Self-supervised learning for medieval handwriting identification: A case study from the Vatican Apostolic Library

Lorenzo Lastilla, Serena Ammirati, Donatella Firmani, Nikos Komodakis, Paolo Merialdo, Simone Scardapane

This is the accepted manuscript of the article published by Elsevier in Information Processing & Management on 9 February 2022, doi: 10.1016/j.ipm.2022.102875, available online at: https://doi.org/10.1016/j.ipm.2022.102875.

Self-supervised learning for medieval handwriting identification: a case study from the Vatican Apostolic Library

Lorenzo Lastilla[a,*], Serena Ammirati[b], Donatella Firmani[c], Nikos Komodakis[d], Paolo Merialdo[e], Simone Scardapane[f]

[a]*Department of Computer, Control and Management Engineering Antonio Ruberti (DIAG), Sapienza University of Rome, Rome, Italy*
[b]*Department of Humanities, Roma Tre University, Rome, Italy*
[c]*Department of Statistical Sciences, Sapienza University of Rome, Rome, Italy*
[d]*Computer Science Department, University of Crete, Crete, Greece*
[e]*Department of Engineering, Roma Tre University, Rome, Italy*
[f]*Department of Information Engineering, Electronics and Telecommunications (DIET), Sapienza University of Rome, Rome, Italy*

Abstract

In this paper, we consider the task of automatically identifying whether different parts of medieval and modern manuscripts can be traced back to the same copyist/scribe, a problem of significant interest in paleography. Currently, the application of deep learning techniques in the context of scribe recognition has been hindered by the lack of a sufficiently large, labeled dataset, since the labeling process is incredibly complex and time-consuming. Here, we propose the first successful application of the recent framework of self-supervised learning to the field of digital paleography, wherein we pretrain a convolutional neural network by leveraging large amounts of unlabeled manuscripts. To this end, we build a novel dataset consisting of both labeled and unlabeled manuscripts for copyist identification extracted from the Vatican Apostolic Library. We show that fine-tuning this model to the task of interest significantly outperforms other baselines, including the common setup of initializing the network

*Corresponding author.
Email addresses: lorenzo.lastilla@uniroma1.it (Lorenzo Lastilla), serena.ammirati@uniroma3.it (Serena Ammirati), donatella.firmani@uniroma1.it (Donatella Firmani), komod@csd.uoc.gr (Nikos Komodakis), paolo.merialdo@uniroma3.it (Paolo Merialdo), simone.scardapane@uniroma1.it (Simone Scardapane)

from general-domain features, or training the model from scratch, also in terms of generalization power. Overall, our results reveal the strong potential of self-supervised techniques in the field of digital paleography, where unlabeled data (i.e., digitized manuscripts) is nowadays available, while labeled data is scarcer.
Keywords: Self-Supervised Learning, Manuscripts, Handwriting Identification

1. Introduction

The automatic analysis of medieval and modern manuscripts based on deep learning techniques is attracting more and more attention in recent years, as it promises to significantly simplify and improve the work of paleographers and other domain experts [1, 2]. Among the many tasks that can be addressed in such a way, the subdivision of the texts into parts belonging to distinct scribes, on the basis of the respective handwriting style, stands out for interest and importance in the overall interpretation of such documents. This operation, also referred to as handwriting identification, relies on the possibility to create a one-to-one correspondence handwriting style-author, which is confirmed by the fact that handwriting – a behavioral and distinctive biometric characteristic of each human being [3, 4] – essentially depends on class and individual factors (respectively, the products of prescribed writing systems and the idiosyncrasies of the individual) [5]. Hence, we can reliably establish a relationship between a handwriting style and an individual when the same distinctive personal writing characteristics are found in multiple writings in sufficient number that the likelihood of accidental coincidence is eliminated [5, 6]. In other words, we can cluster the inscriptions of different scribes, or assess the unique authorship of a group of handwritten texts, based on a set of discriminating elements or features, that is "discrete elements of writing or lettering that vary observably or measurably with its author" [5], which can be mainly gathered into elements of style (text arrangement, dimension, slant, spacing etc.) and execution (abbreviations, alignment, embellishments etc.) [5].

At present, the handwriting identification task for medieval and modern

manuscripts continues to be carried out with traditional methods (and a significant amount of time, costs, and expertise required) by paleographers, which precisely search for such discriminating features across the document to identify the scribes who participated in its realization. Notwithstanding, this operation – which essentially consists of an instance discrimination task – naturally lends itself to the application of computational methods, as proven by the flourishing number of deep learning approaches [7, 8, 9] – based on learnable features instead of the previously mentioned "discriminating elements" – as well as standard pattern recognition methods [10, 11] for handwriting identification proposed in recent years. However, the application of data-driven strategies has found a major obstacle in the lack of structured data, due to the costs, time and expertise required for data labeling.

The present work is placed exactly in the context of the application of deep learning methods to handwriting identification on medieval and modern manuscripts (more precisely, a set of 24 digitized medieval manuscripts selected from the Vatican Apostolic Library [12]). Specifically, the goal of this contribution is to demonstrate for the first time the benefits of using a self-supervised learning approach for the task of handwriting identification on such kind of data. The vastness of the collections of manuscripts examined (and consequently the amount of data available), in fact, is considerable, but most of them are not labeled, i.e., they are devoid of annotations at the level of the single page (or of the single paragraph) about the *copyist*, i.e., the person who physically wrote it. This is indeed the perfect condition for applying self-supervised learning methods, because they learn effective visual representations from a large amount of unlabeled data [13]. These features or representations can then be transferred to the vision task of actual interest (in this case, handwriting identification), which can be performed based on just a few labeled samples [14, 15, 16].

In practice, the proposed methodology consists of two main stages. At first, all the pages contained in the manuscripts – previously preprocessed – are involved in a pretext task (i.e., an ancillary task solved only for the purpose of learning good data representations [16]), according to the reconstruction-based

self-supervised approach described in [15, 17]. This approach is particularly targeted to contextual reasoning, and essentially consists of a teacher-student scheme to train a backbone Convolutional Neural Network (CNN) to reconstruct a Bag of Visual Words (BoW) representation of an image, given as input a perturbed version of that same image. At a later stage, the task of actual interest is faced. In this case, the data we rely on consists of the manuscript pages annotated with the respective copyists only; the available copyists are then split into a background set and an evaluation set (whose samples are never seen during training nor validation). According to the linear evaluation protocol commonly used to assess the learned representations [13, 18, 19, 20, 21], a linear layer is trained on top of the frozen base encoder, with the aim of minimizing a triplet margin loss [22, 23]. Such kind of loss was chosen because it allows us to perform end-to-end learning between the input (which consists of a perturbed version of the original page, according to a given set of transformations) and the desired embedding space [24], to build a distance function able to generalize to never seen classes, and to produce well-separated clusters.

1.1. Contributions

We can sum up the main results and contributions of this work as follows:

1. an original dataset was produced starting from 24 sufficiently homogeneous manuscripts of the Vatican Apostolic Library; it is worth recalling that the examined documents are particularly complex, due to the tendency of copyists to standardize the handwriting style as much as possible while working on the same manuscript;
2. this is the first study that demonstrates the effectiveness of a self-supervised approach for this task, from a performance point of view (evaluated through the Mean Average Precision): indeed, the visual representations learned in a self-supervised fashion outperform the ImageNet [25] ones, as well as the features learned after training the backbone model from scratch (that is, initializing the encoder with random weights);

3. the proposed methodology is able to generalize to scribes not included in the training stage of the downstream task.

The remainder of this paper is divided into the following Sections: Section 2 reviews the most relevant works in the literature, as far as both the application of deep learning methodologies to paleography and handwriting identification and self-supervised learning are concerned; Section 3 presents the dataset, together with the data preprocessing stage; Section 4 covers the proposed methodology and the selected data augmentations; Section 5 includes a presentation of the main aspects and details of the experiments, and a discussion on the obtained results; finally, Section 6 sums up the most relevant outcomes, plus making some conclusive remarks and giving an outlook on future works.

2. Related works

This Section aims at contextualizing the proposed approach for handwriting identification by introducing some relevant works available in the literature: first, the application of deep learning methodologies to paleography and handwriting identification will be treated; then, the most popular self-supervised learning frameworks will be discussed, mainly focusing on the contrastive ones.

2.1. Deep learning applications to paleography and handwriting identification

Digital paleography, intended as the application of computer-based methods for paleography, has highly benefited from the recent development of machine learning and deep learning approaches, as proven by the wide range of case studies and applications of these methodologies. This relatively recent research field covers several areas, going from the development of database systems for manuscript research and editing [26, 27] and solutions for querying large sets of handwritten document images [28, 29], up to the automatic identification of inter-script, inter-scribal, intra-script and intra-scribal variations as well as cultural and textual relevant features [30]. Just to mention the long-established

topic of optical character recognition (OCR) for ancient manuscripts, the authors of [31] realized a comprehensive survey on the application stages of this technique for different writing systems, such as Devanagari [32], Persian [33], and Bengali [34], and, most importantly, for ancient text documents, including Hanja [35], Gurmukhi [36], and Devanagari [37] writing systems. The task of one-shot handwritten character recognition, instead, in which the prediction is made given only a single example of each new character, was accomplished through the successful application of Siamese Convolutional Neural Networks (CNNs) by [38], for the Omniglot dataset [39], and by [40], for the CASIA HWDB1.1 dataset of Chinese handwritten characters [41]. Siamese CNNs have been exploited to tackle the fragment retrieval task too: the authors of [42] propose a fragment matching approach based on 2D Siamese Networks to reassemble pottery pieces (*ostraca*) covered with textual inscriptions; in [43], a self-supervised deep metric learning solution for the association of ancient papyri fragments is proposed, without human intervention for annotating images. The same Siamese Networks strategy was also successfully applied to other paleographic tasks, such as manuscript alignment [44] (which aims at determining the similarities and differences between two versions of a given manuscript), offline [45] and online [46] signature verification.

Coming to the most relevant contributions on the long-standing issue of automatic handwriting identification, which is the task of interest for this work, many studies have been carried out in recent years, aimed both at the conceptualization and theoretical modeling of the problem, and at the application of new methodologies. From a theoretical perspective, in [47] a conceptual model for the description and retrieval of handwriting features in Western medieval script is presented; moreover, the authors of [48] and [49] describe the problem as relatively agnostic with respect to the writing system [48], and essentially overlapping with script classification [49]. On the other hand, if we focus on the solutions proposed over the years for the handwriting identification task, we can list many different case studies and approaches, not always based on machine learning. The authors of [50] propose a handwriting matching tool

based on sparse representation to join fragments of the same scribe, and a paleographic classification tool that matches a given document to a large set of paleographic samples. In [51], a dataset of handwriting on papyri for the task of writer identification is proposed; moreover, a preliminary evaluation is carried out, based on a Normalised local Naïve Bayes nearest-neighbour classifier [52] as a learning- and segmentation-free method for writer identification. The authors of [11], instead, continuing the work of [53], demonstrate through standard pattern recognition methods and statistical tools that two main scribes, each showing different writing patterns, were responsible for the Great Isaiah Scroll. The authors deliberately avoid the extensive use of parameter-dense methods for the classification stage, due to the data scarcity and limited explainability of transfer learning approaches [11], while using deep learning at the level of image processing for manuscript binarization [54]. Furthermore, in [55] two algorithms for writer identification – also based on pattern recognition techniques – were tested on the Arad corpus (consisting of 18 different texts inscribed over 16 Hebrew *ostraca* and dated to ca. 600 BCE), and systematically compared to an independent forensic examination. The authors of [56] applied a three-step solution for line detection, line classification, and page writer identification on the Avila Bible (a medieval manuscript of the XII century). This approach, which is based on deep neural networks trained with transfer learning, proved successful even with a relatively small training dataset. Finally, also for the task of handwriting identification the framework of Siamese Networks provided high accuracy: in [7], an automatic system based on Siamese Networks for dividing a manuscript into similar parts, according to their similarity in writing style, is presented; in [8], the same approach is used to build a writer independent deep learning model, which is trained on several writing styles, and able to achieve high detection accuracy when tested on writing styles not present in training data. The methodology assessment was carried out through cross-validation, based on a set of seven manuscripts (five used as training ones, two as testing ones). In conclusion, the authors of [9] presented a deep learning model, called Papy-S-Net for Papyrus-Siamese-Network, for papyri fragment matching

(that is, an expert uses a fragment as a request element and gets fragments that belong to the same papyrus).

2.2. Self-supervised learning

In recent years, deep learning methods have achieved excellent performances in several computer vision tasks under supervised settings, that is for benchmark datasets or application domains where it is possible (in terms of resources and competence required) to easily gather huge corpora of manually annotated data. However, this is not always the case for many research fields: when dealing with medieval and modern manuscripts, for example, the expertise required for image and text annotation is highly expensive and the process is very demanding and time-consuming. Because of this, one of the key challenges for the computer vision community is to find suitable strategies for learning good image representations from a few labeled examples while making best use of many unlabeled instances [57, 58], which would minimize the dependence on potentially costly corpora of manually annotated data [20].

One possible solution is provided by generative methods to representation learning [59, 60], which typically operate in pixel space and try to build a distribution over data and latent embedding, then using the learned embeddings as image representations [58]. However, pixel-level generation is computationally expensive and may not be necessary for representation learning [13].

Another strategy, which is gradually taking hold, consists of the so-called self-supervised representation learning, which precisely aims to pretrain a deep learning model to extract useful and effective representations of the input data without relying on human annotations [19, 15, 61, 62]; such representations or features can then be transferred to other vision tasks of actual interest (the "downstream tasks", such as image classification or object detection, which often have only a few labeled instances [14]) by fine-tuning the pretrained model [15, 16]. To do so, self-supervised learning methods generally try to solve a pretext task (that is, a task which is not of genuine interest) which creates different types of supervision signals from unlabeled instances based on careful inspection of

underlying regularities in the data [14, 16]. Many solutions have been proposed in this sense, showing that it is possible to learn self-supervised representations that are competitive with supervised ones [62], at least for standard datasets – such as CIFAR10, CIFAR100, STL10 [63], ImageNet [25], Places205 [64], and VOC07 [65] – and for many learning problems – such as few-shot [66, 67] and semi-supervised [68, 69] learning, or training generative adversarial networks [70].

One of the most widespread self-supervised learning approaches, which builds upon the instance discrimination task, considers each image of the dataset (or "instance") and its transformations (resulting from a given set of perturbations or "data augmentations") as a separate class [61], and aims to learn low-dimensional image embeddings that are invariant under the selected perturbations while being discriminative among different classes [61, 62, 17]. Most of the discriminative methods are implemented in a contrastive learning framework: the similarity of representations obtained from different transformations of the same image (positive pairs) is maximized, while spreading representations of views from different images (negative pairs) apart [58, 62, 14, 71]. The similarity of sample pairs is measured by a distance function or contrastive loss [72], which directly operates in the representation space [61]. As previously anticipated, the input is not provided to the network with a specific target; for contrastive losses, the target can vary during training and can be defined in terms of the data representation computed by the network [16, 72]. Since computing all the pairwise comparisons on a large dataset is not practical, most implementations approximate the loss by reducing the number of comparisons to random subsets of images during training [61]. Different solutions were proposed under the contrastive self-supervised learning framework: in [19], context features are constructed as a summary of past input segments, and then contrasted with local features from a future time step [68, 73]. The approach proposed in [74], instead, learns a representation that maximizes the mutual information (MI) among various views of the same scene, with the aim of maximizing the good information while minimizing the noise. In a similar way, [20] introduces a

methodology based on maximizing MI between features extracted – also across multiple scales – from independently-augmented versions of each input. The authors of [57, 13] claim that contrastive learning of visual representations benefits from the composition of multiple data augmentation operations, the introduction of a learnable nonlinear transformation between the representation and the contrastive loss, the application of embedding normalization and temperature parameter before the contrastive cross-entropy loss, large batch sizes, deep and wide networks, and long training. Finally, [16] (later improved in [75] through the expedients contained in [13]) proposes to view contrastive learning as a form of dictionary lookup (maintaining the dictionary as a queue of data samples, to decouple the dictionary size from the batch size) and relies on an online network and a momentum-updated offline network to maintain consistency. Although contrastive methods manage to achieve impressive results, they focus less on other important aspects in representation learning, such as contextual reasoning [17]. Moreover, they often require comparing each example with many other examples to work well, prompting the question of whether using negative pairs is necessary [58].

Among the other self-supervised learning solutions available in the literature, we can mention approaches based on clustering [61, 76, 77, 78] and redundancy-reduction – where the objective function tries to make the cross-correlation matrix computed from twin representations as close to the identity matrix as possible [62]; another line of research was also indicated by [58], whose algorithm iteratively bootstraps the outputs of a momentum-updated offline network to serve as targets for the prediction of an online network. Finally, some self-supervised methods rely on auxiliary handcrafted tasks to learn their representation [79, 80, 18, 81, 82, 83, 84, 85, 86]. In this work, we focused on the reconstruction-based methodology described in [15, 17], which belongs to the latter group of approaches, and is more targeted to contextual reasoning. In particular, this solution relies on a teacher-student scheme to train a CNN to reconstruct a Bag of Visual Words (BoW) representation of an image, given as input a perturbed version of that same image. More details on this methodology

will be provided, however, in Section 4.

3. Case study

The validation of the proposed methodology went through the definition of a relevant dataset with respect to the task of interest. Indeed, we selected 24 manuscripts among the tables for Latin paleography exercises published by the Vatican Apostolic Library in 2004 [87], which collect very recognizable graphic types [88, 89]; in particular, we focused on the documents available in digital format and at high resolution on the Vatican Apostolic Library website [12]. 21 manuscripts out of the 24 selected can be gathered into 3 macro-groups, according to the categorization identified by [87]:

- 7 manuscripts characterized by a variety of regional styles (graphic particularism), namely Vatt. latt. 3313, 5951, 9882 (IX century), 3317 (X century), 4958, 12910 (XI century), and 4939 (XII century);
- 11 manuscripts characterized by a more uniform style (Carolingian minuscule and Gothic minuscule), namely Vatt. latt. 43, 3868, 4965, 5775 (IX century), 378, 579, 653, 8487 (XI century), 42, 620, and 3833 (XII century);
- 3 later manuscripts written using the Gothic minuscule, namely Vatt. latt. 907, 2669 (XIII century), and 588 (XIV century).

In addition to the previous list, 3 manuscripts from the Atlantic Bibles [90] were included – Vatt. latt. 4217 (XI century), 4220, 4221 (XVI century) – whose scribes are clearly identified. Although not particularly extensive, the corpus has both elements of variety (very different scripts) and homogeneity (there are clusters with increasing difficulty, such as the Gothic script one and the Atlantic Bibles one) that allow us to carry out different levels of analysis. In the following Table 1, the number of pages per manuscript is recalled, after the removal of the unnecessary pages (for example, unwritten pages, pages containing too many

drawings, or pictures of the manuscript cover). The final corpus amounts to 8745 pages. It is worth highlighting that the manuscripts are quite heterogeneous in terms of handwriting style and state of conservation, which is convenient for representation learning in terms of generalization power. A critical issue of the corpus consists of the class imbalance, instead, both at the level of samples per manuscript and samples per copyist.

3.1. Data Preprocessing

To prevent the unwritten borders of the pages from being included in the training stage, a coarse cropping region for each manuscript was identified, given that the lines of text are placed approximately in the same position across the manuscript (so, more precisely, two cropping regions were identified per manuscript, one for the left pages – *verso* – and one for the right pages – *recto*).

After this first cropping, each page x_i^m was further cropped to match the size (w_{min}^m, h_{min}^m) of the smallest image in the manuscript m, obtaining the result $\tilde{x}_i^m$ visible in Figure 1. This second cropping was done to prevent different resizing distortions for pages belonging to the same manuscript[1]. The final sizes for the pages of each document are recalled in Table 1.

3.2. Datasets

Starting from the overall group of selected pages, two different datasets were created and involved in the first and the second phase of the workflow (that is, the pretext task and the downstream one) respectively.

3.2.1. Pretext task dataset

As to the first stage of the process, indeed, the 8745 samples – organized in 24 classes corresponding to the available manuscripts – were randomly split into training, validation and test sets according to the ratio 0.8-0.15-0.05 (equal to 6986, 1302, and 457 pages respectively).

[1]Due to copyright issues on the original images, the preprocessed dataset is only available upon request.

Table 1: Number of useful pages, number of copyists, and final image size per manuscript.

Manuscript ID	**Number of pages**	**Number of copyists**	**Final image size**
Vat. lat. 12910	69	0	510×370
Vat. lat. 2669	132	0	1303×894
Vat. lat. 3313	701	0	909×609
Vat. lat. 3317	178	0	1061×787
Vat. lat. 378	256	6	909×565
Vat. lat. 3833	288	0	806×627
Vat. lat. 3868	45	0	1098×975
Vat. lat. 42	228	0	1441×932
Vat. lat. 4217	873	3	1917×1219
Vat. lat. 4220	411	8^*	2054×1302
Vat. lat. 4221	367	8^*	2020×1270
Vat. lat. 43	406	0	731×400
Vat. lat. 4939	369	0	905×468
Vat. lat. 4958	189	0	1008×672
Vat. lat. 4965	304	2	1152×809
Vat. lat. 5775	308	0	1170×854
Vat. lat. 579	296	0	1666×1006
Vat. lat. 588	261	0	654×453
Vat. lat. 5951	310	3	1043×682
Vat. lat. 620	240	0	998×647
Vat. lat. 653	538	4	1279×902
Vat. lat. 8487	1000	3	1260×835
Vat. lat. 907	380	2	920×626
Vat. lat. 9882	596	0	754×386

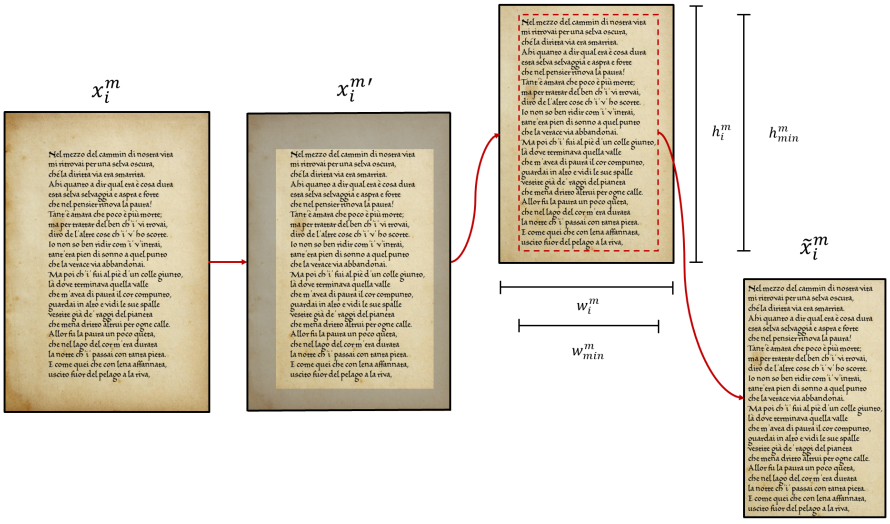


Figure 1: Preprocessing to remove unwritten borders.

3.2.2. Downstream task dataset

For the second and last phase of the workflow, only the annotated pages were selected from the overall corpus. As indicated in Table 1, 31 scribes were identified in 9 manuscripts (Vat. lat. 4220 and Vat. lat. 4221 share the same set of 8 copyists), even though 4 of them were not included in the experiments, because only few lines (3 copyists from Vat. lat. 378) or corrupted pages (1 copyist from Vat. lat. 8487) were attributed to them. Therefore, the final dataset consists of 27 copyists and 3730 annotated pages (each one attributed to a single scribe).

Having defined a subset of the initial corpus for the downstream task, the available copyists were then split into an *evaluation set* (consisting of the 4 scribes from Vat. lat. 653, completely excluded from the training and validation stages of this task) and a *background set* (including the 23 remaining scribes). The number of pages per scribe is recalled in Tables 2 and 3. The choice fell on this partition both to preserve a sufficiently high number of scribes and a good level of variability for the training stage, and to guarantee a meaningful subset for the evaluation stage (thus containing a suitable number of copyists from the

Table 2: Number of pages per copyist (background set).

Copyist	**ID**	**Number of pages**
Vat. lat. 378 – 1	0	38
Vat. lat. 378 – 2	1	33
Vat. lat. 378 – 3	2	177
Vat. lat. 907 – 1	3	356
Vat. lat. 907 – 2	4	24
Vat. lat. 4217 – 1	5	97
Vat. lat. 4217 – 2	6	80
Vat. lat. 4217 – 3	7	160
Vatt. latt. 4220-4221 – 1	8	466
Vat. lat. 4220 – 2	9	44
Vat. lat. 4221 – 3	10	24
Vat. lat. 4221 – 4	11	75
Vat. lat. 4221 – 5	12	12
Vat. lat. 4221 – 6	13	84
Vat. lat. 4221 – 7	14	11
Vat. lat. 4221 – 8	15	61
Vat. lat. 4965 – 1	16	66
Vat. lat. 4965 – 2	17	94
Vat. lat. 5951 – 1	18	100
Vat. lat. 5951 – 2	19	96
Vat. lat. 5951 – 3	20	114
Vat. lat. 8487 – 1	21	854
Vat. lat. 8487 – 2	22	142
Total		**3208**

Table 3: Number of pages per copyist (evaluation set).

Copyist	ID	Number of pages
Vat. lat. 653 – 1	0	42
Vat. lat. 653 – 2	1	58
Vat. lat. 653 – 3	2	202
Vat. lat. 653 – 4	3	220
Total		**522**

same manuscript). Finally, the background set was further randomly split into training and validation sets – for hyperparameter optimization – according to the ratio 0.8-0.2.

4. Proposed methodology

In this Section, the two main stages of the proposed methodology are presented, alongside the different sets of perturbations or data augmentations applied to the images. The core idea of our work can be summarized as follows: as far as the pretext task (the ancillary and preliminary task useful to learn good data representations from unlabeled samples [16]) is concerned, a CNN-based feature extractor (or encoder) is optimized to generate a representation of an instance, which should depend as much as possible on the invariant properties of the instance with respect to a set of random perturbations. The above framework is common to any self-supervised learning approach. The specificity of the self-supervised learning strategy adopted in this work consists in training (with unlabeled data only) the feature extractor to predict the Bag of Visual Words representation of an image given as input a perturbed version of that image [17]. It is worth pointing out that this framework is highly convenient in terms of generalization power, since it relies on a huge corpus of unlabeled data (which could be increased indefinitely) and remains invariant to non-semantically important perturbations of the data. Once self-supervised pretraining is completed, the

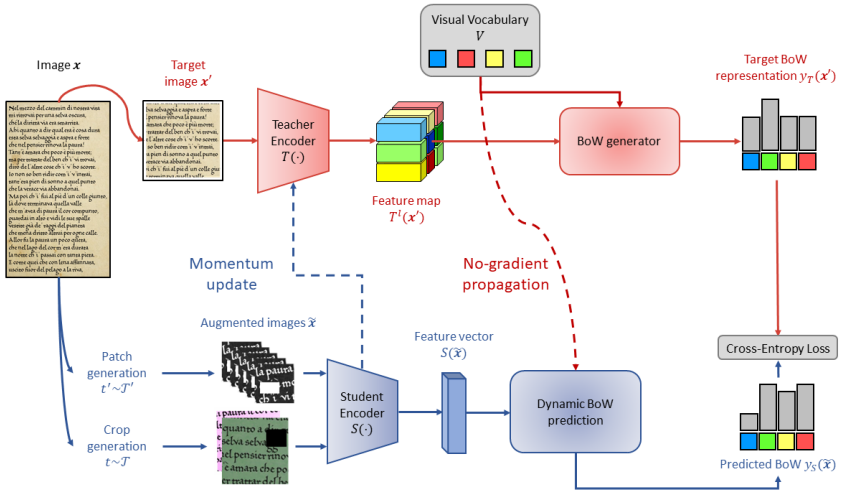


Figure 2: Online Bag of Visual Words (OBoW) reconstruction task.

encoder is involved in the downstream task, that is the real task of interest, which corresponds, in our case, to handwriting identification, and is based on the minimization of a triplet margin loss.

4.1. Pretext task

The BoW reconstruction task, shown in Figure 2, follows the essential lines described in [17]. In particular, a student encoder CNN $S(\cdot)$ – parameterized by θ_S – learns image representations based on the BoW targets generated by a teacher encoder $T(\cdot)$ – parameterized by θ_T (which are an exponential moving average of θ_S [16], being updated at each training iteration according to $\theta_T \leftarrow \alpha \cdot \theta_T + (1 - \alpha) \cdot \theta_S$, where α is a momentum coefficient set to 0.99 in our experiments). Since the two networks share the same architecture, for both of them a ResNet18-based model [91] was chosen.

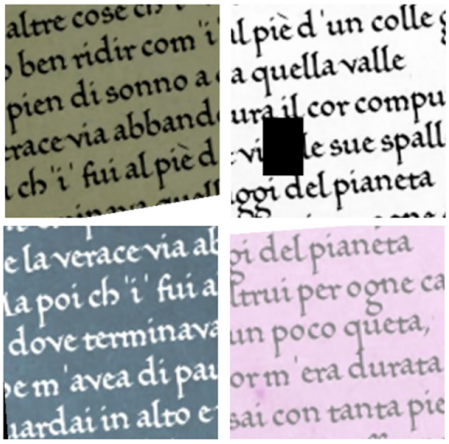
4.1.1. Teacher encoder and vocabulary update

Differently from [17], in this case the teacher receives as input $\mathbf{x}'$ a relatively small portion of the whole sample $\mathbf{x}$, that is a 380×380 random crop of the image (normalized through the mean and standard deviation of the training

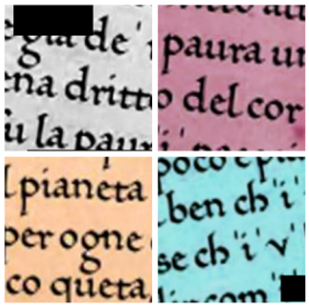
set), and extracts a feature map $T^l(\mathbf{x}') \in \mathbb{R}^{c_l \times h_l \times w_l}$, of spatial size $h_l \times w_l$ with c_l channels, from its l^{th} layer. The $h_l \times w_l$ c_l-dimensional feature vectors are then quantized over a continuously evolving vocabulary $V = [\mathbf{v}_1, ..., \mathbf{v}_K]$ of K visual words of dimension c_l, obtaining a $h_l \times w_l \times K$ quantized feature map (in this work, $K = 8192$). The quantization process, indeed, assigns each feature vector to the K words based on the respective squared Euclidean distances and through soft-assignment codes, computed via Sinkhorn optimization [92], which is a suitable strategy for facing the continuous update of the vocabulary of visual words [17]. The soft-assignment operation depends on a temperature parameter δ_{base}, which was set to $\frac{1}{15}$ in our experiments. Then, the quantized feature map is reduced to a K-dimensional BoW $\tilde{y}_T(\mathbf{x}')$ by channel-wise max-pooling, and, ultimately, converted into a probability distribution over the visual words $y_T(\mathbf{x}')$ by L^1-normalization. The sequence of feature map quantization, channel-wise max-pooling and L^1-normalization is summarized by the *BoW generator* block in Figure 2. As far as the evolution of the vocabulary of visual words V is concerned, the dictionary is treated as a K-sized queue of random features. In particular, V is updated at each training step by replacing the $B < K$ oldest items in the queue with B feature vectors, each of which selected from an image of the B-sized current mini-batch. As to the feature vector sampling strategy, the "local averaging" approach was chosen: given a feature map $T^l(\mathbf{x}')$, first it is locally averaged with a 3×3 kernel, then a feature vector is sampled from it with uniform distribution.

4.1.2. Student encoder and data augmentation schemes

As to the student network, it receives as input a set of perturbed versions $\tilde{\mathbf{x}}$ of the image $\mathbf{x}$, and it is trained to reconstruct the BoW representation $y_T(\mathbf{x}')$ produced by the teacher. Following [17], the data augmentation scheme was specifically designed to produce instances with small (and even no) regions in common with the target image $\mathbf{x}'$, thus forcing the student network to focus on high-level statistics to reconstruct $y_T(\mathbf{x}')$ and to understand and learn spatial dependency between visual parts. Because of this, two kinds of crops were ex-



(a) Examples of crops extracted through $t \sim \mathcal{T}$.



(b) Examples of patches extracted through $t' \sim \mathcal{T}'$.

Figure 3: Examples of input data provided to the student network $S(\cdot)$.

tracted from the page $\mathbf{x}$ through the data augmentation operators $t \sim \mathcal{T}$ and $t' \sim \mathcal{T}'$ respectively, where $\mathcal{T}$ and $\mathcal{T}'$ are two different augmentation families. In particular, the first operator extracts two 270×270 random crops from $\mathbf{x}$ (which cover, at most, $\sim 50\%$ of $\mathbf{x}'$ [17]), and then applies the following set of perturbations: radiometric perturbations (such as color jittering, random grayscale conversion, and random inversion), Gaussian blur, random erasing, and mild geometric distortions (such as random affine and random perspective transformations), to preserve the most important geometric features of the handwriting style. The second operator t', instead, applies the same set of perturbations as t to a 256×256 region of $\mathbf{x}$ (but with slightly different parameters), and then extracts from the obtained image five partially overlapping 150×150 patches (each patch covering, at most, $\sim 16\%$ of $\mathbf{x}'$ [17]). All the obtained crops – which can be referred to as $\tilde{\mathbf{x}} = t(\mathbf{x}) \cup t'(\mathbf{x})$ – are then normalized according to the same statistics as the target image $\mathbf{x}'$. In Figure 3, some examples of crops obtained through the two augmentation schemes are shown (without normalization).

After the crop and patch generation stage, the student encoder extracts C global vector representations $S(t(\mathbf{x}))[c] \in \mathbb{R}^{512}, c = 1, ..., C$ from the C crops, and P global vector representations $S(t'(\mathbf{x}))[p] \in \mathbb{R}^{512}, p = 1, ..., P$ from the P

patches. Then, a linear-plus-softmax layer is applied to $S(\tilde{\mathbf{x}}) = S(t(\mathbf{x})) \cup S(t'(\mathbf{x}))$ to obtain the $(C+P)$ K-dimensional vectors $y_S(\tilde{\mathbf{x}}) = y_S(t(\mathbf{x})) \cup y_S(t'(\mathbf{x}))$, which are the predicted softmax probabilities of the target $y_T(\mathbf{x}')$. Hence, the training loss that is minimized for a single image $\mathbf{x}$ is obtained by composing the cross-entropy losses between the softmax distributions $y_S(\tilde{\mathbf{x}})$ predicted by the student from $\tilde{\mathbf{x}}$ and the BoW distribution $y_T(\mathbf{x}')$ according to Equation 1:

$$\mathrm{CE}(y_S(\tilde{\mathbf{x}}), y_T(\mathbf{x}')) = -\frac{1}{C}\sum_{c=1}^{C}\sum_{k=1}^{K} y_T(\mathbf{x}')[k] \log(y_S(t(\mathbf{x}))[c][k]) - \frac{1}{P}\sum_{p=1}^{P}\sum_{k=1}^{K} y_T(\mathbf{x}')[k] \log(y_S(t'(\mathbf{x}))[p][k]) \quad (1)$$

It is worth recalling that the weights of the linear-plus-softmax layer applied to $S(\tilde{\mathbf{x}})$ are not fixed, but they are updated following the vocabulary of visual words V: a generation network $G(\cdot)$, indeed, takes as input V at each training step and produces the prediction weights. This dynamic form of BoW prediction depends on a parameter κ, which equally scales the magnitudes of all the predicted weights and was fixed to 8 in our experiments.

4.1.3. Multi-level feature extraction

With the aim of forcing the student encoder to learn richer and more powerful representations, the previously exposed methodology includes multi-scale BoW reconstruction targets, as explicitly suggested by [17]. Hence, two feature maps $T^l(\mathbf{x}'), l = \{L-1, L\}$ are extracted by the teacher encoder, with $L-1$ corresponding to the penultimate layer of ResNet (`conv4`) and L to the last layer (`conv5`). As a consequence, a separate vocabulary of size $K = 8192$ is used for each layer, as well as two different weight generation networks. As to the loss $\mathrm{CE}(y_S(\tilde{\mathbf{x}}), y_T(\mathbf{x}'))$, each of the two addends is obtained by averaging the two corresponding terms computed for layers $L-1$ and L.

4.2. Downstream task

As previously anticipated, the downstream task (represented in Figure 4) consists in handwriting identification by minimizing a triplet margin loss [23], which can be seen as learning a distance function useful for discriminating instances belonging to different classes in the embedding space, and able to generalize to never seen copyists and to produce well-separated clusters [24]. For this purpose, the frozen features of the pretrained student encoder $\hat{S}(\cdot)$ (based on a ResNet18 architecture, with an adaptive average pooling layer at the end of the last convolutional block) are used; in other words, the backbone model weights are not updated at this stage. In details, $\hat{S}(\cdot)$ receives as input a batch of B samples and, for each sample $\mathbf{x}'$ (which is a perturbed version of the image $\mathbf{x}$, according to the augmentation scheme discussed below), it extracts a global vector representation $\hat{S}(\mathbf{x}') \in \mathbb{R}^{512}$. Thereafter, a linear layer $L(\cdot)$ – added to the pretrained backbone encoder $\hat{S}(\cdot)$ – is trained to extract more powerful representations or embeddings $L(\hat{S}(\mathbf{x}')) \in \mathbb{R}^k$ of $\mathbf{x}'$ with respect to the task of interest (k, that is the embedding width, is a hyperparameter of the problem). The B embeddings obtained are normalized and finally combined into triplets. A triplet consists of an anchor $\mathbf{a}$, a positive $\mathbf{p}$, and a negative $\mathbf{n}$ sample, where the anchor belongs to the same class as the positive, and the negative to a different one [93, 94]. For some distance function $d(\cdot)$ on the embedding space, we can define the triplet margin loss [22, 94] of the triplet $(\mathbf{a}, \mathbf{p}, \mathbf{n})$ as:

$$TL(\mathbf{a}, \mathbf{p}, \mathbf{n}) = \max(d(\mathbf{a}, \mathbf{p}) - d(\mathbf{a}, \mathbf{n}) + m, 0) \tag{2}$$

where m is a predefined margin (which is another hyperparameter of the optimization problem). The triplet margin loss minimization is then equivalent to making $d(\mathbf{a}, \mathbf{p})$ smaller than $d(\mathbf{a}, \mathbf{n})$ by a predefined margin m, or, alternatively, to pushing similar instances closer while dividing dissimilar ones as much as possible.

According to the implementation of the triplet margin loss exploited in [93], for each batch of B embeddings, the loss is computed as an average over just a subset of all the possible triplets (which amount to B^3), created through a

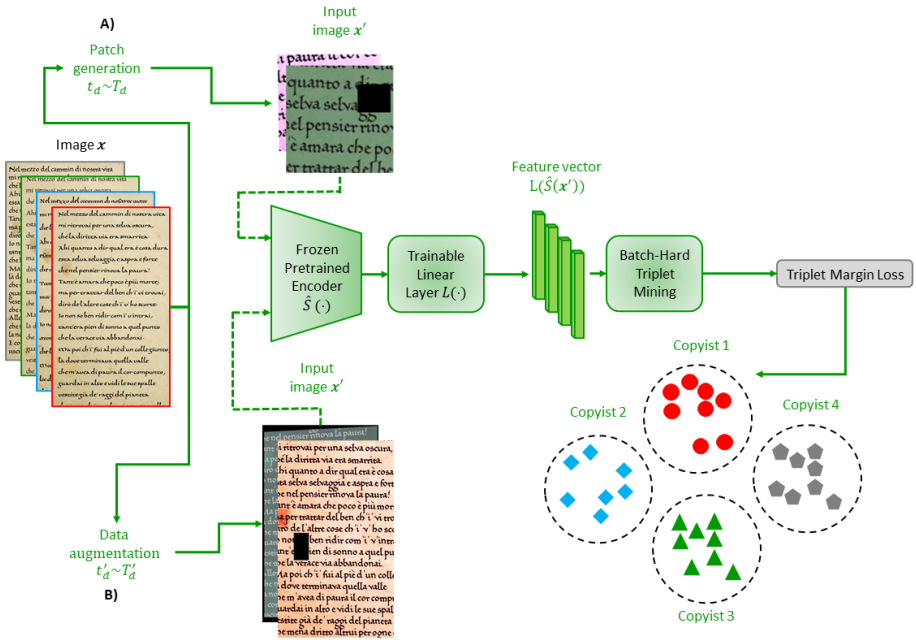


Figure 4: Handwriting identification downstream task, carried out both in mode A (with random crops extracted from the pages) and B (fully convolutional neural network applied to the whole pages).

mining strategy (i.e., the process of finding the best triplets to train on [94]). Indeed, using all the possible combinations can consume a lot of memory, and, theoretically, has the tendency to include a large number of less informative triplets, causing performance to plateau quickly [94]. To overcome this issue, we opted for the online batch-hard triplet mining solution, represented through the corresponding block in Figure 4. According to this approach, the valid triplets only (i.e., triplets where the first and the second element fall into the same class, while the third one belongs to a different class) are first considered. Then, for each anchor, the hardest positive and the hardest negative sample (which provide, respectively, the highest distance $d(\mathbf{a}, \mathbf{p})$ and the lowest distance $d(\mathbf{a}, \mathbf{n})$ within the batch) are selected. This way, we rely essentially on moderate triplets for the loss computation, since they are the hardest within a small subset of the dataset, obtaining the optimal configuration for this kind of task [24, 93]. As to the pairwise distance function $d(\cdot)$ involved in the triplet margin loss computation, the L^2 norm was chosen.

4.2.1. Data augmentation schemes – downstream task

In conclusion, the downstream task was carried out based on two mutually exclusive data augmentation schemes – indicated as **A)** and **B)** in Figure 4. The first scheme involves a transformation $t_d \sim \mathcal{T}_d$, which first extracts a random 380×380 crop from the page $\mathbf{x}$, and then applies the same set of perturbations as $t \sim \mathcal{T}$ to the crop. Finally, the resulting images are normalized based on the statistics computed for the downstream task training set, obtaining $\mathbf{x}'$. Scheme **B)**, instead, incorporates from [95] the concept of "fully convolutional" networks that take input of arbitrary size, and applies through the operator $t'_d \sim \mathcal{T}'_d$ the same set of perturbations (plus normalization) as $t \sim \mathcal{T}$ (but with different parameters) to the whole page $\mathbf{x}$. Because pages coming from different manuscripts have different sizes, however, the data augmentation chain is preceded by a central crop which equalizes the height and width of all images to the minimum values of page height and width across the dataset. Notwithstanding, the k-dimensional embedding $L(\hat{S}(\mathbf{x}'))$ is representative of a much wider portion

of the original page **x** than scheme **A)**.

5. Experiments and results

The following Section deals, firstly, with the choice of the optimization hyperparameters, both for the pretext and for the downstream task, and with the formal definition of a suitable performance metric. Secondly, the results obtained in terms of handwriting identification – through the proposed methodology and the baselines, respectively – will be presented, with the aim of demonstrating the benefits of self-supervised pretraining for the task of interest. Then, further aspects related to the pretext task and the tests in general will be discussed.

5.1. Experimental setup

All the experiments reported here were carried out using a Tesla V100 SXM2 32GB GPU, and involved a ResNet18-based architecture[2].

As to the BoW reconstruction task, in addition to the parameters mentioned in Subsection 4.1, the following choices were made in terms of optimization (after the execution of preliminary tests): the student encoder was trained for 100 epochs; the batch size (which affects the queue-based vocabulary update) was fixed to 64; finally, we made use of Stochastic Gradient Descent (SGD) with learning rate set to 0.03, and progressively adjusted up to the final value of 0.00003 through a cosine scheduler with an initial warmup of 5 epochs. Overall, the pretraining stage lasted approximately 28 hours. At the end of each training epoch an online monitoring of the teacher features and an online validation were also performed, both conceived as a linear classification over the 24 classes, and quantified through the top-1 accuracy metric. Moreover, both the operations involved the samples contained in the pretext task validation set (in this case, the model just receives the 380×380 crops with no perturbations, but normalized according to the training set mean and standard deviation).

[2] At `https://github.com/L9L4/HI-SSL` the code used for the experiments is available.

As to the handwriting identification task, it was faced based on both the augmentation schemes **A)** and **B)**, considering 3 different configurations and thus ending up with 6 tests in total:

1. a linear layer was trained on top of the frozen backbone model pretrained with OBoW;
2. a linear layer was trained on top of the ImageNet frozen features (also in this case, a ResNet18 backbone encoder – but pretrained on the ImageNet dataset – was used);
3. a model initialized with random weights (but characterized by the same architecture as the other two cases) was fully trained from scratch directly on the downstream task using the same setup as Subsection 4.2, without self-supervised pretraining.

For all the 6 tests, the following experimental setup was adopted: the output dimension of the linear layer (embedding width) was fixed to 1024, while the margin m of the triplet margin loss was set to 0.2, based on the value suggested in [23]. The model was trained for 100 epochs with SGD optimization: the learning rate, starting from 0.15, was increased up to 0.6 through a linear warmup for the first 10 epochs, and then decayed with a cosine annealing up to 0.0015. As to the batch size, it was set to 256 for the tests carried out under the data augmentation scheme **A)**, and to 32 for scheme **B)**. These choices correspond to the maximum batch size values allowed for the two configurations and for the single GPU specified above: ensuring a high value of this parameter is indeed of great importance to generate many combinations and thus to identify significant triplets to learn from. The whole set of tests lasted approximately 6 days and 19 hours.

5.2. Performance evaluation

To quantitatively assess the performance of the models obtained, the Mean Average Precision (MAP) was used. This metric, indeed, is characterized by especially good discrimination and stability [96]. Specifically, for each test:

- the model corresponding to the minimum validation loss during training was chosen;

- this model was used to extract the embeddings starting from the samples contained in the evaluation set – more precisely, for case **B)** the model receives the normalized pages as input, while for case **A)**, for each sample, it receives ten normalized 380×380 random crops extracted from the image (similarly to one of the augmentation protocols for evaluating the performance at test time described in [97]), and produces ten embeddings that are averaged to obtain a single vector, better suited to represent the whole page and to minimize the variability of the single crops;

- for the i^{th} embedding from class j ($j = 1, ..., J$), the K-Average Precision $P^{(i,j)}@K = \frac{1}{n_j - 1} \sum_{k=1}^{K-1} P(R_{ik})$ was computed, where K is the number of samples in the evaluation set, n_j is the number of instances belonging to class j, and $P(R_{ik}) = \frac{\sum_{p=1}^{k} \mathbb{1}(R_{ik}^{(p)}=j)}{k}$ is the precision associated to R_{ik}, with R_{ik} the set of ranked retrieval embeddings from the closest up to the k^{th} closest to the i^{th} embedding from class j [96], and $\mathbb{1}(R_{ik}^{(p)} = j)$ the indicator function computed for the p^{th} element of R_{ik};

- the average of the K-Average Precision values was calculated, both by class – $MAP_j = \frac{1}{n_j} \sum_{i=1}^{n_j} P^{(i,j)}@K$ – and on the entire evaluation set – $MAP = \frac{1}{K} \sum_{j=1}^{J} n_j \cdot MAP_j$.

5.3. Handwriting identification

In Table 4, the results obtained for each test in terms of Mean Average Precision are shown[3]. In particular, it is immediately evident that the self-supervised learning based approach is far more effective for the task of interest than making use of the representations learned from ImageNet, or training from scratch

[3] As to the background scribes, the same accuracy assessment as the evaluation ones – described in Subsection 5.2 – was carried out; this time, however, the model corresponding to the minimum training loss was chosen, and the metric was computed based on half of the training set images, to speed up the computation.

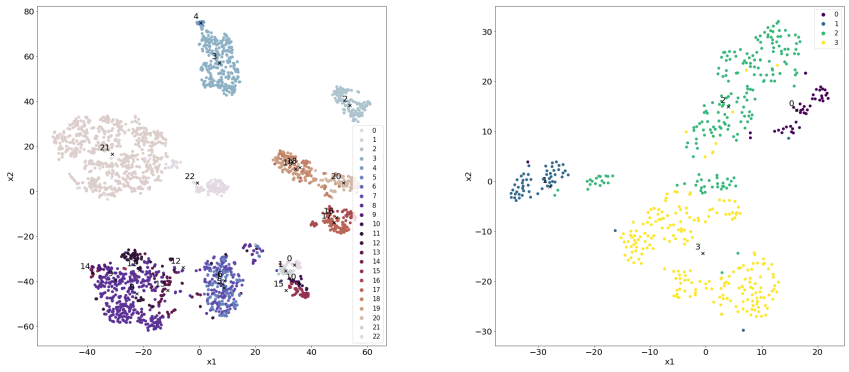
Table 4: Performance obtained for the handwriting identification task for the 6 tests, with respect to the Mean Average Precision metric.

Test ID	**Pretraining**	**Mode**	**Scheme**	**MAP Background Set**	**MAP Evaluation Set**
0	OBoW	Transfer learning	**B)**	**74.8***	72.0
1	ImageNet	Transfer learning	**B)**	69.7	64.9
2	-	Training from scratch	**B)**	60.8	58.8
3	OBoW	Transfer learning	**A)**	71.7	**79.0***
4	ImageNet	Transfer learning	**A)**	63.7	67.5
5	-	Training from scratch	**A)**	48.5	59.1

a new model initialized with random weights, under both the **A)** and **B)** data augmentation schemes. This is true, indeed, both for the background scribes and for the copyists used in the evaluation phase, to test the generalization capacity of the model, achieving, respectively, a MAP of 74.8% for the background set – obtained under the **B)** scheme – and of 79.0% for the evaluation set – **A)** scheme[4]. It is worth noticing how, under the **A)** scheme, the obtained models seem to perform worse on the background set than on the evaluation one. Moreover, despite under-performing on the same set of 23 scribes with respect to scheme **B)**, they seem to show a better generalization power (indicated by the higher values of MAP for the evaluation set). This fact can be attributed to the great variability of new samples produced with the data augmentation of type **A)**, which, for each image, returns new 380×380 random crops from epoch to epoch, while, on the contrary, the other configuration allows the model to just focus on perturbed versions of the same pages, which however keep constant the invariant properties of the original instances.

In Figures 5, 6, 7, 8, 9, and 10, it is possible to visualize the 2D projection of the embeddings of both the background and the evaluation set (together

[4]The best model with respect to the validation loss for test 3 – producing the highest MAP for the evaluation set – is available at https://github.com/L9L4/HI-SSL/blob/main/model/checkpoints_3/Test_3_TL_val_best_model.pth.



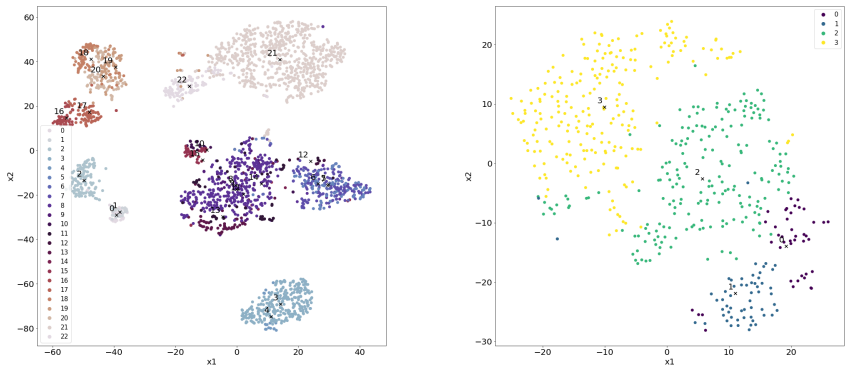
(a) Background set.

(b) Evaluation set.

Figure 5: 2D t-SNE visualization of the embedding distribution for test 0 – OBoW pretraining, augmentation scheme **B)**.

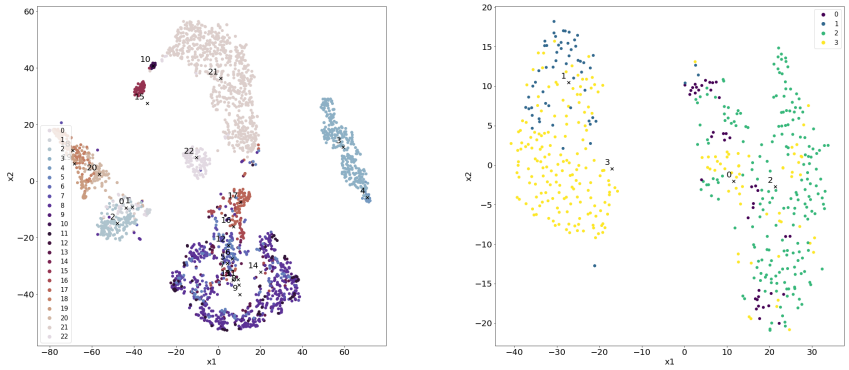
with the respective cluster centroids), after dimensionality reduction through the t-SNE technique [98]. In details, Figures 5 and 8 refer to the embeddings produced after OBoW pretraining, under the augmentations schemes **B)** and **A)** respectively; Figures 6 – scheme **B)** – and 9 – scheme **A)** – refer to the embeddings produced after ImageNet pretraining; finally, Figures 7 and 10 refer to the embeddings produced after training the model from scratch – based, also in this case, respectively on **B)** and **A)** schemes. The embedding 2D projections further confirm the superiority – for the task of interest – of an approach based on self-supervised learning over the baselines considered (transfer learning from ImageNet features and training from scratch), since the clusters obtained are more easily distinguishable.

More specifically, if we focus on the background set (Figures 5a, 6a, 7a, 8a, 9a, and 10a), we can notice some differences from manuscript to manuscript (apart from test 5, which performs clearly worse than any other test, as visible in Figure 10a). Particularly, it is worth highlighting that, for Vat. lat. 378 – scribes 0-2 – and Vat. lat. 907 – scribes 3-4, the proposed methodology (Figures 5a and 8a) is able to produce better clusters than the corresponding



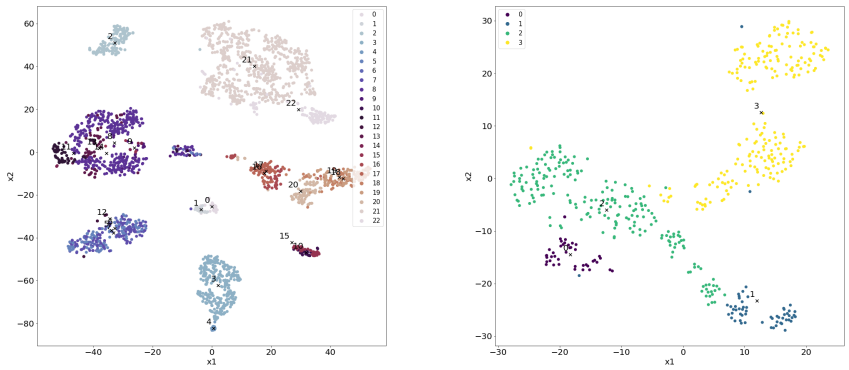
(a) Background set. (b) Evaluation set.

Figure 6: 2D t-SNE visualization of the embedding distribution for test 1 – ImageNet pretraining, augmentation scheme **B)**.



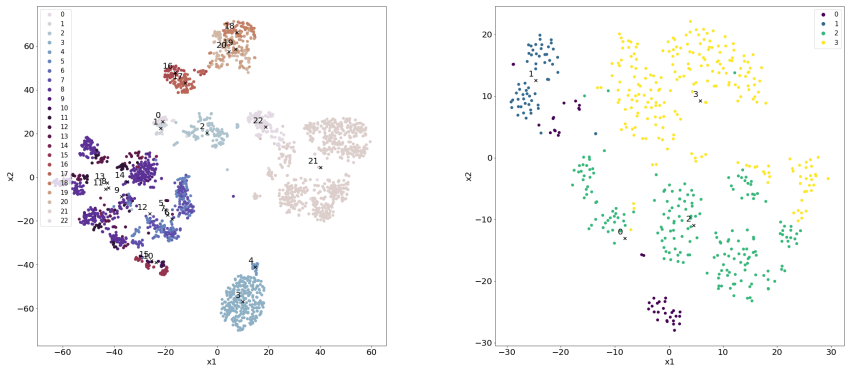
(a) Background set. (b) Evaluation set.

Figure 7: 2D t-SNE visualization of the embedding distribution for test 2 – training from scratch, augmentation scheme **B)**.



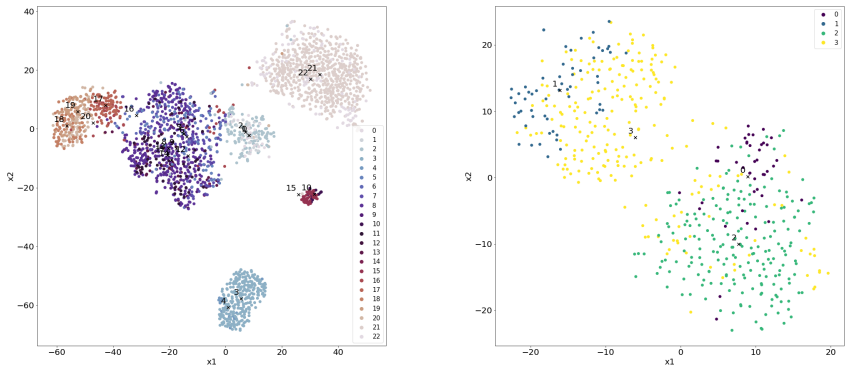
(a) Background set. (b) Evaluation set.

Figure 8: 2D t-SNE visualization of the embedding distribution for test 3 – OBoW pretraining, augmentation scheme **A)**.



(a) Background set. (b) Evaluation set.

Figure 9: 2D t-SNE visualization of the embedding distribution for test 4 – ImageNet pretraining, augmentation scheme **A)**.



(a) Background set.

(b) Evaluation set.

Figure 10: 2D t-SNE visualization of the embedding distribution for test 5 – training from scratch, augmentation scheme **A)**.

baselines; for Vat. lat. 4965 (scribes 16-17), Vat. lat. 5951 (18-20), and Vat. lat. 8487 (21-22), instead, the performance between OBoW (Figures 5a and 8a) and ImageNet pretraining (Figures 6a and 9a) is comparable, but generally higher than training from scratch (Figures 7a and 10a). Finally, the subset of manuscripts including Vat. lat. 4217, Vat. lat. 4220, and Vat. lat. 4221 proved difficult for all the approaches (even if, also in this case, the MAP computed for tests 0 and 3 is higher than the corresponding baselines); these particularly complex cases will be discussed in Subsection 5.5.

If we consider, instead, the embedding projections for the evaluation set obtained through scheme **B)** – Figures 5b, 6b, and 7b, we can say that self-supervised pretrainig positively affects the generation of well-separated clusters with respect to both test 1 and (to a greater extent) 2; indeed, for test 2 (Figure 7b), it is possible to see a small overlap between scribes 1 and 3 (which are clearly separated in the other two cases), and also between scribe 2, part of the samples from scribe 3, and scribe 0 (which is practically indistinguishable from the other two). The situation is almost identical for scheme **A)** – Figures 8b, 9b, and 10b: the only small difference emerges when comparing tests 0 and 3 (Figures 5b and

8b), both resulting from OBoW pretraining, since scheme **A)** ensures a clearer separation between scribes 2 and 3.

5.4. OBoW pretraining

For the OBoW reconstruction task, as anticipated at the beginning of Subsection 5.1, an online monitoring of the teacher features and an online validation were carried out during training, based on the top-1 accuracy metric. The evolution of the top-1 accuracy is represented in Figure 11: the maximum top-1 accuracy for the teacher network (97.0%) was achieved at the 94^{th} epoch, while the maximum validation top-1 accuracy (96.9%) at the 88^{th} epoch. In Figures 12a and 12b, some examples of visual word members from the vocabularies involved in the multi-level feature extraction are shown. In particular, each shown line is associated to a visual word, and contains the 8 image patches (retrieved from the pretext task dataset) with the highest assignment score for that word (based on the state of the vocabulary at the end of training).

5.5. Discussion

The main advantage of the proposed methodology consists of extracting high-level visual representations from a large quantity of raw manuscripts to ultimately obtain a system capable of performing handwriting identification more effectively, even for manuscripts excluded from training for this specific task. The importance of such a methodology is linked to the consideration that, at the present time, one of the critical aspects in deep learning research is the definition of strategies that minimize the costs of annotating data, and therefore, essentially, the amount of data needed to train models on. With respect to the issue of scarcity of annotated data, the validity of the workflow discussed in this contribution is highlighted by considering the difference in terms of Mean Average Precision from possible alternative solutions: as to the **A)** configuration, OBoW pretraining outperforms the concurrent baselines (provided by tests 1 and 2) by $\sim 5\%$ and 14% on the background set respectively, and by $\sim 7\%$ and $\sim 13\%$ on the evaluation set; as to the **B)** configuration, instead, it outperforms

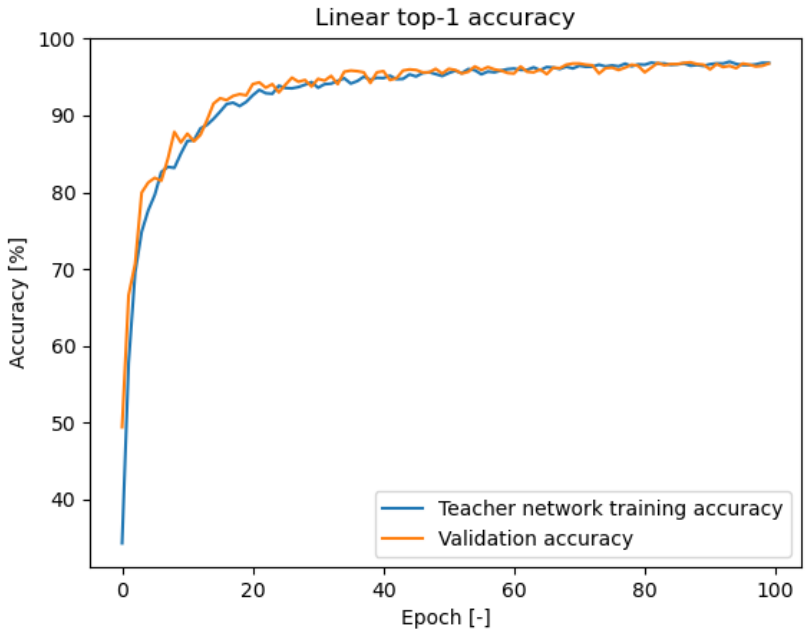
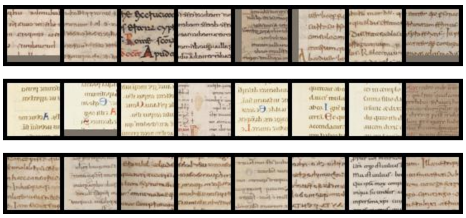


Figure 11: Evolution of the training and validation top-1 accuracy for the OBoW reconstruction task.



(a) Visual word members from the vocabulary associated with ResNet18 layer $L-1$ (`conv4`).

(b) Visual word members from the vocabulary associated with ResNet18 layer L (`conv5`).

Figure 12: Examples of visual word members from the vocabularies involved in the multi-level feature extraction.

the baselines (provided by tests 4 and 5) by 8% and 23% on the background set, and by $\sim$ 11% and $\sim$ 20% on the evaluation set. These results are particularly encouraging in extending the proposed method to any non-annotated manuscript which, once involved in the pretext task, can then be partitioned through zero-shot learning.

At present, however, an explainability analysis of the methodology has not been executed, so it cannot be established with certainty whether the features extracted from the image crops have any connection with those used by paleographers for handwriting identification. Consequently, this approach could not constitute a tool of immediate use, but rather an input to direct the scholar's work. In conclusion, a critical aspect that is worth considering in the analysis of the results is certainly the class imbalance. If we consider Figures 13 and 14, which compare the relative frequency of the copyists and the MAP per class for test 3, however, we fail to observe a clear correlation between the class frequency and the value of the metric. Rather, we observe lower values of MAP

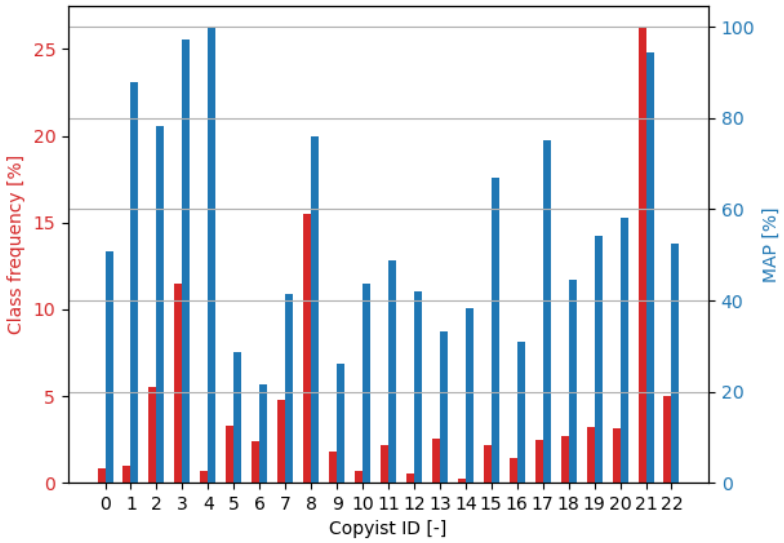


Figure 13: Class frequency distribution compared to the Mean Average Precision computed for the copyists of the background set (test 3).

for some scribes (5-7, 9-14) coming from very specific manuscripts (Vat. lat. 4217, Vat. lat. 4220, and Vat. lat. 4221). This could be explained by the fact that these 3 manuscripts constitute a particularly complex subset, since the copyists who realized them aimed for the maximum handwriting uniformity: thus, the representations extracted from the model, generally sufficient for the other manuscripts, might have been unsuitable to grasp the set of discriminating elements valid for this subgroup.

6. Conclusion

In this paper, we focused on the automatic handwriting identification task for medieval manuscripts, that is the problem of partitioning a manuscript among the copyists who realized it, in the face of the scarcity of large and annotated datasets due to the incredible complexity and amount of time of the labeling process, one of the main factors which hindered the application of deep learn-

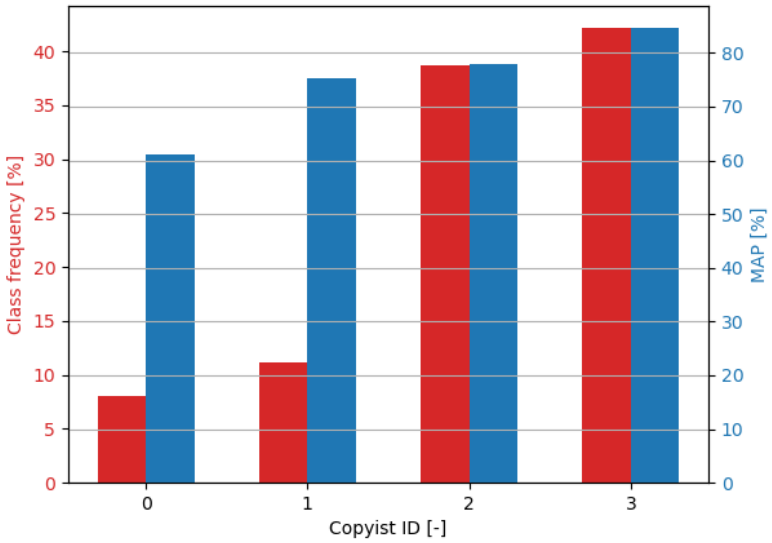


Figure 14: Class frequency distribution compared to the Mean Average Precision computed for the copyists of the evaluation set (test 3).

ing techniques to this domain. Specifically, we provided, to the best of our knowledge, the first empirical validation of the self-supervised framework in the medieval and modern manuscript domain, assessing its capability to learn effective visual representations from a large amount of raw data (that is, the large number of unlabeled digitized manuscripts involved in this study) and then to build a solid starting point for the task of interest, which can be performed based on just a few (and even zero) labeled samples, and with higher precision. The proposed approach was compared with two common setups, namely the network initialization with general-domain (ImageNet) features, and training the full model from scratch, and it turned out to significantly outperform both the baselines, also from a generalization power point of view (which is encouraging in extending the method to other unlabeled manuscripts). Among the main results of this work, we can also mention the creation of an original dataset consisting of 24 sufficiently homogeneous manuscripts from the Vatican

Apostolic Library, including 31 different copyists. Lastly, even if the benefits of self-supervised learning for digital paleography were assessed for the specific task of handwriting identification only, this work naturally lends itself to be extended to many other paleographic problems, such as automatic transcription or text detection: the outlook in this regard is very promising, as suggested by the success of self-supervised learning in a wide range of computer vision tasks [99].

Regarding possible future developments, an explainability analysis of the methodology could be carried out, to try and understand whether the features used to automatically distinguish different copyists have any connection with those used by paleographers for the handwriting identification task. At present, indeed, it is not possible to highlight which parts of a page contribute most to the assignment of that page to a given scribe: the solution of this "black box" effect of our methodology is a key step in making it actually useful and usable for scholars. Nonetheless, we must recall that this aspect might depend on the fact that our self-supervised learning strategy is actually quite generic (i.e., task-agnostic), which is, however, a big advantage from a computational/practical point of view. Another potential analysis in terms of explainability includes testing the Visual Probing approach suggested by [100], where a simple classifier verifies if the visual representations encode a particular property/concept, even though this property was not a direct training objective. Moreover, the subset of scribes whose identification was most difficult will be analyzed in depth, in order to determine the necessary adjustments to the model to extract useful features even in complex cases like this, or to test the inclusion of linguistic features (such as abbreviations) into the pipeline, alongside the visual ones. In addition to these drawbacks (that is, the "black box" effect and the insufficient representations obtainable for complex sets of scribes), it is worth highlighting at least another limitation of the proposed methodology: currently, it is unable to tackle the problem of multigraphism, that is it cannot identify a single scribe writing entirely different scripts [101]. Despite the complexity of this sub-task, it is of considerable interest in the context of digital paleography, given the

frequent attestations of multigraphism in ancient documents [101].

As far as future research opportunities are concerned, a wider experimental setup will be investigated, by considering different metric learning losses and data augmentation schemes, as well as testing new sets of hyperparameters and architectures – as an alternative to standard Computer Vision models we can mention, for example, Vision Transformers (ViT, [102]) or MLP-Mixer [103], which however require to significantly increase the dataset size with respect to the case study considered in this work. Another research topic which is worth mentioning for future works is the so-called "model compression", which consists of the efficient compression and execution of deep neural networks, without significantly compromising accuracy [104], allowing to transfer heavy models to apps on smartphones but also to perform efficient on-device learning [105], thus making these tools even easier to use for paleographers. Several solutions could be tested in this sense, such as tensor decomposition, data quantization, network sparsification [104, 106], and even knowledge distillation [107], where a small student model is trained to mimic (and thus to absorb the knowledge) of a heavy (trained) model. In this regard, it is worth noticing that our self-supervised approach already employs a pair of student-teacher networks, which are used, however, for a different purpose. Finally, it is known that powerful scaling laws exist for self-supervised learning models. In this sense, leveraging such techniques at larger scale (by creating wider corpora of unlabeled data) could provide the starting point for a domain-specific "foundation model" (in the sense of [108]), similar to BERT [109] and GPT [110] models in NLP research, freely exploitable by any researcher interested in solving a digital paleography task with scarce labeled data.

Funding

This work was partially supported by the Data Science Ph.D. course and the 2020 Joint Mobility Call fellowships granted to Lorenzo Lastilla by Sapienza University of Rome, Italy, and by Regione Lazio POR FESR 2014-2020, Project

"Matrices" (36664), Italy, granted to Paolo Merialdo.

Acknowledgements

The authors wish to thank the Physics Department of the Sapienza University of Rome, and particularly the local branch of the National Institute for Nuclear Physics (INFN), for the possibility of carrying out the experiments on the INFN NVIDIA DGX-1 server.

CRediT authorship contribution statement

Lorenzo Lastilla: Data curation, Formal analysis, Investigation, Methodology, Software, Validation, Visualization, Writing – original draft.

Serena Ammirati: Conceptualization, Data curation, Resources.

Donatella Firmani: Conceptualization, Formal analysis, Methodology, Supervision.

Nikos Komodakis: Conceptualization, Formal analysis, Methodology, Supervision, Writing – review & editing.

Paolo Merialdo: Conceptualization, Formal analysis, Funding acquisition, Methodology, Project administration, Resources, Supervision, Writing – review & editing.

Simone Scardapane: Conceptualization, Formal analysis, Investigation, Methodology, Project administration, Supervision, Writing – original draft.

References

[1] E. Nieddu, D. Firmani, P. Merialdo, M. Maiorino, In Codice Ratio: A crowd-enabled solution for low resource machine transcription of the Vatican Registers, Information Processing & Management 58 (5) (2021) 102606. doi:https://doi.org/10.1016/j.ipm.2021.102606.
URL https://www.sciencedirect.com/science/article/pii/S0306457321001035

[2] P. Hu, W. Wang, Q. Li, T. Wang, Touching text line segmentation combined local baseline and connected component for Uchen Tibetan historical documents, Information Processing & Management 58 (6) (2021) 102689. doi:https://doi.org/10.1016/j.ipm.2021.102689. URL https://www.sciencedirect.com/science/article/pii/S0306457321001746

[3] S. Srihari, C. Huang, H. Srinivasan, On the discriminability of the handwriting of twins, Journal of Forensic Sciences 53 (2) (2008) 430–446.

[4] K. M. bin Abdl, S. Z. M. Hashim, Handwriting identification: a direction review, in: 2009 IEEE International Conference on Signal and Image Processing Applications, IEEE, 2009, pp. 459–463.

[5] R. A. Huber, A. M. Headrick, Handwriting identification: facts and fundamentals, CRC press, 1999.

[6] O. Hilton, Scientific examination of questioned documents, CRC press, 1992.

[7] A. Abdalhaleem, B. K. Barakat, J. El-Sana, Case study: Fine writing style classification using siamese neural network, in: 2018 IEEE 2nd International Workshop on Arabic and Derived Script Analysis and Recognition (ASAR), IEEE, 2018, pp. 62–66.

[8] M. Kassis, J. Nassour, J. El-Sana, Writing Style Invariant Deep Learning Model for Historical Manuscripts Alignment, arXiv preprint arXiv:1806.03987.

[9] A. Pirrone, M. B. Aimar, N. Journet, Papy-S-Net: A Siamese Network to match papyrus fragments, in: Proceedings of the 5th International Workshop on Historical Document Imaging and Processing, 2019, pp. 78–83.

[10] A. Durou, I. Aref, S. Al-Maadeed, A. Bouridane, E. Benkhelifa, Writer identification approach based on bag of words with OBI features,

Information Processing & Management 56 (2) (2019) 354–366, advance Arabic Natural Language Processing (ANLP) and its Applications. doi:https://doi.org/10.1016/j.ipm.2017.09.005.
URL https://www.sciencedirect.com/science/article/pii/S0306457316305829

[11] M. Popović, M. A. Dhali, L. Schomaker, Artificial intelligence based writer identification generates new evidence for the unknown scribes of the Dead Sea Scrolls exemplified by the Great Isaiah Scroll (1QIsaa), PloS one 16 (4) (2021) e0249769.

[12] Biblioteca Apostolica Vaticana, Website of the Biblioteca Apostolica Vaticana, https://www.vaticanlibrary.va/en/.

[13] T. Chen, S. Kornblith, M. Norouzi, G. Hinton, A simple framework for contrastive learning of visual representations, in: International conference on machine learning, PMLR, 2020, pp. 1597–1607.

[14] W. Falcon, K. Cho, A framework for contrastive self-supervised learning and designing a new approach, arXiv preprint arXiv:2009.00104.

[15] S. Gidaris, A. Bursuc, N. Komodakis, P. Pérez, M. Cord, Learning representations by predicting bags of visual words, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 6928–6938.

[16] K. He, H. Fan, Y. Wu, S. Xie, R. Girshick, Momentum contrast for unsupervised visual representation learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 9729–9738.

[17] S. Gidaris, A. Bursuc, G. Puy, N. Komodakis, M. Cord, P. Pérez, Online bag-of-visual-words generation for unsupervised representation learning, arXiv preprint arXiv:2012.11552.

[18] R. Zhang, P. Isola, A. A. Efros, Colorful image colorization, in: European conference on computer vision, Springer, 2016, pp. 649–666.

[19] A. v. d. Oord, Y. Li, O. Vinyals, Representation learning with contrastive predictive coding, arXiv preprint arXiv:1807.03748.

[20] P. Bachman, R. D. Hjelm, W. Buchwalter, Learning Representations by Maximizing Mutual Information Across Views, Advances in Neural Information Processing Systems 32 (2019) 15535–15545.

[21] A. Kolesnikov, X. Zhai, L. Beyer, Revisiting self-supervised visual representation learning, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2019, pp. 1920–1929.

[22] K. Q. Weinberger, L. K. Saul, Distance metric learning for large margin nearest neighbor classification., Journal of machine learning research 10 (2).

[23] F. Schroff, D. Kalenichenko, J. Philbin, Facenet: A unified embedding for face recognition and clustering, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 815–823.

[24] A. Hermans, L. Beyer, B. Leibe, In defense of the triplet loss for person re-identification, arXiv preprint arXiv:1703.07737.

[25] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al., Imagenet large scale visual recognition challenge, International journal of computer vision 115 (3) (2015) 211–252.

[26] ICARUS – International Centre for Archival Research, Monasterium.net, `https://www.icar-us.eu/en/cooperation/online-portals/monasterium-net/`.

[27] ICARUS – International Centre for Archival Research, Monasterium Collaborative Archive, `https://www.monasterium.net/mom/home`.

[28] HIMANIS project, HIMANIS — HIstorical MANuscript Indexing for user-controlled Search, https://himanis.hypotheses.org/.

[29] HIMANIS project, Himanis – Chancery Indexing and Search – Huma-Num, http://himanis.huma-num.fr/app/.

[30] T. Hassner, R. Sablatnig, D. Stutzmann, S. Tarte, Digital Palaeography: New Machines and Old Texts (Dagstuhl Seminar 14302), Dagstuhl Reports 4 (7) (2014) 112–134. doi:10.4230/DagRep.4.7.112.
URL http://drops.dagstuhl.de/opus/volltexte/2014/4793

[31] S. R. Narang, M. K. Jindal, M. Kumar, Ancient text recognition: a review, Artificial Intelligence Review 53 (8) (2020) 5517–5558.

[32] D. Singh, J. P. Saini, D. S. Chauhan, Hindi character recognition using RBF neural network and directional group feature extraction technique, in: 2015 International Conference on Cognitive Computing and Information Processing (CCIP), 2015, pp. 1–4. doi:10.1109/CCIP.2015.7100726.

[33] B. Alizadehashraf, S. Roohi, Persian handwritten character recognition using convolutional neural network, in: 2017 10th Iranian Conference on Machine Vision and Image Processing (MVIP), 2017, pp. 247–251. doi:10.1109/IranianMVIP.2017.8342359.

[34] B. Purkaystha, T. Datta, M. S. Islam, Bengali handwritten character recognition using deep convolutional neural network, in: 2017 20th International Conference of Computer and Information Technology (ICCIT), 2017, pp. 1–5. doi:10.1109/ICCITECHN.2017.8281853.

[35] M.-S. Kim, M.-D. Jang, H.-I. Choi, T.-H. Rhee, J.-H. Kim, H.-K. Kwag, Digitalizing scheme of handwritten Hanja historical documents, in: First International Workshop on Document Image Analysis for Libraries, 2004. Proceedings., 2004, pp. 321–327. doi:10.1109/DIAL.2004.1263261.

[36] M. Kumar, S. R. Jindal, M. K. Jindal, G. S. Lehal, Improved recognition results of medieval handwritten Gurmukhi manuscripts using boosting and bagging methodologies, Neural Processing Letters 50 (1) (2019) 43–56.

[37] S. R. Narang, M. Kumar, M. K. Jindal, DeepNetDevanagari: a deep learning model for Devanagari ancient character recognition, Multimedia Tools and Applications 80 (13) (2021) 20671–20686.

[38] G. Koch, R. Zemel, R. Salakhutdinov, et al., Siamese neural networks for one-shot image recognition, in: ICML deep learning workshop, Vol. 2, Lille, 2015.

[39] B. M. Lake, R. Salakhutdinov, J. B. Tenenbaum, Human-level concept learning through probabilistic program induction, Science 350 (6266) (2015) 1332–1338.

[40] H. Li, X. Ren, Y. Lv, One-Shot Chinese Character Recognition Based on Deep Siamese Networks, in: Chinese Intelligent Systems Conference, Springer, 2019, pp. 742–750.

[41] C.-L. Liu, F. Yin, D.-H. Wang, Q.-F. Wang, CASIA online and offline Chinese handwriting databases, in: 2011 International Conference on Document Analysis and Recognition, IEEE, 2011, pp. 37–41.

[42] C. Ostertag, M. Beurton-Aimar, Matching ostraca fragments using a siamese neural network, Pattern Recognition Letters 131 (2020) 336–340.

[43] A. Pirrone, M. Beurton-Aimar, N. Journet, Self-supervised deep metric learning for ancient papyrus fragments retrieval, International Journal on Document Analysis and Recognition (IJDAR) (2021) 1–16.

[44] M. Kassis, J. Nassour, J. El-Sana, Alignment of historical handwritten manuscripts using siamese neural network, in: 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), Vol. 1, IEEE, 2017, pp. 293–298.

[45] S. Dey, A. Dutta, J. I. Toledo, S. K. Ghosh, J. Lladós, U. Pal, Signet: Convolutional siamese network for writer independent offline signature verification, arXiv preprint arXiv:1707.02131.

[46] K. Ahrabian, B. BabaAli, Usage of autoencoders and Siamese networks for online handwritten signature verification, Neural Computing and Applications 31 (12) (2019) 9321–9334.

[47] P. A. Stokes, Modeling Medieval Handwriting: A New Approach to Digital Palaeography, in: J. C. Meister (Ed.), Digital Humanities 2012, University of Hamburg, Hamburg, 2012, pp. 382–385.

[48] P. A. Stokes, Digital Approaches to Paleography and Book History: Some Challenges, Present and Future, Frontiers in Digital Humanities 2 (2015) 5. doi:10.3389/fdigh.2015.00005.
URL https://www.frontiersin.org/article/10.3389/fdigh.2015.00005

[49] D. Stutzmann, C. Tensmeyer, V. Christlein, Writer identification and script classification: two tasks for a common understanding of cultural heritage, OpenX for Interdisciplinary Computational Manuscript Research (2018) 12–15.

[50] L. Wolf, L. Potikha, N. Dershowitz, R. Shweka, Y. Choueka, Computerized paleography: tools for historical manuscripts, in: 2011 18th IEEE International Conference on Image Processing, IEEE, 2011, pp. 3545–3548.

[51] H. Mohammed, I. Marthot-Santaniello, V. Märgner, Grk-papyri: A dataset of Greek handwriting on papyri for the task of writer identification, in: 2019 International Conference on Document Analysis and Recognition (ICDAR), IEEE, 2019, pp. 726–731.

[52] H. Mohammed, V. Mäergner, T. Konidaris, H. S. Stiehl, Normalised local Naïve Bayes nearest-neighbour classifier for offline writer identification,

in: 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), Vol. 1, IEEE, 2017, pp. 1013–1018.

[53] M. A. Dhali, S. He, M. Popović, E. Tigchelaar, L. Schomaker, A digital palaeographic approach towards writer identification in the dead sea scrolls, in: Proceedings of the 6th International Conference on Pattern Recognition Applications and Methods-Volume 1: ICPRAM, Vol. 2017, Scitepress; Setúbal, 2017, pp. 693–702.

[54] M. A. Dhali, J. W. de Wit, L. Schomaker, Binet: Degraded-manuscript binarization in diverse document textures and layouts using deep encoder-decoder networks, arXiv preprint arXiv:1911.07930.

[55] A. Shaus, Y. Gerber, S. Faigenbaum-Golovin, B. Sober, E. Piasetzky, I. Finkelstein, Forensic document examination and algorithmic handwriting analysis of Judahite biblical period inscriptions reveal significant literacy level, Plos one 15 (9) (2020) e0237962.

[56] N. Cilia, C. De Stefano, F. Fontanella, C. Marrocco, M. Molinara, A. Scotto Di Freca, An end-to-end deep learning system for medieval writer identification, Pattern Recognition Letters 129 (2020) 137–143. doi:https://doi.org/10.1016/j.patrec.2019.11.025.
URL https://www.sciencedirect.com/science/article/pii/S0167865519303460

[57] T. Chen, S. Kornblith, K. Swersky, M. Norouzi, G. E. Hinton, Big Self-Supervised Models are Strong Semi-Supervised Learners, Advances in Neural Information Processing Systems 33 (2020) 22243–22255.

[58] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. H. Richemond, E. Buchatskaya, C. Doersch, B. A. Pires, Z. D. Guo, M. G. Azar, et al., Bootstrap your own latent: A new approach to self-supervised learning, arXiv preprint arXiv:2006.07733.

[59] V. Dumoulin, I. Belghazi, B. Poole, O. Mastropietro, A. Lamb, M. Arjovsky, A. Courville, Adversarially learned inference, arXiv preprint arXiv:1606.00704.

[60] J. Donahue, K. Simonyan, Large Scale Adversarial Representation Learning, Advances in Neural Information Processing Systems 32 (2019) 10542–10552.

[61] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, A. Joulin, Unsupervised Learning of Visual Features by Contrasting Cluster Assignments, in: H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, H. Lin (Eds.), Advances in Neural Information Processing Systems, Vol. 33, Curran Associates, Inc., 2020, pp. 9912–9924.
URL https://proceedings.neurips.cc/paper/2020/file/70feb62b69f16e0238f741fab228fec2-Paper.pdf

[62] J. Zbontar, L. Jing, I. Misra, Y. LeCun, S. Deny, Barlow twins: Self-supervised learning via redundancy reduction, arXiv preprint arXiv:2103.03230.

[63] A. Coates, A. Ng, H. Lee, An analysis of single-layer networks in unsupervised feature learning, in: Proceedings of the fourteenth international conference on artificial intelligence and statistics, JMLR Workshop and Conference Proceedings, 2011, pp. 215–223.

[64] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, A. Oliva, Learning Deep Features for Scene Recognition using Places Database, Advances in Neural Information Processing Systems 27 (2014) 487–495.

[65] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, A. Zisserman, The pascal visual object classes (voc) challenge, International journal of computer vision 88 (2) (2010) 303–338.

[66] S. Gidaris, A. Bursuc, N. Komodakis, P. Pérez, M. Cord, Boosting few-shot visual learning with self-supervision, in: Proceedings of the

IEEE/CVF International Conference on Computer Vision, 2019, pp. 8059–8068.

[67] J.-C. Su, S. Maji, B. Hariharan, When does self-supervision improve few-shot learning?, in: European Conference on Computer Vision, Springer, 2020, pp. 645–666.

[68] O. Henaff, Data-efficient image recognition with contrastive predictive coding, in: International Conference on Machine Learning, PMLR, 2020, pp. 4182–4192.

[69] X. Zhai, A. Oliver, A. Kolesnikov, L. Beyer, S4l: Self-supervised semi-supervised learning, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 1476–1485.

[70] T. Chen, X. Zhai, M. Ritter, M. Lucic, N. Houlsby, Self-supervised generative adversarial networks, arXiv preprint arXiv:1811.11212 2.

[71] P. O. O. Pinheiro, A. Almahairi, R. Benmalek, F. Golemo, A. C. Courville, Unsupervised Learning of Dense Visual Representations, in: H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, H. Lin (Eds.), Advances in Neural Information Processing Systems, Vol. 33, Curran Associates, Inc., 2020, pp. 4489–4500.
URL https://proceedings.neurips.cc/paper/2020/file/3000311ca56a1cb93397bc676c0b7fff-Paper.pdf

[72] R. Hadsell, S. Chopra, Y. LeCun, Dimensionality reduction by learning an invariant mapping, in: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06), Vol. 2, IEEE, 2006, pp. 1735–1742.

[73] P. H. Le-Khac, G. Healy, A. F. Smeaton, Contrastive representation learning: A framework and review, IEEE Access.

[74] Y. Tian, D. Krishnan, P. Isola, Contrastive multiview coding, in: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16, Springer, 2020, pp. 776–794.

[75] X. Chen, H. Fan, R. Girshick, K. He, Improved baselines with momentum contrastive learning, arXiv preprint arXiv:2003.04297.

[76] M. Caron, P. Bojanowski, A. Joulin, M. Douze, Deep clustering for unsupervised learning of visual features, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 132–149.

[77] M. Caron, P. Bojanowski, J. Mairal, A. Joulin, Unsupervised pre-training of image features on non-curated data, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 2959–2968.

[78] Y. M. Asano, C. Rupprecht, A. Vedaldi, Self-labelling via simultaneous clustering and representation learning, arXiv preprint arXiv:1911.05371.

[79] A. Dosovitskiy, J. T. Springenberg, M. Riedmiller, T. Brox, Discriminative unsupervised feature learning with convolutional neural networks, Advances in neural information processing systems 27 (2014) 766–774.

[80] C. Doersch, A. Gupta, A. A. Efros, Unsupervised visual representation learning by context prediction, in: Proceedings of the IEEE international conference on computer vision, 2015, pp. 1422–1430.

[81] M. Noroozi, P. Favaro, Unsupervised learning of visual representations by solving jigsaw puzzles, in: European conference on computer vision, Springer, 2016, pp. 69–84.

[82] G. Larsson, M. Maire, G. Shakhnarovich, Learning representations for automatic colorization, in: European conference on computer vision, Springer, 2016, pp. 577–593.

[83] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, A. A. Efros, Context encoders: Feature learning by inpainting, in: Proceedings of the IEEE

conference on computer vision and pattern recognition, 2016, pp. 2536–2544.

[84] C. Doersch, A. Zisserman, Multi-task self-supervised visual learning, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 2051–2060.

[85] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, et al., Photo-realistic single image super-resolution using a generative adversarial network, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 4681–4690.

[86] S. Gidaris, P. Singh, N. Komodakis, Unsupervised Representation Learning by Predicting Image Rotations, in: ICLR (Poster), OpenReview.net, 2018.

[87] P. Cherubini, A. Pratesi, Paleografia latina. Tavole, Vol. 1, Scuola Vaticana di Paleografia, Diplomatica e Archivistica, 2004.

[88] P. Cherubini, A. Pratesi, Paleografia latina. L'avventura grafica del mondo occidentale, Littera Antiqua, 16, Scuola Vaticana di Paleografia, Diplomatica e Archivistica, Città del Vaticano, 2010.

[89] F. Coulson, R. Babcock, The Oxford Handbook of Latin Palaeography, Oxford University Press, USA, 2020.

[90] M. Maniaci, G. Orofino, Le Bibbie atlantiche. Il Libro delle Scritture tra monumentalità e rappresentazione, Centro Tibaldi, Milano, Italia, 2000.

[91] K. He, X. Zhang, S. Ren, J. Sun, Deep Residual Learning for Image Recognition, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770–778. doi:10.1109/CVPR.2016.90.

[92] M. Cuturi, Sinkhorn distances: Lightspeed computation of optimal transport, Advances in neural information processing systems 26 (2013) 2292–2300.

[93] Olivier Moindrot, Triplet Loss and Online Triplet Mining in TensorFlow, https://omoindrot.github.io/triplet-loss (2018).

[94] K. Musgrave, S. Belongie, S.-N. Lim, A Metric Learning Reality Check, in: A. Vedaldi, H. Bischof, T. Brox, J.-M. Frahm (Eds.), Computer Vision – ECCV 2020, Springer International Publishing, Cham, 2020, pp. 681–699.

[95] E. Shelhamer, J. Long, T. Darrell, Fully Convolutional Networks for Semantic Segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence 39 (4) (2017) 640–651. doi:10.1109/TPAMI.2016.2572683.

[96] C. D. Manning, P. Raghavan, H. Schütze, Introduction to Information Retrieval, Cambridge University Press, Cambridge, UK, 2008.

[97] J. S. Chung, A. Nagrani, A. Zisserman, Voxceleb2: Deep speaker recognition, arXiv preprint arXiv:1806.05622.

[98] L. Van der Maaten, G. Hinton, Visualizing data using t-SNE., Journal of machine learning research 9 (11).

[99] A. Jaiswal, A. R. Babu, M. Z. Zadeh, D. Banerjee, F. Makedon, A survey on contrastive self-supervised learning, Technologies 9 (1) (2021) 2.

[100] D. Basaj, W. Oleszkiewicz, I. Sieradzki, M. Górszczak, B. Rychalska, T. Trzcinski, B. Zieliński, Explaining Self-Supervised Image Representations with Visual Probing, in: Z.-H. Zhou (Ed.), Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21, International Joint Conferences on Artificial Intelligence Organization, 2021, pp. 592–598, main Track. doi:10.24963/ijcai.2021/82.
URL https://doi.org/10.24963/ijcai.2021/82

[101] P. A. Stokes, Scribal Attribution across Multiple Scripts: A Digitally Aided Approach, Speculum 92 (S1) (2017) S65–S85. doi:10.1086/693968.
URL https://doi.org/10.1086/693968

[102] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al., An image is worth 16x16 words: Transformers for image recognition at scale, arXiv preprint arXiv:2010.11929.

[103] I. Tolstikhin, N. Houlsby, A. Kolesnikov, L. Beyer, X. Zhai, T. Unterthiner, J. Yung, D. Keysers, J. Uszkoreit, M. Lucic, et al., Mlp-mixer: An all-mlp architecture for vision, arXiv preprint arXiv:2105.01601.

[104] L. Deng, G. Li, S. Han, L. Shi, Y. Xie, Model Compression and Hardware Acceleration for Neural Networks: A Comprehensive Survey, Proceedings of the IEEE 108 (4) (2020) 485–532. doi:10.1109/JPROC.2020.2976475.

[105] H. Cai, C. Gan, L. Zhu, S. Han, TinyTL: Reduce Memory, Not Parameters for Efficient On-Device Learning, in: H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, H. Lin (Eds.), Advances in Neural Information Processing Systems, Vol. 33, Curran Associates, Inc., 2020, pp. 11285–11297.
URL https://proceedings.neurips.cc/paper/2020/file/81f7acabd411274fcf65ce2070ed568a-Paper.pdf

[106] T. Hoefler, D. Alistarh, T. Ben-Nun, N. Dryden, A. Peste, Sparsity in Deep Learning: Pruning and growth for efficient inference and training in neural networks, arXiv preprint arXiv:2102.00554.

[107] J. Gou, B. Yu, S. J. Maybank, D. Tao, Knowledge distillation: A survey, International Journal of Computer Vision 129 (6) (2021) 1789–1819.

[108] R. Bommasani, D. A. Hudson, E. Adeli, R. Altman, S. Arora, S. von Arx, M. S. Bernstein, J. Bohg, A. Bosselut, E. Brunskill, et al., On the Opportunities and Risks of Foundation Models, arXiv preprint arXiv:2108.07258.

[109] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep

bidirectional transformers for language understanding, arXiv preprint arXiv:1810.04805.

[110] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al., Language models are few-shot learners, arXiv preprint arXiv:2005.14165.