IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-7, Issue-1, Jan 2019 E-ISSN: 2347-2693

Devanagari Script Recognition using Capsule Neural Network

U.M. Sawant[1*], R.K. Parkar[2], S.L. Shitole[3], S.P. Deore[4]

[1,2,3]Department of Computer Science, Modern Education Society's College of Engineering, Pune, India
[4]Modern Education Society's College of Engineering, Pune, India

[*]*Corresponding Author: udaysawant998@gmail.com, Tel.: +91-8975583271*

Available online at: www.ijcseonline.org

Accepted: 19/Jan/2019, Published: 31/Jan/2019

***Abstract*—** Handwritten Devanagari Script Recognition has a lot of applications in the field of document processing, automation of postal services, automated cheque processing and so on. Several approaches have been proposed and experimented in the past depending on the type of features extracted and the ways of extracting them. In this paper, we proposed the use of Capsule Neural Networks (CapsNet) for the recognition of Handwritten Devanagari script, which is an advancement over the Convolutional Neural Networks (CNN) in terms of spatial relationships between the features. Capsule Neural Networks follow the principle of equivariance unlike the convolutional neural networks which follow the invariance property. CapsNet uses the dynamic routing by agreement method for passing data to higher capsules. CapsNet uses vector format for data representation. It can recognize similar characters in a more efficient manner as compared to CNN. Thus by using the advantages of CapsNet we are aiming to achieve better classification rate. We collected 100 samples of each of the 48 Devanagari characters and 10 Devanagari digits, and performed scaling, rotation and mirroring operations on these images. Hence, our dataset consists of total 29000 images.

***Keywords*—** *Capsule Networks, Dynamic routing, Devanagari Script Recognition, Convolutional Neural Networks*

I. INTRODUCTION

Handwritten character recognition is the conversion of images from paper documents, photographs, printed text or touch-screens into machine interpretable format, which is possible by either scanning the document, using a photo of a document, or from subtitle text superimposed on an image. Character recognition for the Devanagari script has been experimented by researchers using various Machine Learning techniques and their accuracies have been measured and used for further improvements in this field. Earlier versions of OCR had to be trained with images of each character with different fonts. But the advanced systems are capable of producing a higher degree of recognition accuracy for most fonts, and also support a variety of file format inputs for the digitized images. One of the most common benchmarks to determine how well an algorithm performs is to train it on the MNIST handwritten database. The accuracy that it achieves on MNIST gives a fair idea of how efficient an algorithm is, as compared to the other algorithms.

Devanagari script recognition has been tried and tested using various methods such as SVM, MLP-NN, KNN, Random Forest, CNN and so on. All these techniques have provided very good recognition accuracies on various datasets. Convolutional Neural Networks (CNN) turned out to be the most popular of all other methods due to its ability to build its own features from raw signals and provides higher optimization. The goal is to train a CNN to be as accurate as possible when labeling handwritten digits (ranging from 0-9) and handwritten characters. A CNN consists of an input and an output layer, as well as multiple hidden layers. The hidden layers of a CNN typically consist of convolutional layers, ReLU layer i.e. activation function, pooling layers, fully connected layers and normalization layers.

CNNs perform very well while classifying images that are very close to the data set. But if the images have a change of pose in terms of rotation, tilt or some different orientation then CNNs have poor performance. This problem was solved by Hinton's Capsule Neural Network also called CapsNet. A Capsule Neural Network is particularly good at handling different types of visual stimulus and encoding things like pose (position, size, and orientation), deformation, velocity, albedo, hue, texture etc. CapsNet provides a dynamic routing between capsules that delivers rotational and other invariances, which can improve the recognition accuracy even when the scripts are not in their most conventional format. CapsNet has also provided great results on the MNIST database. In a regular neural network you keep on adding more layers. But in CapsNet you would add more layers inside a single layer. A capsule is a nested set of neural layers, in which we add more layers inside a single layer. It uses the dynamic routing or routing by agreement

method for routing between the capsules. Dynamic routing basically means that a lower level capsule will pass the input to a higher level capsule only if it knows that the higher level capsule agrees with the input [1].

The use of CapsNet for Devanagari script recognition has the potential to provide very high accuracy because of its ability to map spatial relationships among the features, which overcomes the drawback in CNN as this information gets lost or ignored in the Max Pooling stage of CNN.

Rest of the paper is organized as follows, Section I contains the introduction of Devanagari Script Recognition using Capsule Neural Network, Section II contains the related work in the field of Capsule Neural Network and Handwritten Character Recognition. Section III contains detailed description about the Devanagari Script, Section IV describes the limitations of Convolutional Neural Networks. Section V gives brief explanation about the concept of Capsule Neural Networks and Section VI deals with working of Capsule Neural Networks. Section VII concludes research works with future directions.

II. RELATED WORK

Paul Gader et al. presented a word recognition algorithm that employed dynamic programming and neural network based inter-character compatibility scores and showed that inter-character information leads to significant improvement in performance [2].

M. Blumenstein et al. used the CEDAR benchmark database and achieved recognition results above 80% by using the conventional method of transforming images to pbm and storing them in a multidimensional matrix. They performed zoning using a window and also formatted the length of strokes. A feature input vector was created which stored all vertical, horizontal, diagonal lines and their lengths plus information of intersections. In contrary to the popular approach of a direction feature, a transition feature was used. It used Backpropagation and Radial Basis Function networks. For experimentation purposes the architectures were modified varying the number of inputs, outputs, hidden units, hidden layers and various learning terms [3].

Abdelhak Boukharouba and Abdelhak Bennia present an efficient handwritten digit recognition system based on support vector machines (SVM). A novel feature set based on transition information in the vertical and horizontal directions of a digit image combined with the famous Freeman chain code is proposed. The main contribution of this work focuses on feature extraction where a novel feature set based on transition information in the vertical and horizontal directions of a digit image combined with the well-known chain code histogram (CCH) is discussed and compared with others in the literature. Their scheme was evaluated on 80,000 handwritten samples of Persian numerals and achieved promising results [4].

Shalaka Deore and Leena Ragha have performed recognition of Handwritten Devanagari characters. By using online and offline data they calculated moments based on Neural Networks. The database used consisted of 100 samples of each 12 characters. Total 1200 characters were present [5].

Punam Ingale has given an insight on Digital image processing. She has explained the fundamental steps in Digital image processing such as image acquisition, preprocessing, multiresolution processing, image compression, etc. Applications of Digital image processing such as Character Recognition and Signature Verification have also been described in detail [6].

P. Umorya and R. Singh have presented a survey on image segmentation. Dividing an image into many regions is the segmentation process, which provides a way to find a particular region of point inside an image. Various segmentation techniques such as Edge based segmentation, Fuzzy based segmentation, ANN based segmentation, etc. have been explained [7].

III. DEVANAGARI SCRIPT

An estimated 300 million people in India use the Devanagari script for documentations. Many official languages in India such as Hindi, Marathi, Sanskrit, Nepali, Konkani and Sindhi use Devanagari as the script for written purposes. Devanagari has 47 primary characters out of which 14 are vowels and 33 are consonants. It also consists of 9 digits. Unlike many other scripts, Devanagari does not have the concept of upper or lower case letters. There are also compound characters in the Devanagari script, which is basically a vowel following a consonant. These compound characters can take modified shapes depending on whether the vowel is placed on top, bottom, left or right of the consonant. They are called as modifiers or ‘matras’. The writing style of Devanagari is from left to right.

An automated recognition of Handwritten Devanagari script is not as easy as its Latin counterpart and must overcome its own set of challenges. Different people have different writing styles. The Devanagari script contains characters that have loops and strokes, which can sometimes get overlooked by the writer unintentionally or out of habit. So while training the recognition system, it is important to take such cases into consideration as well so as to avoid the misinterpretation or incorrect recognition of such characters. Some Devanagari characters and digits are simply mirror images of each other and requires appropriate feature extraction techniques to avoid confusion for the classifiers.

Devanagari script also has modifiers or 'matras' that add to the complexity of the recognition process.

IV. LIMITATION OF CONVOLUTION NEURAL NETWORK

Often, a single feature extraction method alone is not sufficient to obtain good discrimination power. An obvious solution is to combine features from different feature extraction methods. To overcome this, Convolutional Neural Network (CNN) was introduced as it has the ability to build its own features. But in the max pooling stage of Convolutional Neural Network, a lot of information regarding spatial relationships among the different features extracted is lost or ignored. Hence it may only consider the presence or absence of the features and not their order.

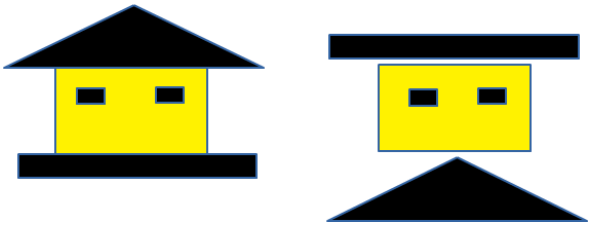

Figure 1. CNN considers both images as a house

Max Pooling is a crutch that made CNN work surprisingly well, achieving outstanding performance in many areas. But although CNN works well due to max pooling, it is losing valuable information in this process. As shown in Fig. 1, CNN will consider both images on the left and right as a house, even though the image on the right is not a correct representation of a house. It only considers the features like the presence of a roof, walls, windows and base, but do not consider the pose and orientation of these features.

V. CAPSULE NEURAL NETWORKS

The limitations of CNN mentioned above can be overcome by using Capsule Neural Networks. It considers a variety of spatial information among features like the pose, orientation, size and so on and uses this information to accurately classify each element in the image. Even if the element to be classified is rotated, tilted or presented in a somewhat unconventional form, the Capsule Networks will be able to correctly identify. In Capsule Networks, the features are represented as vectors instead of scalars, and these vectors can help carry more relative and relational information. Vectors are useful as they help to encode more information like [likelihood, orientation, size] as opposed to only the activation function in CNN.

The dynamic routing or routing by agreement property in Capsule Networks is a major improvement over the max pooling in CNN. It provides a routing between capsules such that the lower level capsule will pass the input to a higher level capsule only when the higher level capsule agrees with the input. Using routing by agreement, we only pass useful information and throw away the data that would add noise to the results. This gives a much smarter selection than just choosing the largest number, like in max pooling.

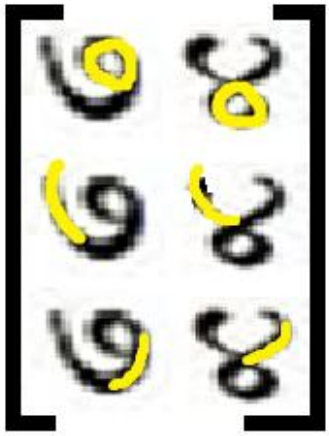

Figure 2. Routing by agreement

Capsule Networks do not require huge number of images to be trained like CNN. It can generalize well using much less training data. They can also handle ambiguity well, hence they perform well on crowded scenes. But it must still improve its performance on backgrounds. CNN may require extra components or layers to determine the object to which a part belongs to. But Capsule Networks give you the hierarchy of parts, as capsules are nothing but a nested set of layers that add inside a single layer.

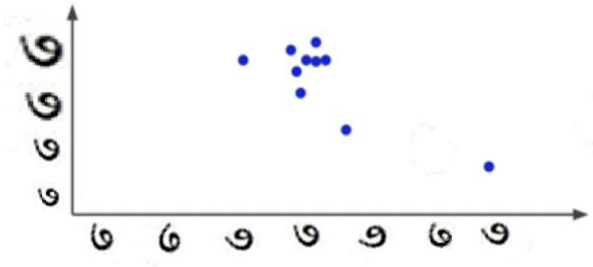

Figure 3. Clustering by agreement where x-axis represents orientation and y-axis represents scaling

VI. METHODOLOGY

Capsule j processes total input vector S_j and outputs activation vector V_j, such that the direction of feature is

preserved and the magnitude is "squashed" to a value between 0 and 1.

A. Equations

The value of V_j is represented in equation form as follows:

$$V_j = \frac{||S_j||^2}{1 + ||S_j||^2} \cdot \frac{S_j}{||S_j||}$$

In the above equation, $\frac{||S_j||^2}{1+||S_j||^2}$ denotes magnitude and $\frac{S_j}{||S_j||}$ denotes direction.

S_j is calculated by processing all the vectors connected to it by previous layer *i*. This process can be denoted as:

$$\hat{u}_{j|i} = w_{ij} . u_i$$

We multiply u_i with w_{ij} which is a weight matrix, whose multiplication with u_i helps to generate a "prediction" vector *i.e.* $\hat{u}_{j|i}$ to approximate V_j. $\hat{u}_{j|i}$ from each capsule is weighted by a coupling coefficient C_{ij}, which is calculated via the dynamic routing algorithm. Hence, the sum of coupling coefficients between capsule *i* and all possible capsules *j* in the next layer equals 1.

$$S_j = \sum_i C_{ij} \ . \hat{u}_{j|i}$$

The inputs for C_{ij} is based on b_{ij} which is log prior probabilities that capsule *i* and *j* are coupled.

$$C_{ij} = \frac{exp(b_{ij})}{\sum_k exp(b_{ik})}$$

The agreement a_{ij} between the prediction vector and activation vector should be maximized;

$$maximize\ a_{ij} = \hat{u}_{j|i} . V_j$$

To update coupling coefficients, b_{ij} is iteratively incremented by a_{ij} through iterations of dynamic routing algorithm.

VII. CONCLUSION AND FUTURE SCOPE

The recognition of Devanagari Script is difficult because of its various strokes and loops and the variations that come along with it in terms of writing styles and shape discrimination between the characters. We have presented a novel approach for recognition of Handwritten Devanagari Script by using the Capsule Networks. Routing by agreement in Capsule Networks helps to overcome the limitations of Max Pooling in Convolutional Neural Networks and recognize similar characters more accurately as compared to Convolutional Neural Networks.

In future work, we are planning to enhance our proposed model by introducing Concavity feature extraction technique. Concavities are structural features based on measurements of character concavities. Lastly, we can extend this model for use in online character recognition.

REFERENCES

[1] Sara Sabour, Nicholas Frosst, Geoffrey Hinton , "*Dynamic Routing Between Capsules*", In the Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, **USA, 2017.**

[2] Paul Gader, Magdi Mohamed, "Handwritten Word Recognition with Character and Inter-Character Neural Networks", IEEE Transactions On Systems, Man, And Cybernetics-Part B: Cybernetics-Part B: Cybernetics, Vol.**27,** Issue.**1**, pp. **158-164, 1997**.

[3] M. Blumenstein, B. Verma, H. Basli, "A Novel Feature Extraction Technique for the Recognition of Segmented Handwritten Characters", In the Proceedings of ICDAR, pp. **137-141, 2003**.

[4] Abdelhak Boukharouba, Abdelhak Bennia, "Novel feature extraction technique for recognition of handwritten digits", Applied Computing and Infomatics, pp. **20-26, 2017**.

[5] Shalaka Deore, Leena Ragha, "Moment Based Online and Offline Handwritten Character Recognition", CiiT International Journal of Biometrics and Bioinfomatics, Vol.**3,** Issue.**3,** pp. **111-115, 2011**.

[6] Punam Ingale, "The importance of Digital Image Processing and its applications", International Journal of Scientific Research in Computer Science and Engineering, Vol.**6,** Issue.**1,** pp. **31-32, 2018.**

[7] P. Umorya, R. Singh, "A Comparative Based Review on Image Segmentation of Medical Image and its Technique", International Journal of Scientific Research in Computer Science and Engineering, Vol.**5**, Issue.**2**, pp.**71-76, 2017**.

Authors Profile

U. M. Sawant is a student currently studying in Modern Education Society's College of Engineering, Pune. He is pursuing Bachelors in Computer Engineering in a 4 year program.

R. K. Parkar is a student currently studying in Modern Education Society's College of Engineering, Pune. He is pursuing Bachelors in Computer Engineering in a 4 year program.

S. L. Shitole is a student currently studying in Modern education Society's College of Engineering, Pune. He is pursuing Bachelors in Computer Engineering in a 4 year program.

S. P. Deore is an Assistant Professor currently pursuing her Ph.D.[CSE]. She has published her works in 11 Journals and 7 National Conferences.