SIMILARITY-BASED TEXT RECOGNITION BY DEEPLY SUPERVISED SIAMESE NETWORK

Ehsan Hosseini-Asl[*]
Electrical and Computer Engineering
University of Louisville
Louisville, KY 40292, USA
ehsan.hosseiniasl@louisville.edu

Angshuman Guha
Captricity, Inc.
Oakland, CA 94612 USA
angshumang@captricity.com

ABSTRACT

In this paper, we propose a new text recognition model based on measuring the visual similarity of text and predicting the content of unlabeled texts. First a Siamese convolutional network is trained with deep supervision on a labeled training dataset. This network projects texts into a similarity manifold. The Deeply Supervised Siamese network learns visual similarity of texts. Then a K-nearest neighbor classifier is used to predict unlabeled text based on similarity distance to labeled texts. The performance of the model is evaluated on three datasets of machine-print and hand-written text combined. We demonstrate that the model reduces the cost of human estimation by $50\% - 85\%$. The error of the system is less than 0.5%. The proposed model outperform conventional Siamese network by finding visually-similar barely-readable and readable text, e.g. machine-printed, handwritten, due to deep supervision. The results also demonstrate that the predicted labels are sometimes better than human labels e.g. spelling correction.

1 INTRODUCTION

Optical Character Recognition (OCR) is traditionally used to convert images of machine printed text into textual content. Intelligent Character Recognition (ICR) is used to do the same for images of handwritten text. State-of-the-art OCR engines can work well, but only for clean data and where the OCR engine can be adjusted to deal with a single font or a small set of fonts. State-of-the-art ICR engines are significantly worse.

For a real-life application of high-accuracy character recognition involving both machine print and handwriting, one has to develop one's own OCR/ICR engine. This *typically* requires plenty of character-segmented data, as well as labeling at the character level. This is a very expensive proposition in most real-world situations, if not an impossible one. To avoid the character segmentation cost, Keeler et al. (1991) proposed learning character segmentation and recognition simultaneously from un-segmented data. This does not work well in practice beyond numeric characters and for large vocabularies. There has been some work at limited-vocabulary whole word recognition, see, for example, Lavrenko et al. (2004). To avoid character segmentation, LeCun et al. (1998) proposed a graph transformer network with Viterbi search. These kinds of models cannot compete in performance (training time) with modern deep neural nets that afford efficient implementations, for example, using GPU. Bunke et al. (2004) handled the segmentation problem using HMM-based recognition using the Viterbi algorithm. The present authors have experienced (convolutional) neural nets consistently out-performing HMMs in at least two domains, both with large quantities of industrial data: online handwriting recognition and speech recognition. Recently there also has been work involving innovative models where n^{th} characters are predicted for an input word of a fixed-size input image. For instance, one model might predict the first character, another model might predict the second character, and so on. See for example, Jaderberg et al. (2014).

Our goal is for a practical system that organically incorporates human labeling with machine learning in an active learning paradigm, to achieve very low error rates ($\leq 0.5\%$) while minimizing the amount of necessary human labeling.

[*]The work was performed during an internship at Captricity, Inc.

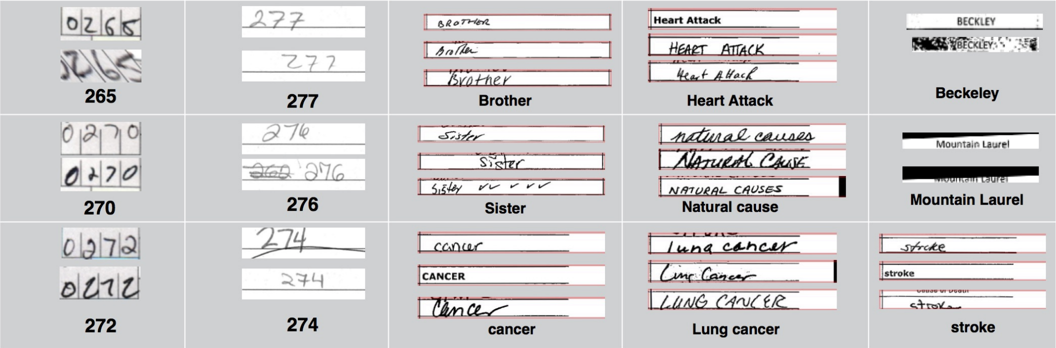


Figure 1: Examples of similar text. The content of barely-readable text can be predicted from similar texts.

Siamese Network (SN) is a type of end-to-end metric learning approach, consisting of a neural network that learns a discriminative function. SNs are trained by learning a similarity metric between pairs of data. SN models are applied to signature verification Bromley et al. (1993), digit recognition Hadsell et al. (2006), face recognition Chopra et al. (2005), Speech feature extraction Chen & Salman (2011), and Speech keyword detection Grangier & Bengio (2007). In this paper, we propose and discuss a novel method of text recognition that does not require character-segmented data. by predicting the content of a text using similar labeled texts. We use a Siamese Convolutional Network to learn the similarity between text images in a low-dimensional Euclidean space. To account for similar machine-printed and handwritten text, the SN is regularized by supervision in hidden layers Weston et al. (2012); Lee et al. (2014). Then a k-nearest neighbor algorithm is employed to predict the label based on most similar labeled texts. To account for unseen labels in test data, an interactive k-nearest neighbor algorithm with human annotation is employed for label prediction, and reducing the error. See Fig. 1 for examples of similar images i.e. with the same text content.

2 Proposed Model

In this section, we propose a model of text recognition based on learning similarity of images. In section 2.1, a Siamese network is proposed for learning similarity of text, and section 2.2 describes a text recognition framework.

2.1 Learning Text Similarity

To train a model to be able to learn the similarity between texts, a Siamese network is used as in Chopra et al. (2005); Hadsell et al. (2006). The Siamese network is trained to project the images into a feature space, where similar images are projected with short mutual Euclidean distance, and dissimilar images are projected with large mutual Euclidean distances. Training of the Siamese network is based on minimizing the contrastive loss of a pair of images,

$$\mathcal{L}(\mathbf{W}) = (1-Y) * \frac{1}{2} D^2 + \frac{1}{2} * Y * max(0, m - D)^2 \tag{1}$$

where $W = \{\{w^1, ..., w^n\}, w^o\}$ are the weights of the hidden layers and output layer of the Siamese network, Y is the label of paired images, i.e. 0 if similar and 1 if dissimilar, D_w is the Euclidean distance between a pair of images, and m is the desired Euclidean distance between a pair of dissimilar images. Siamese networks have shown promising results in learning similarity of the handwritten digits dataset, MNIST. However, in complicated cases of long text, capturing similarities between two texts is infeasible using a single loss function in the output layer of Siamese network. The performance of contrastive loss L is dependent on feature extraction of the hidden layers, where it should capture the similarities in a hierarchical way, to enable the output layer to extract features which can clearly represent the similarities of long and complex text. In order to boost the performance of the Siamese network for learning similarity of long text, we used the method of deep supervision proposed in Lee et al. (2014), where several contrastive loss functions are used for hidden and output layers, to improve the discriminativeness of feature learning, as shown in Fig. 2.

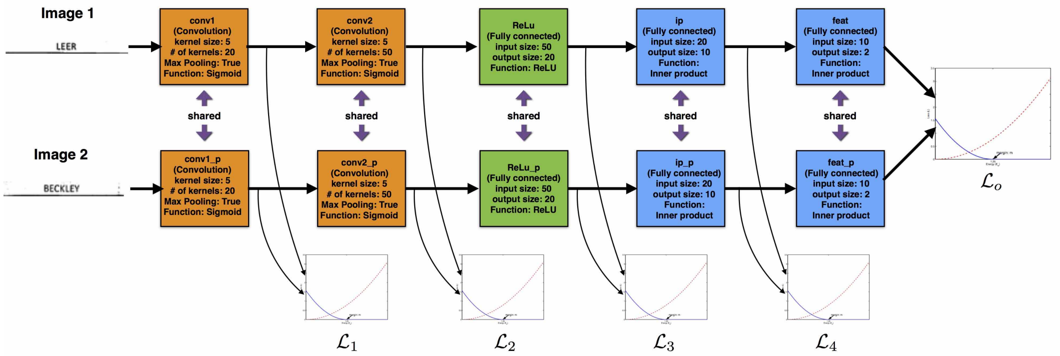


Image 1
LEER
Image 2
BECKLEY
conv1 (Convolution) kernel size: 5 # of kernels: 20 Max Pooling: True Function: Sigmoid
conv2 (Convolution) kernel size: 5 # of kernels: 50 Max Pooling: True Function: Sigmoid
ReLu (Fully connected) input size: 50 output size: 20 Function: ReLU
ip (Fully connected) input size: 20 output size: 10 Function: Inner product
feat (Fully connected) input size: 10 output size: 2 Function: Inner product
shared
shared
shared
shared
shared
conv1_p (Convolution) kernel size: 5 # of kernels: 20 Max Pooling: True Function: Sigmoid
conv2_p (Convolution) kernel size: 5 # of kernels: 50 Max Pooling: True Function: Sigmoid
ReLu_p (Fully connected) input size: 50 output size: 20 Function: ReLU
ip_p (Fully connected) input size: 20 output size: 10 Function: Inner product
feat_p (Fully connected) input size: 10 output size: 2 Function: Inner product
$\mathcal{L}_o$
$\mathcal{L}_1$
$\mathcal{L}_2$
$\mathcal{L}_3$
$\mathcal{L}_4$

Figure 2: Deeply Supervised Siamese Network (DSSN) for learning text similarity of machine-printed and handwritten text.

Therefore, the proposed method is called Deeply Supervised Siamese Network (DSSN) and it is trained using the combined contrastive loss,

$$\mathcal{L}_{DSSN}(\mathbf{W}) = \sum_{l=1}^{n-1} \lambda_l \mathcal{L}_l(\mathbf{w}^l) + \mathcal{L}_o(\mathbf{W}) \tag{2}$$

where l indicates the index for hidden layer, o is the output layer. Eq. 2 indicates that the loss $\mathcal{L}_l$ of each hidden layer is only the function of weights of that layer, i.e. $\mathbf{w}^l$. The DSSN network generates a Similarity Manifold, where similar texts are projected with short mutual Euclidean distances. The next section describes the text recognition model based on the Similarity Manifold. The ADADELTA method of gradient descent (Zeiler (2012)) is used to update the parameters of DSSN.

2.2 Text Recognition by Text Similarity

This section describes a text recognition framework to predict the label of text using the DSSN model developed in the previous section. The text recognition model is based on feature extraction of text using proposed DSSN network. We use a K-nearest neighbor algorithm to predict the label of text images in test data, based on similarity distance to the labeled text in train data. The predicted label is compared with human estimation as shown in Fig. 3(a).

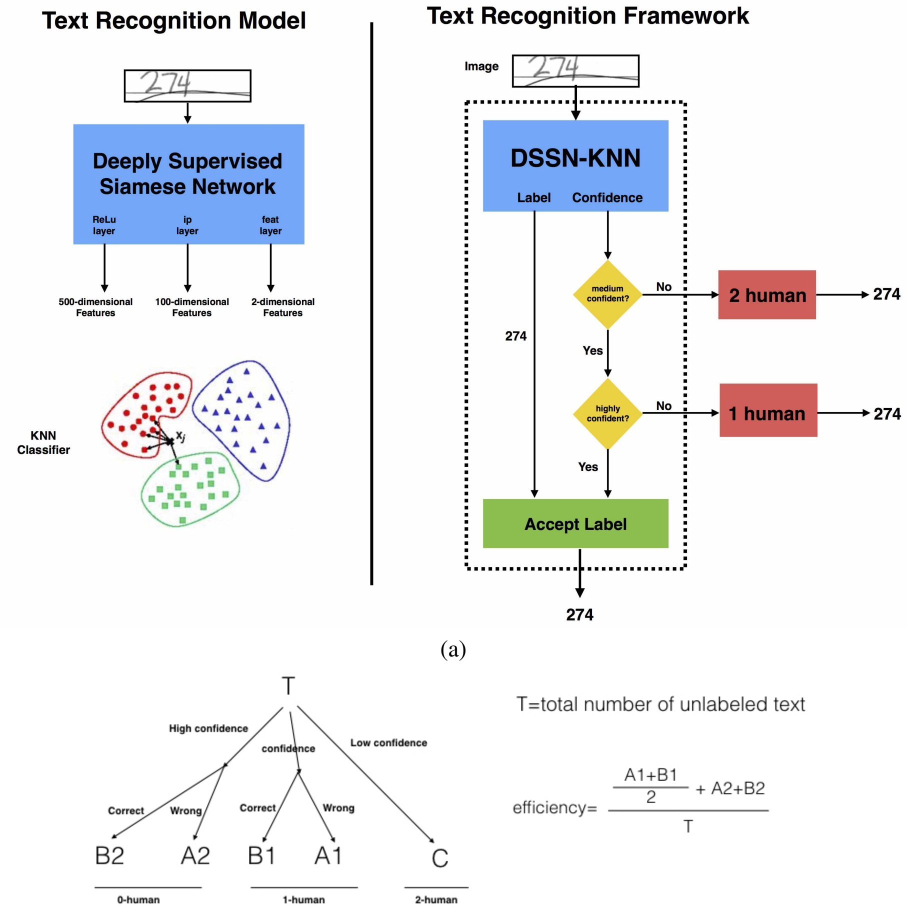
Our human-based model for text label prediction is based on voting of two humans on a text image. The proposed framework in Fig. 3 (a) is motivated by our goal of reducing the cost of human estimations while maintaining a low error rate. As shown in Fig. 3(a), the predicted label of DSSN-KNN is accompanied by a *confidence* value. We choose two parameters, θ_1 and θ_2, such that the confidence value can be classified as *highly confident*, *confident* and *not confident*. If the model's prediction confidence is high, we omit the required two human estimations. However, if the prediction is only confident, we validate the predicted label of DSSN-KNN with one human estimation. The parameters θ_1 and θ_2 are chosen by tuning the model's performance on the training set (or one can use a validation set).

To measure the performance of DSSN-KNN in reducing the human estimation, we define an efficiency metric as shown in Fig. 3(b),

$$\texttt{efficiency} = \frac{\frac{A_1+B_1}{2} + A_2 + B_2}{T} \tag{3}$$

where T is the total number of text samples, A_1 and B_1 are the number of confidently wrong and confidently correct predictions, and A_2 and B_2 are the number of high-confident wrong and high-confident correct predictions, respectively.

Note that the efficiency metric definition implicitly assumes a low rate of disagreement between two humans labeling the same image or between a human and the DSSN-KNN model. If this rate is



(a)

(b)

Figure 3: (a) Proposed Framework for interactive text recognition based on text similarity. (b) Efficiency metric to measure the reduction in human estimation of label.

1% (which is what we see in practice, see AC column in Table 4), the metric will overcount the reduction in the required number of human estimates by $\sim 1\%$. In the case of disagreement, extra human estimates will be needed to resolve conflicts.

The DSSN-KNN model can be used in one of two modes: ROBOTIC and ASSISTIVE.

ROBOTIC mode is suggested by Fig. 3 – (i) for high confidence predictions, we skip human labeling, (ii) for medium confidence predictions, we ask for human confirmation and (iii) for low confidence predictions, we discard the prediction and ask for at least two human estimates.

ASSISTIVE mode is to ignore θ_2 (high confidence threshold) – (i) for high and medium confidence predictions, we ask for human confirmation and (ii) for low confidence predictions, we discard the prediction and ask for at least two human estimates.

By definition, ASSISTIVE mode results in zero error from the DSSN-KNN model. But efficiency is lower because $\{A, B\}_2$ are folded into $\{A, B\}_1$ in the numerator of Fig. 3(b). On the other hand, ROBOTIC mode has higher efficiency at the cost of some DSSN-KNN errors unchecked by humans. We want this error to be under 0.5%. The detail of ASSISTIVE and ROBOTIC mode is explained in Algorithm 1.

Algorithm 1 Interactive Text Recognition by DSSN-KNN

```
Data: text
Output: label
Extract the feature of new text image from hidden layer of DSSN
Predict the label and confidence using k-nearest-neighbor algorithm
if mode=ROBOTIC then
    if $\theta > \theta_2$ then
        Output=label
    else if $\theta_1 < \theta < \theta_2$ then
        verify the label with 1 human
        if DSSN-KNN and human disagree then
            get another human label
            if two humans agree then
                Update KNN dictionary
                Output=human label
            else
                Output=label
            end if
        else
            Output=label
        end if
    else
        Label the text with 2 human
        Update KNN dictionary
        Output=human label
    end if
else // mode=ASSISTIVE
    if $\theta_1 < \theta$ then
        verify the label with 1 human
        if DSSN-KNN and human disagree then
            get another human label
            if two human agree then
                Update KNN dictionary
                Output=human label
            else
                Output=label
            end if
        else
            Output=label
        end if
    else
        Label the text with 2 human
        Update KNN dictionary
        Output=human label
    end if
end if
```

3 EXPERIMENTS

In this section, we design several experiments to evaluate the performance of our proposed model of text recognition. First, the DSSN-KNN model is pretrained on MNIST data with same same number of layers and filters. The MNIST pretraining is employed to extract edge-like filters to capture complex variation in handwritten text. Then the trained filters are used to initialize the DSSN for each dataset, and then fine-tuned based on similarity within each datasets, by minimizing the loss function of Eq. 2. We have chosen 80% for training and 20% for testing. We select minibatch size of 10 paired texts, containing 5 similar pairs and 5 dissimilar pairs, to train the Similarity manifold. Then based on Algorithm 1, in an online paradigm, data are read from test set one by one, and the predicted label is compared to truth label (as human) based on the confidence measure. The dataset is collected from form digitization. Then gussian blurring and laplacian are used to remove noise and enhance the contrast before training the model. We used Caffe, Jia et al. (2014), and Theano, Bastien et al. (2012), on Amazon EC2 g2.8xlarge instances with GPU GRID K520 for our experiments. First we apply some metrics to evaluate the performance of DSSN in learning the similarity manifold in section 3.1. Then the performance of DSSN-KNN is evaluated for text recognition of three hand-written text datasets in section 3.2.

3.1 Similarity Manifold Evaluation

In order to evaluate the performance of proposed DSSN for text recognition, we evaluated the trained similarity manifold for detecting similar and dissimilar texts. For this purpose, we implemented two separate experiments for non-numeric and numeric texts.

The non-numeric dataset contains 8 classes, where two major classes dominate in sample count. We found that most of the human-labeled 'blanks' are actually not blank, and contain some text from the two major classes. This misclassified text in training data hurts the performance of DSSN. The numeric dataset contains 9 classes, including 'blank' and numbers. The dominant classes are 'blank, '2016', '2018', '2014', and '2020'.

To investigate the distribution of text in the similarity manifold, the feature spaces of hidden layers and output layer are visualized in Fig. 4 and Fig. 5. Fig.4 shows the visualization of texts based on the 50- and 20-dimensional features extracted in 'conv2' and 'ReLu' layers, respectively. (Visualization of multidimensional data is done using a technique called t-SNE. See Van der Maaten & Hinton (2008).) It demonstrates that the three major classes are well-separated e.g. 'LEER, "BECKLEY' and 'Mountain Laurel'. Fig. 5 depicts the distribution of all texts in 'feat' layer, where each region is expanded for better visualization. Fig. 5 demosntrates the effect of deep supervision in forming natural clusters of texts with same label in hidden embedding. Accordingly, some boxes contain texts belonging to only one class, e.g. 2, 3, 5, 8, 9, 10, 11. The '2014' class is mixed with other classes of '2018' and '2016', as shown in boxes 1, 4, 6, 7, 13. The 'blank' shreds in box 12 which are combined with '2016' texts are mis-labeled texts – reducing the clustering performance of the DSSN model.

In order to evaluate the similarity manifold, several random pairs of images are selected from the test set and feed-forwarded through the DSSN. Then, the Euclidean distance between the paired images is computed based on the output of 'feat' layer. We choose a decision threshold, θ, such that $0.9 * FN + 0.1 * FP$ is minimized over the training set. Here FP is the false positive rate (similar images predicted as dissimilar) and FN is the false negative rate (dissimilar images predicted as similar). We weigh FN more than FP because the former increases efficiency at the cost of accuracy while the latter does not hurt accuracy. Table 1 shows the results for the model initialized by MNIST data, and after fine-tuning on the training dataset.

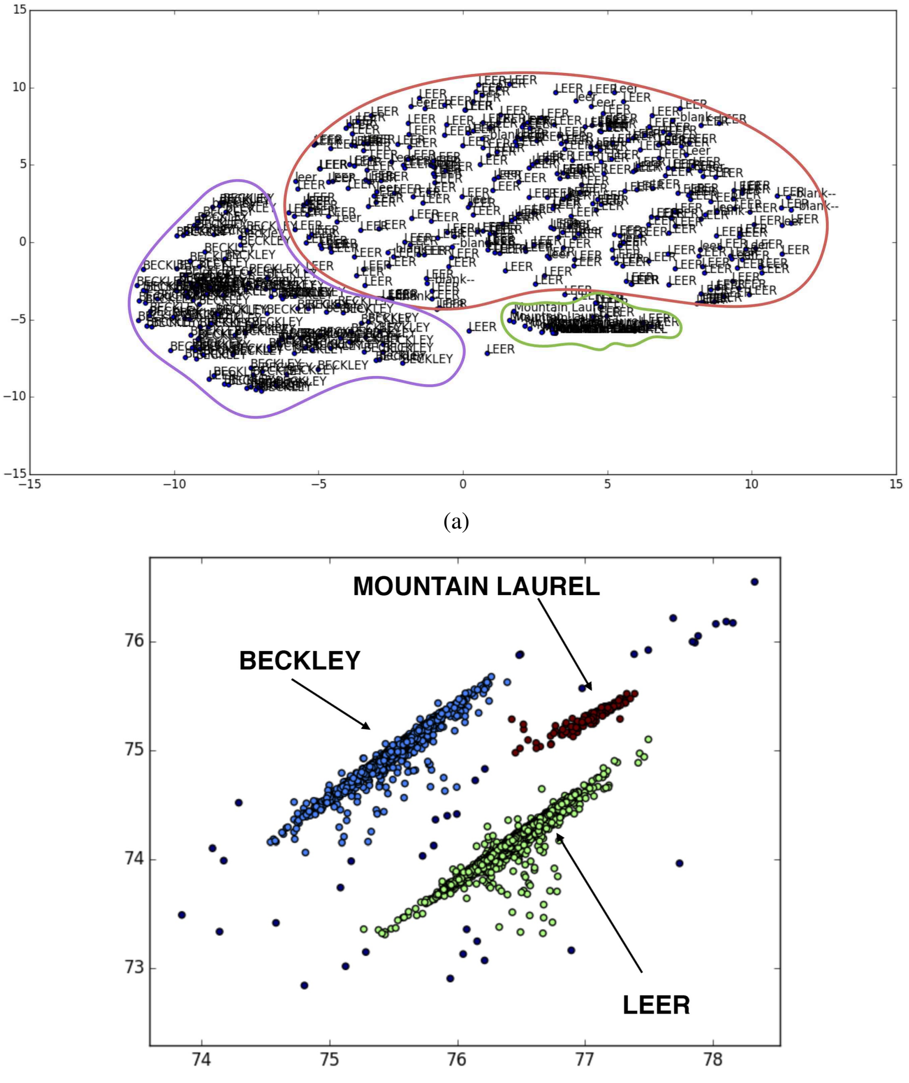
Table 1: Similarity prediction in Similarity manifold based on Euclidean Distance for non-numeric data.

DSSN	FN	FP	Error
Pretrained by MNIST	21.63%	7.58%	14.60%
After Fine-tuning	4.61%	1.89%	3.25%

Table 2: Text Clustering evaluation in Similarity manifold of different layers of DSSN in machine-printed texts

Dataset Type	Adjusted Rand Index		
	feat layer	ip layer	ReLu layer
non-numeric text	0.91	0.95	0.95
numeric text	0.96	0.93	0.96

To further evaluate the similarity manifold, a clustering algorithm is applied on texts and the clustered texts are evaluated based on truth labels in Table 2. For this test, we don't need parallel networks of DSSN. We use the extracted features from hidden and output layers for clustering of the



(a)

(b)

Figure 4: Similarity manifold visualization of machine-printed non-numeric texts in (a) hidden and (b) output layer, using t-SNE projection.

text. Several clustering algorithms were implemented: K-means, spectral clustering, DBSCAN and agglomerative clustering. To have a better evaluation of features in each layer, we applied clustering algorithms on the features of the ’ReLu’, ’ip’, and ’feat’ layers. The number of clusters for K-means and spectral clustering were set to 8. For DBSCAN and Agglomerative algorithms, the number of clusters was based on the similarity distance between text samples. The clustering performance is measured using Adjusted Rand Index (Hubert & Arabie (1985), Rand (1971)), which measures the similarity between clustered text and truth clusters formed by the truth labels. Table 2 shows the best clustering algorithm performance, which was agglomerative clustering on 3 layers of DSSN network.

3.2 Text Recognition Evaluation

In section 3.1, the similarity manifold learned by DSSN was evaluated for clustering and similarity prediction. This section focuses on performance of the proposed DSSN-KNN framework, as shown in Fig. 3 for text recognition. The trained DSSN model was tested on three difficult hand-written

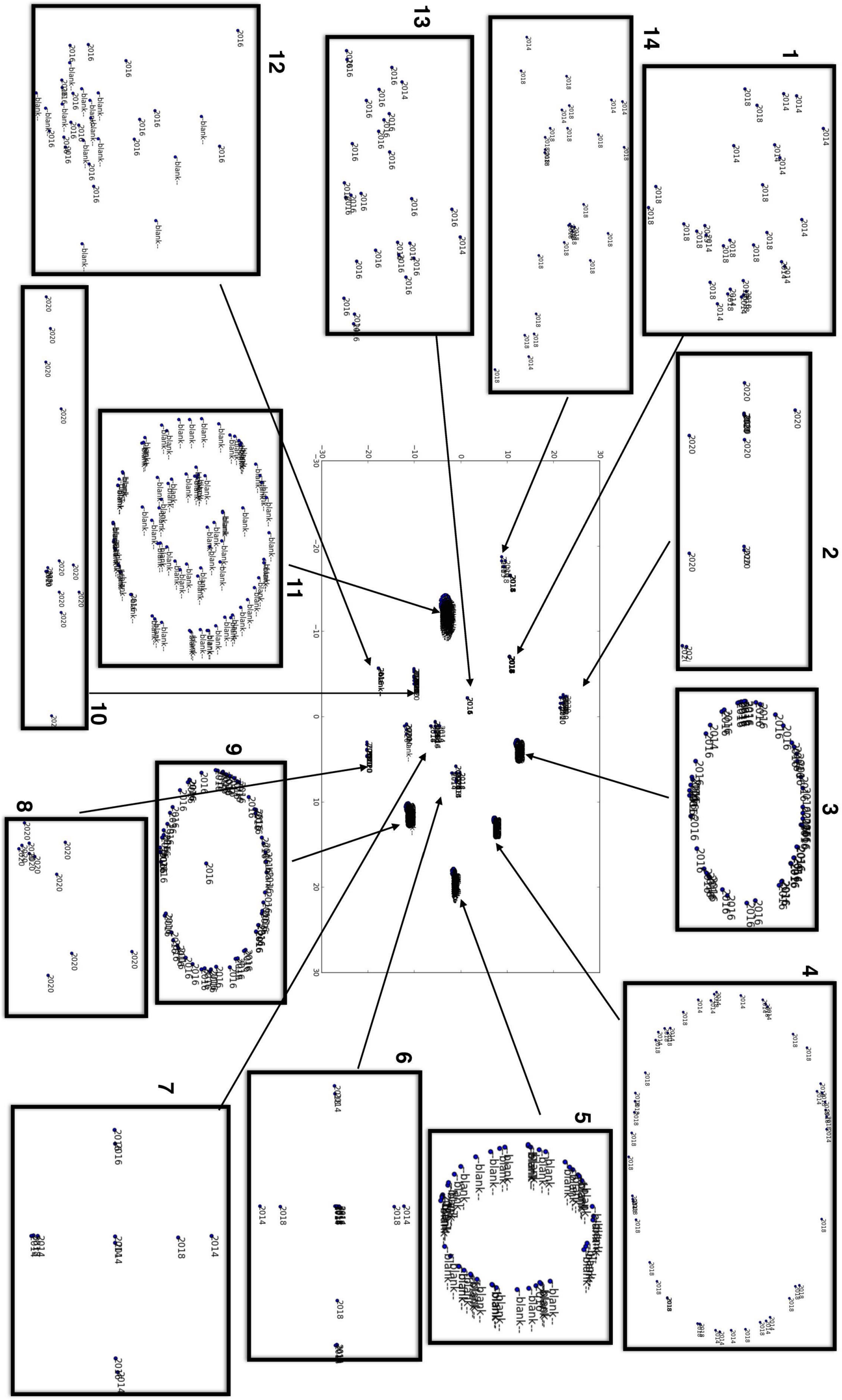


Figure 5: Similarity manifold visualization of machine-printed numeric text using t-SNE projection.

Table 3: Hand-written text image datasets

Dataset#1 (Short text - Unit)	**Total data**	**Train Data**	**Test Data**
No. of Images	90010	72008	18002
No. of Labels	1956	1722	827
No. of Unseen Labels	11%	–	**234**
No. of blank Images	50592	40517	10075

Dataset#2 (Short text - Non-Numeric)	**Total data**	**Train Data**	**Test Data**
No. of Images	89580	71664	17916
No. of Labels	1612	1321	459
No. of Unseen Labels	18%	–	**291**
No. of blank Images	84143	67309	16834

Dataset#3 (Short text - Numeric and Non-Numeric)	**Total data**	**Train Data**	**Test Data**
No. of Images	89461	71568	17893
No. of Labels	3124	2540	792
No. of Unseen Labels	18.69%	–	**584**
No. of blank Images	82864	66328	16534

datasets. These datasets included hand-written and machine printed text with many variations of translation, scale and image patterns for each class. The number of texts and unique classes in each dataset are listed in Table 3.

The text recognition performance of DSSN-KNN on the three datasets is listed in Table 5, where the reduction in human estimation is computed. The performance of DSSN-KNN is measured by Accuracy (AC), Accuracy of DSSN-KNN High-Confidence predicted labels (HCAC), Accuracy of medium-confident predicted labels validated by a human (HVAC), False Negative labels (FN), and High-Confidence False Negatives (HCFN). In order to select the confidence and high-confidence thresholds (θ_1 and θ_2) for each dataset, we did a grid search over the two thresholds to minimize High Confidence False Negative (HCFN). The chosen thresholds for each dataset and the error values are shown in Table 4.

Table 4: Text recognition performance on each dataset with respect to θ_1 and θ_2 to achieve HCFN$\leq$ 0.5%

Dataset	Method	Model	θ_1	θ_2	efficiency	AC	HCAC	HVAC	FN	HCFN
Dataset#1	ROBOTIC	SN	0.94	0.99	0.48	0.97	0.97	0.27	0.01	0.0124
		DSSN	0.94	0.99	**0.50**	**0.99**	**0.99**	**0.98**	**0.0049**	**0.0033**
	ASSISTIVE	SN	0.95	1	0.24	0.97	-	0.97	0.01	0
		DSSN	0.95	1	**0.27**	**0.99**	-	**0.99**	**0.0047**	0
Dataset#2	ROBOTIC	SN	0.94	0.99	0.73	0.79	0.81	0.98	0.4461	0.4367
		DSSN	0.94	0.99	**0.87**	**0.99**	**0.99**	**0.98**	**0.00407**	**0.0027**
	ASSISTIVE	SN	0.95	1	0.33	0.56	-	0.56	0.44	0
		DSSN	0.95	1	**0.45**	**0.99**	-	**0.99**	**0.0039**	0
Dataset#3	ROBOTIC	SN	0.94	0.99	0.82	0.89	0.89	0.85	0.11	0.1087
		DSSN	0.94	0.99	**0.86**	**0.99**	**0.99**	**0.98**	**0.0030**	**0.0016**
	ASSISTIVE	SN	0.95	1	0.41	0.89	-	0.89	0.11	0
		DSSN	0.95	1	**0.45**	**0.99**	-	**0.99**	**0.0029**	0

Some of the text images where DSSN-KNN produces high confidence errors are shown in Fig. 6. It is evident that most of the example pairs are, in fact, mutually visually similar, and the ”errors” can be attributed to human errors in ground truth. Interestingly, DSSN-KNN sometimes predicts better-than-human labels, for example, spelling corrections.

Table 5: Human-less estimation using proposed DSSN-KNN text recognition model.

Dataset	Type	No. of labeled Images	**Human-less efficiency**	
			ROBOTIC	ASSISTIVE
			No-human/1-human(efficiency%)	1-human(efficiency%)
Dataset #1	machine & hand	18002	8196/1659(50.31%)	9789(27.19%)
Dataset #2	machine & hand	17916	14739/1808(87.31%)	16475(45.98%)
Dataset #3	machine & hand	17893	14509/1706(85.85%)	16130(45.07%)

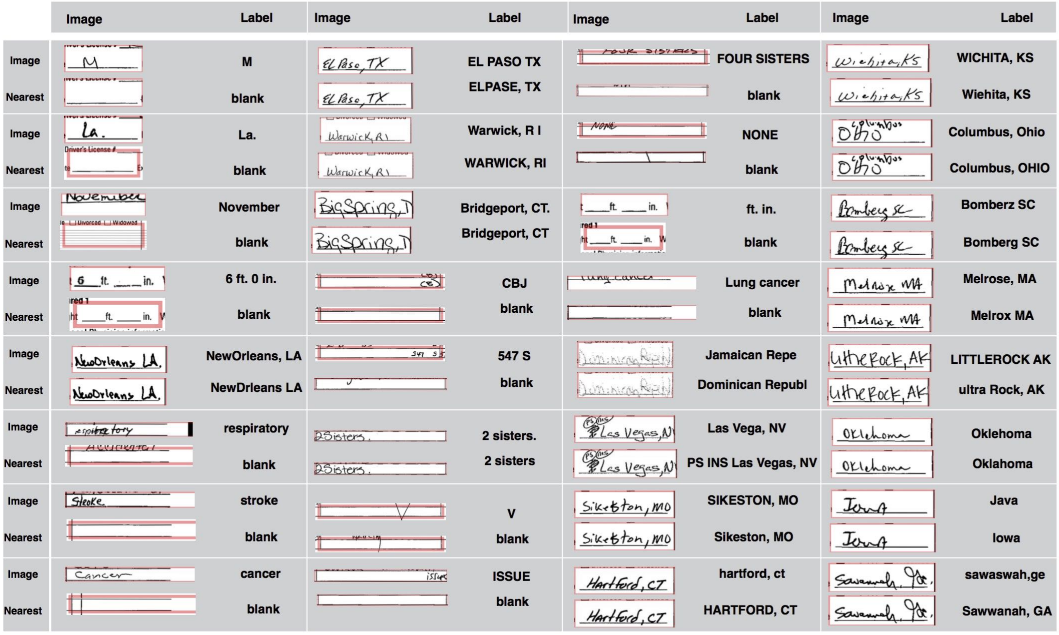


Figure 6: Texts with HCFN error (where DSSN-KNN produce high confidence wrong prediction). The nearest neighbor text in Similarity Manifold chosen by KNN is shown.

4 Conclusion

In this paper, we proposed a new text recognition model based on visual similarity of text images. A Deeply Supervised Siamese Network is trained along with a K-nearest neighbor classifier, to predict labels of text images. The performance of the proposed model is evaluated for accuracy and reduction of human cost of labeling. The results show that the average value of human-less efficiency on successful field is: $25 - 45\%$ in ASSISTIVE mode with NO error, and $50 - 85\%$ in ROBOTIC mode with $< 0.5\%$ error. Observed errors are explainable. Predicted labels are sometimes better than human labels e.g. spell corrections. Some of the false negative errors we count are in whitespace and irrelevant punctuation (the ”real” error is lower than reported here).

References

Bastien, Frédéric, Lamblin, Pascal, Pascanu, Razvan, Bergstra, James, Goodfellow, Ian J., Bergeron, Arnaud, Bouchard, Nicolas, and Bengio, Yoshua. Theano: new features and speed improvements. Deep Learning and Unsupervised Feature Learning NIPS 2012 Workshop, 2012.

Bromley, Jane, Bentz, James W, Bottou, Léon, Guyon, Isabelle, LeCun, Yann, Moore, Cliff, Säckinger, Eduard, and Shah, Roopak. Signature verification using a siamese time delay neural network. *International Journal of Pattern Recognition and Artificial Intelligence*, 7(04):669–688, 1993.

Bunke, Horst, Bengio, Samy, and Vinciarelli, Alessandro. Offline recognition of unconstrained handwritten texts using hmms and statistical language models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(6):709–720, 2004.

Chen, Ke and Salman, Ahmad. Extracting speaker-specific information with a regularized siamese deep network. In *Advances in Neural Information Processing Systems*, pp. 298–306, 2011.

Chopra, Sumit, Hadsell, Raia, and LeCun, Yann. Learning a similarity metric discriminatively, with application to face verification. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pp. 539–546. IEEE, 2005.

Grangier, David and Bengio, Samy. Learning the inter-frame distance for discriminative template-based keyword detection. In *INTERSPEECH*, pp. 902–905, 2007.

Hadsell, Raia, Chopra, Sumit, and LeCun, Yann. Dimensionality reduction by learning an invariant mapping. In *Computer vision and pattern recognition, 2006 IEEE computer society conference on*, volume 2, pp. 1735–1742. IEEE, 2006.

Hubert, Lawrence and Arabie, Phipps. Comparing partitions. *Journal of classification*, 2(1):193–218, 1985.

Jaderberg, Max, Simonyan, Karen, Vedaldi, Andrea, and Zisserman, Andrew. Synthetic data and artificial neural networks for natural scene text recognition. *arXiv preprint arXiv:1406.2227*, 2014.

Jia, Yangqing, Shelhamer, Evan, Donahue, Jeff, Karayev, Sergey, Long, Jonathan, Girshick, Ross, Guadarrama, Sergio, and Darrell, Trevor. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014.

Keeler, James D, Rumelhart, David E, Leow, W. K., Lippmann, R. P., Moody, J. M., and Touretzky, D. S. Self-organizing integrated segmentation and recognition neural network. In *Neural Information Processing Systems*, volume 3, pp. 557–563, 1991.

Lavrenko, Victor, Rath, Toni M, and Manmatha, R. Holistic word recognition for handwritten historical documents. In *Document Image Analysis for Libraries, 2004. Proceedings. First International Workshop on*, pp. 278–287. IEEE, 2004.

LeCun, Yann, Bottou, Léon, Bengio, Yoshua, and Haffner, Patrick. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Lee, Chen-Yu, Xie, Saining, Gallagher, Patrick, Zhang, Zhengyou, and Tu, Zhuowen. Deeply-supervised nets. *arXiv preprint arXiv:1409.5185*, 2014.

Rand, William M. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical association*, 66(336):846–850, 1971.

Van der Maaten, Laurens and Hinton, Geoffrey. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(2579-2605):85, 2008.

Weston, Jason, Ratle, Frédéric, Mobahi, Hossein, and Collobert, Ronan. Deep learning via semi-supervised embedding. In *Neural Networks: Tricks of the Trade*, pp. 639–655. Springer, 2012.

Zeiler, Matthew D. Adadelta: An adaptive learning rate method. *arXiv preprint arXiv:1212.5701*, 2012.