GAP Method for Keyword Spotting

Varsha Thakur[1], HimaniSikarwar[2]
[1,2]*Department Of computer science and engineering, Rajasthan College of Engineering for Women, Jaipur*

Abstract: deep neural network which is the baseline for all embracing machine vision tasks also has achieved a huge success in the field of document image analysis. Nowadays Convolutional neural networks (convents, cnns) have been established as the state-of-the art models in a wide range of image processing. The tasks of searching for relevant word images given an image or text query, known in the related literature as keyword spotting or word spotting (KWS), is no exception to this rule. Several variants of convolutional networks for keyword spotting have been proposed and employed with success.In this work, we present a CNN architecture that is PHOCNet with some pooling layers which is able to outperform on handwritten documents with less parameters and computational power.
Keywords: CNN, GAP, Maxout Function, Query-by-Example (QbE).

I. INTRODUCTION AND RELATED WORK

Convolutional neural networks (convents, CNNs) have been established as the state-of-the art models in a wide range of vision tasks. The tasks of searching for relevant word images given an image or text query, known in the related literature as keyword spotting or word spotting (KWS), is no exception to this rule. Several variants of convolutional networks for keyword spotting have been proposed and employed with success.

In [1] the CNN was pretrained and adapted to perform the word spotting and recognition task however; this approach has come with several short comings [2]. [3] They made the use of pyramidal histogram of characters (PHOC) embedding to form a segmentation free word spotting. Although originally learned using Fisher vector as image features, some adaptions using CNNs to learn the attributes have also been proposed.[4][5]

The Limitations of using CNNs or any other deep learning architecture are: as we know neural networks are scandalous for credibly being 'data hungry' in the sense they require huge amount of training data, computational resources and time taken is also more. Also as a positive side CNNs that were used for some specific tasks such as image classification, have been made available as part of toolkits. [6][7]

In [8][9] author came up with the idea of CNN architecture with pyramidal histogram of characters (PHOC) specially designed for word spotting. PHOCNet is a typical feed –forward convolutional network where convolutional layers and max-pooling layers embraced together, topped by fully connected layers leading to a stack of sigmoid outputs. For classification, the feature maps of the last convolutional layer are vectorized and fed into fully connected layers followed by a softmax logistic regression layer. This structure aqueduct the convolutional structure with traditional neural network classifier. [10] It tends the convolutional layers as feature extractors, and the resulting feature is classified in a conventional way. However, the fully connected layers are liable to overfitting, thus hampering the generalization ability of the overall network.

In this paper, we proposed a method with another strategy called global pooling [11] to replace the established fully connected layers in CNN. Instead of adding fully connected layers on top of the feature maps, we take the average of each feature map, and the resulting vector is provided directly into the softmax layer. This layer is very useful in comparison to others, one advantage of global average pooling over the fully connected layers is that it is more indigenous to the convolution structure by enforcing correspondences between feature maps and categories. Thus the feature maps can be easily interpreted as categories confidence maps. Another advantage is that there is no such parameter to optimize in the global average pooling thus overfitting is avoided at this layer. Furthermore, global average pooling aggregate out the spatial information, thus it is more robust to spatial translations of the input. Also we can see global average pooling as a structural regularize that explicitly accomplishes feature maps to be confidence maps of conceits (categories). This is made possible by the Multi-layer Perceptron (mlpconv) layers, as they make better approximation to the confidence maps than GLMs.

The rest of the paper is organized as follows. In section II we review the network architectural choices that we define. The proposed framework and methodology is described detailed in section III. We describe our experimental setup/datasets and discuss the result in section IV. After that we draw the conclusions and future work in section V.

II. ARCHITECTURAL CHOICES OF NETWORK

The PHOCNet model achieved a big success for keyword spotting results in tests with various document collections. While the model as a whole has shown to work very well, we assert that several choices regarding its architecture, the way it is trained, moreover the way the trained model and datasets is used for keyword spotting, is worth considering further also there are a lot differences in between a very new PHOCNet and traditional models in that it

uses the SPP layer which is the network head. The SPP layer brings the special oddity of transforming variablesize [12] inputs into fixed-size outputs. This characteristic leads to the evidently important advantage of not resizing the input image to a fixed size because as we have seen in many approaches [13] [14] [15] each word image has to be cropped to a unit width and height which almost always distort the imageas resizing will change the aspect ratio and scale of the input. Although, this lead is not much clear with other aspects. We focus on different strategies concerning the layering structure and especially with less computational power so we define some of the layers to overcome with the problem of overfitting where a tensor of dimension $h \times w \times d$ is reduced with $1 \times 1 \times d$which also take the input of any size.

III. PROPOSED FRAMEWORK

3.1 CNNArchitecture of element-

CNN architecture comprises of different input output layers and also with some hidden layers, these layers consist of convolutional layers, pooling layers, Activation layers and some functions. Basically the architecture is divided into 3 parts: feature maps, pooling layers and classification part.From input to output we can divide the PHOCNet model into 2 segments: a) the convolutional network which is the backbone of this architecture and b) layering part where some of these are used to minimize over fitting by reducing the total number of parameters. These convolutional layers consist of a number of filters which are lapped together with an input image. The output has a number of feature maps which can be input to another layer of CNN. For each feature map produced by appealing one of those filters in the convolutional layers to the input. So as for non-linearity the output of convolutional layers is passed through an activation function $f(x)$ so as to make the network more powerful and add an ability to learn from the complex and complicated form data and representations.
It is also needed to perform back propagation optimization strategy while propagating backwards in the network to evaluate the gradient of error (loss) and to reduce errors by using gradient descend or by any other optimization technique with respect to weight. Traditionally, many approaches have used the sigmoid or logistic function which has the slow convergence and dropped out for deep neural network due to the vanishing gradient problem. To avoid this problem some CNN architectures have made use of the Rectified linear units $R(x) = max\,(0,x) i.e. if\ x < 0, R(x) = 0\ and\ if\ x >= 0\ R(x) = x$ which is easier and efficient but the limitations are that it should only be used within Hidden layers of neural network.
Another problem with ReLu is that some gradients can be delicate during training and die. To fix this problem another improvement was introduced called Maxout function which enhanced the accuracy of dropout and improves the optimization. It also improves bagging training style on deeper layer. The maxout network is so named because each maxout unit chooses the maximum value within a group of linear pieces as the activation.[16] So that the network is linear almost everywhere, which resembles the ReLU network. However, the maxout units compare values of a group of candidate pieces, while the ReLUs only compare the value of a single piece with 0.
In this the input of activation function is divided into k unit groups and maximum response is recorded.
It has the k linear models and the output is the maximum value in k model from given input x.
It is written as,

$M_i(x) = max_{j \in [1,k]} z_{ij}$

Where

$z_{ij} = x^T p_{\dots ij} + a_{ij}$

$P \in S^{d \times m \times k}\ and\ a \in S^{m \times k}$

m: Number of hidden layers

d: Size of input vector(x)

k: Number of linear models

After applying the activation function this attuned field is expanded by using the pooling layers, which operates on each feature map independently. The most common approach is used in pooling is max pooling. A max pooling layer performs down-sampling by dividing the input into rectangular pooling regions, and computing the maximum of each region which is further passed to the next layer.The output from the convolutional layers performs the high-level features in the data although that output could be flattened and connected to output layers, so figuring this a fully-connected layer is used to learn there feature which works on same principle as the Multi-layer Perceptron

(MLP). .
For image classification task the training is carried out with softmax function first, the function is given by

$$sf(x) = \frac{e^{x_b}}{\sum_{a=1}^{k} e^{xa}} = v'$$

to the output of last layer of CNN to generate the output whose loss is calculated by the cross entropy as cross - entropy loss increases as the predicted probability diverges from the actual label.

3.2 Method

Convnet architectures typically consist different type of layers which divides are model into 3 parts : i) some convolutional layers , helpful for extracting the features ; ii) pooling layers, followed by an activation function ; iii) the classification part which is a fully connected layers completed by loss function. The numbers of circumstances are made for the design model. We only use the 3×3 convolutional layers followed by the maxout function, will also help in case of dead neurons. 2×2 max pooling layers are same as shown in other architecture, followed again by 3×3 convolutional layer and 2×2 max pooling layers. There is another type of pooling that is sometimes called global pooling used to replace some of the other layers. So here we are using only 2 fully connected layers. And the resulting vector is directly fed into the softmaxlayer.

3.2.1 Global Pooling

It is used to replace the fully connected layers in CNN. Since the first fully-connected layer consumes most of the energy, we decide to compress its weight matrix first. The idea is to generate one feature map for each proportionate category of the classification task in last layer. Instead of adding fully connected layers on the top of the feature maps, we only take the average/max of each feature map, and the resulting vector is fed directly into the softmax layer. The reason of using this is these fully connected layers are very large which ended up with increased network size and it's weight, requires more computational power. The advantage of using this is that there is no parameter to optimize thus over fitting is avoided at this layer.

$$x^l = \frac{1}{N} \sum_{i,j} y_{ij}^l$$

Where l is the each feature map and y^l is the measured distance.

There are two forms of Global pooling layers. In one form a single layer is completely replaces the fully connected layers. In other form, a single global layer feeds its output to one or more fully connected layers. Similar to max pooling layers, GAP layers are used to reduce the spatial dimensions where a tensor of dimension $h \times w \times d$ is reduced with 1×1×d sized dimension. GAP layers simply reduce the each $h \times w$ feature map to single number.

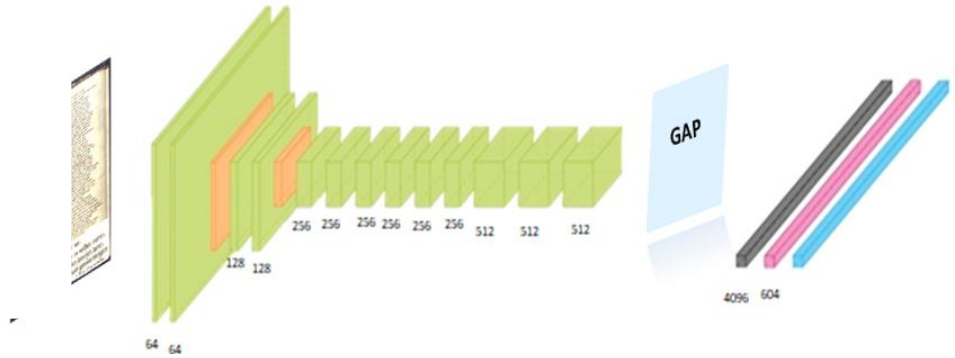

Figure 1.Proposed architecture of the Model.All the layers before GAP are same as PHOCNet; instead of ReLu function we use maxout activation function. GAP is used for reducing the problem of overfitting

3×3 convolutional layers + maxout

Max pooling layer

fully connected layer

fully connected layer

Softmax

3.3 Working of Global pooling layer-

We proposed a method for both the parameter level and less computational power. The layering part of this model is same as PHOCNet architecture but instead of using ReLu we use maxout function for the dead neurons, if available. By means of global average pooling on an input feature map is to compute the average value of all the elements in the feature map.

We already discovered in last part that global pooling layer can replace fully-connected layer .we can either replace all the layers or simply do the work with replacement of first fully connected layer and thus can reduce the problem of overfitting caused by fully connected layer still it is a quite tough task to pre trained without fully connected layer so first, we add a global average pooling layer before the first fully connected layer to perform the down sampling on each input feature maps of that fully connected layer so that we can shrink the weight matrix. Also it has the advantage of preserving the easy fine-tuning property and the capability of a pre-trained model with conventional structures.

1×1×(total number of input maps after layering of GAP), this is the output feature maps (from the last convolutional layer) which is down sampled with GAP feature maps. For which input vector, weight of row and matrix needs to be same with input elements and if not, we will adjust the matrix according to changed input element. And the length of each row modified to 1×1×(depth of feature map GAP). Then it will pass to first and the output will fed into softmax layer.

IV. EXPERIMENTS

4.1 Datasets

The Keyword Spotting experiments are assessed on the various databases for example, George Washington dataset, MNIST Datasets, and IFN/ENIT datasets which may cause challenges on training due to present to their limited training sets, here we are going to perform an experiment on challenging IAM dataset .The IAM dataset contains a total number of 115,320 words written by 657 different writers.it was first published in [22].IAM dataset has been initially proposed for assessing handwriting recognition methods. In recent times this dataset has become conceivably the most popular as well as reliable for keyword spotting techniques. Which can be used to train and test handwritten text recognition and to perform writer identification and verification experiments. This contains large number of comprised words, as well as their variety in writing style makes it ideal for training and testing deep neural networks.Our point of convergence is for QbE spotting scenario so that the images those are only appeared once or stop words are keep out. The retrieval list is computed by the nearest neighbor search by using cosine distance function. The performance is assessed in terms of mean Average Precision (MAP), by calculated the mean average for all the queries.

4.2 Tools and Training

The method is performed with the Keras toolkit which provides open source and high level neural network framework written in Python works with backend Tensor flow It was developed to make implementing deep learning models as fast and easy as possible for research and development.

Training is fulfilled with assuming a cross-entropy loss. And optimized by Adam.The generalized rule of error for output layers defined as,

$$E_i^0 = \frac{1}{2}\sum_{k=1}^{k}\left[X_k^i - Y_k(Z_{ki}^0)\right]$$

Where

$$Z_{ki}^0 = \sum_{j=1}^{j} Y_j\left((Z_{ji}^d)\, S_{ij}\right)$$

and

$$Y_k(Z_{ki}^0) = f(Z_{ki}^0) = \frac{1}{1+e^{-z_{ki}^0}}$$

$$E_i^h = \frac{1}{2}\sum_{j=1}^{j}\left[X_k^i - Y_k(Z_{ki}^0)\right]$$

Also we use a batch with size of 10 images with same size as using batch of different sizes do not utilize the GPU capabilities. Networks are trained for 10, 0000 iterations.

V. RESULT AND DISCUSSION

List shows the result for the different experiments sprint on the dataset. The overall performance of our setup which compared with the other state-of-the art approaches, seen in Table1. This method with comparison to other PHOCNet, has achieved a merciless act. In point of fact our setup exceeds the available SpottingNet approaches in different assessments.

Table -1 Result of experiments for IAM Dataset in MAP (%)

Method	MAP (%)
Softmax CNN	48.67
PHOCNet	72.51
Semantic	81.58
Deep PHOCNet	81.50
GAP CNN	82.01

There is a number of engrossing views to make from the method as in our method we refute the perception of CNNs that they always needs a huge amount of training data as we dropped some of the layering part and already done with the dead neurons with the maxout network.

For the IAM, training done in less than 12 hours and the estimated representation for the given word image in less than 25ms and the parameters are exacts the same size.We separate train, validation1, validation2 and test in ratio 20: 3: 3: 6 and perform 60,000 epochs on our deep network.Also we have seen that with change in aspect ratio of an image does not affect the result anyhow. So this might be the more advisable choice of keeping the initial size as it is without cropping the image, even though it has greater training requirements.

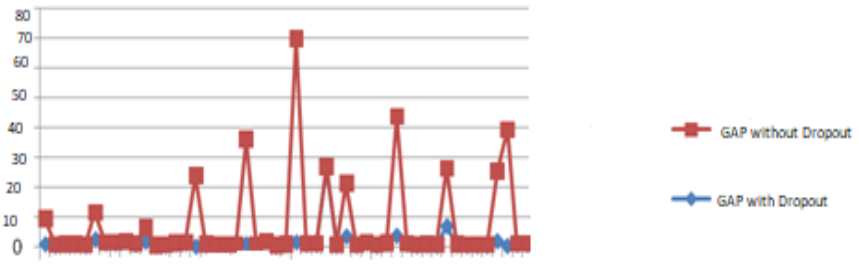

Figure2.The MAP valuesare shown in the graph for IAM Dataset with the standard error

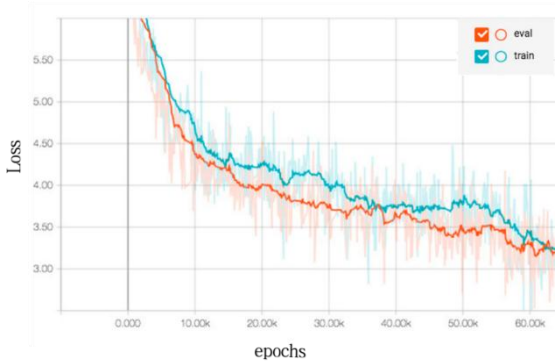

Figure3.Graph shows the values for the eachepoch with Loss

VI. CONCLUSION AND FUTURE WORK

We have studied and learned a number of important characteristics concerning the model, architecture and training of a variety of CNN (convnets) used for keyword spotting. In our experiment we introduced the use of Global average pooling (GAP) for the PHOCNet, a deep CNN designed for word spotting. This presented experiment helps us draw some perceptive conclusions.

With our work we give the new baseline for KWS system by replacing some of fully connected CNN layers with GAP. The useof GAP thus allows networks to function with less computational power and togeneralize the better performance. So in our experiment we have proposed a model where we assimilated our conclusions on the better architecture and the strategies for the training. Also the proposed model enhanced by employing learning to construct a better version of KWS system and gives us the new baseline on the IAM datasets.
Furthermore this work can also extend by replacing some of the layers also, by using other activation functions which enhance the performance with other datasets. As future work, we would like to examine the performance of more advanced entirety strategies.

VII. REFERENCES

[1] Sharma and K. PramodSankar, "Adapting off-the-shelf CNNs for Word Spotting & Recognition," in International Conference on Document Analysis and Recognition, 2015, pp. 986–990.
[2] Sebastian Sudholt, Gernot A. Fink, "PHOCNet: A Deep Convolutional Neural Network for Word Spotting in Handwritten Documents",2017
[3] George Retsinas, GiorgosSfikas, Nikolaos Stamatopoulos1, Georgios Louloudis1 and Basilis Gatos1 "Exploring critical aspects of CNN-based Keyword Spotting. A PHOCNet study",2018
[4] J. Almaz´an, A. Gordo, A. Forn´es, and E. Valveny, "Word Spotting and Recognition with Embedded Attributes," Transactions on Pattern Analysis and Machine Intelligence, vol.36, no.12, pp. 2552–2566, 2014.
[5] Saad ALBAWI , Tareq Abed MOHAMMED Saad AL-ZAWI, "Understanding of a Convolutional Neural Network" , ICET2017, Antalya, Turkey 978-1-5386-1949-0/17/$31.00 ©2017 IEEE
[6] T. M. Rath and R. Manmatha, "Word Spotting for Historical Documents," IJDAR, vol. 9, pp. 139–152, 2007.
[7] ZhuoyaoZhong, Weishen Pan, Lianwen Jin, Harold Mouchère, "SpottingNet: Learning the Similarity of Word Images with Convolutional Neural Network for Word Spotting in Handwritten Historical Documents", 2016 IEEE
[8] PriyankaSherkhane, Prof. DeepaliVora,"Survey of Deep Learning Software Tools", 2017
[9] M. Rusinol, D. Aldavert, R. Toledo, and J. Llad´os, "Efficient segmentation-free keyword spotting in historical document collections," Pattern Recognition, vol. 48, no. 2, pp. 545–555, 2015.
[10] D. Aldavert, M. Rusinol, R. Toledo, and J. Llados, "Integrating Visual and Textual Cues for Query-by-String Word Spotting," in International Conference on Document Analysis and Recognition, 2013, pp. 511–515.
[11] Ting-Yun Hsiao, Yung-Chang Chang†, and Ching-Te Chiu , "Filter-based Deep-Compression with Global Average pooling for convolutional network", IEEE Workshop on Signal Processing Systems, 2018
[12] M. Rusi˜nol, D. Aldavert, R. Toledo, and J. Llad´os, "Towards Query-bySpeech Handwritten Keyword Spotting," in International Conference on Document Analysis and Recognition, 2015, pp. 501–505.
[13] L. Rothacker and G. A. Fink, "Segmentation-free query-by-string word spotting with bag-of-features HMMs," in International Conference on Document Analysis and Recognition, Nancy, France, 2015.
[14] S. Sudholt and G. A. Fink, "A Modified Isomap Approach to Manifold Learning in Word Spotting," in 37th German Conference on Pattern Recognition, ser. Lecture Notes in Computer Science, Aachen, Germany, 2015.
[15] Frinken, A. Fischer, R. Manmatha, and H. Bunke, "A Novel Word Spotting Method Based on Recurrent Neural Networks," Transactions on Pattern Analysis and Machine Intelligence, vol. 34, pp. 211–224, 2012.
[16] Xiangang Li, XihongWu,"Improving long short-term memory networks using max out units for large vocabulary speech recognition",2017 IEEE
[17] K. He, G. Gkioxari, P. Doll´ar, and R. Girshick, "Mask RCNN," arXiv preprint arXiv:1703.06870, 2017.
[18] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," CoRR, vol. abs/1405.3531, 2014.
[19] C. Tensmeyer and T. Martinez, "Document image binarization with fully convolutional neural networks," International Conference on Document Analysis and Recognition (ICDAR), 2017.
[20] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," arXiv preprint arXiv:1412.6980, 2014.
[21] T. Wilkinson and A. Brun, "Semantic and verbatim word spotting using deep neural networks," in Proceedings of the 15th International Conference on Frontiers in Handwriting Recognition (ICFHR), 2016, pp. 307–312.
[22] U.Marti and H. Bunke . A Full english sentence databses for off-line handwriting recognition . In proc. Of the 5th int. conf. on document analysis and recognition, pages 705-708,1999.
[23] Hongxi Wei, GuanglaiGao, "Visual Language Model for Keyword Spotting on Historical Mongolian Document Images" , IEEE 2017
[24] Leydier M, Aldavert D, Toledo R, Llados J (2011) Browsing heterogeneous document collections by a segmentation-free word spotting method. In: International conference on document assay and recognition, 63–67
[25] Czuni L, Kiss P, Gal M, Lipovits A (2013) Local feature based word spotting in handwritten archive documents. In: 2013 11th international workshop on content-based multimedia indexing (CBMI), 179–184
[26] SehlaLoussaief, AfefAbdelkrim, "Deep Learning vs. Bag of Features in Machine Learning for Image Classification", IEEE 2018
[27] Sharma A, Sankar KP (2015) Adapting off-the-shelf cnnsforword spotting and recognition. In: 13th international conference on document assay and recognition (ICDAR), 986–990, x
[28] Sudholt S, Fink GA (2016) PHOCNet: a deep convolutional neural network for word spotting in handwritten documents. https:// arxiv.org/pdf/1604.00187.pdf
[29] George Retsinas and GiorgosSfikas , " Exploring critical aspects of CNN based keyword spotting system", 2018 IEEE
[30] ZhuoyaoZhong, Weishen Pan, Lianwen Jin , "SpottingNet: Learning the Similarity of Word Images with Convolutional Neural Network for Word Spotting in Handwritten Historical Documents" , 2015 IEEE
[31] Tomas Wilkinson and Anders Brun , "Semantic and Verbatim Word Spotting using Deep Neural Networks" , 2016 IEEE
[32] JinhuanWang,YuleiHuang,Wei Wei, Jing Li, "Character Recognition System Design based on Image Feature Extraction And QNN", 2018 2nd IEEE (IMCEC 2018)

[33] P. Krishnan, K. Dutta, and C. V. Jawahar, "Deep feature embedding for accurate recognition and retrieval of handwritten text," ICFHR, 2016, pp. 289–294.
[34] J. Almaz´an, A. Gordo, A. Forn´es, and E. Valveny, "Word spotting and recognition with embedded attributes," IEEE Transactions in Pattern Analysis and Machine Intelligence, vol. 36, no. 12, pp. 2552–2566, 2014.
[35] S. Han, J. Pool, J. Tran, and W. Dally, "Learning both weights and connections for efficient neural network," in Advances in Neural Information Processing Systems, 2015, pp. 1135–1143.
[36] H. Li, A. Kadav, I. Durdanovic, H. Samet, and H.P. Graf, "Pruning filters for efficient convnets," arXiv preprint arXiv: 1608.08710, 2016.
[37] E. L. Denton, W. Zaremba, J. Bruna, Y. LeCun, and R. Fergus, "Exploiting linear structure within convolutional networks for efficient evaluation," in Advances in Neural Information Processing Systems, 2014, pp. 1269–1277.
[38] Y.-D. Kim, E. Park, S. Yoo, T. Choi, L. Yang, and D. Shin, "Compression of deep convolutional neural networks for fast and low power mobile applications," arXiv preprint, 2015
[39] Rusiñol, M., et al., Efficient segmentation-free keyword spotting in historical document collections. Pattern Recognition, 2015. 48(2): p. 545-555.
[40] Zhang, X. and C.L. Tan, "Handwritten word image matching based on Heat Kernel Signature. Pattern Recognition", 2015. 48(11): p. 3346-3356.
[41] Rath, T.M. and R. Manmatha. "Word image matching using dynamic time warping". 2003 IEEE.
[42] Moghaddam, R.F. and M. Cheriet. Application of Multi-Level Classifiers and Clustering for Automatic Word Spotting in Historical Document Images. in 2009
[43] M. Lin, Q. Chen,andS.Yan,"Network in network," arXiv preprint arXiv:1312.4400, 2013. [10] J. Xue, J. Li, and Y. Gong, "Restructuring of deep neural network acoustic models with singular value decomposition." in Inter speech, 2013, pp.2365–2369.
[44] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in Proceedings of the 22nd ACM international conference on Multimedia. ACM, 2014, pp. 675–678.