International Journal of Current Engineering and Technology
E-ISSN 2277 – 4106, P-ISSN 2347 – 5161
©2021 INPRESSCO®, All Rights Reserved
Available at http://inpressco.com/category/ijcet

Research Article

Handwritten Marathi Character recognition using Deep learning

Prashant Misal, Prof.A.J.Patankar and Dr. Nitika Singhi

Department of Computer Engineering Alard college of Engineering & Mgmt Pune, India

Received 10 Nov 2020, Accepted 10 Dec 2020, Available online 01 Feb 2021, **Special Issue-8 (Feb 2021)**

Abstract

A normal human can easily recognize any written or typed or scanned text, numbers, etc., but when it comes to a machine, it is difficult to find out what exactly that given text or numbers. It will be difficult to recognize a handwritten digit for a machine. Many machine learning methods were used to fix the handwritten digit recognition issue. It is growing in more convoluted domains, so its training complexity is also increasing. To overcome this complexity problem, many algorithms have been implemented. In this project, the Convolutional Neural Network (CNN) with transfer learning and dropout methods, these approaches do use for recognition of the isolated handwritten alphabets. Transfer learning and dropouts are used to reduce the overall computation time of the proposed system. The customized Transfer learning and dropout techniques with CNN, to decreases the required number of epochs for training. It is used to identify Marathi Characters in the Devnagari handwritten digital database to predict the Scripts. We will try to achieve Maximum accuracy in short time.

Keywords: *Handwritten Character Recognition, Convolutional Neural Network;*

Introduction

The Indic scripts have a number of consonants and vowels, which represents a distinctive sound. Based on the articulatory mechanism used to produce the corresponding sound, these consonants are falls into different groups such as velar, palatal, retroflex, dental, labial, and a few others. The order of the voiceless and voiced plosives followed by the corresponding nasal sound is used to arrange them as unaspirated and aspirated.

अ	आ	इ	ई	उ	ऊ	ए	ऐ	ओ	औ	अं	अः

Fig. 1: Devanagari vowels.

क	ख	ग	घ	ङ	च	छ	ज	झ	ञ	ट	ठ
ड	ढ	ण	त	थ	द	ध	न	प	फ	ब	भ
म	य	र	ल	व	स	ष	श	ह	क्ष	त्र	ज्ञ

Fig. 2: Devanagari consonants.

प	पा	पि	पी	पु	पू	पे	पै	पो	पौ	पं	पः

Fig. 3 Derived forms of consonant “pa” when wrapped with vowels

The Vowels are also arranged according to the articulation of sounds and their short and long duration, figure 1 shows the vowels of devanagari scripts where fig. 2 denotes a consonant/conjunct. There is a corresponding modifier symbol called a matra for each modifiers. A matra can be attached to a full consonant or a consonant cluster to form a conjunct. Fig. 3 denotes an example of matra,kana,ukar, anuswar,and velanti. [1]

A pure consonant, or a sequence of the pure consonants, followed by a full consonant forms a consonant cluster—or a conjunct. Depending on how elaborate the text is that the user is composing Conjuncts, regardless of how formed, are all equivalent, and the user can decide which form to use. Even the individual consonant symbols can have different, but equivalent, shapes. Some of the consonants with the nukta diacritic behave as an independent consonant with a slightly different sound. Some of the conjuncts along with their constituents. The visual shapes of the conjuncts can be completely different from their constituents. The second consonant glyph and attached to the first consonant vertically or horizontally.[1]

Literature Survey

A. Image Analysis

Let take example of swans Image for creating a neural network model that is capable of recognizing swans in images. The swan image has certain characteristics that can be used to help determine whether a swan is present or not, e.g. its long neck, its white color.[2]

A swan has certain characteristic features that can be used for recognition purposes.

Fig. 4: Image Analysis

B. Problem with Traditional neural network

We will assume that you are familiar with traditional neural networks known as the multilayer perceptron (MLP).These are modeled on the human brain, whereby neurons are stimulated by connected nodes and are only activated when a certain threshold value is reached.[2] There are several drawbacks of MLP's, especially when it comes to image processing. MLPs use one perceptron for each input (e.g. pixel in an image, multiplied by 3 in RGB example). The amount of weights rapidly becomes unmanageable for large images. For a 224 x 224 pixel image with three color channels there are around 150,000 weights which must trained, So the result, difficulties may arise while training and over fitting can occur.

Another example of same problem is that MLPs react differently to an input (image/images) and its shifted version they are not translation invariant. e.g. if an picture of a cat appears in the top left of the image in one picture and the bottom right of another picture, the MLP will try to correct itself and assume that a cat will always appear in these section of the image.

It means MLPs are not the best idea to use for image processing for big size images. One of the main problems is that spatial information is lost when the image is flattened into an MLP. Nodes that are close together are important because they help to define the features of an image

Convolution Neural Network((ConvNet/CNN): This is an Deep Learning algorithm which will take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other. The pre-processing required in a convolution is much lower as compared to other existing classification algorithms. While in primitive methods filters are hand-engineered, with enough training, CNN have the ability to learn these filters/characteristics.[3] The architecture of a CNN is analogous to that of the connectivity pattern of Neurons in the Human Brain and was inspired by the organization of the Visual Cortex. Individual neurons respond to stimuli only in a restricted region of the visual field known as the Receptive Field. A collection of Receptive fields overlap to cover the entire visual area.

By application of relevant filters a CNN is able to successfully capture the Spatial and Temporal dependencies in an image. Due to the reduction in the number of parameters involved and reusability of weights, The architecture performs a better fitting to the image dataset. In other words, the network can be trained to understand the sophistication of the image better.

CNN have three major layers to build a system

- *Convolution Layer*: In this layer Convolutional layers filters are applied to the original image, or to other feature maps in a deep Convolution neural network. This is where most of the user-specified parameters are in the network. The most important parameters are the number of kernels and the size of the kernels.
- *Pooling layer:* Pooling layers are similar to convolutional layers, but they perform a specific function such as max pooling, which takes the maximum value in a certain filter region, or average pooling, which takes the average value in a filter region. pooling layer is used to reduce the dimensionality of the network.
- *Fully connected layer:* Fully connected layers should placed before the classification output of a CNN and is used to flatten the results before classification. This is similar to the output layer of an MLP.

Proposed Methodology

In our proposed mechanism should achieve the following goals: Deep Convolutional Neural Network have shown superior results to traditional shallow networks in many recognition tasks. Keeping distance with the regular approach of character recognition by Deep CNN, we explore the use different architecture and different techniques

To present the design of proposed approach and algorithms.

To present the practical analysis proposed algorithms and evaluate its performances.

To present the comparative analysis of existing and proposed approach in order to achieve the accuracy of character recognition.

This section describes the research related Marathi Devanagari Script. [4]

After deciding the dimension hand written character was recorded from 10 persons, twenty four alphabets (10 persons, 24 alphabets, 10 per alphabet) in all total database size was 2400.

After eliminating out of defined parameter alphabets images and noisy data, we used 2000 data set in total. We used MS paint application for recording the handwritten character, each character is of dimension 32x32.

The images are gray level and 8bpp. While recording we do manually consider rotation and shift. We then normalized each image by value 255 to have dynamic range of 0-1.The NN we consider is dense network with 1024 input nodes followed by two hidden layers with down sampling factor of 2 and final output layer is 24 (number of alphabets).The activation for the hidden layer is Relu and the output activation is tanh. We also added the dropout of 10% at each layer to avoid data over fitting. While training we flatten each normalized alphabet image to single vector of size 1024 to feed as input. The corresponding output is vector of size 24 with one hot representation. For training we used Adam as optimizer with learning rate 0.0001 and binary cross entropy as a loss function. We have considered 80:20 as training and validation split of the data. We train the network for 500 epochs with the batch size of 8.while testing we follow the same flow and apply max operator to the output probabilities to decide the predicted alphabet.

Wide variety of information which has been conventionally stored on paper is now converted to electronic form for better storage and perspicacious processing. Representation of documents as images is withal undesirable because it does not sanction the utilizer to edit or probe the document. We have taken our research interest in developing OCR systems because of number of potential applications in business and industry. OCR is technology which sanctions machine to apperceive text from an image. It is consequential for computerizing printed text so that they can be probed electronically, stored compactly or utilized for machine processing like translation or text to verbalization conversion[6]

The different layers for training and best result we willl consider,

Dropout layer: Generally, use a small dropout value of 15%-30% of neurons with 15% providing a good starting point.

A probability too low has minimal effect and a value too high results in under-learning by the network.
we are likely to get best performance when dropouts are used on a larger network, giving the model more of an opportunity to learn independent representations.
Application of dropout at each layer of the network has shown good results when we use dropout on incoming (visible) as well as hidden units.
Use a large learning rate with decay and a large momentum. Increase your learning rate by a factor of 15 to 100 and use a high momentum value of 0.9 or 0.99.
Constrain the size of network weights. A large learning rate can result in very large network weights. Introducing a constraint on the size of network weights such as maxnorm regularization with a size of 4 or 5 has been shown to improve results.

Dense Layer: A dense layer is just a regular layer of neurons in a convolution neural network. Each neuron receives input from all the neurons in the previous layer, which densely connected. The layer has a weight matrix W, a bias vector b, and the activations of previous layer a.A dense layer represents a matrix vector multiplication. (assuming your batch size is 1) The values in the matrix are the trainable parameters which get updated during back propagation. uT.W, W∈ Rn×mu, T.W,W∈Rnxm, So you get a m dimensional vector as output. A dense layer thus is used to change the dimensions of your vector. Mathematically speaking, it applies a rotation, scaling, translation transform to your vector.[7]

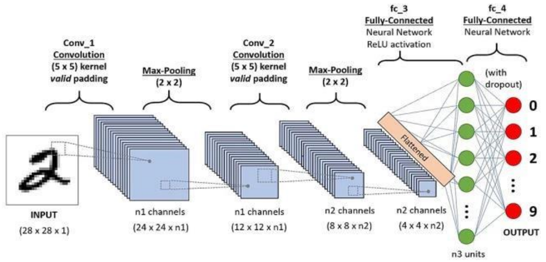

Fig.5: Proposed methodology

C. Architecture

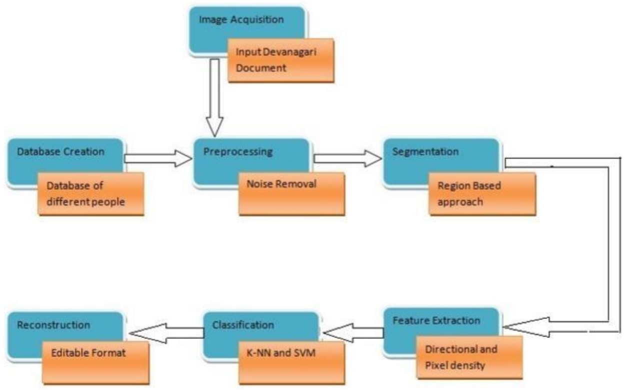

Fig 6: System Architecture

This research work is aim to developed method for recognition of handwritten Marathi Devanagari script by devising computationally efficient method for segmentation the complete process of optical character recognition is not a single step process.[8] It starts from the preprocessing and ends with the classification and recognition steps as shown in figure

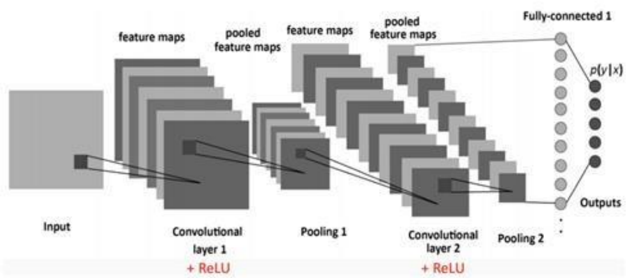

Fig 7: CNN Based Architecture

As our work is focused on Off-line recognition of Devanagari document, we have taken input in the form of handwritten documents which are written by Pen independent of color. It starts from preprocessing of handwritten document and convert it into editable form. We don't have to bother about the type of data, it may be in different shape and type. But in the modern age, the electronic presence is growing, sanctioning one to cogitate having more control over the data being acquired. The final output is present in doc or notepad format, we can give our comfortable texture format.

D. Algorithms

CNN Algorithm for the step by step procedure is as below, Step 1: Pretrain the filter, and initialize the filter size pixel as P1 × P2.

Step 2: Enter the image dataset for training. Process the image of the training set into the same picture as the filter size, and read the data to form the image data matrix X. Step 3: Initialize the weight w (l) i,j and bias bi and invoke the kernel function def Kernel() provided by TensorFlow to initialize parallel operations.

Step 4: The Conv2d is used for two-dimensional convolution operation to obtain the first layer convolution feature matrix X (1).

Step 5: The first layer convolution feature matrix X (1) is used as the input data of the pool layer. Use Formula (5) for the pool operation to obtain the feature matrix X (2).

Step 6: Use the SGD optimizer function expressed in Formula (4) to derive the learning rate of the top-down tuning optimizer, and use the weights in TensorFlow and the update-biased interface to update the weight wi and the bias bi , thus obtaining the feature matrix X. Algorithms 2018, 11, 28 8 of 15

Step 7: Generate the second convolution following Steps 4, 5, and 6 to derive the feature matrix X (4). Step 8: Merge the feature matrix X (4) into a column vector as the input of the neuron at the full-joint layer, multiply it with the weight matrix plus the bias, and then use the Leaky ReLU activation function to obtain the eigenvector b1.

Step 9: Use the eigenvector of the fully connected stratum as the input of the dropout layer, compute the output probability of the neuron in the dropout layer using Formula , and the eigenvector b2 is obtained.

Step 10: Use the eigenvector b2 as the input and the Softmax classifier output to achieve the results[9]

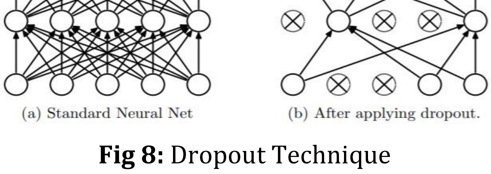

Fig 8: Dropout Technique

The dropout is introduced into the CNN to improve the normalization capability of the network. Any neuron in a neural network is temporarily discarded by the probability p, and the formula is as follows:

$$p = (p_a = 1|x)$$
$$= \sum_{i,j \in B_a} \exp(w^{(l)}{}_{i,j} * x_j{}^{(l-1)} + b_i{}^{(l)}) / (1 + \exp \sum_{i,j \in B_a} w^{(l)}{}_{i,j} * x_j{}^{(l-1)} + b_i{}^{(l)})$$

Below given is approaches we used to prevent our model from overfitting.[10] Dropout: Dropout simply refers to dropping out units; units representing both hidden and visible in the deep network. We temporarily remove the random units from the network along with all its inlet and outlet connections. For each training iterations, there will be new lighter network that remains after dropping the random units from the common denser architecture which will be sampled and trained. Each unit is retained with the fixed probability of p independent of other units and we set 0.5 for p, the number being optimal choice for most of the cases. [11] Transfer Learning: The basic premise of transfer learning is simple: take a model trained on a large dataset and transfer its knowledge to a smaller dataset. For object recognition with a CNN, we freeze the early convolutional layers of the network and only train the last few layers which make a prediction. The idea is the convolutional layers extract general, low-level features that are applicable across images — such as edges, patterns, gradients — and the later layers identify specific features within an image such as eyes or wheels Following is the general outline for transfer learning for object recognition: • Load in a pre-trained CNN model trained on a large dataset • Freeze parameters (weights) in model's lower convolutional layers • Add custom classifier with several layers of trainable parameters to model • Train classifier layers on training data available for task • Fine-tune hyper parameters and unfreeze more layers as needed This approach has proven successful for a wide range of domains. It's a great tool to have in your arsenal and generally the first approach that should be tried when confronted with a new image recognition problem. [12]

Result and Discussions

Here we tested the dataset with different handwritten Marathi Devnagari alphabets. The alphabets are used with different version fonts and handwriting so test the verity of Marathi alphabets. The data set is downloaded and created manually. The results are taken in four different ways. 1. Using MLP but no dropouts 2. Using MLP and dropouts 3. Using CNN and no dropout 4. Using CNN and with dropout. For the CNN we used 3 convolution layers. As discuss above training data set had 2000 images and corresponding one hot representation of output. The training accuracy we achieved is 97%. For testing we have created 10 images of each alphabet (1 per person per alphabet), in all 240 images. The test accuracy we observed is 95.83%. We also consider effect of noise by adding Gaussian noise with sigma 0.3, the test accuracy observed in that vase is 87.5% Based on the above experiment and output result the MLP is gives good result for small images and accuracy also good. But for when image size is big then time and out is not as per expectation. But compare to MLP, we got best result from CNN for small and large size images. In CNN results when we check without dropout is fast but problem is it stores result in memory and real-time recognition accuracy is less as compare to CNN with dropout and transfer learning V. CONCLUSIONS Based on our experiments we learned that the traditional MLP will give good results for the small image size and accuracy will be high. But for Big size images this method will not work. For big size images we need to consider CNN which will be more accurate and fast result provides. We learned the limitations of CNN i.e. overfitting hence we used dropout techniques which gives the more better results in actual (real time) scenarios. But the limitation of this paper is we only considered the basic set of alphabets and we can have very large scope for word and sentence recognition. Also the Devnagari numbers are not included in this paper which will be again more complex level for research. The sample data for machine training is key for deep learning research, her we have very high volume of data used for more accurate results.

References

[1] Acharya, Shailesh, Ashok Kumar Pant, and Prashnna Kumar Gyawali. "Deep learning based large scale handwritten Devanagari character recognition." 2015 9th International Conference on Software, Knowledge, Information Management and Applications (SKIMA). IEEE, 2015.

[2] https://towardsdatascience.com/simple-introduction-to-convolutionalneural-networks-cdf8d3077bac cPGCON 2020 (Post Graduate Conference for Computer Engineering

[3] https://towardsdatascience.com/a-comprehensive-guide-toconvolutional-neural-networks-the-eli5-way-3bd2b1164a53

[4] V. J. Dongre and V. H. Mankar, “A review of research on Devnagari character recognition,” arXiv preprint arXiv:1101.2491, 2011.

[5] https://www.mdpi.com/1999-4893/11/3/28/pdf [6] Kaur, Harpreet, and Simpel Rani. "Handwritten Gurumukhi Character Recognition Using Convolution Neural Network." International Journal of Computational Intelligence Research 13.5 (2017): 933-943

[7] https://towardsdatascience.com/transfer-learning-with-convolutionalneural-networks-in-pytorch-dd09190245ce

[8] Patil, Bhushan Dayaram, et al. "Material segmentation in image volumes." U.S. Patent No. 10,438,350. 8 Oct. 2019

[9] https://www.mdpi.com/1999-4893/11/3/28/pdf. [10] Reddy, S. . G., K. Thammi Reddy, and V. Valli Kumari. "Optimization of Deep Learning using Various Optimizers, Loss Functions and Dropout."

[11] https://medium.com/@amarbudhiraja/https-medium-comamarbudhiraja-learning-less-to-learn-better-dropout-in-deep-machinelearning-74334da4bfc5

[12] https://towardsdatascience.com/transfer-learning-with-convolutionalneural-networks-in-pytorch-dd09190245ce

[13] Abhijit Janardan Patankar ,Dr. Kshama V. Kulhalli ,Dr. Kotrappa irbi,” Emotweet: Sentiment Analysis tool for twitter”2016 IEEE Inter-national Conference on Advances in Electronics, Communication and Computer Technology (ICAECCT)

[14] Abhijit J. Patankar, Kotrappa Sirbi, Kshama V. Kulhalli ”Preservation of Privacy using Multidimensional K-Anonymity Method for NonRelational Data”IJRTE 2019