# Unsupervised Feature Learning for Writer Identification and Writer Retrieval

Vincent Christlein*, Martin Gropp*, Stefan Fiel†, and Andreas Maier*
* Pattern Recognition Lab, Friedrich-Alexander-Universität Erlangen-Nürnberg, 91058 Erlangen, Germany
† Computer Vision Lab, TU Wien, 1040 Vienna, Austria,
vincent.christlein@fau.de, martin.gropp@fau.de, fiel@caa.tuwien.ac.at, andreas.maier@fau.de

***Abstract*—Deep Convolutional Neural Networks (CNN) have shown great success in supervised classification tasks such as character classification or dating. Deep learning methods typically need a lot of annotated training data, which is not available in many scenarios. In these cases, traditional methods are often better than or equivalent to deep learning methods. In this paper, we propose a simple, yet effective, way to learn CNN activation features in an unsupervised manner. Therefore, we train a deep residual network using surrogate classes. The surrogate classes are created by clustering the training dataset, where each cluster index represents one surrogate class. The activations from the penultimate CNN layer serve as features for subsequent classification tasks. We evaluate the feature representations on two publicly available datasets. The focus lies on the ICDAR17 competition dataset on historical document writer identification (Historical-WI). We show that the activation features trained without supervision are superior to descriptors of state-of-the-art writer identification methods. Additionally, we achieve comparable results in the case of handwriting classification using the ICFHR16 competition dataset on historical Latin script types (CLaMM16).**

***Keywords*-unsupervised feature learning; writer identification; writer retrieval; deep learning; document analysis**

## I. Introduction

The analysis of historical data is typically a task for experts in history or paleography. However, due to the digitization process of archives and libraries, a manual analysis of a large data corpus might not be feasible anymore. We believe that automatic methods can support people working in the field of humanities. In this paper, we focus on the task of *writer identification* and *writer retrieval*. Writer identification refers to the problem of assigning the correct writer for a query image by comparing it with images of known scribal attribution. For writer retrieval, the task consists of finding all relevant documents of a specific writer. Additionally, we evaluate our method in a classification task to classify historical script types.

We make use of deep Convolutional Neural Networks (CNN) that are able to create powerful feature representations [1] and are the state-of-the-art tool for image classification since the AlexNet CNN of Krizhevsky et al. [2] won the ImageNet competition. Deep-learning-based methods achieve also great performance in the field of handwritten documents classification, e. g., dating [3], word spotting [4], or handwritten text recognition [5]. However, such methods typically require a lot of labeled data for each class. We face another problem in the case of writer identification, where the writers of the training set are different from those of the test set in the typical used benchmark datasets. On top of that, current datasets have only one to five images per writer. While a form of writer adaptation with exemplar Support Vector Machines (E-SVM) is possible [6], CNN training for each query image would be very cost-intensive. Thus, deep-learning-based methods are solely used to create robust features [7]–[9]. In these cases, the writers of the training set serve as surrogate classes. In comparison to this supervised feature learning, we show that deep activation features learned in an unsupervised manner can *i)* serve as better surrogate classes, and *ii)* outperform handcrafted features from current state-of-the-art methods.

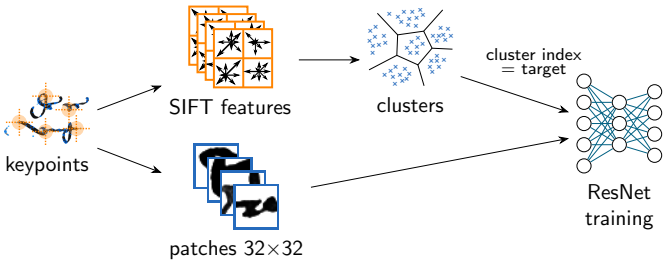


Fig. 1: Overview of the unsupervised feature learning. At SIFT keypoint locations, SIFT descriptors and image patches are extracted. The cluster indices of the clustered SIFT descriptors represent the targets and the corresponding patches as input for the CNN training.

In detail, our contributions are as follows:

- We present a simple method for feature learning using deep neural networks without the need of labeled data. Fig. 1 gives an overview of our method. First SIFT descriptors [10] are computed on the training dataset, which are subsequently clustered. A deep residual network (ResNet) [11] is trained using patches extracted from each SIFT location (keypoint) using the cluster membership as target. The activations of the penultimate layer serve as local feature descriptors that are subsequently encoded and classified.
- We thoroughly evaluate all steps of our pipeline using a publicly available dataset on historical document writer identification.
- We show that our method outperforms state-of-the-art in the case of writer identification and retrieval.

- Additionally, we evaluate our method for the classification of medieval script types. On this task, we achieve equally good results as the competition winner.

The rest of the paper is organized as follows. Sec. II gives an overview over the related work in the field of unsupervised feature learning, and writer identification. The unsupervised feature learning and encoding step is presented in Sec. III. The evaluation protocol is given in Sec. IV, and the results in Sec. V. Sec. VI gives a summary and an outlook.

## II. Related Work

We focus our evaluation on the task of writer identification and retrieval. Method-wise, writer identification / retrieval can be divided into two groups: *statistical* methods (a. k. a. textural methods [12]) and *codebook*-based methods. The differentiation lies in the creation of the global descriptor which is going to be compared, or classified, respectively. Global statistics of the handwriting are computed in the former group, such as the width of the ink trace, or the angles of stroke directions [13], [14]. More recently, Nicolaou et al. [15] employed local binary patterns that are evaluated densely at the image.

Conversely, codebook-based descriptors are based on the well-known Bag-of-(Visual)-Words (BoW) principle, i. e., a global descriptor is created by encoding local descriptors using statistics obtained by a pre-trained dictionary. Fisher vectors [16], VLAD [17] or self organizing maps [18] were employed for writer identification and retrieval. Popular local descriptors for writer identification are based on Scale Invariant Feature Transform [10] (SIFT), see [6], [16], [19], [20]. However, also handcrafted descriptors are developed that are specifically designed to work well on handwriting. One example is the work by He et al. [18], who characterize script by computing junctions of the handwriting. In contrast, the hereinafter presented work learns the descriptors using a deep CNN. In previous works the writers of the training datasets have been used as targets for the CNN training [7]–[9]. While the output neurons of the last layer were aggregated using sum-pooling by Xing and Qiao [9], the activation features of the penultimate layer were encoded using Fisher vectors [8] and GMM supervectors [7]. In contrast, we do not rely on any writer label information, but use cluster membership of image patches as surrogate targets.

Clustering has also been used to create unsupervised attributes for historical document dating in the work of He et al. [21]. However, they use handcrafted features in conjunction with SVMs. Instead, we learn the features in an unsupervised manner using a deep CNN.

The most closely related work comes from Dosovitskiy et al. [22], where surrogate classes are created by a variety of image transformations such as rotation or scale. Using these classes to train a CNN, they generate features, which are invariant to many transformations and are advantageous in comparison to handcrafted features. They also suggest to cluster the images in advance to apply their transformations on each cluster image, and then use the cluster indices as surrogate classes. A similar procedure is applied by Huang et al. [23] to discover shared attributes and visual representations. In comparison to the datasets used in the evaluation of Dosovitskiy et al. and Huang et al., we have much more training samples available since we consider small handwriting patches. Thus, an exhaustive augmentation of the dataset is not necessary; instead, one cluster directly represents a surrogate class. Another interesting approach for deep unsupervised feature learning is the work of Paulin et al. [24], where Convolutional Kernel Networks (CKN) are employed. CKNs are similar to CNNs but are trained layer-wise to approximate a particular non-linear kernel.

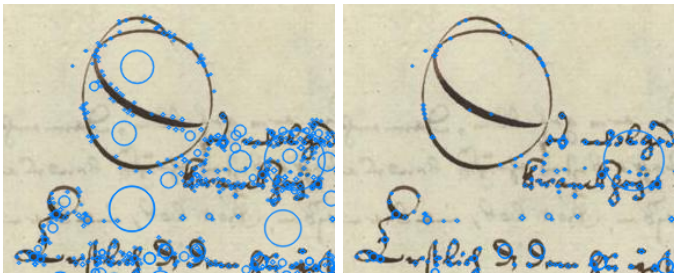

Fig. 2: Excerpt of an image of the Historical-WI dataset. Left: Original SIFT keypoints, right: restricted SIFT keypoints.

## III. Methodology

Our goal is to learn robust local features in an unsupervised manner. These features can then be used for subsequent classification tasks such as writer identification or script type classification. Therefore, a state-of-the-art CNN architecture is employed to train a powerful patch representation using cluster memberships as targets. A global image descriptor is created by means of VLAD encoding.

### A. Unsupervised Feature Learning

First, SIFT keypoints are extracted. At each keypoint location a SIFT descriptor and a $32 \times 32$ patch is extracted. The SIFT descriptors of the training set are clustered. While the patches are the inputs for the CNN training, the cluster memberships of the corresponding SIFT descriptors are used as targets. Cf. also Fig. 1 for an overview of the feature learning process.

SIFT keypoint localization is based on blob detection [10]. The keypoints rely on finding both minima and maxima in the Difference-of-Gaussian (DoG) scale space, and in addition to document coordinates also contain information about rotation and "size", i. e., their location in scale space. The keypoints commonly occur between text lines, as can be seen in Fig. 2. These gratuitous locations can be filtered out either afterwards by analyzing the keypoint size or using the binarized image as mask. Another possibility is to restrict the SIFT keypoint algorithm on finding only minima in the scale space, thus, obtaining only dark on bright blobs. We employ this technique to mainly obtain patches containing text, further referred to R-SIFT (restricted SIFT). Note that we also filter keypoints positioned at the same location to always obtain distinct input patches.

For an improved cluster association, we also normalize the SIFT descriptors by applying the Hellinger kernel [25]. In practice, the Hellinger normalization of SIFT descriptors consists of an element-wise application of the square root, followed by an $l_1$ normalization. This normalization effectively helps to reduce the occurrence of visual bursts, i. e., dominating bins in the SIFT descriptor, and has been shown to improve image recognition [25] and writer identification / retrieval [6]. The descriptors are dimensionality-reduced from 128 to 32 dimensions and whitened using principal component analysis (PCA) to lower the computational cost of the clustering process.

For clustering we use a subset of 500k randomly chosen R-SIFT descriptors of the training set. We use the mini-batch version of $k$-means [26] for a fast clustering. After the clustering process, we filter out descriptors (and corresponding patches) that lie on the border between two clusters. Therefore, the ratio $\rho$ between the distances of the input descriptor $\boldsymbol{x}$ to the closest cluster center $\boldsymbol{\mu}_1$ and to the second closest one $\boldsymbol{\mu}_2$ is computed, i. e.:

$$\rho = \frac{\|\boldsymbol{x} - \boldsymbol{\mu}_1\|_2}{\|\boldsymbol{x} - \boldsymbol{\mu}_2\|_2} \tag{1}$$

If this ratio is too large, the descriptor is removed. In practice, we use a maximum allowed ratio of $0.9$.

Given the $32 \times 32$ image patches and their cluster memberships, a deep CNN is trained. We employ a deep residual network [11] (ResNet) with 20-layers. Residual networks have shown great results in image classification and object recognition. A ResNet consists of residual building blocks that have two branches. One branch has two or more convolutional layers and the other one just forwards the result of the previous layer, thus bypassing the other branch. These building blocks help to preserve the identity and allow training deeper models. As the residual building block, we use the pre-resnet building block of [27]. For training, we follow the architectural design and procedure of He et al. [11] for the CIFAR10 dataset. Following previous works [7]–[9], we use the activations of the penultimate layer as feature descriptors. Note that typically the features of the penultimate layer are most distinctive [28], but other layers are possible, too [24]. In our case, the penultimate layer is a pooling layer that pools the filters from the previous residual block. It consists of 64 hidden nodes resulting in a feature descriptor dimensionality of 64.

### B. Encoding

A global image descriptor is created by encoding the obtained CNN activation features. We use VLAD encoding [29], which can be seen as a non-probabilistic version of the Fisher Kernel. It encodes first order statistics by aggregating the residuals of local descriptors to their corresponding nearest cluster center. VLAD is a standard encoding method, which has already been used for writer identification [17]. It has also successfully been used to encode CNN activation features for classification and retrieval tasks [24], [30].

Formally, a VLAD is constructed as follows [29]. First, a codebook $\boldsymbol{D} = \{\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_K\}$ is computed from random descriptors of the training set using $k$-means with $K$ clusters. Every local image descriptor $\boldsymbol{x}$ of one image is assigned to its nearest cluster center. Then, all residuals between the cluster center and the assigned descriptors are accumulated for each cluster:

$$\boldsymbol{v}_k = \sum_{\boldsymbol{x}_t:\ \mathrm{NN}(\boldsymbol{x}_t)=\boldsymbol{\mu}_k} (\boldsymbol{x}_t - \boldsymbol{\mu}_k)\ , \tag{2}$$

where $\mathrm{NN}(\boldsymbol{x}_t)$ refers to the nearest neighbor of $\boldsymbol{x}_t$ in the dictionary $\boldsymbol{D}$. The final VLAD encoding is the concatenation of all $\boldsymbol{v}_k$:

$$\boldsymbol{v} = \left(\boldsymbol{v}_1^\top, \ldots, \boldsymbol{v}_K^\top\right)^\top\ . \tag{3}$$

We use *power* normalization [29] instead of the more recent *intra* normalization [31]. The former one is preferable, since we employ keypoints for the patch extraction instead of a dense sampling [32]. In power-normalization, the normalized vector $\hat{\boldsymbol{v}}$ follows as:

$$\hat{\boldsymbol{v}}_i \coloneqq \mathrm{sign}(\boldsymbol{v}_i)|\boldsymbol{v}_i|^\rho \qquad \forall i = \{1, \ldots, |\boldsymbol{v}|\}, 0 < \rho \leq 1\ , \tag{4}$$

where we set $\rho$ to $0.5$. Afterwards, the vector is $l_2$-normalized.

Similar to the work of Christlein et al. [17], multiple codebooks are computed from different random training descriptors. For each of these codebooks a VLAD encoding is computed. The encodings are subsequently decorrelated and optionally dimensionality reduced by means of PCA whitening. This step has been shown to be very beneficial for writer and image retrieval [17], [33], [34]. We refer to this approach as multiple codebook VLAD, or short m-VLAD.

### C. Exemplar SVM

Additionally, we train linear support vector machines (SVM) for each individual query sample. Such an Exemplar SVM (E-SVM) is trained with only a single positive sample and multiple negative samples. This method was originally proposed for object detection [35], where an ensemble of E-SVMs is used for each object class. Conversely, E-SVMs can also be used to adapt to a specific face image [36] or writer [6]. In principle, we follow the approach of Christlein et al. [6] and use E-SVMs at query time. Since we know that the writers of the training set are independent from those of the test set, an E-SVM is trained using the query VLAD encoding as positive sample and all the training encodings as negatives. This has the effect of computing an individual similarity for the query descriptor.

The SVM large margin formulation with $l_2$ regularization and squared hinge loss $h(x) = \max(0, 1 - x)^2$ is defined as:

$$\underset{\boldsymbol{w}}{\operatorname{argmin}} \frac{1}{2}\|\boldsymbol{w}\|_2^2 + c_p h(\boldsymbol{w}^\top \boldsymbol{x_p}) + c_n \sum_{\boldsymbol{x_n} \in \mathcal{N}} h(-\boldsymbol{w}^\top \boldsymbol{x_n})\ , \tag{5}$$

where $\boldsymbol{x_p}$ is the single positive sample and $\boldsymbol{x_n}$ are the samples of the negative training set $\mathcal{N}$. $c_p$ and $c_n$ are regularization parameters for balancing the positive and negative costs. We chose to set them indirectly proportional to the number of samples such that only one parameter $C$ needs to be cross-validated in advance, cf. [36] for details.

Unlike the work of Christlein et al. [6], we do not rank the other images according to the SVM score. Instead, we use the linear SVM as feature encoder [37], [38], i. e., we directly use

the normalized weight vector as our new feature representation for $\boldsymbol{x}$:

$$\boldsymbol{x} \mapsto \hat{\boldsymbol{x}} = \frac{\boldsymbol{w}}{\|\boldsymbol{w}\|_2} \,. \tag{6}$$

The new representations are ranked according to their cosine similarity.

## IV. Evaluation Protocol

The focus of our evaluation lies on writer identification and retrieval, where we thoroughly explore the effects of different pipeline decisions. Additionally, the features are employed for the classification of medieval handwriting. In the following subsections the datasets, evaluation metrics and implementation details are presented.

### A. Datasets

The method proposed is evaluated on the dataset of the "ICDAR 2017 Competition on Historical Document Writer Identification" (Historical-WI) [39]. The test set consists of 3600 document images written by 720 different writers. Each writer contributed 5 pages to the dataset, which have been sampled equidistantly of all available documents to ensure a high variability of the data. The documents have been written between the $13^{th}$ and $20^{th}$ century and contain mostly correspondences in German, Latin, and French. The training set contains 1182 document images written by 394 writers. Again, the number of pages per writer is equally distributed.

Additionally, the method is evaluated on a document classification task using the dataset for the ICFHR2016 competition on the classification of medieval handwritings in Latin script (CLaMM16) [40]. It consists of 3000 images of Latin scripts scanned from handwritten books dated between 500 and 1600 CE. The dataset is split into 2000 training and 1000 test images. The task is to automatically classify the test images into one of twelve Latin script types.

### B. Evaluation Metrics

To evaluate our method, we use a leave-one-image-out procedure, where each image in the test set is used once as a query and the system has to retrieve a ranked list of documents from the remaining images. Ideally, the top entries of these lists would be the *relevant* documents written by the same scribe as the query image.

We use several common metrics to assess the quality of these results. *Soft Top N* (*Soft-N*) examines the $N$ items ranked at the top of a retrieved list. A list is considered an "acceptable" result if there is *at least one* relevant document in the top $N$ items. The final score for this metric is then the percentage of acceptable results. *Hard Top N* (*Hard-N*), by comparison, is much stricter and requires *all* of the top $N$ items to be relevant for an acceptable result.

*Precision at N* ($p@N$) computes the percentage of relevant documents in the top $N$ items of a result. The numbers reported for $p@N$ are the means over all queries.

The *average precision* (AP) measure considers the average $p@N$ over all positions $N$ of relevant documents in a result.

TABLE I: Using writers from the Historical-WI training dataset as targets for the feature computation. The evaluation is carried out using the Historical-WI test set.

| | p@1 | p@2 | p@3 | p@4 | mAP |
|---|---|---|---|---|---|
| Writers (LeNet) | 66.22 | 57.10 | 48.71 | 41.70 | 44.89 |
| Writers (ResNet) | 67.36 | 58.38 | 49.81 | 42.85 | 46.11 |

Taking the mean AP over all queries finally yields the *Mean Average Precision* score (mAP).

Since for $N = 1$ *Hard-N*, *Soft-N*, and $p@N$ are equivalent, we record these scores only once as *TOP-1*.

### C. Implementation Details

If not stated otherwise, the standard pipeline consists of $5000$ cluster indices as surrogate classes for $32 \times 32$ patches. The patches were extracted from the binarized images in the case of the Historical-WI dataset, and from the grayscale images in the case of the CLaMM16 dataset. The patches are extracted around the restricted SIFT keypoints (see Sec. III-A). We extract RootSIFT descriptors and apply a PCA for whitening and reducing the dimensionality to 32. These vectors are then used for the clustering step. A deep residual network (number of layers $L = 20$) is trained using stochastic gradient descent with an adaptive learning rate (i. e., if the error increases, the learning rate is divided by 10), a Nesterov momentum of $0.9$ and a weight decay of $0.0001$. The training runs for a maximum of 50 epochs, stopping early if the validation error (20k random patches not part of the training procedure) increases. Note that the maximum epoch number is sufficient given the large number of handwriting patches (480k). The activations of the penultimate layer are used as local descriptors. They are encoded using m-VLAD with five vocabularies. The final descriptors are PCA-whitened and compared using the cosine distance.

For the comparison with the state of the art, we also employ linear SVMs. The SVM margin parameter $C$ is cross-evaluated in the range $[10^{-5}, 10^{4}]$ using an inner stratified 5-fold cross-validation for script type classification. In the case of writer identification / retrieval a 2-fold cross-validation is employed, i. e., the training set is split into two writer-independent parts to have more E-SVMs for the validation.

## V. Results

First, the use of writers as surrogate classes is evaluated, similar to the work of Christlein et al. [7] and Fiel et al. [8]. Afterwards, our proposed method for feature learning, different encoding strategies and the used parameters are evaluated and eventually compared to the state-of-the-art methods.

### A. Writers as Surrogate Classes

A natural choice for the training targets are the writers of the training set. This has been successfully used by recent works for smaller, non-historical benchmark datasets such as the ICDAR 2013 competition dataset for writer identification [7],

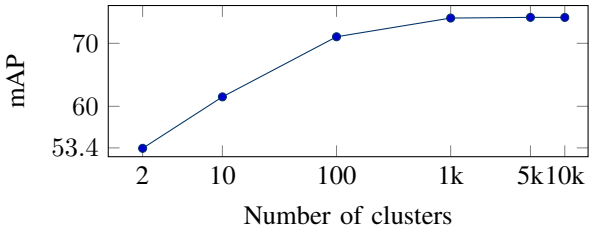


Fig. 3: Evaluation of the number of surrogate classes (clusters) using the Historical-WI test data.

TABLE II: Comparison of different encoding methods evaluated on the Historical-WI test test.

| | TOP-1 | mAP |
|---|---|---|
| Cl-S + Sum | 63.9 | 42.6 |
| Cl-S + FV | 76.9 | 57.6 |
| Cl-S + SV | 83.4 | 63.7 |
| Cl-S + VLAD | 82.6 | 63.6 |
| Cl-S + m-VLAD | 88.3 | 74.1 |
| Cl-S + m-VLAD$_{400}$ | 87.6 | 73.2 |

[8]. Thus, we employ the same scheme also for Historical-WI. On one hand, we employ the LeNet architecture used by Christlein et al. [7], i. e., two subsequent blocks of a convolutional layer, followed by a pooling layer, and a final fully connected layer before the target layer with its 394 nodes. On the other hand, we employ the same architecture we propose for our method, i. e., a residual network (ResNet) with 20 layers.

Tab. I reveals that the use of writers as the surrogate class does not work as intended. Independent of the architecture, we achieve much worse results than a standard approach using SIFT descriptors or Zernike moments, cf. Tab. V.

### B. Influence of the Encoding Method

For the following experiments, we now train our network using the cluster indices as surrogate classes (denoted as Cl-S). Babenko et al. [28] states that sum-pooling CNN activation features is superior to other encoding techniques such as VLAD or Fisher Vectors. In Tab. II, we compare sum-pooling to three other encoding methods: *I)* Fisher vectors [41] using first and second order statistics, which have also been employed for writer identification [16]. We normalize them in a manner similar to the proposed VLAD normalization, i. e., power normalization followed by an $L_2$ normalization. *II)* GMM supervectors [42], which were used for writer identification by Christlein et al. [6], normalized by a Kullback-Leibler normalization scheme. *III)* the proposed VLAD encoding [17].

Tab. II shows that sum pooling (Cl-S + Sum) performs significantly worse than other encoding schemes. While Fisher vectors (Cl-S + FV) trail the GMM supervectors (Cl-S + SV) and VLAD encoding (Cl-S + VLAD / m-VLAD / m-VLAD$_{400}$), GMM supervectors perform slightly better than the average of the non-whitened version of the five VLAD encodings (Cl-S + VLAD). However, when using the m-VLAD approach (Cl-S + m-VLAD), i. e., jointly decorrelating the five VLAD encodings by PCA whitening, we achieve a much higher precision. Even if we incorporate a dimensionality reduction to 400 components (Cl-S + m-VLAD$_{400}$) during the PCA whitening process, the results are significantly better than other encoding schemes with 6400 dimensions in case of the GMM supervectors, or 12 800 in case of the Fisher vectors.

TABLE III: Comparison of different parameters used for the unsupervised feature learning step evaluated on the Historical-WI test set.

| | TOP-1 | mAP |
|---|---|---|
| Cl-S (Baseline: $\rho = 0.9$, $L = 20$) | 88.3 | 74.1 |
| Cl-S ($L = 44$) | 88.2 | 74.3 |
| Cl-S ($\rho = 1.0$) | 87.3 | 72.4 |

TABLE IV: Evaluation of different sampling strategies evaluated on the Historical-WI test set. Bin. refers to the binarized images. SIFT and R-SIFT to the SIFT keypoint extraction method and the restricted keypoint extraction, respectively, cf. Sec. III-A.

| | TOP-1 | mAP |
|---|---|---|
| Cl-S (Baseline: Bin. / R-SIFT) | 88.3 | 74.1 |
| Cl-S (Bin. / SIFT) | 88.6 | 74.8 |
| Cl-S (Gray / R-SIFT) | 87.1 | 71.6 |
| Cl-S (Gray / SIFT) | 87.7 | 72.3 |

### C. Parameter Evaluation

Fig. 3 plots the writer retrieval performance given different numbers of surrogate classes that are used for clustering, and the training targets, respectively. Interestingly, even a small number of 2 clusters is sufficient to produce better results than using the writers as surrogate classes. When using more than 1 000 clusters, the results are very similar to each other with a peak at 5 000 clusters.

To evaluate the importance of the number of layers, we employed a much deeper residual network consisting in total of 44 layers (instead of 20). Since the results in Tab. III show that the increase in depth (Cl-S ($L = 44$)) produces only a slight improvement in terms of mAP, and comes with greater resource consumption, we stick to the smaller 20 layer deep network for the following experiments.

Next, we evaluate the influence of the parameter $\rho$, which is used to remove patches that do not clearly fall into one Voronoi cell computed by $k$-means, cf. Sec. III-A. When using a factor of 1.0 (instead of 0.9), and thus, not removing any patches, the performance drops from 74.1% mAP to 72.4% mAP.

### D. Sampling Importance

Finally, we also evaluate the impact of the proposed restricted SIFT keypoint computation (R-SIFT) in comparison to standard SIFT, as well as the influence of binarization (bin.) in comparison to grayscale patches (gray). We standardize the grayscale patches to zero mean and unit standard deviation. Tab. IV shows that binarization is in general beneficial for an improvement in precision. This is even more astonishing considering that

TABLE V: Comparison with state-of-the-art evaluated on the Historical-WI test set.

| Method | Top-1 | Hard-2 | Hard-3 | Hard-4 | Soft-5 | Soft-10 | p@2 | p@3 | p@4 | mAP |
|---|---|---|---|---|---|---|---|---|---|---|
| SIFT + FV [16] | 81.4 | 63.8 | 46.2 | 27.7 | 87.6 | 89.3 | 74.0 | 66.7 | 59.0 | 62.2 |
| C-Zernike + m-VLAD [17] | 86.0 | 71.4 | 56.8 | 37.7 | 90.3 | 91.7 | 79.9 | 73.6 | 66.4 | 69.2 |
| Cl-S | 88.6 | 77.1 | 64.7 | 46.8 | 92.2 | 93.4 | 83.8 | 78.9 | 72.3 | 74.8 |
| Cl-S + E-SVM-FE | **88.9** | **78.6** | **67.5** | **49.1** | **92.7** | **93.8** | **84.8** | **80.5** | **74.0** | **76.2** |

several images belong to the same handwritten letter. Thus, the background information should actually improve the results. A possible explanation could be that binary image patches are easier to train with, thus resulting in a better representation. When comparing SIFT with its restricted version (R-SIFT), the former consistently outperforms the restricted version by about 0.7% mAP. It seems that completely blank patches do not harm the CNN classification. This might be related to the clustering process, since all these patches typically end up in one cluster. Furthermore, the training patches, which are extracted, are more diverse. Also keypoints located right next to the contour are preserved, cf. Fig. 2.

In summary, we can state that 1) m-VLAD encoding is the best encoding candidate. 2) Our method is quite robust to the number of clusters. Given enough surrogate classes, the method outperforms other surrogate classes that need label information. 3) The removal of descriptors (and corresponding patches) using a simple ratio criterion seems to be beneficial. 4) Deeper networks do not seem to be necessary for the task of writer identification. 5) Patches extracted at SIFT keypoint locations computed on binarized images are preferable to other modalities.

### *E. Comparison with the state of the art*

We compare our method with the state-of-the-art methods of Fiel et al. [16] (SIFT + FV) and Christlein et al. [17] (C-Zernike + m-VLAD). While the former one uses SIFT descriptors that are encoded using Fisher vectors [41], the latter relies on Zernike moments evaluated densely at the contour that are subsequently encoded using the m-VLAD approach. Tab. V shows that our proposed method achieves superior results in comparison to these methods. Note that the encoding stage of the Contour-Zernike-based method is similar to ours (Cl-S). It differs only in the way of post-processing, where we use power normalization in preference to intra normalization [31]. However, the difference in accuracy is very small, see [17]. It follows that the improvement in performance just relies on the better feature descriptors. The use of Exemplar SVMs for feature encoding gives another improvement of nearly 1.5% mAP.

Additionally, we evaluate the method on the classification of medieval Latin script types. Tab. VI shows that our method is slightly, but not significantly, better than state-of-the-art methods [40] (Soft-5: 98.1%). Possible reasons are: a) the text areas in the images are not segmented, i. e., the images contain much more non-text elements such as decorations, which might lower the actual feature learning process; b) the images are not binarized, which proves beneficial, cf. Sec. V-D; c) one can train here on average with 166 instances per class, while only an exemplar classifier is trainable in the case of writer identification.

TABLE VI: Comparison with state-of-the-art evaluated on the CLaMM16 test set. The numbers for the first four rows are taken from [40].

| Method | TOP-1 |
|---|---|
| DeepScript | 76.5 |
| FRDC-OCR | 79.8 |
| NNML | 83.8 |
| FAU | 83.9 |
| Cl-S + SVM | **84.1** |

## VI. Conclusion

We have presented a simple method for deep feature learning using cluster memberships as surrogate classes for local extracted image patches. The main advantage is that no training labels are necessary. All necessary training parameters have been evaluated thoroughly. We show that this approach outperforms supervised surrogate classes and traditional features in the case of writer identification and writer retrieval. The method achieves also comparable results to other methods on the task of classification of script types.

As a secondary result, we found that binarized images are preferable to grayscale versions for the training of our proposed feature learning process. In the future, we want to investigate this further, e. g., by evaluating only single handwritten lines instead of full paragraphs to investigate the influence of inter-linear spaces. Activations from other layers than the penultimate one are also worth to be examined. Another idea relates to the use of the last neural network layer, i. e., the predicted cluster membership for each patch. Since VLAD encoding relies on cluster memberships, this could be directly incorporated in the pipeline.

## References


[1] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "CNN features off-the-shelf: An astounding baseline for recognition," in *2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, jun 2014, pp. 512–519. 1

[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Advances In Neural Information Processing Systems 25*. Curran Associates, Inc., 2012, pp. 1097–1105. 1

[3] F. Wahlberg, T. Wilkinson, and A. Brun, "Historical manuscript production date estimation using deep convolutional neural networks," in *2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, Shenzhen, oct 2016, pp. 205–210. 1

[4] M. Jaderberg, A. Vedaldi, and A. Zisserman, "Deep features for text spotting," in *Computer Vision – ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part IV*. Cham: Springer International Publishing, 2014, pp. 512–528. 1

[5] T. Bluche, H. Ney, and C. Kermorvant, "Feature extraction with convolutional neural networks for handwritten word recognition," in *2013 12th International Conference on Document Analysis and Recognition*, Buffalo, aug 2013, pp. 285–289. 1

[6] V. Christlein, D. Bernecker, F. Hönig, A. Maier, and E. Angelopoulou, "Writer identification using GMM supervectors and exemplar-SVMs," *Pattern Recognition*, vol. 63, pp. 258–267, 2017. 1, 2, 3, 5

[7] V. Christlein, D. Bernecker, A. Maier, and E. Angelopoulou, "Offline writer identification using convolutional neural network activation features," in *Pattern Recognition: 37th German Conference, GCPR 2015, Aachen, Germany, October 7-10, 2015, Proceedings*. Springer International Publishing, 2015, vol. 9358, pp. 540–552. 1, 2, 3, 4, 5

[8] S. Fiel and R. Sablatnig, "Writer identification and retrieval using a convolutional neural network," in *Computer Analysis of Images and Patterns: 16th International Conference, CAIP 2015, Valletta, Malta, September 2-4, 2015, Proceedings, Part II*. Cham: Springer International Publishing, 2015, pp. 26–37. 1, 2, 3, 4

[9] L. Xing and Y. Qiao, "Deepwriter: A multi-stream deep CNN for text-independent writer identification," in *2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, Shenzhen, oct 2016, pp. 584–589. 1, 2, 3

[10] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, nov 2004. 1, 2

[11] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, jun 2016, pp. 770–778. 1, 3

[12] M. Bulacu and L. Schomaker, "Automatic handwriting identification on medieval documents," in *14th International Conference on Image Analysis and Processing (ICIAP 2007)*, Modena, sep 2007, pp. 279–284. 2

[13] A. Brink, J. Smit, M. Bulacu, and L. Schomaker, "Writer identification using directional ink-trace width measurements," *Pattern Recognition*, vol. 45, no. 1, pp. 162–171, jan 2012. 2

[14] S. He and L. Schomaker, "Delta-n hinge: Rotation-invariant features for writer identification," in *2014 22nd International Conference on Pattern Recognition*, Stockholm, aug 2014, pp. 2023–2028. 2

[15] A. Nicolaou, A. D. Bagdanov, M. Liwicki, and D. Karatzas, "Sparse radial sampling LBP for writer identification," in *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*, Nancy, aug 2015, pp. 716–720. 2

[16] S. Fiel and R. Sablatnig, "Writer identification and writer retrieval using the Fisher vector on visual vocabularies," in *Document Analysis and Recognition (ICDAR), 2013 12th International Conference on*, Washington DC, aug 2013, pp. 545–549. 2, 5, 6

[17] V. Christlein, D. Bernecker, and E. Angelopoulou, "Writer identification using VLAD encoded contour-zernike moments," in *Document Analysis and Recognition (ICDAR), 2015 13th International Conference on*, Nancy, aug 2015, pp. 906–910. 2, 3, 5, 6

[18] S. He, M. Wiering, and L. Schomaker, "Junction detection in handwritten documents and its application to writer identification," *Pattern Recognition*, vol. 48, no. 12, pp. 4036–4048, 2015. 2

[19] V. Christlein, D. Bernecker, F. Hönig, and E. Angelopoulou, "Writer identification and verification using GMM supervectors," in *Applications of Computer Vision (WACV), 2014 IEEE Winter Conference on*, mar 2014, pp. 998–1005. 2

[20] R. Jain and D. Doermann, "Combining local features for offline writer identification," in *2014 14th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, Heraklion, sep 2014, pp. 583–588. 2

[21] S. He, P. Samara, J. Burgers, and L. Schomaker, "Historical document dating using unsupervised attribute learning," in *12th IAPR Workshop on Document Analysis Systems*, no. April, 2016, pp. 36–41. 2

[22] A. Dosovitskiy, P. Fischer, J. T. Springenberg, M. Riedmiller, and T. Brox, "Discriminative unsupervised feature learning with exemplar convolutional neural networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 9, pp. 1734–1747, 2016. 2

[23] C. Huang, C. C. Loy, and X. Tang, "Unsupervised learning of discriminative attributes and visual representations," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, jun 2016, pp. 5175–5184. 2

[24] M. Paulin, J. Mairal, M. Douze, Z. Harchaoui, F. Perronnin, and C. Schmid, "Convolutional patch representations for image retrieval: An unsupervised approach," *International Journal of Computer Vision*, vol. 121, no. 1, pp. 149–168, 2016. 2, 3

[25] R. Arandjelovic and A. Zisserman, "Three things everyone should know to improve object retrieval," in *2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Providence, jun 2012, pp. 2911–2918. 3

[26] D. Sculley, "Web-scale k-means clustering," in *World Wide Web, 19th International Conference on*, ser. WWW '10. New York: ACM, apr 2010, pp. 1177–1178. 3

[27] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *Computer Vision – ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV*. Springer International Publishing, 2016, pp. 630–645. 3

[28] A. Babenko and V. Lempitsky, "Aggregating local deep features for image retrieval," in *2015 IEEE International Conference on Computer Vision (ICCV)*, vol. 11-18-Dece, Boston, MA, dec 2015, pp. 1269–1277. 3, 5

[29] H. Jégou, F. Perronnin, M. Douze, J. Sánchez, P. Pérez, and C. Schmid, "Aggregating local image descriptors into compact codes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 9, pp. 1704–1716, sep 2012. 3

[30] Y. Gong, L. Wang, R. Guo, and S. Lazebnik, "Multi-scale orderless pooling of deep convolutional activation features," in *Computer Vision – ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part VII*. Springer International Publishing, 2014, vol. 8695, pp. 392–407. 3

[31] R. Arandjelovic and A. Zisserman, "All about VLAD," in *2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Portland, jun 2013, pp. 1578 – 1585. 3, 6

[32] X. Peng, L. Wang, X. Wang, and Y. Qiao, "Bag of visual words and fusion methods for action recognition: Comprehensive study and good practice," *Computer Vision and Image Understanding*, vol. 150, pp. 109–125, may 2015. 3

[33] H. Jégou and O. Chum, "Negative evidences and co-occurences in image retrieval: The benefit of PCA and whitening," in *Computer Vision – ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part II*. Springer Berlin Heidelberg, oct 2012, pp. 774–787. 3

[34] E. Spyromitros-Xioufis, S. Papadopoulos, I. Kompatsiaris, G. Tsoumakas, and I. Vlahavas, "A comprehensive study over VLAD and product quantizationin large-scale image retrieval," *Multimedia, IEEE Transactions on*, vol. 16, no. 6, pp. 1713–1728, oct 2014. 3

[35] T. Malisiewicz, A. Gupta, and A. a. Efros, "Ensemble of exemplar-SVMs for object detection and beyond," in *IEEE International Conference on Computer Vision (ICCV)*, Barcelona, nov 2011, pp. 89–96. 3

[36] N. Crosswhite, J. Byrne, O. M. Parkhi, C. Stauffer, Q. Cao, and A. Zisserman, "Template adaptation for face verification and identification," *CoRR*, vol. abs/1603.0, 2016. 3

[37] J. Zepeda and P. Pérez, "Exemplar SVMs as visual feature encoders," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 07-12-June, Boston, jun 2015, pp. 3052–3060. 3

[38] T. Kobayashi, "Three viewpoints toward exemplar SVM," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, jun 2015, pp. 2765–2773. 3

[39] S. Fiel, F. Kleber, M. Diem, V. Christlein, G. Louloudis, N. Stamatopoulos, and B. Gatos, "ICDAR 2017 competition on historical document writer identification (Historical-WI)," in *2017 14th International Conference on Document Analysis and Recognition*, Kyoto, Japan, nov 2017. 4

[40] F. Cloppet, V. Eglin, V. C. Kieu, D. Stutzmann, N. Vincent, V. Églin, V. C. Kieu, D. Stutzmann, and N. Vincent, "ICFHR2016 competition on the classification of medieval handwritings in latin script," in *2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, Shenzhen, oct 2016, pp. 590–595. 4, 6

[41] J. Sánchez, F. Perronnin, T. Mensink, and J. Verbeek, "Image classification with the Fisher vector: Theory and practice," *International Journal of Computer Vision*, vol. 105, no. 3, pp. 222–245, 2013. 5, 6

[42] W. M. Campbell, D. E. Sturim, and D. A. Reynolds, "Support vector machines using GMM supervectors for speaker verification," *Signal Processing Letters, IEEE*, vol. 13, no. 5, pp. 308–311, may 2006. 5