applied sciences

MDPI

Article

Automatic CNN-Based Arabic Numeral Spotting and Handwritten Digit Recognition by Using Deep Transfer Learning in Ottoman Population Registers

Yekta Said Can * and M. Erdem Kabadayı

College of Social Sciences and Humanities, Koç University, Rumelifeneri Yolu, 34450 Sarıyer, Istanbul, Turkey; mkabadayi@ku.edu.tr

* Correspondence: ycan@ku.edu.tr

Received: 3 July 2020; Accepted: 3 August 2020; Published: 6 August 2020

Abstract: Historical manuscripts and archival documentation are handwritten texts which are the backbone sources for historical inquiry. Recent developments in the digital humanities field and the need for extracting information from the historical documents have fastened the digitization processes. Cutting edge machine learning methods are applied to extract meaning from these documents. Page segmentation (layout analysis), keyword, number and symbol spotting, handwritten text recognition algorithms are tested on historical documents. For most of the languages, these techniques are widely studied and high performance techniques are developed. However, the properties of Arabic scripts (i.e., diacritics, varying script styles, diacritics, and ligatures) create additional problems for these algorithms and, therefore, the number of research is limited. In this research, we first automatically spotted the Arabic numerals from the very first series of population registers of the Ottoman Empire conducted in the mid-nineteenth century and recognized these numbers. They are important because they held information about the number of households, registered individuals and ages of individuals. We applied a red color filter to separate numerals from the document by taking advantage of the structure of the studied registers (numerals are written in red). We first used a CNN-based segmentation method for spotting these numerals. In the second part, we annotated a local Arabic handwritten digit dataset from the spotted numerals by selecting uni-digit ones and tested the Deep Transfer Learning method from large open Arabic handwritten digit datasets for digit recognition. We achieved promising results for recognizing digits in these historical documents.

Keywords: numeral spotting; historical document analysis; convolutional neural networks; deep transfer learning; handwritten digit recognition

1. Introduction

Historical documents are valuable sources for analyzing historical, social, and economic perspectives of the past. In order to provide immediate access to researchers and to the public, digitization processes of these archives have been carried out in recent decades including non-European handwritten archival collections [1]. Nevertheless, especially during maintenance periods, access to these archives could be restricted. Information retrieval and extraction are only possible through the digitalization processes. Page segmentation, keyword, number and symbol spotting, optical character recognition (OCR) and handwritten text recognition (HTR) are among the most applied techniques for these documents [2].

In page segmentation, the document is analyzed by separating the image into different areas such as graphics, backgrounds, decorations, and texts via page segmentation algorithms [3]. Historical document layout analysis is more difficult when compared to modern document processing since there

are more issues to be dealt with: degrading documents, digitization errors, and different layout types, respectively [4]. Consequently, it is challenging to apply page segmentation on historical documents by using rule-based or projection-based methods [3]. Page segmentation can be applied before OCR, HTR and keyword spotting techniques in some cases that is why the page segmentation processes gain importance for the accurate digitization of historical manuscripts. The errors in the page segmentation process affect the output of these processes, which are used the digitalize the handwritten or printed manuscripts [2].

Keyword Spotting (KWS) is another widely used technique for information retrieval from historical documents. There are a lot of different types of keyword spotting. The keyword can be a word, symbol, or a numeral. Another widely known distinction is whether the spotting is done Query-by-Example/Query-by-String [5]. In QbE, the query is provided as a word image example, whereas, in QbS, it is provided as a character string. Other significant distinctions are training-based/training-free; i.e., whether the spotting technique requires or not to be trained on annotated images, and segmentation-based/segmentation-free; i.e., whether the spotting technique is applied to the whole page images or just to segmented images/parts of the whole page [5]. Usually, a training-based method decodes images and spots the most proper keyword position during training. Training-based keyword spotting methods are evaluated as more practical and they overcome multi-writers and multi-fonts issues [6].

Arabic scripts are widely adopted in manuscripts of different countries and cultures, e.g., Ottoman, Arabic, Urdu, Kurdish and Persian [7]. These scripts can be written in different ways, which complicates the page segmentation, keyword spotting, HTR and OCR processes. It is a cursive script in which combined letters form ligatures [7]. Moreover, the Arabic words can consist of dots and diacritics, which makes it even more difficult to extract information [7]. These properties might not cause problems for digit recognition since digits are isolated, but, when keyword spotting and handwritten text recognition algorithms are applied, they will create additional challenges.

Several methods have been proposed, and high identification accuracies are reported for the English handwritten digits [8,9]. Recently, researchers also proposed numeral spotting [10] and handwritten digit recognition systems for Arabic scripts on different datasets ([11–13]). These studies achieved accuracies above 90%. However, the used datasets are created recently, and they do not suffer from the mentioned problems of the historical documents.

In this study, we first automatically spotted the Arabic numerals from the very first series of population registers of the Ottoman Empire conducted in the mid-nineteenth century and recognized these numbers. The household numbers, registered individual ids and ages are written red in the studied documents. We implemented a red color filter to discriminate numerals from the document to take advantage of the structure of the registers. We further trained a CNN-based segmentation scheme for spotting these numerals. Our numeral spotting technique is both training-based and segmentation-based. In the second part, we formed a small Arabic digit dataset from the spotted numerals by selecting uni-digit ones and tested the Deep Transfer Learning (DTL) methods from the models trained in large open datasets for digit recognition. We also compared these results obtained by training and testing a system by using our dataset. We obtained promising results for recognizing Arabic digits in these historical documents.

We organized the rest of the paper as follows. The literature on historical document page segmentation, keyword spotting and Arabic digit recognition will be provided. We described the structure of the formed databases for spotting numerals and digit recognition in Section 3. Our numeral spotting technique and digit recognition method are described in Section 4. In Section 5, the experimental results and discussion are presented. We mention the conclusion and future works of this research in Section 6.

2. Related Works

Arabic document page segmentation has also been studied by using traditional machine learning (ML) techniques. Hesham et al. [7] proposed an automatic layout analysis scheme for Arabic manuscripts. They further appended a line segmentation support. Text and non-text areas were differentiated by using the Support Vector Machine (SVM) algorithm. They also identified words and lines.

Artificial Neural Networks were further tested on Arabic document layout analysis schemes. Bukhari et al. [14] differentiated the central body and the side manuscript by applying the Multilayer Perceptron (MLP) classifier. A dataset is created which includes 38 historical document images and they achieved 95% classification accuracy. Long Short Term Memory (LSTM) and CNN are employed for document page segmentation of scientific manuscripts written in English in [15,16]. Amer et al. developed a CNN-based document page segmentation scheme for Arabic newspapers and Arabic printed manuscripts. They obtained approximately 90% accuracy in detecting text and non-text areas. CNNs have also been employed for historical document layout analysis [2,3,17]. The page segmentation algorithms are important because they could be applied prior to keyword spotting, HTR and OCR techniques in some studies (as in our work) and, therefore, their performance is critical.

There are very few Arabic handwriting keyword spotting studies in the literature [6]. Some QbE studies ([18–20]) are proposed for the historical Arabic documents and used a matching method adjusted to the Arabic script. QbS approaches [21,22] used the HMM technique for keyword spotting in handwritten Arabic manuscripts. They were standard HMM KWS applications without taking the particular properties of the Arabic script into account. A spotting scheme is developed specifically for Arabic handwritten digits/symbols achieved an overall precision of 80% and 83.3% recall [10]. Another prominent keyword spotting research conducted on both historical Arabic dataset VML and George Washington datasets. Barakat et al. [23] applied a convolutional siamese network that uses two identical convolutional networks to rank the similarity between two word images. In this way, they developed a system which is more robust against different writing styles and is able to recognize out of vocabulary words.

After spotting the numerals, Arabic digits should be recognized for information retrieval from the historical manuscripts. Arabic digit recognition is a well-studied topic in the literature [13] (see Table 1). Melhaoui et al. proposed an Arabic digit recognition scheme that used multi-layer perceptron and K-nearest neighbor classifiers [24]. They run tests on the dataset include 600 Arabic digits with 200 testing images and 400 training images. They achieved 99% recognition accuracy on this small database. The HODA dataset was used for testing Persian (which is based on Arabic scripts) handwritten digit recognition systems in the literature [25–27]. Takruri et al. [28] proposed a three-level classifier that uses Support Vector Machine, Fuzzy C Means, and Unique Pixels for the classification of handwritten Arabic digits. They achieved 88% accuracy on the dataset containing 3510 images. Sawy et al. also achieved 88% accuracy by using CNN on the public ADBase dataset [13]. Kateeb et al. used the same dataset (ADBase) and applied the Dynamic Bayesian Network technique for digit recognition. They achieved 85.26% accuracy. Ashiquzzaman et al. achieved 97.3% accuracy by using MLP with appropriate activation and regularization functions on the public CMATERDB 3.3.1 Arabic handwritten digit dataset [29]. They further improved their system accuracy by using data augmentation and dropout to 99.4% [12]. However, as mentioned before, these studies were carried out on modern open datasets, and they did not need to alleviate the low-quality data issues of historical manuscripts. To the best of our knowledge, our study is the first to develop a CNN-based Arabic numeral spotting and handwritten digit recognition system for historical documents by using deep transfer learning methods.

Table 1. The comparison of our study with the Arabic handwritten digit recognition studies on different datasets.

Article	Dataset	Dataset Type	Method	Accuracy (%)
[13] [2017]	ADBase	Modern Arabic	CNN + LeNet50	88
[25] [2018]	HODA	Modern Persian	CNN + LeNet	97.38
[11] [2014]	ADBase	Modern Arabic	Dynamic Bayesian Network	85.26
[26] [2019]	HODA	Modern Persian	CNN + CapsNet	99.87
[27] [2017]	HODA	Modern Persian	CNN + AlexNet	99.44
[30] [2020]	CmaterDb3.3.1	Modern Arabic	CNN	99.76
[31] [2017]	CmaterDb3.3.1	Modern Arabic	Boltzmann Machine + CNN	98.59
[12] [2019]	CmaterDb3.3.1	Modern Arabic	CNN	99.4
Our Study [2020]	HODA	Modern Persian	CNN	99.47
Our Study [2020]	ADBase	Modern Arabic	CNN	99.34
Our Study [2020]	Ottoman Registers 1840s	Historical Arabic Scripts	CNN, Deep Transfer Learning from HODA and ADBase Datasets	80

3. Structure of the Ottoman Registers

In this research, we concentrated on the Nicaea district registers, NFS.d. 1411 and 1452. They are digitally available at the Turkish Presidency State Archives of the Republic of Turkey—Department of Ottoman Archives in jpeg format. We strive to implement an automatic reading method for registers from different precincts of the empire, which are obtained in the mid-nineteenth century. These registers include comprehensive demographic data on the male population in the households, i.e., names, occupations, ages and family relations. Females were neither counted nor recorded in these records. The registers were cataloged and gradually provided for research since 2011. There are approximately 11,000 registers. In this research, we study the generic characteristics of these manuscripts. The size of a digitized page in the recordings was 2210 $\times$ 3000 pixels. The first object type is the symbol marking the beginning of a populated place. It is seen in most of the registers and can mark the end of a previous village and start of a new one (see Figure 1). The other objects are individuals or households counted in the register, and they include demographic information about them. If an individual is the first person of a household, in the top of the object, there are two numbers (right and left) showing the number of the household and individual. Otherwise, they put only one number on top of the object showing the number of the individual inside the populated place. In the last line of all objects, the age of individuals is written. These registers sometimes updated by drawing a line on the people when they go to military service or decease. Sometimes the updates mistakenly connect the individual with a neighboring person, which might result in malfunctions in the information retrieval algorithms [4] (see Figure 2).

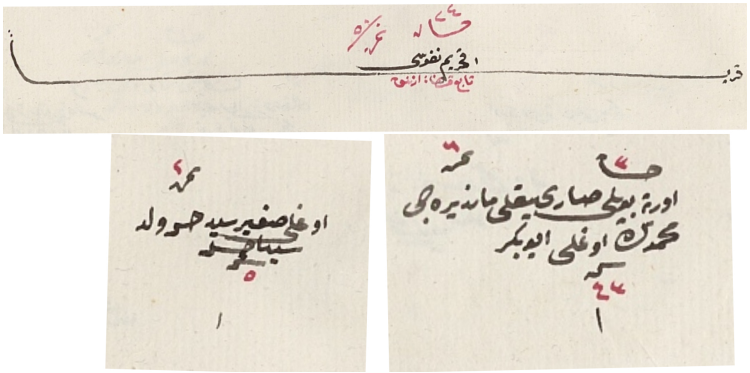

Figure 1. A population start, an individual and a household image samples are demonstrated. In the top, the populated place starting object, in the right bottom, a household and in the left bottom of the image, an individual object is shown.

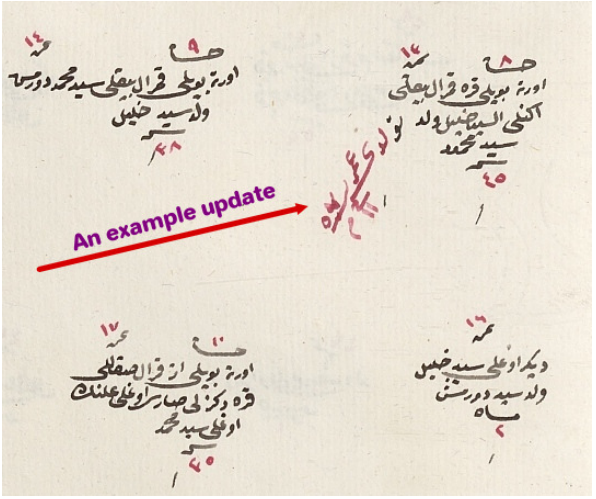

Figure 2. An example update drawn in red color is shown.

4. Methodology

In this section, we will describe our numeral spotting and handwritten digit recognition systems separately.

4.1. Automatic Numeral Spotting in Ottoman Registers

4.1.1. Red Color Filtering

As shown in Figure 1, the numerals are written in red color in the majority of these registers. However, not only the numerals but also the updates are written in red and we have to distinguish them from the numerals. In order to spot them easily, we applied a red color filter on the documents. We converted the image from RGB representation to HSV. The upper and lower limits for the red color used in these historical documents are determined by trial and error. Lower HSV thresholds were selected as (170;70;50), whereas the upper HSV thresholds were determined as (180,255,255). An example original image of the register and the filtered one is shown in Figure 3. The mask background color was chosen as black.

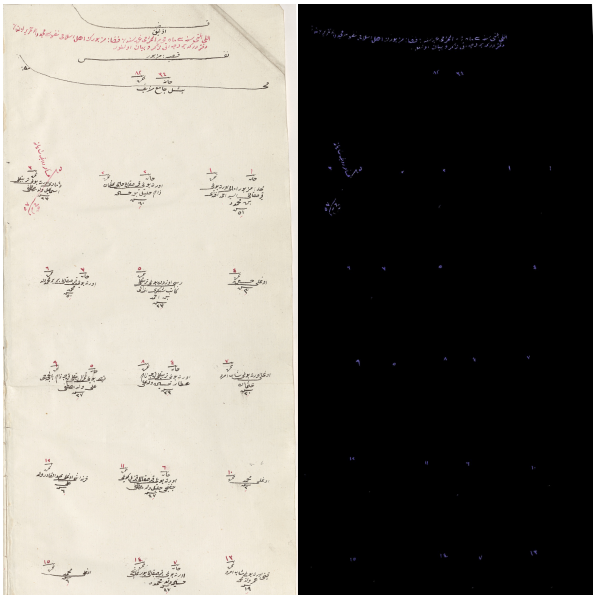

Figure 3. In the left, the original register image is shown. In the right, the resulting image after applying a red color mask is shown.

4.1.2. Creating an Annotated Dataset for Numeral Spotting

In order to employ the dhSegment toolbox [17] for page segmentation, we formed a dataset with annotations. Two classes were created. The first class is the background, which is the area other than numeral regions. We marked this area as black. The second class is the numeral region, and these fields were marked with green. We marked 50 pages of registers that belong to the Nicaea district with the described labels. In those pages, there were approximately 5000 numerals. A sample original image and marked version are presented in Figure 4.

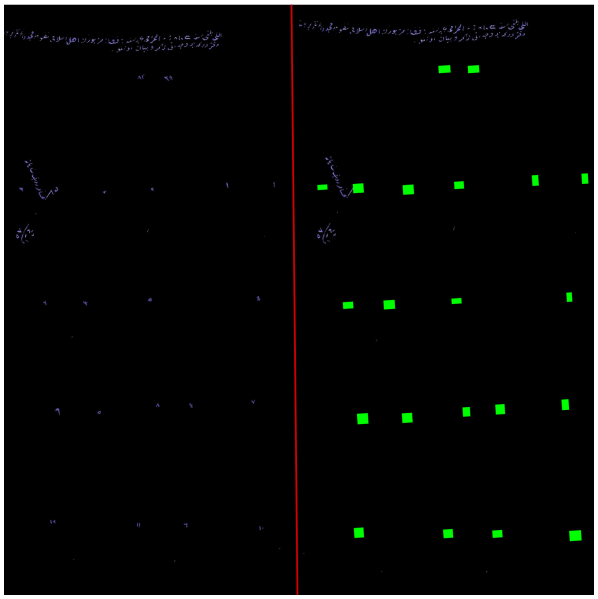

Figure 4. In the left, the red filtered register image is shown. In the right, the numerals marked and annotated for training the CNN model.

4.1.3. Creating a 3-Class Annotated Dataset

We also annotated a 3-class dataset. Numerals, register updates and background are the target classes. We aim to analyze the effect of adding register update class to the numeral spotting performance. Numerals were colored as green; updates were colored as red. The background is black which is the same as the 2-class annotation (see Figure 5).

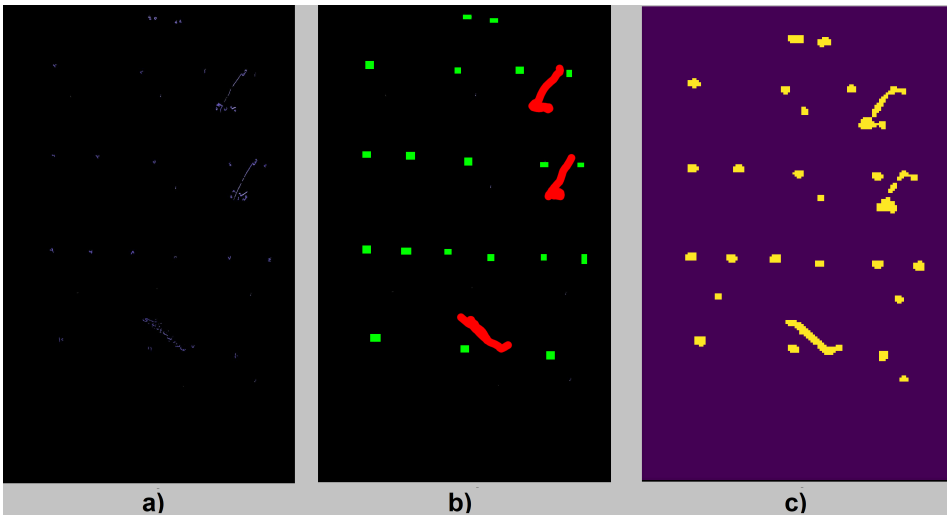

Figure 5. In part (**a**), the masked figure is shown. In part (**b**), the 3-class annotated image and in part (**c**), the prediction of our model (updates and numerals vs. background) are shown.

4.1.4. Training a CNN Model for Numeral Spotting

For training a CNN model for numeral spotting, we used the dhSegment toolbox. It trained a model by using a pretrained Resnet-50 architecture [32]. L2 regularization was employed with 10^{-6} weight decay [17]. Xavier initialization [33] and Adam optimizer [34] were used. Batch renormalization [35] was also applied to prevent a lack of diversity problem. The dhSegment toolbox also downsized pictures and arranged them into 300×300 patches for better fitting into the memory and giving support to batch training. By adding the margins, they prevented the border effects. By using pre-measured weights in the network, they decreased the training time considerably [17]. The training process employs several on-the-fly data augmentation procedures such as scaling (coefficient from 0.8 to 1.2), rotation (from -0.2 to 0.2 rad) and mirroring. Lastly, the toolbox outputs the probabilities of pixels that belong to classified object types. For further details of the toolbox, the paper explaining this toolbox [17] could be examined. For 2-class, a binary matrix comprises of the probabilities that a pixel belongs to the class is created. Pixels could be connected, and components should be created by analyzing this matrix. Connected component analysis tool [17] is applied for forming objects. We can measure the performance of our system after the objects are created for these classes. We presented predicted raw binarized image with the original manuscript and masked image in Figure 6. CPU is used to train the model. It took three hours to train a model for a hundred images. Testing an image, on the other hand, lasted for approximately 10 s.

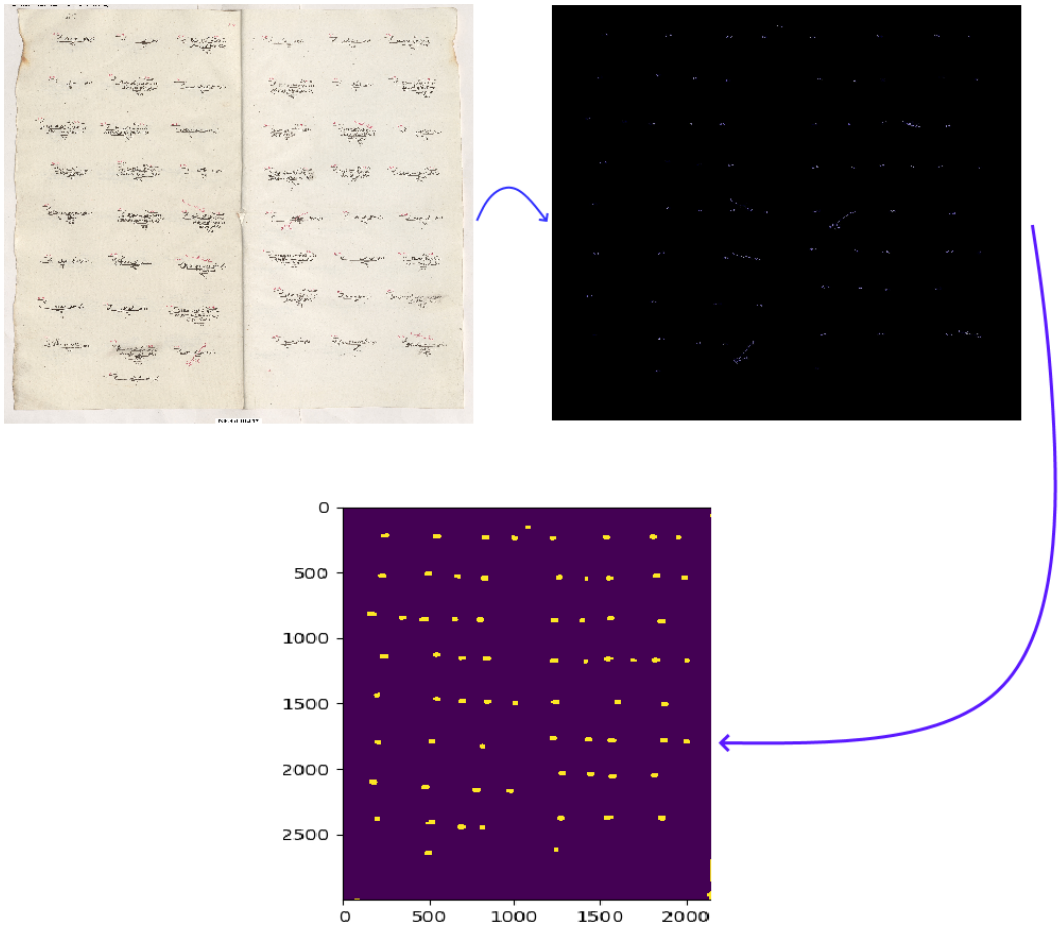

Figure 6. The complete processing of numeral spotting is shown. First, a red mask is applied to the original image. The masked image is shown in the middle. Lastly, a binary prediction image for spotting numerals is created.

4.2. Automatic Digit Recognition Using Deep Transfer Learning

For Arabic digit recognition in the historical registers, we first created a small dataset of 50 digits obtained from the spotted one-digit numerals. The dataset is balanced in terms of digit types. We then used two large datasets for using deep transfer learning. The first one is HODA dataset [36] which includes 70,000 images. The second one is AHDBase [37] which is also composed of 70,000 digits written by 700 participants. Lastly, we presented results obtained by dividing our local dataset (see Figure 7 for samples) into 80% training and 20% test sets.

Figure 7. All ten-digit samples from our local data. Fifty digits are selected from the output of the numeral spotting system.

4.2.1. Applying DTL on HODA and AHDBase Datasets

We pretrained two different CNN architectures by using the HODA and AHDBase datasets and tested on our local numeral dataset. These datasets have both 60,000 training samples and 10,000 test samples. By using the training samples as inputs to the CNN architecture (see Figure 8), we obtained 128 features (a vector of 128 numbers) in the final Conv2D layer for both datasets. For each convolutional layer, we applied batch normalization, maxPooling and dropout processes. As the dropout ratio, we used 0.2. MaxPooling layers used pool size as two. To prepare the model for feature extraction, we pretrained the model with all layers by using both HODA and AHDBase datasets and removed the last layers outside the rectangle in Figure 7 which provided the above mentioned 128 features. We then provided our test samples as inputs for this model to predict the 128 features. After extracting these features of our local dataset by using this pretrained "transferred cropped model", we applied different machine learning classifiers (MultiLayer Perceptron (MLP) with one hidden layer (with 100 nodes), kNN with k = 3, Random Forest with 100 trees (RF), Support Vector Machine (SVM) with a radial kernel (cost = 1 and kernel degree = 3) and Linear Discriminant Analysis (LDA)) to them to obtain Arabic handwritten digit recognition results. These classifiers are selected as representatives of the most commonly applied classifier types. The WEKA toolkit [38] was used for applying these classifiers. We used the default parameter settings in the WEKA package.

4.2.2. CNN-Based Handwritten Arabic Digit Recognition on the Local Data

We separated the local data into 80% training and 20% test sets. The local dataset is balanced in terms of digit types. Then, we applied the CNN architecture shown in Figure 8 to train a model for the local data and tested on the remaining separate test set.

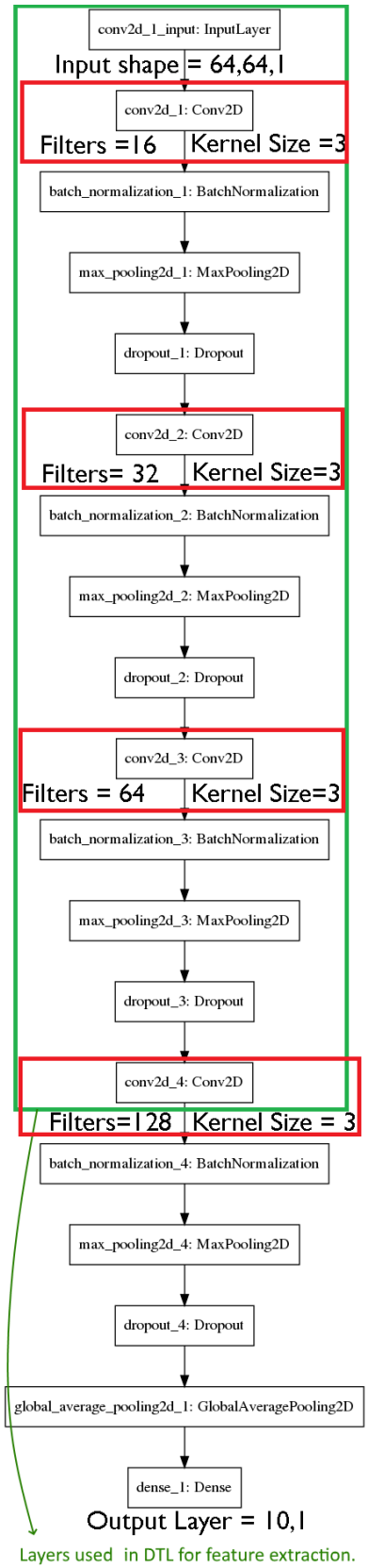

Figure 8. The CNN architecture is shown. The layers in the green rectangle are used for feature extraction in DTL. In order to train a model for local data, all layers are used. Conv2D stands for 2D Convolutional Layer.

5. Experimental Results and Discussion

5.1. Metrics

In order to evaluate our numeral spotting system performance, five metrics were used. Four of them are low-level metrics, and the last metric is a high-level one that we defined for our numeral spotting system. Low-level metrics are pixel-wise precision, recall and f-measure and Intersection over Union. These are widely employed for detecting objects in different image processing applications [39]. We further defined a high-level counting error metric to assess the performance of our numeral spotting method.

5.1.1. Pixel-Wise Precision, Recall and F-Measure

We first used the pixel-wise precision, recall and f-measure metrics. They are computed for each page in the test set and averaged over all pages. They can be calculated as:

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive} \tag{1}$$

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative} \tag{2}$$

$$F_{measure} = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{3}$$

5.1.2. Intersection over Union

We further computed the Intersection over Union (IoU) metric. The actual area of the segmented objects can be called as the ground truth whereas the connected areas are formed by connecting the adjacent pixel classifications belong to the same class can be called as the prediction area. The IoU can be computed by dividing the intersection of these two areas into the union of these areas.

5.1.3. High-Level Numeral Spotting Error

This last metric is specific to our application for spotting numerals in registers. It could be defined as the percentage of mistakenly classified numerals over the count of numerals from the ground truth. The predicted numeral count is the number of numerals predicted by our model. The ground truth of numeral count is the actual number of numerals in the dataset counted by our team. This metric is named as Numeral Spotting Error (NSE).

$$NSE = || \frac{PredictedNumeralCount - GroundTruthNumeralCount}{GroundTruthNumeralCount} || \tag{4}$$

5.2. Numeral Spotting Results and Discussion

The registers used in the case study are from the Nicaea district. All 50 pages are divided into 80% training and 20% test. The pixel-wise precision, recall, f-measure, IoU, and high-level numeral spotting error results are presented in Table 2. Note that the first four metrics are presented for 2-class classification (background vs. numeral). The last metric is the accuracy of spotting the numerals in the manuscripts. We successfully spotted the numerals in the documents with 96.06% ($1 -$ NSE = %3.94) high-level accuracy. Although the IoU metric is relatively low, the performance of the spotting system shows that the documents are suitable for automatic segmentation processes after the red color mask. We further tested a 3-class classifier (see Table 3). When we added register update class, we obtained lower f-measure classes as expected. The lowest performance is achieved when recognizing the register updates. However, since our main focus is to spot numerals, numeral spotting performance is more important. We obtained 0.61 f-measure score while recognizing only numerals and 0.67 f-measure

score while recognizing numerals with updates vs. background which are close to the 2-class f-measure score 0.72.

Table 2. The performance of our numeral spotting model (numerals vs. background) is presented with different metrics.

IoU (%)	NSE (%)	Pixel-Wise Precision	Pixel-Wise Recall (%)	Pixel-Wise F-Measure (%)
49.82	3.94	0.7089	0.7535	0.7247

Table 3. The performance of our 3-class numeral spotting model (numerals and updates vs. background, numerals vs. background, updates vs. backgrounds) is presented with different metrics. PW stands for pixel-wise.

Classification Type	PW Precision	PW Recall (%)	PW F-Measure (%)
Numerals vs. Background	0.4753	0.8704	0.6126
Numerals and Updates vs. Background	0.6164	0.7538	0.6751
Updates vs. Background	0.9009	0.1574	0.2681

5.3. Digit Recognition Results using Deep Transfer Learning

5.3.1. Applying DTL on HODA and AHDBase Datasets

We first trained the CNN model for the training part of the datasets (60,000 images) and tested in the remaining images (10,000 test image) for validating our model. We obtained above 99% accuracy for both datasets (99.47% for HODA Dataset and 99.34% for AHDBase dataset respectively). After that, we applied the DTL method for feature extraction and tested different classifiers with our local dataset. The digit recognition results by applying different classifiers on the features extracted from the local datasets are presented in Table 4. The results of AHDBase are always higher than HODA dataset because their number representation is similar to our dataset. However, since the HODA dataset is created in Iran, the number representation is different from our dataset (see numbers in Figure 9). Zero corresponds to our five, six is similar to two in our historical manuscripts, four and five are totally different in our dataset which is responsible for relatively lower accuracies. A maximum of 72.4% accuracy is achieved by using MLP classifier in AHDBase features. MLP is the most successful classifier on both datasets. We interpreted this results as the neural network structure helps MLP to learn CNN extracted features better. RF and kNN are other successful classifiers on these features for both datasets. We also extracted the same 128 features from our local dataset and tested the accuracies and area under ROC curve results (see Table 5). The accuracies are higher than the DTL results from the modern datasets (AHDBase and HODA) which shows that they could not capture the properties of these historical manuscripts successfully via the DTL method.

Figure 9. HODA representations for Arabic digits are demonstrated [40].

Table 4. Results obtained by applying different classifiers to extracted features by using deep transfer learning. DTL from AHDBase and HODA datasets are shown in separate columns.

Method	DTL from AHDBase	DTL from HODA
LDA	58.6	46.2
RF	62.1	58.6
MLP	72.4	65.5
SVM	57.9	49.0
kNN	64.1	62.1

Table 5. Results obtained by applying different classifiers to extracted features by using deep transfer learning. Results obtained with applying DTL from the local CNN architecture are shown.

Method	Accuracy	F-Measure	Area under ROC
kNN	84.61	0.829	0.964
RF	89.74	0.897	0.970
MLP	92.31	0.923	0.977
SVM	89.74	0.897	0.944
LDA	92.31	0.923	0.958

5.3.2. CNN-Based Handwritten Arabic Digit Recognition on the Local Data

We tested the CNN architecture on the local test dataset. We observed the accuracies of training and test set for 100 epochs (see Figures 10 and 11). We obtained 80% accuracy on the separate test set which is promising and outperformed the DTL accuracies. Both datasets (HODA and AHDBase) are recorded recently (after the 2000s). Therefore, their quality is higher than manuscripts recorded in the 1840s. Therefore, when we trained and tested a CNN model in our local dataset, the system also learns the properties of historical documents. However, when the DTL method is applied from these modern datasets, they could not capture the properties of these historical manuscripts. That could explain the relatively lower performance of DTL techniques. These results are also higher than learning CNN directly from the local data (80%), which shows the advantage of using DTL based feature extraction in our dataset.

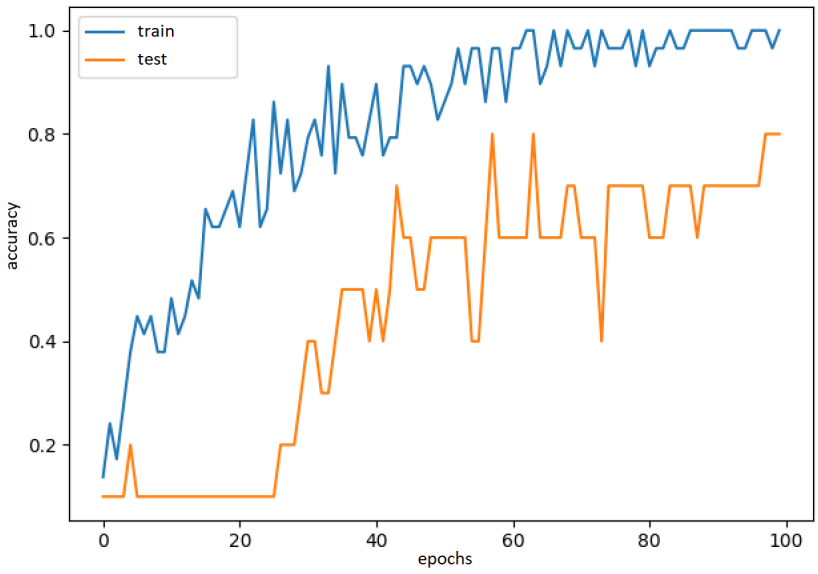

Figure 10. Training and test accuracies of CNN-based handwritten Arabic digit recognition system by epochs are shown. The local dataset is separated to 80% training and 20% test sets.

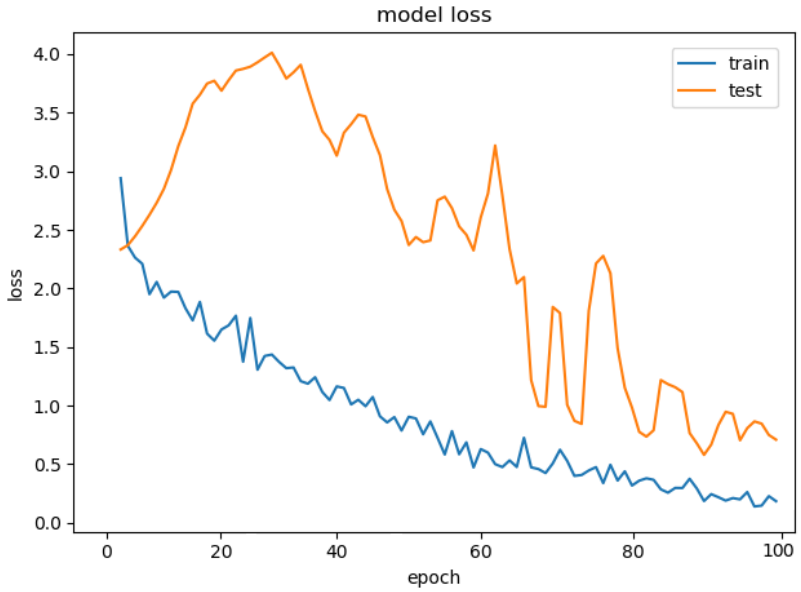

Figure 11. Training and test model loss of CNN-based handwritten Arabic digit recognition system by epochs are shown. The local dataset is separated to 80% training and 20% test sets.

6. Conclusions

In this study, we implemented an automatic Arabic numeral spotting system to a selection of the very first series of population registers of the Ottoman Empire conducted in the mid-nineteenth century. We took advantage of the property of population registers that numerals are written in red color. After applying a red color mask, we developed a CNN-based numeral spotting system. We further formed a small Arabic digit dataset from the detected numerals by selecting uni-digit ones and tested the Deep Transfer Learning (DTL) methods from the models trained in large open datasets for digit recognition. We also compared these results with the CNN architecture trained and tested on the local dataset. For numeral spotting, we obtained 96.06% accuracy which shows that numerals in these historical population registers could be spotted after applying a red filter. After spotting these numerals, we presented the Arabic handwritten digit recognition results by applying DTL from the substantial datasets and a trained CNN architecture on the local dataset. The CNN architecture is trained on the local dataset and tested on the separate test set outperforms DTL methods with the digit recognition accuracy of 80%. This could be explained by the unique properties and the fact that the degradation of historical documents could not be detected when DTL from modern datasets is used. DTL, by using the AHDBase dataset results are always higher than using HODA dataset because its digits are similar to the digits used in the Ottoman population registers. In fact, four digits of the HODA dataset are totally different from the digits of historical Ottoman population registers. The best accuracy obtained by applying DTL with AHDBase is 72% (CNN + MLP) which is lower than CNN alone in the local dataset.

We believe that the contribution of this article will be useful for researchers studying Arabic handwritten digit recognition. From these promising results, we plan to increase the size of the local dataset and carry on further tests. As future works, we plan to develop a keyword spotting system for handwritten text recognition in these population registers in order to detect further personal information belonging to registered individuals such as names, family relations within households, and occupations.

Author Contributions: Y.S.C. is the main writer of the manuscript. He performed the curation and development of the dataset and of the software and conducted the analysis. M.E.K. organized the preparation of the archival sources and initial data gathering. He has provided historical context and information regarding late Ottoman

population registers, and contributed to the conceptualization of the case study. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the European Research Council (ERC) project: "Industrialisation and Urban Growth from the mid-nineteenth century Ottoman Empire to Contemporary Turkey in a Comparative Perspective, 1850–2000" under the European Union's Horizon 2020 research and innovation program Grant Agreement No. 679097, acronym UrbanOccupationsOETR. M. Erdem Kabadayı is the principal investigator of UrbanOccupationsOETR.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Kim, M.S.; Cho, K.T.; Kwag, H.K.; Kim, J.H. Segmentation of handwritten characters for digitalizing Korean historical documents. In *Document Analysis Systems VI*; Marinai, S., Dengel, A.R., Eds.; Springer: Berlin/Heidelberg, Germany, 2004; pp. 114–124.
2. Wick, C.; Puppe, F. Fully convolutional neural networks for page segmentation of historical document images. In Proceedings of the IEEE 2018 13th IAPR International Workshop on Document Analysis Systems (DAS), Vienna, Austria, 24–27 April 2018; pp. 287–292.
3. Xu, Y.; He, W.; Yin, F.; Liu, C.L. Page segmentation for historical handwritten documents using fully convolutional networks. In Proceedings of the IEEE 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), Kyoto, Japan, 9–15 November 2017; Volume 1, pp. 541–546.
4. Can, Y.S.; Kabadayı, M.E. CNN-based page segmentation and object classification for counting population in Ottoman archival documentation. *J. Imaging* **2020**, *6*, 32. [CrossRef]
5. Puigcerver, J.; Toselli, A.H.; Vidal, E. ICDAR2015 competition on keyword spotting for handwritten documents. In Proceedings of the 2015 13th International Conference on Document Analysis and Recognition (ICDAR), Tunis, Tunisia, 23–26 August 2015; pp. 1176–1180.
6. Rouhou, A.C.; Kessentini, Y.; Kanoun, S. Hybrid HMM/DNN system for Arabic handwriting keyword spotting. In *International Conference on Image Analysis and Recognition*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 216–227.
7. Hesham, A.M.; Rashwan, M.A.; Al-Barhamtoshy, H.M.; Abdou, S.M.; Badr, A.A.; Farag, I. Arabic document layout analysis. *Pattern Anal. Appl.* **2017**, *20*, 1275–1287. [CrossRef]
8. Niu, X.X.; Suen, C.Y. A novel hybrid CNN–SVM classifier for recognizing handwritten digits. *Pattern Recognit.* **2012**, *45*, 1318–1325. [CrossRef]
9. Tissera, M.D.; McDonnell, M.D. Deep extreme learning machines: supervised autoencoding architecture for classification. *Neurocomputing* **2016**, *174*, 42–49. [CrossRef]
10. Nobile, N.; He, C.L.; Sagheer, M.W.; Lam, L.; Suen, C.Y. Digit/symbol pruning and verification for Arabic handwritten digit/symbol spotting. In Proceedings of the 2011 International Conference on Document Analysis and Recognition, Beijing, China, 18–21 September 2011; pp. 648–652.
11. AlKhateeb, J.H.; Alseid, M. DBN—Based learning for Arabic handwritten digit recognition using DCT features. In Proceedings of the 2014 6th International Conference on Computer Science and Information Technology (CSIT), Amman, Jordan, 27–28 November 2014; pp. 222–226. [CrossRef]
12. Ashiquzzaman, A.; Tushar, A.K.; Rahman, A.; Mohsin, F. An efficient recognition method for handwritten Arabic numerals using CNN with data augmentation and dropout. In *Data Management, Analytics and Innovation*; Balas, V.E., Sharma, N., Chakrabarti, A., Eds.; Springer: Singapore, 2019; pp. 299–309.
13. El-Sawy, A.; EL-Bakry, H.; Loey, M. CNN for handwritten Arabic digits recognition based on LeNet-5. In *Proceedings of the International Conference on Advanced Intelligent Systems and Informatics 2016*; Hassanien, A.E., Shaalan, K., Gaber, T., Azar, A.T., Tolba, M.F., Eds.; Springer: Cham, Switzerland, 2017; pp. 566–575.
14. Bukhari, S.S.; Breuel, T.M.; Asi, A.; El-Sana, J. Layout analysis for arabic historical document images using machine learning. In Proceedings of the IEEE 2012 International Conference on Frontiers in Handwriting Recognition, Bari, Italy, 18–20 September 2012; pp. 639–644.
15. Breuel, T.M. Robust, simple page segmentation using hybrid convolutional MDLSTM networks. In Proceedings of the 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), Kyoto, Japan, 9–15 November 2017; Volume 1, pp. 733–740. [CrossRef]

16. Augusto Borges Oliveira, D.; Palhares Viana, M. Fast CNN-based document layout analysis. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 1173–1180.
17. Ares Oliveira, S.; Seguin, B.; Kaplan, F. dhSegment: A generic deep-learning approach for document segmentation. In Proceedings of the 2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR), Niagara Falls, NY, USA, 5–8 August 2018; pp. 7–12. [CrossRef]
18. Brik, Y.; Chibani, Y.; Hadjadji, B.; Zemouri, E.T. Keyword-guided Arabic word spotting in ancient document images using Curvelet descriptors. In Proceedings of the IEEE 2014 International Conference on Multimedia Computing and Systems (ICMCS), Marrakesh, Morocco, 14–16 April 2014; pp. 57–61.
19. Kassis, M.; El-Sana, J. Automatic synthesis of historical arabic text for word-spotting. In Proceedings of the IEEE 2016 12th IAPR Workshop on Document Analysis Systems (DAS), Santorini, Greece, 11–14 April 2016; pp. 239–244.
20. Zirari, F.; Ennaji, A.; Nicolas, S.; Mammass, D. A methodology to spot words in historical arabic documents. In Proceedings of the IEEE 2013 ACS International Conference on Computer Systems and Applications (AICCSA), Ifrane, Morocco, 27–30 May 2013; pp. 1–4.
21. Wshah, S.; Kumar, G.; Govindaraju, V. Multilingual word spotting in offline handwritten documents. In Proceedings of the IEEE 21st International Conference on Pattern Recognition (ICPR2012), Istanbul, Turkey, 23–26 August 2012; pp. 310–313.
22. Khayyat, M.; Lam, L.; Suen, C.Y. Arabic handwritten word spotting using language models. In Proceedings of the IEEE 2012 International Conference on Frontiers in Handwriting Recognition, Bari, Italy, 18–20 September 2012; pp. 43–48.
23. Barakat, B.K.; Alasam, R.; El-Sana, J. Word spotting using convolutional siamese network. In Proceedings of the 2018 13th IAPR International Workshop on Document Analysis Systems (DAS), Vienna, Austria, 24–27 April 2018; pp. 229–234.
24. Lekhal, F.; El Hitmy, M.; Melhaoui, O.E. Arabic numerals recognition based on an improved version of the loci characteristic. *Int. J. Comput. Appl.* **2011**, *24*, 36–41.
25. Dehghanian, A.; Ghods, V. Farsi handwriting digit recognition based on convolutional neural networks. In Proceedings of the 2018 6th International Symposium on Computational and Business Intelligence (ISCBI), Basel, Switzerland, 27–29 August 2018; pp. 65–68.
26. Ghofrani, A.; Toroghi, R.M. Capsule-based Persian/Arabic robust handwritten digit recognition using EM routing. In Proceedings of the 2019 4th International Conference on Pattern Recognition and Image Analysis (IPRIA), Tehran, Iran, 6–7 March 2019; pp. 168–172.
27. Farahbakhsh, E.; Kozegar, E.; Soryani, M. Improving persian digit recognition by combining data augmentation and AlexNet. In Proceedings of the 2017 10th Iranian Conference on Machine Vision and Image Processing (MVIP), Isfahan, Iran, 22–23 November 2017; pp. 265–270.
28. Takruri, M.; Al-Hmouz, R.; Al-Hmouz, A. A three-level classifier: Fuzzy C Means, Support Vector Machine and unique pixels for Arabic handwritten digits. In Proceedings of the 2014 World Symposium on Computer Applications Research (WSCAR), Sousse, Tunisia, 18–20 January 2014; pp. 1–5.
29. Ashiquzzaman, A.; Tushar, A.K. Handwritten Arabic numeral recognition using deep learning neural networks. In Proceedings of the 2017 IEEE International Conference on Imaging, Vision Pattern Recognition (icIVPR), Dhaka, Bangladesh, 13–14 February 2017; pp. 1–4. [CrossRef]
30. Ahamed, P.; Kundu, S.; Khan, T.; Bhateja, V.; Sarkar, R.; Mollah, A.F. Handwritten Arabic numerals recognition using convolutional neural network. *J. Ambient Intell. Humaniz. Comput.* **2020**. [CrossRef]
31. Alani, A.A. Arabic handwritten digit recognition based on restricted Boltzmann machine and convolutional neural networks. *Information* **2017**, *8*, 142. [CrossRef]
32. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
33. Glorot, X.; Bengio, Y. Understanding the difficulty of training deep feedforward neural networks. In Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, Sardinia, Italy, 13–15 May 2010; pp. 249–256.
34. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
35. Ioffe, S. Batch renormalization: Towards reducing minibatch dependence in batch-normalized models. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 1945–1953.

36. Hoda Dataset. Available online: http://farsiocr.ir (accessed on 30 June 2020).
37. AHDBase Dataset. Available online: http://datacenter.aucegypt.edu/shazeem/ (accessed on 30 June 2020).
38. Holmes, G.; Donkin, A.; Witten, I.H. WEKA: A machine learning workbench. In Proceedings of the ANZIIS '94—Australian New Zealand Intelligent Information Systems Conference, Brisbane, Australia, 29 November–2 December 1994; pp. 357–361.
39. Jobin, K.V.; Jawahar, C.V. Document image segmentation using deep features. In *Computer Vision, Pattern Recognition, Image Processing, and Graphics*; Rameshan, R., Arora, C., Dutta Roy, S., Eds.; Springer: Singapore, 2018; pp. 372–382.
40. Hoda Dataset Persian Digits Demonstration [Online]. Available online: https://github.manzik.com/Persian-Handwritten-Digit-Recognizer/JS%20Interactive/ (accessed on 30 June 2020).