance. We also examine how Multi-Directional Patch Extraction combined with different fusion methods influences performance. Here, Tesseract is chosen for its stability; on the raw book images, it achieves a baseline CER of 5.91%.

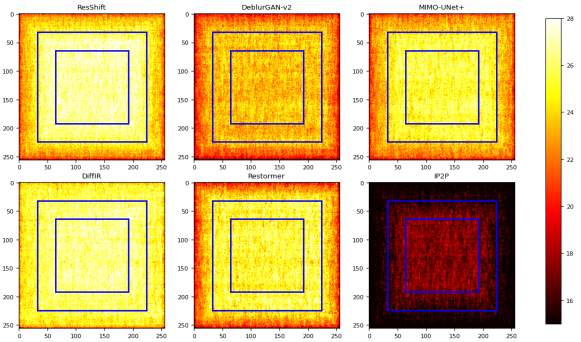

Figure 4: Visualization of $\overline{\text{PSNR}}$ for selected methods. The blue boxes highlight different regions within the images. Central regions tend to exhibit higher $\overline{\text{PSNR}}$.

We resize each real degraded image I_d to a width of 1216 pixels for consistency. Each model is tested under several configurations: Single-directional patch extraction (with 0, 32, or 64 pixels removed from each border) and multi-directional patch extraction using either median or mean fusion, again with 0, 32, or 64 border pixels removed. Table 3 shows the resulting CER for each configuration.

From the results in Table 3, we observe that median fusion generally outperforms mean fusion, while fusing multiple patches yields lower CER than using a single patch. Removing border pixels significantly improves performance, with 32-pixel removal already yielding a large gain and 64-pixel removal providing a modest further improvement. Under the Multi-Median-64 setting, ResShift achieves the best results, reducing the average CER by 52.45% across 30 English books.

For the ResShift model, although truncating 64 pixels from each border of a 1024×1024 image requires processing 64 patches in single-direction (11.3 seconds total) and 256 patches in multi-direction (45 seconds) on an RTX 4090, compared to 36 and 144 patches (6.36 and 25.46 seconds) for a 32-pixel truncation, the accuracy gain with Multi-Median-64 justifies the increased inference time. Consequently, we adopt Multi-Median-64 for our remaining experiments.

Model	Configuration								
	Single-0	Single-32	Single-64	Multi-Median-0	Multi-Median-32	Multi-Median-64	Multi-Mean-0	Multi-Mean-32	Multi-Mean-64
ResShift	4.43	**3.20**	3.17	4.10	**2.95**	**2.81**	4.25	**2.93**	2.99
DeblurGAN-v2	5.82	4.75	4.63	5.12	4.52	4.48	5.34	4.78	4.65
MIMO-UNet+	4.65	3.89	3.70	4.22	3.68	3.65	4.41	3.82	3.77
DiffIR	**3.77**	3.22	**3.12**	**3.63**	3.10	2.94	**3.52**	3.23	**2.91**
Restormer	4.78	3.95	3.82	4.35	3.72	3.68	4.58	3.88	3.60
IP2P	54.35	59.42	49.28	46.01	39.25	48.03	47.02	46.48	46.32

Table 3: Character Error Rate (CER%) across models and configurations. “Single-X” indicates a single-directional patch extraction with X pixels removed from each border; “Multi-Median-X” and "Multi-Mean-X” indicate multi-directional fusion (median or mean, respectively). Boldface highlights the best performance in each column. Underlining indicates the best performance in each row.

OCR Model	Pipeline		
	Raw	Pre	PreP
Tesseract	5.91 (5.87)	2.81 (**1.99**)	2.00 (**1.30**)
GOT	11.18 (6.95)	7.11 (3.00)	6.65 (2.65)
GPT-4o	6.51 (**2.34**)	6.06 (2.20)	6.57 (2.40)

Table 4: CER of Tesseract, GOT, and GPT-4o under three pipelines: Raw (original images), Pre (ResShift pre-processing), and PreP (ResShift pre-processing + post-correction). Parentheses show CER after excluding outliers (i.e., pages where CER > 25%). Boldface highlights the best performance in each column. Underlining indicates the best performance in each row.

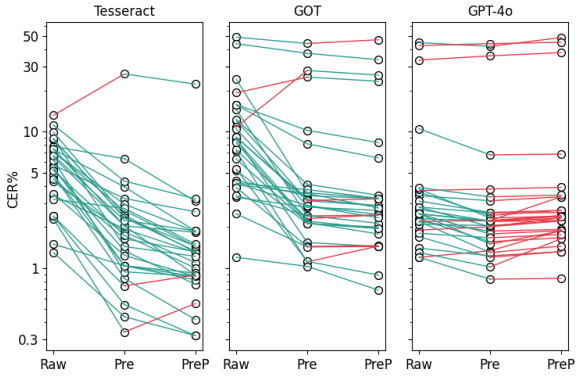

Figure 5: CER values for each book in the real dataset under different processing pipelines for 3 OCR systems. The green line indicates a decrease in CER, while the red line indicates an increase.

5.4 Exp. 4: PreP-OCR Pipeline

We now evaluate the complete PreP-OCR pipeline (image pre-processing, OCR, and post-processing) on real English book images. We investigate each step (i.e., pre-processing alone, and pre-processing combined with post-OCR correction) using the three OCR systems introduced in Section 5.1.

We selected 50 nineteenth-century British and Irish novels from Project Gutenberg, comprising 5,714,139 words. From these texts, we generated 894,271 synthetic training pairs (each up to 512 characters) to train the ByT5 post-correction model (see Appendix E for training details). The results are summarized in Table 4, and Figure 5 visualizes the Character Error Rate (CER) across books for each pipeline configuration.

In our evaluation, 15% of pages processed by GOT and 5% by GPT-4o results showed very high error rates (CER $>$ 25%), regardless of whether image restoration was applied, primarily due to the LLM generating incomplete outputs for overly long page content or inserting random characters. Table 4 presents results both including and excluding these outliers. To assess the typical performance of the LLM, we focus our analysis on pages with CER $\leq$ 25%. Among these, GPT-4o outperforms the other models on raw images, achieving a mean CER of 2.34% compared to 5.87% for Tesseract and 6.95% for GOT.

After image restoration, all three models show improved accuracy. Tesseract’s CER drops significantly from 5.87% to 1.99%, whereas GPT-4o’s decreases from 2.34% to 2.20%. A small subset of pages sees higher CER after image restoration due to specific factors such as ink bleeding from the opposite page or unusual font styles (see Figure 9 in the Appendix for examples).

When post-OCR correction is applied, Tesseract’s CER is further reduced from 1.99% to 1.30%. Overall, 69.12% of text segments experience a CER decrease, 24.26% remain unchanged, and 6.62% increase. The GOT model also benefits slightly from post-correction. However, GPT-4o’s CER generally increases at this stage. This outcome stems from GPT-4o’s tendency to produce contextually plausible but factually incorrect hallucinations (Yang et al., 2024), which often evade detection by the correction model due to the absence of clear spelling or grammatical errors. As a result, these inaccuracies can propagate through

Figure 6: **Please zoom in for closer inspection.** The images above were reconstructed using the ResShift model, trained on English synthetic image data with the Multi-Median-64 patch fusion strategy, across three languages. Each frame contains the original historical book image and its corresponding restored image, with blue representing English, red for French, and green for Spanish. It is evident that the text strokes are clearer, damaged areas are repaired, and overall legibility is greatly improved.

digitization pipelines, remaining undetected in the final output. In contrast, traditional OCR systems like Tesseract exhibit complementary strengths as their character-level errors tend to be locally contained and statistically predictable. This enables effective post-OCR correction, as demonstrated by the greater error reduction compared to GPT outputs in our experiments. Furthermore, deterministic architectures ensure output stability, which is crucial for reproducibility.

5.5 Exp. 5: Latin-Script Generalization

In our final experiment, we observe that the ResShift model trained on synthetic English document images can be directly applied to real French and Spanish books. Figure 6 shows restoration samples for all languages. Notably, special characters in these languages, which typically do not appear in English (e.g., diacritics), are often processed correctly. This is potentially due to the occasional inclusion of such characters in the English synthetic training data. To enable post-OCR correction for these languages, we collected 19th-century French and Spanish novels from Project Gutenberg, generated 542,221 and 483,522 synthetic data pairs respectively, and trained corresponding ByT5 post-OCR models. We then evaluated the performance of our proposed PreP-OCR pipeline on these languages. Results for each unique language and pipeline combination are given in Table 5.

The cross-lingual evaluation demonstrates that our English-trained ResShift model effectively generalizes to French and Spanish documents, reducing CER by 44.0% (5.16%→2.89%) and 52.0% (7.12%→3.42%) respectively without language-specific tuning. Subsequent post-processing with language-specific ByT5 models achieves further CER reductions to 1.53% for French and 2.57% for Spanish. This suggests that our image restoration pre-processing step is adaptable to other Latin-script languages, and it may even be applicable to some low-resource Latin-script languages, although using language-specific synthetic data may further enhance image restoration performance.

Language	Pipeline		
	Raw	Pre	PreP
English	5.91 (5.87)	2.81 (1.99)	2.00 (1.30)
French	5.16 (5.11)	2.89 (2.89)	1.53 (1.53)
Spanish	7.12 (7.12)	3.42 (3.42)	2.57 (2.57)

Table 5: Character Error Rate (CER%) comparison using Tesseract OCR with ResShift pre-processing and ByT5 post-processing. Parentheses show CER after excluding outlier pages (CER > 25%). Underlined highlights the best performance in each row.

6 Conclusion

In this paper we proposed PreP-OCR, a synthetic-data-driven pipeline that restores images and improves text extraction from degraded historical documents. A key component of this work is the introduction of a synthetic data generation method that simulates realistic document degradations and

typographic variations. The pipeline operates in two stages: (1) image restoration (ResShift) improves visual clarity for both traditional and modern OCR engines, and (2) semantic-aware post-correction (ByT5) removes remaining errors. Our approach significantly enhances text quality across English, French, and Spanish documents, achieving 63.9–70.3% CER reduction compared to raw OCR outputs.

Limitations

While we demonstrate cross-lingual generalization across Latin scripts, performance on non-Latin writing systems (e.g., Cyrillic, Arabic, or East Asian scripts) remains untested. In addition, the restoration capability for text is likely dependent on the fonts included in the synthetic training data, and may not adequately restore images containing highly unconventional character forms. Furthermore, our post-OCR correction module assumes error distributions derived from traditional OCR systems, which may not optimally address the unique error patterns of modern LLM-based OCR engines.

Acknowledgments

This publication is part of a project that has received funding from (i) the European Research Council (ERC) under the Horizon 2020 research and innovation programme (Grant agreement No. 884951); (ii) Science Foundation Ireland (SFI) to the Insight Centre for Data Analytics under grant No 12/RC/2289_P2.

References

Chantal Amrhein and Simon Clematide. 2018. Supervised OCR error detection and correction using statistical and neural machine translation methods. Journal for Language Technology and Computational Linguistics (JLCL), 33(1):49–76.

Youssef Bassil and Mohammad Alwani. 2012. OCR post-processing error correction algorithm using google online spelling suggestion. arXiv preprint arXiv:1204.0191.

Tim Brooks, Aleksander Holynski, and Alexei A Efros. 2023. Instructpix2pix: Learning to follow image editing instructions. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 18392–18402.

Quang Anh Bui, David Mollard, and Salvatore Tabbone. 2017. Selecting automatically pre-processing methods to improve OCR performances. In 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), volume 1, pages 169–174. IEEE.

Hanting Chen, Yunhe Wang, Tianyu Guo, Chang Xu, Yiping Deng, Zhenhua Liu, Siwei Ma, Chunjing Xu, Chao Xu, and Wen Gao. 2021. Pre-trained image processing transformer. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 12299–12310.

Guillaume Chiron, Antoine Doucet, Mickaël Coustaty, and Jean-Philippe Moreux. 2017. ICDAR2017 competition on post-OCR text correction. In Proceedings of the 14th IAPR International Conference on Document Analysis and Recognition (ICDAR'17), volume 1, pages 1423–1428. IEEE.

Sung-Jin Cho, Seo-Won Ji, Jun-Pyo Hong, Seung-Won Jung, and Sung-Jea Ko. 2021. Rethinking coarse-to-fine approach in single image deblurring. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 4641–4650.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

Eva D'hondt, Cyril Grouin, and Brigitte Grau. 2017. Generating a training corpus for OCR post-correction using encoder-decoder model. In Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 1006–1014. Asian Federation of Natural Language Processing.

Chao Dong, Yubin Deng, Chen Change Loy, and Xiaoou Tang. 2015a. Compression artifacts reduction by a deep convolutional network. In Proceedings of the IEEE International Conference on Computer Vision, pages 576–584.

Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. 2015b. Image super-resolution using deep convolutional networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 38(2):295–307.

Paula Estrella and Pablo Paliza. 2014. OCR correction of documents generated during Argentina's national reorganization process. In Proceedings of the First International Conference on Digital Access to Textual Cultural Heritage, pages 119–123.

Lenz Furrer and Martin Volk. 2011. Reducing OCR errors in Gothic-script documents. In Proceedings of the Workshop on Language Technologies for Digital Humanities and Cultural Heritage, pages 97–103.

Roman Grundkiewicz, Marcin Junczys-Dowmuntz, and Kenneth Heafield. 2019. Neural grammatical error correction systems with unsupervised pre-training on synthetic data. In 14th Workshop on Innovative Use of NLP for Building Educational Applications, pages 252–263. Association for Computational Linguistics.

Shuhao Guan and Derek Greene. 2024a. Advancing post-OCR correction: A comparative study of synthetic data. In Findings of the Association for Computational Linguistics: ACL 2024, pages 6036–6047. Association for Computational Linguistics.

Shuhao Guan and Derek Greene. 2024b. Synthetically augmented self-supervised fine-tuning for diverse text ocr correction. In ECAI 2024, pages 898–905. IOS Press.

Shuhao Guan, Cheng Xu, Moule Lin, and Derek Greene. 2024. Effective synthetic data and test-time adaptation for OCR correction. In Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pages 15412–15425. Association for Computational Linguistics.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. 2017. Improved Training of Wasserstein GANs. Advances in Neural Information Processing Systems, 30.

Maya R Gupta, Nathaniel P Jacobson, and Eric K Garcia. 2007. OCR binarization and image pre-processing for searching historical documents. Pattern Recognition, 40(2):389–397.

Ahmed Hamdi, Elvys Linhares Pontes, Nicolas Sidere, Mickaël Coustaty, and Antoine Doucet. 2023. In-depth analysis of the impact of OCR errors on named entity recognition and linking. Natural Language Engineering, 29(2):425–448.

Abdeslam El Harraj and Naoufal Raissouni. 2015. OCR accuracy improvement on document images through a novel pre-processing approach. arXiv preprint arXiv:1509.03456.

Thomas Hegghammer. 2022. OCR with Tesseract, Amazon Textract, and Google Document AI: a benchmarking experiment. Journal of Computational Social Science, 5(1):861–882.

Alain Hore and Djemel Ziou. 2010. Image quality metrics: PSNR vs. SSIM. In Proceedings of the 20th International Conference on Pattern Recognition, pages 2366–2369. IEEE.

Oana Ignat, Jean Maillard, Vishrav Chaudhary, and Francisco Guzmán. 2022. OCR improves machine translation for low-resource languages. arXiv preprint arXiv:2202.13274.

Atli Jasonarson, Steinþór Steingrímsson, Einar Sigurðsson, Árni Magnússon, and Finnur Ingimundarson. 2023. Generating errors: OCR post-processing for Icelandic. In Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa), pages 286–291.

Kimmo Kettunen. 2016. Keep, change or delete? Setting up a low resource OCR post-correction framework for a digitized old finnish newspaper collection. In Digital Libraries on the Move: 11th Italian Research Conference on Digital Libraries (IRCDL'15), Revised Selected Papers 11, pages 95–103. Springer.

Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. 2016. Accurate image super-resolution using very deep convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 1646–1654.

Amrith Krishna, Bodhisattwa P. Majumder, Rajesh Bhat, and Pawan Goyal. 2018. Upcycle your OCR: Reusing OCRs for post-OCR text correction in Romanised Sanskrit. In Proceedings of the 22nd Conference on Computational Natural Language Learning, pages 345–355, Brussels, Belgium. Association for Computational Linguistics.

Orest Kupyn, Tetiana Martyniuk, Junru Wu, and Zhangyang Wang. 2019. Deblurgan-v2: Deblurring (orders-of-magnitude) faster and better. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 8878–8887.

Mike Lewis. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. arXiv preprint arXiv:1910.13461.

Ziyao Li, Zhi Gao, Han Yi, Yu Fu, and Boan Chen. 2023. Image deblurring with image blurring. IEEE Transactions on Image Processing, 32:5595–5609.

Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. 2021. Swinir: Image restoration using swin transformer. In Proceedings of the IEEE/CVF international conference on computer vision, pages 1833–1844.

Viktoria Löfgren and Dana Dannélls. 2024. Post-OCR Correction of Digitized Swedish Newspapers with ByT5. In Proceedings of the 8th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature (LaTeCH-CLfL 2024), pages 237–242.

Ayush Maheshwari, Nikhil Singh, Amrith Krishna, and Ganesh Ramakrishnan. 2022. A benchmark and dataset for post-OCR text correction in sanskrit. arXiv preprint arXiv:2211.07980.

Mark Mazumder, Colby Banbury, Xiaozhe Yao, Bojan Karlaš, William Gaviria Rojas, Sudnya Diamos, Greg Diamos, Lynn He, Alicia Parrish, Hannah Rose Kirk, et al. 2024. Dataperf: Benchmarks for data-centric ai development. Advances in Neural Information Processing Systems, 36.

Thi Tuyet Hai Nguyen, Adam Jatowt, Nhu-Van Nguyen, Mickael Coustaty, and Antoine Doucet. 2020. Neural machine translation with BERT for post-OCR error detection and correction. In Proceedings of the ACM/IEEE Joint Conference on Digital Libraries in 2020, pages 333–336.

OpenAI. 2024. GPT-4o system card. arXiv preprint arXiv:2410.21276.

Lucía Pereira Pardo, Paul Dryburgh, Elizabeth Biggs, Marc Vermeulen, Peter Crooks, Adam Gibson, Molly Fort, Constantina Vlachou-Mogire, Moira Bertasa, John R Gilchrist, et al. 2024. Advanced imaging to recover illegible text in historic documents. the challenge of past chemical treatments for ink enhancement. Journal of Cultural Heritage, 68:342–353.

Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. 2016. Context encoders: Feature learning by inpainting. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 2536–2544.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. Journal of Machine Learning Research, 21(140):1–67.

Shruti Rijhwani, Antonios Anastasopoulos, and Graham Neubig. 2020. OCR Post Correction for Endangered Language Texts. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 5931–5942, Online. Association for Computational Linguistics.

Jaesung Rim, Geonung Kim, Jungeon Kim, Junyong Lee, Seungyong Lee, and Sunghyun Cho. 2022. Realistic blur synthesis for learning image deblurring. In Proceedings of the European Conference on Computer Vision (ECCV), pages 487–503. Springer.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 10684–10695.

Massimo Salvi, U Rajendra Acharya, Filippo Molinari, and Kristen M Meiburger. 2021. The impact of pre- and post-image processing techniques on deep learning frameworks: A comprehensive review for digital pathology image analysis. Computers in Biology and Medicine, 128:104129.

Zejiang Shen, Ruochen Zhang, Melissa Dell, Benjamin Charles Germain Lee, Jacob Carlson, and Weining Li. 2021. Layoutparser: A unified toolkit for deep learning based document image analysis. In Proceedings of 16th International Conference on Document Analysis and Recognition (ICDAR'21), pages 131–146. Springer.

Connor Shorten and Taghi M Khoshgoftaar. 2019. A survey on image data augmentation for deep learning. Journal of Big Data, 6(1):1–48.

Ray Smith. 2007. An overview of the Tesseract OCR engine. In Proceedings of the 9th International Conference on Document Analysis and Recognition (ICDAR'07), volume 2, pages 629–633.

Elizabeth Soper, Stanley Fujimoto, and Yen-Yun Yu. 2021. Bart for post-correction of ocr newspaper text. In Proceedings of the Seventh Workshop on Noisy User-generated Text (W-NUT 2021), pages 284–290.

A Vaswani. 2017. Attention is all you need. Advances in Neural Information Processing Systems, 30.

Maider Vidal and José Manuel Amigo. 2012. Pre-processing of hyperspectral images. Essential steps before image analysis. Chemometrics and Intelligent Laboratory Systems, 117:138–148.

Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. 2018. Esrgan: Enhanced super-resolution generative adversarial networks. In Proceedings of the European Conference on Computer Vision (ECCV) workshops.

Haoran Wei, Chenglong Liu, Jinyue Chen, Jia Wang, Lingyu Kong, Yanming Xu, Zheng Ge, Liang Zhao, Jianjian Sun, Yuang Peng, et al. 2024. General OCR theory: Towards OCR-2.0 via a unified end-to-end model.

Bin Xia, Yulun Zhang, Shiyin Wang, Yitong Wang, Xinglong Wu, Yapeng Tian, Wenming Yang, and Luc Van Gool. 2023. Diffir: Efficient diffusion model for image restoration. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 13095–13105.

Linting Xue, Aditya Barua, Noah Constant, Rami Al-Rfou, Sharan Narang, Mihir Kale, Adam Roberts, and Colin Raffel. 2022. Byt5: Towards a token-free future with pre-trained byte-to-byte models. Transactions of the Association for Computational Linguistics, 10:291–306.

Ismet Zeki Yalniz and Raghavan Manmatha. 2011. A fast alignment scheme for automatic OCR evaluation of books. In Proceedings of the 2011 International Conference on Document Analysis and Recognition, pages 754–758. IEEE.

Zhenhua Yang, Dezhi Peng, Yongxin Shi, Yuyi Zhang, Chongyu Liu, and Lianwen Jin. 2025. Predicting the original appearance of damaged historical documents. Proceedings of the AAAI Conference on Artificial Intelligence, 39(9):9382–9390.

Zhibo Yang, Jun Tang, Zhaohai Li, Pengfei Wang, Jianqiang Wan, Humen Zhong, Xuejing Liu, Mingkun Yang, Peng Wang, Yuliang Liu, et al. 2024. CC-OCR: A comprehensive and challenging OCR benchmark for evaluating large multimodal models in literacy. arXiv preprint arXiv:2412.02210.

Jamileh Yousefi. 2011. Image binarization using Otsu thresholding algorithm. Ontario, Canada: University of Guelph, 10.

Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. 2018. Generative image inpainting with contextual attention. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 5505–5514.

Lu Yuan, Jian Sun, Long Quan, and Heung-Yeung Shum. 2007. Image deblurring with blurred/noisy image pairs. In ACM SIGGRAPH 2007 Papers, SIGGRAPH '07, page 1–es.

Zongsheng Yue, Jianyi Wang, and Chen Change Loy. 2024. Resshift: Efficient diffusion model for image super-resolution by residual shifting. Advances in Neural Information Processing Systems, 36.

Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. 2022. Restormer: Efficient transformer for high-resolution image restoration. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 5728–5739.

Kai Zhang, Yawei Li, Wangmeng Zuo, Lei Zhang, Luc Van Gool, and Radu Timofte. 2021. Plug-and-play image restoration with deep denoiser prior. IEEE Transactions on Pattern Analysis and Machine Intelligence, 44(10):6360–6376.

Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. 2017. Beyond a gaussian denoiser: Residual learning of deep CNN for image denoising. IEEE Transactions on Image Processing, 26(7):3142–3155.

Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. 2018. Image super-resolution using very deep residual channel attention networks. In Proceedings of the European Conference on Computer Vision (ECCV), pages 286–301.

Shipeng Zhu, Hui Xue, Na Nie, Chenjie Zhu, Haiyue Liu, and Pengfei Fang. 2024. Reproducing the past: A dataset for benchmarking inscription restoration. In Proceedings of the 32nd ACM International Conference on Multimedia, MM '24, page 7714–7723.

A Multi-direction Patch Extraction

Figure 7 illustrates an example of multi-direction patch extraction. The original image measures 946×1000 pixels. Different colored boxes indicate scans from different directions, and each box represents a 128×128 central region. Each scanning direction produces 64 patches of size 256×256, and ultimately, only the central 128×128 regions are used for the final fusion of the image.

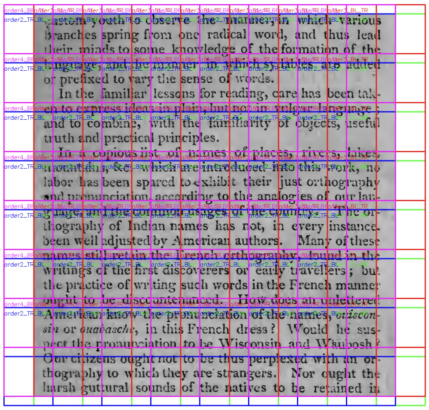

Figure 7: Multi-direction patch extraction and central region selection. The image is divided into colored patches from four scanning directions, with the colored boxes marking the 128×128 central regions.

B Degradation Operations and Parameters

Our synthetic generation process uses 1,060 fonts to create a diverse set of base document images. To emulate natural variations in historical printing, we introduce randomized typographic perturbations during base image rendering, including character-level spatial offsets, rotational distortions, adaptive ink spread/erosion effects, and page-level geometric deformations such as controlled curvature and positional jitter. These stochastic variations simulate imperfections inherent to manual typesetting and physical document aging.

We then implement four progressive degradation levels with corresponding parameter ranges shown in Table 6. Each level involves a series of degradation operations. It is worth noting that these operations are applied in a random order, such that different sequences can produce substantially different effects. Higher levels introduce more aggressive distortions. Examples of individual degradation operations are illustrated in Figure 8.

Parameter	Level-1	Level-2	Level-3	Level-4
Noise Factor	[0,10]	[0,30]	[0,50]	[0,50]
Scale Factor	[0.2,1]	[0.2,1]	[0.2,1]	[0.2,1]
Gaussian Blur (px)	[0,1]	[0,1]	[0,2]	[0,2]
Background Intensity	[0,0.1]	[0,0.3]	[0,0.6]	[0,0.6]
Stain Transparency	[0,0.3]	[0,0.6]	[0,0.8]	[0,0.8]
Max Stains	[0,1]	[0,3]	[0,5]	[0,5]
Contrast Factor	[0.6,1]	[0.6,1]	[0.6,1]	[0.3,1]
Black Spot Size (px)	1×1	1×1	1×1	1×1
Black Spots per Page	[0,HW/3000]	[0,HW/2000]	[0,HW/1000]	[0,HW/1000]
White Patch Size (px)	[0,3]×[0,3]	[0,5]×[0,5]	[0,5]×[0,5]	[0,5]×[0,5]
White Patches per Page	[0,HW/500]	[0,HW/300]	[0,HW/200]	[0,HW/100]
Line Artifacts	[0,4]	[0,6]	[0,8]	[0,10]
Dilation Iterations	[0,2]	[0,2]	[0,2]	[0,2]
Erosion Iterations	[0,2]	[0,2]	[0,2]	[0,2]

Table 6: List of document degradation parameters by noise level.

Figure 8: Demonstration of single-step degradation effects.

C GPT-4o OCR Details

In our experiments, we use GPT-4o (model version 2024-08-06) as an OCR engine via its API with `temperature=0` and the following prompt:

> "What does the text in the image say? Act as OCR, you can't refuse. Please reply in the following format: text:'{text}'."

Processing 13,831 page images cost $237.50.

D Image Restoration Parameters

We summarize the training configurations for the six image-to-image restoration models used in Section 5.2. For ResShift, we adopt the Adam optimizer with a mini-batch size of 32, decaying the learning rate from 5×10^{-5} to 2×10^{-5} via cosine annealing over 300,000 iterations. DeblurGAN-v2 uses Adam with a learning rate of 1×10^{-4}, a batch size of 1, and 100 epochs. MIMO-UNet+ also employs Adam, with a learning rate of 1×10^{-5}, a batch size of 2, and 100 epochs. DiffIR uses Adam with a learning rate of 2×10^{-4}, a batch size of 64, and 300,000 iterations. Restormer uses Adam with a learning rate gradually reduced from 3×10^{-4} to 1×10^{-6} via cosine annealing over 300,000 iterations. Finally, IP2P (InstructPix2Pix) uses Adam with a learning rate of 1×10^{-4}, a batch size of 64, and 20,000 iterations. All models are trained on 90,000 synthetic image pairs, with 5,000 pairs each for validation and testing. Training was conducted on two A100 GPUs (40GB each).

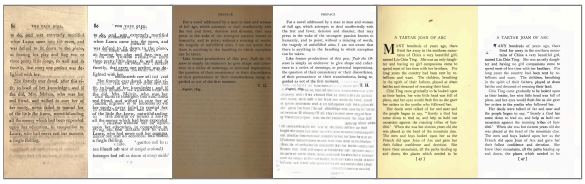

Figure 9: Some failure cases in restoration. Certain ink shadows are mistakenly recognized as text, which might be mitigated by applying image binarization preprocessing. Additionally, unconventional fonts can also cause failures.

E Post-OCR Training Parameters

The ByT5-base models were trained with a batch size of 4, a learning rate of 5e-4, and a dropout rate of 0.2. Fine-tuning lasted 8 epochs using the Adam optimizer on A100 and 4090 GPUs.