Generating Synthetic Handwritten Historical Documents With OCR Constrained GANs

Lars Vögtlin*, Manuel Drazyk*, Vinaychandran Pondenkandath,
Michele Alberti, and Rolf Ingold

Document Image and Voice Analysis Group (DIVA)
University of Fribourg, Switzerland
{firstname.lastname}@unifr.ch

Abstract. We present a framework to generate synthetic historical documents with precise ground truth using nothing more than a collection of unlabeled historical images. Obtaining large labeled datasets is often the limiting factor to effectively use supervised deep learning methods for Document Image Analysis (DIA). Prior approaches towards synthetic data generation either require human expertise or result in poor accuracy in the synthetic documents. To achieve high precision transformations without requiring expertise, we tackle the problem in two steps. First, we create template documents with user-specified content and structure. Second, we transfer the style of a collection of unlabeled historical images to these template documents while preserving their text and layout. We evaluate the use of our synthetic historical documents in a pre-training setting and find that we outperform the baselines (randomly initialized and pre-trained). Additionally, with visual examples, we demonstrate a high-quality synthesis that makes it possible to generate large labeled historical document datasets with precise ground truth.

Keywords: OCR · CycleGAN · synthetic data · historical documents.

1 Introduction

Large labeled datasets play a major role in the significant performance increases seen in DIA and computer vision over the last decade. These datasets – often containing millions of labeled samples – are typically used to train deep neural networks in a supervised setting, achieving state-of-the-art performance in tasks such as text line segmentation [2], Optical Character Recognition (OCR) [4] or layout analysis [24]. However, such methods are much more challenging to train in settings where no labeled data is available. The size of labeled datasets is limited to a few hundred or thousand samples – as is often the case with historical documents [14,7].

Common strategies to deal with limited labeled data include (1) transfer-learning, (2) synthesizing artificial data, or (3) unsupervised learning. In (1)

---

* Both authors contributed equally to this work.

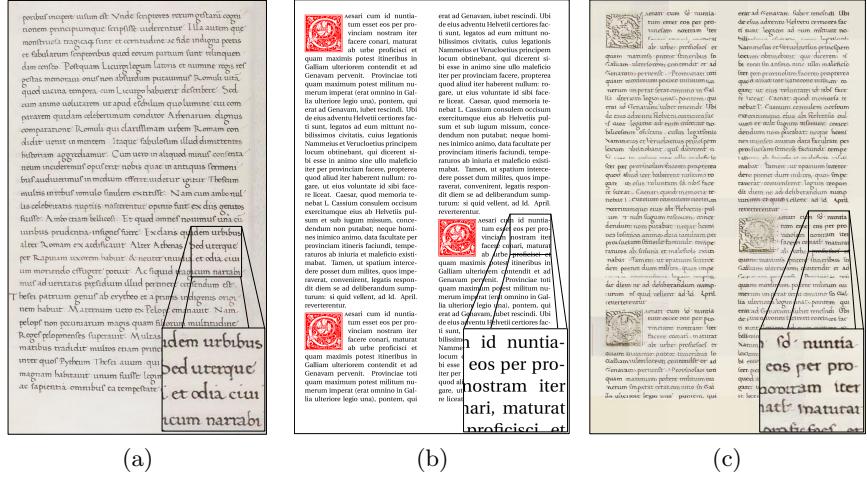|		
(a)	(b)	(c)

Fig. 1: Inputs for the second step of our framework and the output of the network. (a) represents the style template for our output document. (b) shows a source document that was generated using LATEX. (c) shows the corresponding transformed version of the template image (b). The transformation between (b) and (c) preserves overall structure and content.

typical procedure is to train a deep neural network on similar data and then fine-tune this network on the small labeled target dataset. The success depends on having datasets similar enough to the target dataset to perform pre-training. (2) has been an active area of DIA research. Baird [3], Kieu et al. [16] and Seuret et al [25] focus on degrading real document images using defect models to augment datasets. Other tools such as DocEmul [5] and DocCreator [14] aim to create synthetic document images using a combination of user-specified structure, background extraction, degradation methods, and other data augmentation approach. However, such approaches still require human expertise in designing appropriate pipelines to generate realistic documents. When large unlabeled datasets are available for the target task, a common practice is to use unsupervised learning methods such as autoencoders [20] to learn representations. However, recent work [1] shows that autoencoders trained for reconstruction are not useful for this task. Another possibility is to use unlabeled data in a Generative Adversarial [10,32] setting to synthesize artificial data that looks similar in appearance to the unlabeled data.

More recent work in document image synthesis has used deep learning, and Generative Adversarial Network (GAN) based approaches. But these approaches [23,29,12,15] result in various issues: the produced data matches the overall visual style of historical documents but fail to produce meaningful textual content; they require paired datasets, which defeats the purpose of using unlabeled data; only create text of fixed length.

In this paper, we present a framework to generate historical documents without relying on human expertise or labeled data[1]. We approach this problem in two steps. First, we create template document images that contain user-specified content and structure using LaTeX[2]. Second, using the user-specified template documents and a collection of unlabeled historical documents, we learn a mapping function to transform a given template document into the historical style while preserving the textual content and structure present in the template document.

We evaluate the usefulness of our synthetically generated images by measuring the performances of a deep learning model on the downstream task of OCR. Specifically, we measure the performances of this model when (1) trained only on the target dataset (St. Gall [9]), (2) pre-trained on a similar dataset (IAM Handwriting database [19]) and then fine-tuned on the target dataset and finally when (3) pre-trained on our synthetic images and then fine-tuned on the target dataset. This will allow us to compare against a standard supervised baseline as well as a reasonable transfer learning baseline. Our empirical experiments show that, the model pre-trained on our synthetic images (see point 3 above) is outperforming the supervised and transfer learning baselines by 38% and, respectively, 14% lower Character Error Rate (CER).

Main Contribution

This paper extends the existing work on synthetic document generation by providing a general framework that can produce realistic-looking historical documents with a specific style and textual content/structure. We introduce a two-step CycleGAN based process that leverages two Text Recognizer (TR) networks to condition the learning process. This additional signal let us overcome the main limitations of previous work and enable us to obtain significantly better performance measured on a robust set of benchmarks.

2   Datasets

In this work, we use three datasets: the user-specified template document dataset (source domain dataset; see Section 2.1); a dataset of real unlabeled historical documents (target domain dataset) whose style which we want to learn in the transformation function; a dataset of real labeled historical documents (evaluation dataset) with transcription ground truth that we use to evaluate our methods.

2.1   Source Domain Dataset

We create a collection of template documents with user-specified content and structure as Pondenkandath et al. [23]. Our template document images are gen-
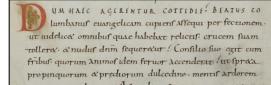
---

[1] https://github.com/DIVA-DIA/Generating-Synthetic-Handwritten-Historical-Documents

[2] This can be done with any other word processing tool such as MS Word.

(a) HBA

(b) StGall

(c) IAM

Fig. 2: We use the HBA dataset (a) as the target historical style, the Saint Gall dataset (b) for evaluating our synthetic data in a pre-training setting and the IAM Handwriting Database (c) as a pre-trained baseline.

Fig. 3: Samples from the hand annotated subset of the HBA dataset used for validation purposes.

erated based on the specifications from LaTeX files; they define the layout, font, size, and content (see Figure 1). As a text, we use the *Bellum Gallicum* [8] with a one or two-column layout. Additionally, we populated each document with different decorative starting letters. The advantage of this technique is that we have very precise ground truth, which is the transcription of the document and the exact position of the word on the page. This dataset contains 455 document images with a resolution of 2754×3564.

2.2 Target Domain Dataset

The target domain dataset refers to the collection of historical documents whose style we aim to learn in the transformation function. To create this dataset, we use the historical documents present in the Historical Book Analysis Competition (HBA) 1.0 dataset [21]. The HBA dataset comprises 11 books, where 5 are manuscripts and 6 books are printed, containing 4436 scanned historical document images. These books were published between the 13th and 19th
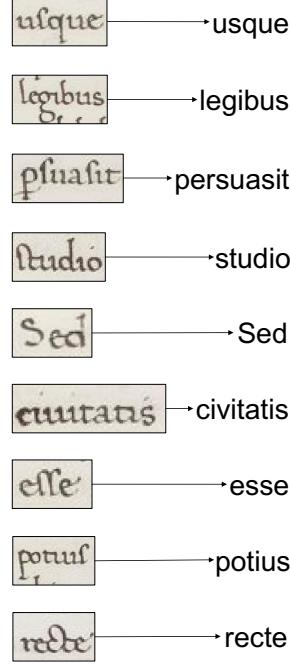
centuries and are written in different scripts and languages. We use one book of this dataset; the handwritten Latin book "Plutarchus, Vitae illustrium virorum". This book contains 730 colored pages with a resolution of 6158×4267 (see Figure 2a) from which we filtered out 120 pages (blank, binding, and title pages), leaving us with 600 pages that contain only text. To validate the best model for our downstream evaluation task, we hand-labeled 350 individual word crops from this book.

2.3   Evaluation Dataset

As part of the evaluation process, we use two different datasets. Our evaluation protocol involves pre-training a Handwritten Text Recognition (HTR) model using synthetic data generated using our method, and then evaluating it in a fine-tuning setting on the St. Gall dataset [9] (see Figure 2b). The Saint Gall dataset includes 40 pages of labeled historical handwritten manuscripts containing 11'597 words and 4'890 word labels. Each image has a resolution of 3328×4992 with quality of 300dpi.

To compare our synthetic data pre-training against pre-training on a real handwritten dataset, we pre-train an HTR model on the IAM Handwriting Database [19] (see Figure 2c). This HTR model (pre-trained on the IAM Handwriting Database) is then evaluated similarly in a fine-tuning setting on the St. Gall dataset. The IAM Handwriting Database contains 1'539 handwritten scanned pages with 115'320 words and 6'625 unique words. The word images are normalized and in black-white colorization.

3   Method

Our method uses a CycleGAN formulation, along with HTR models, to further constrain the synthesis process. To train the CycleGAN, we use unpaired collections of user-specified template images (source domain) and real historical images (target domain). The source domain documents specify the content and overall structure, and the target domain documents exemplify the style we want in our final synthetic historical documents.

Pondenkandath et al. [23] have shown that using only the CycleGAN formulation with the source and target domain datasets is enough to produce synthetic documents that appear stylistically similar to the target domain. However, they do not contain the content or structure specified in the source domain documents. We add a loss term using HTR models that aim to read user-specified content from the synthesized historical documents to address this issue. After completing training, we obtain a generator that transforms any given template image to a corresponding synthetic historical version.
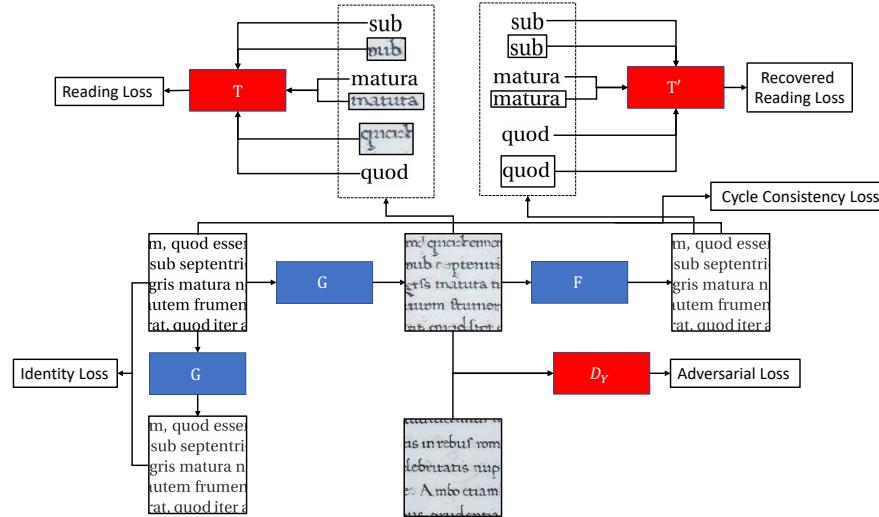
Fig. 4: The CycleGAN architecture presented in [32] with two additional TRs T and T' and the five different loss terms.

3.1   Model Architecture

Our model architecture is based on the CycleGAN formulation. It uses the cycle-consistency loss to transform an image from a given source domain to a target domain in a bi-directional fashion. This architecture introduces two main challenges. First, generating text in the target domain that is human-readable at the character and word levels is difficult due to the under-constrained nature of the CycleGAN architecture for our task. Second, CycleGANs are prone to emergent GAN steganography [31]; where the generators in a GAN can learn to hide information from the discriminator within the synthesized image and use it for perfect reconstruction.

To tackle the first problem of generating human-readable text, we introduce two HTR models T and T' to our architecture (see Figure 4). We aim to adjust for the under-constrained nature of the CycleGAN by adding additional loss terms based on this HTR model. We adopt the bi-directional Long Short-Term Memory (LSTM) and Connectionist Temporal Classification (CTC) based HTR architecture used by the winners of the text recognition competition at ICFHR'18 [26].

The first HTR model T evaluates the quality of the characters or words produced by transforming a source domain template image to the target historical domain. To do this, it takes as input the synthetic images produced by the source-to-target generator G as well as the textual content and location information from the template document images. The second HTR model T' evaluates the quality of the reconstructed source domain documents (produced by the

target-source generator F) by comparing the reconstructed image against the same textual content and location information as T.

The second challenge is overcoming the tendency of CycleGANs to hide information about the input within the generated synthetic image [6]. This tendency arises naturally due to cooperation between the generators and is potentially exacerbated by the presence of the HTR models. To minimize the cyclic consistency loss as well as the loss introduced by the TRs, the generator G attempts to hide information that can be effectively decoded by generator F to produce good reconstructions, as well as information that allows the TRs to recover the textual content. These results in synthetic documents that do not satisfy the constraints of our synthesis process, yet produce very low reconstruction losses and HTR losses. In some of our preliminary experiments, the generator places the encoded template document into the target document by adding or subtracting the encoded value from each pixel. The influence on the image is so small that it is nearly impossible for humans to detect, and it is even challenging to be detected by the style discriminator. Allowing the CycleGAN to cheat prevents it from learning the correct mapping from the target domain back to the source domain, negatively affecting the style representation learned by the GAN.

To prevent the CycleGAN from creating this hidden embedding, we add Gaussian noise to the synthetic document images. This low-frequency noise disturbs the encoded message of the generator, making it much harder to cheat by using steganography. This noise effectively prevents the network from cheating, as a much stronger signal would be needed, which would manipulate the appearance of the image in a way that is more easily detected by the human eye as well as the style discriminator, and thus would achieve a much lower performance score.

3.2   Loss Functions

We train with a loss objective that consists of five different loss terms (see Figure 4). The identity loss, the adversarial loss, and the cycle consistency loss are the loss terms presented in the original CycleGAN paper [32]. To solve the readability problem described in Section 3.1, we introduce two additional loss terms using the HTRs system, the reading loss, and the recovered reading loss. The identity, adversarial, and cycle consistency loss are calculated in both directions, but the reading loss terms are just calculated once per cycle.

Formally, we aim to learn mappings between two domains X and Y with the help of N training samples $x \in X$ and M samples $y \in Y$. Each document image x is composed of pairs of its word images and the corresponding word text (ground truth) $x = ((x_1, z_1), (x_2, z_2), ..., (x_n, z_n))$ where $n = |x|$ and $|x|$ is the amount of words in a document.

The data distributions are denoted as $x \sim p_{data}(x)$ and $y \sim p_{data}(y)$. We also define a projection function α where $\alpha_1(x)$ refers to the first and $\alpha_2(x)$ to the second element of the tuple. The transformation functions of generators G (source-target) and F (target-source) are denoted respectively by $g : X \to Y$ and $f : Y \to X$. Additionally, we have two adversarial discriminators D_x and

D_y. The task of D_x is to distinguish the images of $\{x\}$ and $\{f(y)\}$, and in the same fashion D_y learns to differentiate between $\{y\}$ and $\{g(x)\}$.

Identity Loss This loss term [32,28] is used to regularize both generators to function as identity mapping functions when provided with real samples of their respective output domains. Zhu et al. [32] observed that in the absence of this identity loss term, the generators G and F were free to change the tint between the source and target domains even without any need to do it. The identity loss is defined as follows:

$$\mathcal{L}_{\text{identity}}(G,F) = \mathbb{E}_{x \sim p_{\text{data}}(x)}[\|G(\alpha_1(x)) - \alpha_1(x)\|_1]$$
$$+ \mathbb{E}_{y \sim p_{\text{data}}(y)}[\|F(y) - y\|_1]. \tag{1}$$

Adversarial Loss The adversarial loss [10] shows how well the mapping function g can create images $g(x)$ which looks similar to images in the domain Y, while the discriminator D_y aims to tries to distinguish between images from $g(x)$ and real samples from Y. g tries to minimize this objective against D_y, which tries to maximize it, i.e. $min_g max_{D_y} \mathcal{L}_{\text{GAN}}(g, D_Y, X, Y)$. As we use a CycleGAN, this loss is applied twice, once for g and its discriminator D_y, as well as for f and the discriminator D_x.

$$\mathcal{L}_{\text{GAN}}(g, D_Y, X, Y) = \mathbb{E}_{y \sim p_{\text{data}}(y)}[\log D_Y(y)]$$
$$+ \mathbb{E}_{x \sim p_{\text{data}}(x)}[\log(1 - D_Y(g(\alpha_1(x))))]. \tag{2}$$

Cycle Consistency Loss The cycle consistency loss [32] further restricts the freedom of the GAN. Without it, there is no guarantee that a learned mapping function correctly maps an individual x to the desired y. Hence, for each pair $(x_i, z_i) \in x$ the cycleGAN should be able to bring the image x_i back into the original domain X, i.e. $x_i \rightarrow g(x_i) \rightarrow f(g(x_i)) \approx x_i$. As the nature of the cycleGAN is bidirectional the reverse mapping must also be fulfilled, i.e. $y \rightarrow f(y) \rightarrow g(f(y)) \approx y_i$.

$$\mathcal{L}_{\text{cyc}}(g,f) = \mathbb{E}_{x \sim p_{\text{data}}(x)}[\|f(g(\alpha_1(x))) - \alpha_1(x)\|_1]$$
$$+ \mathbb{E}_{y \sim p_{\text{data}}(y)}[\|g(f(y)) - y\|_1]. \tag{3}$$

Reading Loss and Recovered Reading Loss As described in Section 3.1 and shown in Figure 4, we use the reading loss to ensure that the GAN produces readable images, i.e. images containing valid Latin characters. The TRs T and T' are trained with a CTC loss [11,18], which is well suited to tasks that entail challenging sequences alignments.

$$\mathcal{L}_{CTC}(\mathbf{x}, \mathbf{y}) = -\ln p(\mathbf{x}|\mathbf{y}). \tag{4}$$

To calculate the reading loss, the template word text z_i and the corresponding transformed word image $G(x_i)$ is passed to the TRs T and T'. The loss evaluates the mapping g to our target domain Y at a character level.

This discriminator evaluates the readability of the reconstructed image. Hence, its input is a word text from the source domain z_i and the respective reconstruction $f(g(x_i))$. As above, we calculate the CTC-loss on a word level x_i. Since the documents all have a different length, the per word losses for each document are summed up and divided by the length of the document $|x|$.

The two reading loss terms are combined to form the overall reading loss defined as

$$\mathcal{L}_{\text{reading}}(g, f) = \mathbb{E}_{x \sim p_{\text{data}}(x)} \left[\frac{\sum_{v,w \in s(g,x)} \mathcal{L}_{CTC}(\alpha_2(v), w)}{|x|} \right]$$
$$+ \left[\frac{\sum_{v,w \in s(f(g,x))} \mathcal{L}_{CTC}(\alpha_2(v), w)}{|x|} \right] \tag{5}$$

where $s(h, u) = \{(u_i, h(u_i)) | i = 1, ..., |u|\}$ and h represents the transformation function and u all word image and ground truth pair of a document.

Combined Loss The different loss term are weighted with $\lambda_{\text{cyc}} = 10$, $\lambda_{\text{read}} = 1$, and $\lambda_{\text{id}} = 5$ as suggested by Zhu et al. [32] and Touvron et al.[30] and summed up to form the overall loss objective:

$$\mathcal{L}_{\text{total}}(g, f, D_X, D_Y) = \mathcal{L}_{\text{GAN}}(g, D_Y, X, Y) + \mathcal{L}_{\text{GAN}}(f, D_X, Y, X)$$
$$+ \lambda_{cyc} \times \mathcal{L}_{\text{cyc}}(g, f) + \lambda_{id} \times \mathcal{L}_{\text{Identity}}(g, f)$$
$$+ \lambda_{read} \times \mathcal{L}_{\text{reading}}(g, f). \tag{6}$$

The combined loss is used in a min-max fashion, the generator tries to minimize it, and the discriminators aim to maximize it:

$$g^*, f^* = \arg \min_{g,f} \max_{D_x, D_Y} \mathcal{L}_{\text{total}}(g, f, D_X, D_Y). \tag{7}$$

4 Experimental Setup

Model Architecture To achieve the goal of learning a transformation from source domain X to target domain Y using unpaired collections of images, we use an architecture based on the CycleGAN [32] framework. The generators G and F are each 24 layers deep Convolutional Neural Network (CNN) architectures with 11.3 million parameters. The discriminators D_x and D_y are based on the PatchGAN architecture [13], and have 5 layers and 2.7 million parameters each. Our Text Recognizer (TR) networks T and T' are based on the winning HTR model from the ICFHR2018 competition [26]. Both these networks contain 10 convolutional and batch normalization layers followed by 2 bi-directional LSTM layers for a total of 8.3 million parameters. For all architectures, we apply the preprocessing steps (e.g. resizing) as suggested in their respective publications. The data gets min-max normalized.

Task The first step in our two-stage method is to create the source domain dataset images as described in Section 2.1. The structure and content of these documents are specified using LaTeX. In the second step, we use the source domain dataset files along with a collection of unlabeled historical document images (see Section 2.2) to train our CycleGAN and TR networks. In the training process, we learn a mapping function g that transforms source domain documents to the target domain as well as a mapping function f, which works in the other direction. The TR networks are trained simultaneously to recover the user-specified content from $g(x)$ and $f(g(x))$. After completing training, we use the generator G to transform document images from the source domain to the target domain while preserving content and structure.

Pre-processing Due to GPU memory constraints, we use to train our models using image patches of size 256×256. These image patches are randomly cropped from the document images and fed into the CycleGAN architecture. The TR networks T and T' receive individual words cropped (128×32) from $g(x)$ and $f(g(x))$ respectively. Additionally we add Gaussian Noise to $g(x)$ as described in Section 3.1.

Training Procedure We train the CycleGAN and TR components of our system simultaneously. The models are trained for 200 epochs using the Adam optimizer [17] with a learning rate of 2×10^{-4} and a linear decay starting at 100 epochs. The optimizer uses 5×10^{-5} weight decay and 0.5, 0.999 beta values for the generators and discriminators, respectively. We use a batch size of 1 to facilitate the varying amount of words per patch that is fed to T and T'.

Evaluation Procedure We evaluate the quality of the synthetic historical documents produced with our method qualitatively and quantitatively. We first evaluate the synthetic historical documents produced qualitatively with a visual inspection, highlighting the successfully transformed and key limitations of the produced synthetic documents.

We use synthetic historical documents produced with our method in a pre-training setting to provide a quantitative evaluation. We generate 70'000 synthetic words in the historical style of the target domain dataset and use these words to train a new TR network called $\mathcal{R}_{\mathrm{syn}}$. We then fine-tune $\mathcal{R}_{\mathrm{syn}}$ using various subsets (10%, 20%, 50%, and 100%) of the training data from the St. Gall dataset (see Section 2.3) and evaluate its text recognition performance on the test set. As baselines, we compare $\mathcal{R}_{\mathrm{syn}}$ against $\mathcal{R}_{\mathrm{base}}$ and $\mathcal{R}_{\mathrm{IAM}}$. $\mathcal{R}_{\mathrm{base}}$ is randomly initialized and then trained directly on the St. Gall dataset in a similar manner as $\mathcal{R}_{\mathrm{syn}}$. $\mathcal{R}_{\mathrm{IAM}}$ is pre-trained on the IAM Handwriting Database (see Section 2.3) and fine-tuned on the St. Gall dataset.

To determine the best performing pre-trained models of $\mathcal{R}_{\mathrm{syn}}$ and $\mathcal{R}_{\mathrm{IAM}}$, we train both networks until convergence and select the best performing model based on validation score from the hand-labeled subset of HBA (see Figure 3

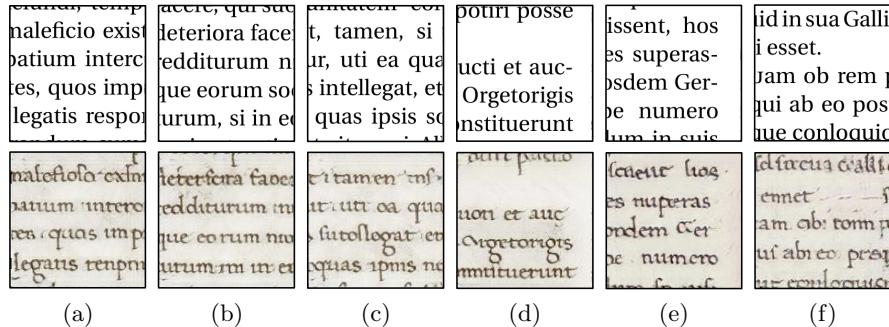| (a) | (b) | (c) | (d) | (e) | (f) |

Fig. 5: Examples of template documents (upper row) and their corresponding synthetic historical images (bottom row). We can see that most characters are accurately transformed to the historical style while remaining readable. Notable exceptions include 'o' → 'a' and 's' → 'n'.

and validation split of the IAM Handwriting Database. The performance of these three models is compared on the test split of the St. Gall dataset using the Character Error Rate (CER) and Word Error Rate (WER) metrics [22].

5   Results

We use two ways to evaluate the results of our generative model: a qualitative visual inspection and a qualitative evaluation. We use a qualitative human-based approach to evaluate the output from a visual perspective and a qualitative approach to measure the influence of our generated data on a downstream text recognition task.

5.1   Visual Analysis

As we can see in Figure 5, the synthetic historical documents generated using our method achieve a high degree of similarity to documents from the target domain (see Section 2.2. The two primary goals of our approach were to preserve structure and content during the transformation of the source domain document into the target domain.

From Figure 1, we can observe that the generator preserves the location of the text from the source domain to the target domain, resulting in the overall structure in the synthetic document matching the input document structure. In most cases, the transformation preserves the number of characters, words, and lines from the source document. However, we observe that on rare occasions, our approach results in synthetic documents where two letters in the source document are combined into a single letter (*legatis* in Figure 5a) or a single letter is expanded into multiple letters (*rem* in Figure 5f). We can also see from Figure 1c
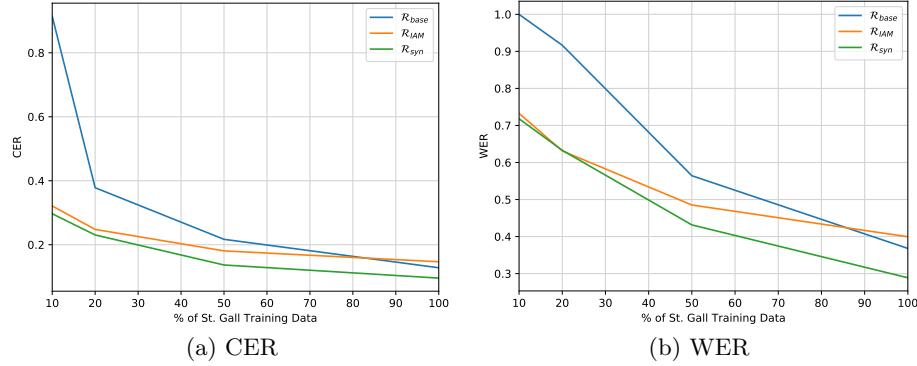
(a) CER

(b) WER

Fig. 6: We can see that the network pre-trained with synthetic data (in green) outperforms the two baselines (orange and blue) in all categories and for both metrics CER and WER.

that our approach is not very effective at transforming the large decorative characters at the beginning of paragraphs. The color of these decorative characters is transformed to the historical style, but they appear slightly distorted. This effect can be viewed as a side-effect of our training procedure, which does not emphasize transforming the decorative elements apart from the general style discrimination provided by D_x and D_y. Additionally, we see artifacts where the patches are stitched together because they are generated individually with 10% overlap and then combined by averaging.

Considering the preservation of textual content, our approach successfully transforms most individual characters to the style of the target domain dataset. Individual words are readable and require some effort to distinguish from real historical image samples – even to expert eyes. However, our approach struggles with the transformation of certain letters. From Figure 5a, we can see that the character 'o' is mistransformed into an 'a'. However, the shape and appearance of these two letters are very similar and often hard to distinguish. Our approach also has problems transforming the letter 's'. This character is sometimes transformed into the character 'n', for e.g., in the word *superas* in Figure 5e, the first 's' is transformed into 'n', however the second 's' is correctly preserved. Despite these small mistakes, we can observe that overall the method produces a very faithful transformation of the source document into the target historical style while preserving content and structure.

5.2   Quantitative Evaluation

In Figure 6 we visualize the empirical results of our experiments where we compare our proposed approach against a purely supervised method and a transfer learning baseline method, with respect to the fraction of labels used in the target

dataset. This way, we can assess the performances of those methods in the conditions of arbitrarily (and here, controlled) small datasets. We recall that small datasets are the common scenario in this domain, as opposed to more mainstream computer vision domains. As expected, with a small amount of data, the pre-trained methods ($\mathcal{R}_{\mathrm{syn}}$ and $\mathcal{R}_{\mathrm{IAM}}$) vastly outperform the baseline ($\mathcal{R}_{\mathrm{base}}$). This margin decreases as we train on large proportions of training data from St. Gall, however, $\mathcal{R}_{\mathrm{syn}}$ consistently achieves the lowest CER (see Figure 6a), and is narrowly beat by $\mathcal{R}_{\mathrm{IAM}}$ only when considering the WER at the 20% subset (see Figure 6b). On average, $\mathcal{R}_{\mathrm{syn}}$ has a 38% lower CER and a 26% lower WER compared to the model trained only on the St. Gall dataset, and a 14% lower CER and 10% lower WER compared to the model pre-trained on the IAM Handwriting Database.

Interestingly, when using the entire training set, $\mathcal{R}_{\mathrm{base}}$ achieves a lower error rate than $\mathcal{R}_{\mathrm{IAM}}$, which could be attributed to stylistic differences between the IAM Handwriting Database and the St. Gall dataset. Similar to observations from Studer et al. [27], the benefits of pre-training on a different domain could decrease when more training data is available from the actual task. Therefore, the stylistic similarity of the synthetic historical images and documents from the St. Gall dataset could explain the lower error rates of $\mathcal{R}_{\mathrm{syn}}$ compared to $\mathcal{R}_{\mathrm{base}}$.

6   Conclusion

We presented a two-step framework for generating synthetic historical images that appear realistic. The two steps are (1) creating electronic user-defined datasets (e.g., with LaTeX) for which the text content is known, and then feed it to step (2) where we use an improved CycleGAN based deep learning model to learn the mapping to a target (real) historical dataset. Differently from previous works in the field, our approach leverages two Text Recognizer (TR) networks to constrain the learning process further to produce images from which the text can still be read. The outcome of the process is a model capable of synthesizing a user-specified template image into historical-looking images. The content is known, i.e., we have the perfect ground truth for all the synthetic data we generate. These synthetic images — which come with a OCR ground truth — can then be used to pre-train models for downstream tasks. We measured the performances of a standard deep learning model using images created with our approach as well as other existing real historical datasets. We show that our approach consistently outperforms the baselines through a robust set of benchmarks, thus becoming a valid alternative as a source dataset for transfer learning. This work extends the already conspicuous work on the field of synthetic document generation. It distinguishes itself for providing the ground truth and high-quality synthetic historical images. Finally, the images generated with our methods are still distinguishable from real genuine ones due to small imperfections. Therefore we envisage that further work would improve upon our open-source implementation.

Acknowledgment

References

1. Alberti, M., Seuret, M., Ingold, R., Liwicki, M.: A Pitfall of Unsupervised Pre-Training (2017), arXiv: 1703.04332
2. Alberti, M., Vögtlin, L., Pondenkandath, V., Seuret, M., Ingold, R., Liwicki, M.: Labeling, cutting, grouping: An efficient text line segmentation method for medieval manuscripts. In: 2019 International Conference on Document Analysis and Recognition (ICDAR). pp. 1200–1206. IEEE
3. Baird, H.S.: Document Image Defect Models. In: Baird, H.S., Bunke, H., Yamamoto, K. (eds.) Structured Document Image Analysis, pp. 546–556. Springer (1992)
4. Bluche, T., Louradour, J., Messina, R.: Scan, Attend and Read: End-to-End Handwritten Paragraph Recognition with MDLSTM Attention. In: 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR). vol. 01, pp. 1050–1055
5. Capobianco, S., Marinai, S.: DocEmul: A Toolkit to Generate Structured Historical Documents. In: 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR). vol. 01, pp. 1186–1191
6. Chu, C., Zhmoginov, A., Sandler, M.: CycleGAN, a master of steganography (2017)
7. Clausner, C., Pletschacher, S., Antonacopoulos, A.: Aletheia - An Advanced Document Layout and Text Ground-Truthing System for Production Environments. In: 2011 International Conference on Document Analysis and Recognition. pp. 48–52
8. Edwards, H.J.: Caesar: The Gallic War. Harvard University Press Cambridge (1917)
9. Fischer, A., Frinken, V., Fornés, A., Bunke, H.: Transcription alignment of Latin manuscripts using hidden Markov models. In: Proceedings of the 2011 Workshop on Historical Document Imaging and Processing. pp. 29–36. HIP '11, Association for Computing Machinery (2011)
10. Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative Adversarial Networks (2014)
11. Graves, A., Fernández, S., Gomez, F., Schmidhuber, J.: Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks. In: Proceedings of the 23rd International Conference on Machine Learning. pp. 369–376. ICML '06, Association for Computing Machinery (2006)
12. Guan, M., Ding, H., Chen, K., Huo, Q.: Improving Handwritten OCR with Augmented Text Line Images Synthesized from Online Handwriting Samples by Style-Conditioned GAN. In: 2020 17th International Conference on Frontiers in Handwriting Recognition (ICFHR). pp. 151–156 (2020)
13. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)

14. Journet, N., Visani, M., Mansencal, B., Van-Cuong, K., Billy, A.: DocCreator: A New Software for Creating Synthetic Ground-Truthed Document Images **3**(4), 62
15. Kang, L., Riba, P., Wang, Y., Rusiñol, M., Fornés, A., Villegas, M.: GANwriting: Content-Conditioned Generation of Styled Handwritten Word Images. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.M. (eds.) Computer Vision – ECCV 2020. pp. 273–289. Lecture Notes in Computer Science, Springer International Publishing
16. Kieu, V.C., Visani, M., Journet, N., Domenger, J.P., Mullot, R.: A character degradation model for grayscale ancient document images. In: Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012). pp. 685–688 (2012)
17. Kingma, D.P., Ba, J.: Adam: A Method for Stochastic Optimization (2017)
18. Li, H., Wang, W.: Reinterpreting CTC training as iterative fitting **105**, 107392 (2020)
19. Marti, U.V., Bunke, H.: The IAM-database: An English sentence database for offline handwriting recognition **5**(1), 39–46 (2002)
20. Masci, J., Meier, U., Cireşan, D., Schmidhuber, J.: Stacked Convolutional Auto-Encoders for Hierarchical Feature Extraction. In: Honkela, T., Duch, W., Girolami, M., Kaski, S. (eds.) Artificial Neural Networks and Machine Learning – ICANN 2011. pp. 52–59. Lecture Notes in Computer Science, Springer (2011)
21. Mehri, M., Héroux, P., Mullot, R., Moreux, J.P., Coüasnon, B., Barrett, B.: HBA 1.0: A pixel-based annotated dataset for historical book analysis. In: Proceedings of the 4th International Workshop on Historical Document Imaging and Processing. pp. 107–112. HIP2017, Association for Computing Machinery (2017)
22. Märgner, V., Abed, H.E.: Tools and metrics for document analysis systems evaluation. In: Doermann, D., Tombre, K. (eds.) Handbook of Document Image Processing and Recognition, pp. 1011–1036. Springer London (2014)
23. Pondenkandath, V., Alberti, M., Diatta, M., Ingold, R., Liwicki, M.: Historical Document Synthesis with Generative Adversarial Networks. In: 2019 International Conference on Document Analysis and Recognition Workshops (ICDARW). vol. 5, pp. 146–151 (2019)
24. Scius-Bertrand, A., Voegtlin, L., Alberti, M., Fischer, A., Bui, M.: Layout analysis and text column segmentation for historical vietnamese steles. In: Proceedings of the 5th International Workshop on Historical Document Imaging and Processing. pp. 84–89. HIP '19, Association for Computing Machinery (2019)
25. Seuret, M., Chen, K., Eichenbergery, N., Liwicki, M., Ingold, R.: Gradient-domain degradations for improving historical documents images layout analysis. In: 2015 13th International Conference on Document Analysis and Recognition (ICDAR). pp. 1006–1010 (2015)
26. Strauß, T., Leifert, G., Labahn, R., Hodel, T., Mühlberger, G.: ICFHR2018 Competition on Automated Text Recognition on a READ Dataset. In: 2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR). pp. 477–482
27. Studer, L., Alberti, M., Pondenkandath, V., Goktepe, P., Kolonko, T., Fischer, A., Liwicki, M., Ingold, R.: A Comprehensive Study of ImageNet Pre-Training for Historical Document Image Analysis. In: 2019 International Conference on Document Analysis and Recognition (ICDAR). pp. 720–725 (2019)
28. Taigman, Y., Polyak, A., Wolf, L.: Unsupervised Cross-Domain Image Generation (2016)
29. Tensmeyer, C., Brodie, M., Saunders, D., Martinez, T.: Generating Realistic Binarization Data with Generative Adversarial Networks. In: 2019 International Conference on Document Analysis and Recognition (ICDAR). pp. 172–177 (2019)
30. Touvron, H., Douze, M., Cord, M., Jégou, H.: Powers of layers for image-to-image translation (2020), arXiv:2008.05763

31. Zhang, K.A., Cuesta-Infante, A., Xu, L., Veeramachaneni, K.: SteganoGAN: High Capacity Image Steganography with GANs (2019), arXiv:1901.03892
32. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV) (2017)