Improving Handwritten Word Synthesis for Annotation-free Word Spotting

Fabian Wolf
Department of Computer Science
TU Dortmund University
44227 Dortmund, Germany
fabian.wolf@cs.tu-dortmund.de

Kai Brandenbusch
Department of Computer Science
TU Dortmund University
44227 Dortmund, Germany
kai.brandenbusch@cs.tu-dortmund.de

Gernot A. Fink
Department of Computer Science
TU Dortmund University
44227 Dortmund, Germany
gernot.fink@cs.tu-dortmund.de

***Abstract*—*Annotation-free* word spotting aims at retrieving relevant word images from a document collection without the need of a manually labeled training dataset. As annotated data is usually scarce in the application scenarios of a word spotting system, transfer learning and *annotation-free* methods became increasingly popular. One possibility to alleviate the annotation problem is to train on synthetically generated word images. Therefore, a common approach is to render word images from electronic fonts and to vary the synthesis parameters randomly. In this work, we show that an *annotation-free* word spotting method benefits from an adapted synthesis procedure. We investigate the influence of the choice of the underlying vocabulary and the combination of synthesis and data augmentation. Furthermore, we present a method to adapt the style of the synthesized word images to the target dataset. We evaluate the proposed changes to the synthesis procedure on three benchmark datasets and improve performances considerably.**

I. Introduction

Information retrieval from digital document collections is a challenging problem. Especially when it comes to historic document image collections, traditional recognition based approaches, such as optical character recognition, may not offer satisfactory results. In these cases, word spotting is a viable alternative [1]. Instead of providing a fixed recognition result, a word spotting system returns a ranked list of most probable occurrences relevant to a query word. As the result constitutes a list of different alternatives, interpretation remains with the user, which is often desired from the perspective of a historian.

The field of word spotting on historic document collections was strongly influenced by the uprise of neural networks and a multitude of neural methods emerged [2]–[4]. While being an extremely powerful tool, all of these approaches share a common drawback. As they rely on a neural network trained in a fully-supervised fashion, a large quantity of labeled data is required. This contradicts the application scenario, which is the first exploration of a so far unknown document collection, where it can be assumed that no set of annotated, representative training material is available.

Even though being less performant, methods not relying on any training material recently received more attention by the research community [5], [6]. These more traditional, feature based approaches do not employ any machine learning techniques and therefore they do not achieve competitive performances compared to training based methods. While the requirement of labeled training data limits application, the need of a preparatory training phase usually does not. Therefore, the crucial question w.r.t. the application to historic document collections is not whether a method requires training, but rather if it requires manually created annotations.

As shown in [7], an *annotation-free* method does not necessarily need to neglect the use of training and machine learning techniques. In context of the analysis of handwritten documents, synthetically generated word images are often combined with weakly-supervised or with transfer learning techniques [3], [7]–[9]. The automatic generation of word images potentially offers the possibility to generate an infinite number of labeled training samples. This improves performance but still does not close the gap between *annotation-free* and fully-supervised approaches.

In this work, we show that an *annotation-free* word spotting method, which relies on a synthetic dataset, benefits from adapting the synthesis procedure w.r.t. the target dataset. Fig. 1 presents an overview of the system and the different synthesis components. We investigate the selection of the underlying vocabulary and augmentation methods. We show that a vocabulary which strongly intersects with the lexicon of the target dataset improves performances. This holds also true for data augmentation although a possibly infinite number of training samples can be synthesized. We furthermore investigate the question whether the style of the synthesized images can be adapted to the target dataset without requiring annotated data. Based on an initial synthetic dataset, we predict a selection of fonts and slant angles, which are then used to generate an adapted training dataset w.r.t. the target domain.

II. Related Work

A. Word Spotting

The problem of word spotting has been of great interest for the document image research community and a large number of different approaches emerged [1]. *Segmentation-based* methods [2], [3], [5] require a preparatory, independent segmentation step. In contrast, *segmentation-free* approaches [4], [6] work on entire document images [4], [6]. Another distinction is made on basis of the query representation. *Query-by-example* (QbE) denotes the case where the query is represented by an exemplar word image. *Query-by-string*

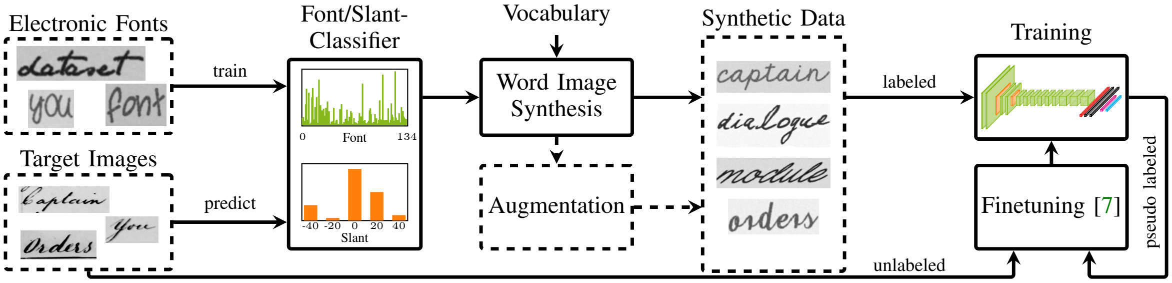


Fig. 1: *Annotation-free* training scheme for the TPP-PHOCNet: Word images are synthesized based on font and slant angle prediction for the target dataset. An initial model is trained with generated labeled images and finetuned in a weakly-supervised fashion on the unlabeled target dataset.

(QbS) uses a string representation, which is often a desideratum as the user does not need to manually search for an exemplary occurrence of the query word. As the availability of training data is often crucial especially in the context of historic document collections, methods can be distinguished in *annotation-free* [5]–[7] and *annotation-based* [2]–[4].

A large number of diverse models has been applied to the problem of word spotting. Methods range from bag-of-feature approaches and support vector machines to sequence models such as hidden Markov models or recurrent neural networks [1]. Today, the predominant models for word spotting are based on neural networks, showing exceptional performances on most benchmark datasets [2]–[4]. As these methods heavily rely on labeled data, their application is often limited by the availability of an annotated training set. This motivates the investigation of traditional feature based approaches [5], [6]. By designing a feature representation that is capable of encoding the visual appearance of handwriting, a direct application is possible and no annotated data is required.

An alternative to heuristic methods based on feature design that still limits the required amount of manually created annotations is offered by weakly-supervised and by transfer learning approaches. In [3] and [8] a synthetic dataset is used to pretrain a neural network. Combining the synthetic dataset with a limited number of annotated samples greatly improves performances. Based on the same synthetic dataset, [7] proposed an entirely *annotation-free* word spotting approach. The method is based on an attribute CNN similar to [2], which is initially trained on the synthetic dataset proposed in [3]. Word recognition is performed for every sample of the unlabeled target dataset with the initial model. This is possible as the attribute vectors estimated by the attribute CNN may be mapped to a previously defined lexicon by a nearest neighbour search in the attribute space. The recognition result is then considered a pseudo label for the formerly unlabeled training sample. A subset of the pseudo labeled samples is selected based on a confidence measure [10]. Training is then continued with the respective pseudo labels to further improve the model. As shown in [7], iteratively reestimating the pseudo labels and finetuning the model leads to significant performance improvements. The only additional information required for this *annotation-free* method is an approximated lexicon that does not need to fit the target dataset precisely.

B. Handwritten Word Synthesis

Neural networks strongly changed the field of computer vision, and they have been proven to give high performances on diverse tasks. These performance gains can be explained to a large extent by the availability of huge annotated training datasets of high quality. As these datasets need to be created manually and are often not available, several works investigate the use of automatically generating labeled images [11], [12]. These synthetic datasets are created with the aim to represent the variations observed in real data.

Cursive handwriting follows a distinctive set of rules making it a suitable domain to generate synthetic images for. Krishnan and Jawahar proposed a synthetic dataset of handwritten word images in [3]. Therefore, they rendered word images from a set of electronic fonts that resemble handwriting. In order to increase the dataset's variability, style defining parameters such as slant angle or stroke width are varied randomly. This synthesis approach is used in various methods to train neural networks. Synthetic data can be combined with manually labeled training samples to achieve state-of-the-art word spotting performances as shown in [3] and [8]. In [7], an initial word spotting model is trained on a synthetic dataset which is then adapted to the target domain in a weakly-supervised fashion with pseudo labeling. Another application of the synthesis approach was presented in [9]. The authors trained a word recognizer on a fully synthetically generated dataset. To improve recognition results, the model is trained on the synthetic dataset with an adversarial domain adaptation approach exploiting unlabeled data from the target domain.

In contrast to the designed approach of [3], generative adversarial networks (GANs) constitute a learned model to synthesize images. GANs have been also investigated to generate images of cursive handwriting [13], but no significant performance gains were reported when using the generated images as data augmentation or as a training set.

III. Methods

Our word spotting system is based on the TPP-PHOCNet proposed in [2] and uses the same hyperparameterization.

An initial model that does not rely on manually annotated training samples can be derived by training on a synthetic dataset. In order to achieve state-of-the-art *annotation-free* word spotting performances, the initial model is finetuned following the weakly-supervised training procedure presented in [7]. In this work, we consider the generation of an adapted synthetic dataset and its combination with different augmentation techniques. Sec. III-A discusses the general synthesis approach, which is then combined with the augmentation methods presented in Sec. III-B. In Sec. III-C, we propose an approach that allows to adapt the style of the synthetically generated word images to a target dataset.

A. Word Synthesis

Our synthesis approach follows the methods proposed in [3]. We select a set of 134 electronic fonts that resemble handwriting. Given a string, a grey scaled word image is rendered. The appearance of the word image is defined by a limited number of parameters, namely slant and skew angle, stroke width and the distance between characters (kerning). In order to generate a synthetic dataset, we generate a fixed number of samples for each word in a predefined vocabulary and randomly vary the style defining parameters. We randomly select a slant angle of $[-40, -20, 0, 20, 40]$ and vary the skew angle between -2 and 2 degrees. The interval of the stroke width lies within the limits of $[1, 3]$ pixels. As we aim our synthesis at cursive handwriting only small kerning values of zero or one are applied. After rendering, we randomly select a value for the background and foreground pixels from uniform distributions with limits of $[180, 255]$ and $[0, 100]$. Finally, a Gaussian filter is used to smooth the generated word image.

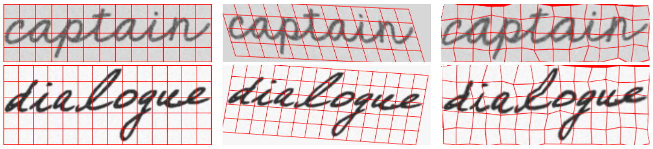
B. Data Augmentation

While word image synthesis from electronic fonts allows the creation of a potentially infinite amount of training samples, their variability is limited to the previously described parameters. In order to tackle the limited variability of training data, we apply different methods of data augmentation. Fig. 2 shows an example of the augmentation methods used in this work.

As a basic approach, we rescale the image randomly by a factor in $[1, 2)$ to simulate the size variability of handwriting. Additionally, we employ more advanced augmentation techniques to mimic other variations of handwriting such as slant, shear or rotation. Those can be achieved by applying random affine transformations to the word images.

For this work we utilize the transformation described in [2]. For every word image we select three control points at fixed relative coordinates. These points are perturbed by multiplying each coordinate value with a random factor drawn from a uniform distribution with limits $[0.9, 1.1]$. The transformation is then defined by the homography to obtain the perturbed points from the initial ones.

While affine transformations are limited to variations of the entire image, the transformation proposed by Wigington et al. [14] mimics variations in the writing style of single characters in a word using a random warp grid distortion. A regular grid of control points is placed on the image with a fixed interval. Then each control point is perturbed in x and y direction by adding a randomly sampled value from a normal distribution with zero mean. Finally the image is warped according to the perturbed control points. For the regular grid, we choose an interval of 25 pixels and a standard deviation of 1.7 pixels for sampling from the normal distribution.



(a) Synthetic image (b) Homography-Aug. (c) Grid-Aug.

Fig. 2: Different augmentations applied to a synthetic sample.

C. Style Adaptation

Considering the synthesis parameters discussed in Sec. III-A, the selection of a font and slant angle strongly defines what can be considered the style of a writer or document. Instead of randomly choosing these parameters, we propose to predict them for a target dataset. In a first step, a synthetic dataset is generated with uniform distributions. This dataset is then used to train two neural networks predicting the font and slant angle of a given word image.

Font classification has been of interest for the document image analysis community, especially for optical character recognition, as it allows to select designated models depending on the predicted font. We adapt the approaches presented in [15] and [16] and use a convolutional neural network to classify the font of a given word image. Each font used in the synthesis procedure is considered a class. As a backbone architecture, we use the residual network ResNet50 [17] with 134 output neurons corresponding to the set of fonts. In order to not rescale the differently sized word images, we remove the batch normalization layers and we train the network with pseudo batches, meaning that each sample is processed individually but weights are updated after a batch of training images. The network is trained for $40,000$ iterations with a batch size of 64 using ADAM optimization and cross-entropy loss. The same approach is taken for predicting slant angles. Each angle used in the synthesis procedure is considered a class, resulting in a ResNet50 architecture with five output neurons. Training follows the same hyperparameters as used for training the font predictor network.

Both networks are exclusively trained on synthetically generated word images. In order to adapt the synthesis procedure to an unknown dataset, we predict a font and slant angle for each sample from the target dataset. The histograms of predicted fonts and slant angles constitute an approximate distribution of which fonts and slant angles result in visual similar word images. An adapted synthetic dataset is then generated by replacing the uniform distributions of the synthesis procedure described in Sec. III-A with the distributions approximated by the font and slant angle predictors.

TABLE I: Intersection between synthesis vocabulary and actual lexicon in percent.

Vocabulary	Size	GW	IAM	BT15
IIITHWS [3]	10k	8.7	9.7	9.5
IIITHWS [3]	90k	81.4	87.4	84.6
Ours	10k	67.1	55.3	51.5

TABLE II: Performances after training on datasets with different vocabularies. Results reported as mAP [%].

Vocabulary	Size	GW		IAM		BT15	
		QbE	QbS	QbE	QbS	QbE	QbS
IIITHWS [3]	10k	54.4	63.7	28.9	56.5	46.3	-
IIITHWS [3]	90k	54.9	64.7	27.7	54.7	41.8	-
Ours	10k	62.8	71.1	33.5	59.6	48.5	-
Lexicon	1k - 9k	**64.5**	**74.4**	**34.1**	**60.4**	**50.2**	-

IV. Experiments

In our experiments, we use the synthesized datasets for training a TPP-PHOCNet [2]. The TPP-PHOCNet is trained with ADAM optimization, a batch size of 10 and binary cross entropy to predict a pyramidal histogram of characters (PHOC) with levels $[2, 3, 4, 5]$ for an input image. We observe that our networks converge rather fast and stop training after one epoch. *Segmentation-based* word spotting is then performed by ranking all images in the test set w.r.t. their cosine similarity to the query word representation.

Following the standard approach from the literature, we measure performance with mean average precision (mAP) [1]. We evaluate our models on well established benchmark datasets, which are George Washington (GW) [18], the IAM database [19] and the 2015 Bentham dataset (BT15) [20]. For George Washington and IAM, we follow the protocols proposed in [21]. On the Bentham dataset, we stick to the training-free *segmentation-based* protocol used in the respective competition. Note that our model is capable to perform *query-by-string* on Bentham and we do not report performance numbers only due to the lack of a protocol and test annotations.

A. Synthesis Vocabulary

In a first experiment, we evaluate the influence of the vocabulary used for synthesizing word images. We compare the 10K and 90K vocabularies of the IIITHWS dataset [3] to our own and a presumably optimal vocabulary. Our own vocabulary differs from the approach of [3] as it is not based on the Hunspell dictionary but contains the most common $10,000$ English words based on the Exquisite Corpus [22] which was generated from multiple modern sources of text. In order to obtain an upper bound of a vocabulary perfectly representing the target dataset, we derive lexica from the dataset annotations. In case of Bentham, we use the annotations of the line-based competition track, as the considered dataset does not provide any labels. Note, that in a practical application of an *annotation-free* word spotting system the perfect vocabulary is not available.

TABLE III: Comparison of different augmentation techniques. Results reported as mAP [%].

Method	GW		IAM		BT15	
	QbE	QbS	QbE	QbS	QbE	QbS
None	60.5	69.5	26.7	51.6	47.3	-
Rescale	62.8	71.1	33.5	59.6	48.5	-
Homography	58.8	69.8	38.5	63.1	**55.6**	-
Grid	**69.2**	**72.3**	**39.0**	**64.1**	54.1	-

The intersection between the vocabularies and the lexica of the benchmark datasets is shown in Tab. I. For training the TPP-PHOCNet, we generate one million word images from each vocabulary with an equal number of images per word. We observe that training converges for all vocabularies for less than one epoch, leading us to the conclusion that generating more word images does not improve performances also for bigger vocabularies. As the PHOC only represents the presence and absence of characters in defined parts of a word, one could assume that the choice of vocabulary has only little influence when learning to predict the PHOC as long as all attributes are well represented in the training data. In our experiments, we refute this assumption showing that the performance of the trained models varies by a large margin depending on the choice of vocabulary cf. Tab. II.

The IIITHWS 10K only has a very little intersection of less than $10\,\%$ with every dataset. Increasing the vocabulary to $90,000$ words (IIITHWS 90K) leads to a significant intersection with all lexica. While only leading to a minimal improvement in performance for George Washington, the mAP decreases for Bentham and IAM. Thus, enlarging the intersection but adding many irrelevant words w.r.t. the datasets does not improve the PHOC prediction. In contrast, when selecting a smaller (10k words) vocabulary more carefully we achieve results comparable with the presumably optimal choice of a vocabulary with an intersection of 100%.

B. Augmentation

Besides varying the synthesis parameters, we also implemented augmentation techniques as described in Sec. III-B. The influence of the different augmentations during training is shown in Tab. III.

By randomly rescaling the training images, we were able to increase the performances especially for the IAM database. Even though we randomly vary the synthesis parameters while rendering the font images, additional variation introduced by the homography transformation improves performances on IAM and Bentham. Both datasets have multiple authors and thus high variation in writing style, which is better represented in the training data using random affine transformations. In contrast, the George Washington dataset shows a very homogeneous writing style and augmenting entire word images leads to a slightly decreased performance.

Both augmentation methods change the entire word image in a uniform way. However, naturally occurring variations in handwriting affect single characters within a word in a

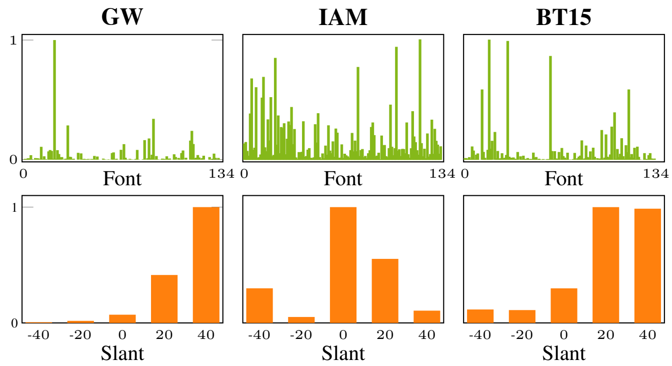


Fig. 3: Distributions of predicted fonts (green) and slant angles (orange). For comparability, each histogram is normalized such that the maximal frequency corresponds to one.

slightly different way even for a single writer. Simulating this variations by the grid augmentation results in better performances for all three datasets emphasizing the importance of character level augmentation. For this reason, we apply random warp grid distortion in further experiments.

C. Style Adaptation

In order to adapt the writing style of the synthesized word images to a target dataset, we follow the approach presented in Sec. III-C. Based on our proposed vocabulary, we create a dataset of one million images with randomly distributed synthesis parameters. A font and slant predictor network is trained with the previously discussed hyperparameterization. We test the classification performance on another $100,000$ generated images, giving us an accuracy of 95.6% for fonts and 96.1% for slant angles.

Each sample of the target datasets is classified w.r.t. a font and slant angle. Fig. 3 top row shows the distributions of font classes for the considered datasets. We observe that the distributions resemble the different characteristics of the datasets. Being a single writer dataset with a rather homogeneous appearance, a single font dominates the distribution for the George Washington dataset. For the IAM database, numerous fonts are frequently predicted which can be explained by the high number of contributing writers. The Bentham collection was written by a few writers and is therefore not as homogeneous as the George Washington dataset and far from being as diverse as the IAM database. Again this characteristic is resembled by the distribution of predicted fonts.

While our approach seems to capture specific dataset characteristics it remains an open question, whether frequently predicted fonts are also visually similar to those occurring in the dataset. Fig. 4 shows three images from the real dataset for which we synthesized corresponding word images using the most and least frequently predicted fonts. Qualitatively, we observe that our approach favours fonts that have a similar style as the real dataset, validating the assumption that a high prediction frequency corresponds to visual similarity in style.

Predicting slant angles also results in plausible distributions, as shown in Fig. 3 bottom row. The George Washington and Bentham dataset are usually written with a rather strong tilt to the right, which results in a frequent prediction of a slant angle of 20 or 40 degrees. In contrast, the modern IAM dataset contains many writers with only minimal slant. These characteristics are captured by the resulting distribution.

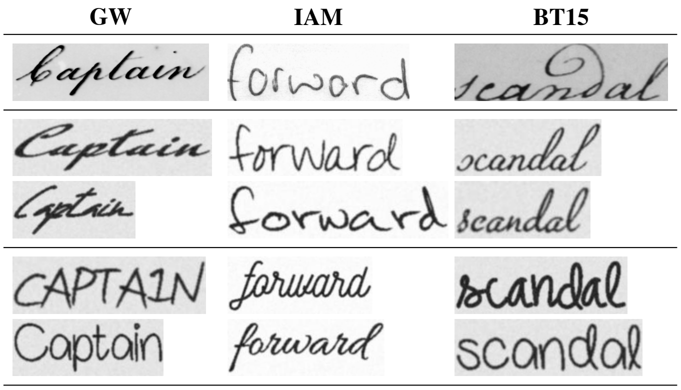


Fig. 4: Comparison of real with synthesized images. Top row shows original images. Second and third row images are rendered from the two most frequently predicted fonts. Bottom rows correspond to least predicted fonts.

In order to generate an adapted synthetic dataset, we replace the random selection of font and slant angle by sampling w.r.t. the previously derived distributions. We do not make any changes to the underlying vocabulary and we train a TPP-PHOCNet for one epoch on the adapted dataset applying grid augmentation. Even though the synthetic training data is generated for a specific dataset, performances decrease slightly on all three benchmarks, cf. Tab. IV. By using the predicted distributions, the synthesis procedure focuses on a smaller number of fonts, which removes variance from the generated dataset. As the rendering process is still quite limited and no font is able to precisely resemble any of the datasets, the initial model probably benefits from the higher variance following the random synthesis procedure.

Nonetheless, our goal is not to apply the model trained exclusively on synthetic data, but to use it as an initialization for the weakly-supervised training scheme proposed in [7]. Therefore, we finetune the initial model following the same hyperparameterization as [7] using the synthesis vocabulary as a lexicon for pseudo labeling and sigmoid activations as a confidence measure. Despite the fact that the initial model performs worse, the model trained on the adapted dataset still serves as a better performing initialization for the following weakly-supervised training scheme, cf. Tab. IV bottom half. We observe performance gains for all benchmarks except for *query-by-example* spotting on the IAM database. In general, we argue that style adaptation is more suitable for datasets with a limited number of writers and a distinctive style.

D. Comparison with the Literature

The considered *annotation-free* word spotting system strongly benefits from the changes made to the synthesis

TABLE IV: Evaluation of style adapted datasets with and without finetuning. Results reported as mAP [%].

Style Adaptation	Annotation-free Finetuning	GW		IAM		BT15	
		QbE	QbS	QbE	QbS	QbE	QbS
No	No	69.2	72.3	39.0	64.1	54.1	-
Yes	No	59.6	66.8	35.3	61.6	52.2	-
No	Yes	87.9	88.6	**68.6**	85.4	82.1	-
Yes	Yes	**89.2**	**91.0**	67.5	**85.9**	**82.8**	-

TABLE V: Comparison with the literature. Results reported as mAP [%]. Methods marked with (*) require annotated data.

Method	GW		IAM		BT15	
	QbE	QbS	QbE	QbS	QbE	QbS
Ours	89.2	91.0	67.5	85.9	**82.8**	-
Wolf et al. [7]	83.2	82.3	62.6	81.0	76.3	-
Retsinas et al. [5]	77.1	-	28.1	-	58.4	-
Zagoris et al. [24]	69.2	-	-	-	44.0	-
Sfikas et al. [23]	58.3	-	13.2	-	41.5	-
Sudholt et al.* [2]	97.9	**97.9**	85.5	93.4	-	-
Krishnan et al.* [3]	**98.2**	-	**92.4**	**94.0**	-	-

procedure and compares quite favorably to the state of the art. Tab. V compares our model to other results from the literature. The most direct comparison can be made to [7] as the only differences to our work is the adapted synthetic dataset and the inclusion of grid augmentation. The improvements from the adapted synthesis result in a performance gain and the state-of-the-art feature based approaches presented in [5], [23], [24] are clearly outperformed. Our proposed synthesis approach reduces the performance gap between *annotation-free* and the best performing, fully-supervised deep learning approaches, which come at the cost of requiring a representative, manually labeled training dataset.

V. Conclusions

In this work, we show that *annotation-free* word spotting performances based on synthetic data may be improved by taking the target dataset into consideration during synthesis. Despite the character based approach, which offers some independence between different attributes, the synthesis vocabulary strongly influences performances. In this case, it is beneficial to employ a vocabulary that highly overlaps with the dataset's lexicon without introducing to many distractors. The synthesis approach offers the possibility to generate an infinite number of labeled training samples. Nonetheless, our experiments show that a combination with traditional data augmentation still improves performances, indicating that the simple synthesis approach is not able to mimic all possible variations present in handwriting. Finally, we propose a method that allows to adapt the style of the synthetic dataset to the target dataset, without the requirement of any manually labeled samples. We are able to show that the derived model can be successfully used as an initialization for a weakly-supervised training scheme, giving state-of-the-art *annotation-free* word spotting performances.

References

[1] A. P. Giotis, G. Sfikas, B. Gatos, and C. Nikou, "A survey of document image word spotting techniques," *Pattern Recognition*, vol. 68, pp. 310–332, 2017.

[2] S. Sudholt and G. A. Fink, "Attribute CNNs for word spotting in handwritten documents," *IJDAR*, vol. 21, no. 3, pp. 199–218, 2018.

[3] P. Krishnan and C. V. Jawahar, "Hwnet v2: an efficient word image representation for handwritten documents," *IJDAR*, vol. 22, no. 4, pp. 387–405, 2019.

[4] A. H. Toselli, V. Romero-Gomez, J. Sánchez, and E. Vidal-Ruiz, "Making two vast historical manuscript collections searchable and extracting meaningful textual features through large-scale probabilistic indexing," in *ICDAR*, Sydney, NSW, Australia, 2019, pp. 108–113.

[5] G. Retsinas, G. Louloudis, N. Stamatopoulos, and B. Gatos, "Efficient learning-free keyword spotting," *TPAMI*, vol. 41, no. 7, pp. 1587–1600, 2019.

[6] E. Vats, A. Hast, and A. Fornés, "Training-free and segmentation-free word spotting using feature matching and query expansion," in *ICDAR*, Sydney, NSW, Australia, 2019.

[7] F. Wolf and G. A. Fink, "Annotation-free learning of deep representations for word spotting using synthetic data and self labeling," *CoRR*, vol. abs/2003.01989, 2020.

[8] N. Gurjar, S. Sudholt, and G. A. Fink, "Learning deep representations for word spotting under weak supervision," in *DAS*, Vienna, Austria, 2018, pp. 7–12.

[9] L. Kang, M. Rusiñol, A. Fornés, P. Riba, and M. Villegas, "Unsupervised writer adaptation for synthetic-to-real handwritten word recognition," *CoRR*, vol. abs/1909.08473, 2019, accepted to WACV 2020.

[10] F. Wolf, P. Oberdiek, and G. A. Fink, "Exploring confidence measures for word spotting in heterogeneous datasets," in *ICDAR*, Sydney, NSW, Australia, 2019, pp. 583–588.

[11] A. Rozantsev, V. Lepetit, and P. Fua, "On rendering synthetic images for training an object detector," *Comput. Vis. Image Underst.*, vol. 137, pp. 24–37, 2015.

[12] G. Ros, L. Sellart, J. Materzynska, D. Vázquez, and A. M. López, "The SYNTHIA dataset: A large collection of synthetic images for semantic segmentation of urban scenes," in *CVPR*, Las Vegas, NV, USA, 2016, pp. 3234–3243.

[13] E. Alonso, B. Moysset, and R. O. Messina, "Adversarial generation of handwritten text images conditioned on sequences," in *ICDAR*, Sydney, NSW, Australia, 2019, pp. 481–486.

[14] C. Wigington, S. Stewart, B. L. Davis, B. Barrett, B. L. Price, and S. Cohen, "Data augmentation for recognition of handwritten words and lines using a CNN-LSTM network," in *ICDAR*, 2017, pp. 639–645.

[15] C. Tensmeyer, D. Saunders, and T. Martinez, "Convolutional neural networks for font classification," in *ICDAR*, 2017, pp. 985–990.

[16] V. Storchan and J. Beauschene, "Data augmentation via adversarial networks for optical character recognition/conference submissions," in *ICDAR*, pp. 184–189.

[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, Las Vegas, NV, USA, 2016, pp. 770–778.

[18] T. M. Rath and R. Manmatha, "Word spotting for historical documents," *IJDAR*, vol. 9, no. 2-4, pp. 139–152, 2007.

[19] U. Marti and H. Bunke, "The IAM-database: an english sentence database for offline handwriting recognition," *IJDAR*, vol. 5, no. 1, pp. 39–46, 2002.

[20] J. Puigcerver, A. Toselli, and E. Vidal, "ICDAR2015 competition on keyword spotting for handwritten documents," in *ICDAR*, Nancy, France, 2015, pp. 1176–1180.

[21] J. Almazán, A. Gordo, A. Fornés, and E. Valveny, "Word spotting and recognition with embedded attributes," *TPAMI*, vol. 36, no. 12, pp. 2552–2566, 2014.

[22] R. Speer, J. Chin, A. Lin, S. Jewett, and L. Nathan, "Luminosoinsight/wordfreq: v2.2," Oct. 2018. [Online]. Available: https://doi.org/10.5281/zenodo.1443582

[23] G. Sfikas, G. Retsinas, and B. Gatos, "Zoning aggregated hypercolumns for keyword spotting," in *ICFHR*, Niagara Falls, NY, USA, 2016, pp. 283–288.

[24] K. Zagoris, I. Pratikakis, and B. Gatos, "Unsupervised word spotting in historical handwritten document images using document-oriented local features," *IEEE Trans. on Image Processing*, vol. 26, no. 8, pp. 4032–4041, 2017.