Dynamic Routing Using Inter Capsule Routing Protocol Between Capsules

Sanjib Kumar Sahu
GGS Indraprastha University
Delhi, India,
sahu_sanjib@rediffmail.com

Pankaj Kumar
USICT, GGS Indrapastha University
Delhi, India,
pankaj786067@gmail.com

Amit Prakash Singh
USICT,GGS Indrapastha University
Delhi, India
amit@ipu.ac.in

***Abstract* - A capsule is a combination of multiple neurons, which designed to analyze specific feature representations in an image, a capsule network resembles with CNN multiple layer network model, in which convolutional and ReLu function is followed by max pooling which help in reducing the input values for processing. However, we observed that for each capsule in layer ℓ to layer (ℓ +1), we need to calculate a prediction vector, whose magnitude defines the detected feature in an image and the orientation define the pose of the object in an image. Now calculating the prediction vector for each capsule in the lower layer is prolonged. This is because, in capsule network, we are not securing the gained knowledge for future use. In this paper, we are introducing a protocol named as ICRP (Inter Capsule Routing Protocol), which uses FDM (Feature Detection Matrix) to store the learned information at every iteration in the capsule network. This reduces the processing time for feature detection in an image gradually, by searching the similar feature in FDM before processing the network.**

Keywords – Capsule networks, Neural Networks, CNN, Softmax, ReLU, Decoder, MNIST, ICRP, FDM

I. INTRODUCTION

In a neural network, CNN is currently the state of art for all image detection and computer vision problems and they have performed exceptionally well in image representation by detecting objects in an image, reading handwritten digits, feature extractions with different representational format. As CNN based on simple fact and mechanism in which the neural network is trained over a large datasets, and make itself capable of predicting the output using the translated replicas of learned feature detectors, this provide an advantage of using the learned weight values at one position in an image to the other positions which is further used to interpret selected features within an image. CNN uses three key concepts which are, shared weight and biases, local receptive fields and activation and pooling. Rectified linear unit ReLU, is the activation function used in Conv net for transformation of output of each neuron to the highest positive value and mark zero for negative values, further the output of the activation step is transformed using the pooling concept. Pooling reduces the dimensionality of the featured map by condensing the output of small regions of the neuron, into a single output by selecting the maximum value in each subset. However, CNN has some limitations which biased these layers to detect only the features irrespective of their spatial orientation in an image. A paper by Hinton in NIPS explains this limitation and instead suggested a solution to this problem by introducing a new concept called as capsule networks.

Capsule network is the idea given by Hinton in his paper "Dynamic Routing between Capsules" et al. (2017) [10]. A capsule is a collection of multiple neurons in a hidden layer of a neural network, which uses intensity vector and pose matrix to identify the features and orientation of an object in an image respectively. To overcome the limitation of CNN by capsule network focus on two major features of image representation. First is to use equivariance instead of invariance for an object detection in an image which allows to consider all the active pixel intensity and forward the output to selective capsule in the upper layer which has a higher probability score to detect similar feature.

Second, is the spatial orientation, which is represented by the pose matrix for an object in an image, it provides the orientation of an object with respect to other objects in the image. Unlike CNN, capsule net is better in understanding the spatial orientation of features in an image which result in better image representation and feature detection. Each hidden layer is divided in multiple capsules and each capsule is a collection of multiple neurons, which have associated activity vector which represent the instantiation parameters of specific a type of entity object in an image.

In this paper, we present a protocol through which information gain by one capsule is available for other capsule in a layer to process, which speeds up the learning mechanism and reduce the training time for each layer of capsule. As Hinton stated in his paper, an activity vector has two components, magnitude and pose. Magnitude defines the probability of the availability of specific features in the upper layer capsule and pose defines the spatial orientation of that object in the image, with respect to other objects. Using this concept, a coupling coefficient

0-7695-6405-4/18/$31.00 ©2018 IEEE
DOI 10.1109/UKSim.2018.00012

vector $n_{i/j}$ is defined as:

$$n_{i/j} = \frac{\exp(b_{ij})}{\sum_k \exp(b_{ik})} \quad (1)$$

This is defined by routing softmax function, which discrimatively learn the log prior with same time as all the other weights, this coefficient defines that couple i mapped to couple j, for a specific feature in an image. Using (1), coupling coefficient calculate the weighted sum of all the input from primary layer vectors and their respective weights, which is also called as the prediction vectors. Let say if the prediction vector is $p_{i/j}$, so this defines as:

$$\vec{p}_{i/j} = \sum_i W(i) * u(i) \quad (2)$$

The prediction vector is defined as the magnitude of the activity vector which gives the probability that a capsule c_i in lower layer, should be routed to capsule c_j in upper layer, for a specific feature, this can be calculated using (2). Initially, the activity vector is associated with zero weights and it passes all the information to the next level capsules, but as the system start learning the features and adjusting the weights based on ReLU activation function, the probability of identifying the similar set of feature in the upper layer of capsule increases.

Now using (3), for all the capsule in the second layer, we define the set of input vectors which should be between 0 and sum to 1 and all the set of inputs I is defines as

$$I = \sum_i n_{i/j} * \vec{p}_{i/j} \quad (3)$$

Using the squashing function (4), the given vector output of capsule j as v_j. , this iterative dynamic routing with capsule is the example of routing by agreement.

$$v_j = \frac{||I||^2}{1+||I||^2} \frac{I}{||I||} \quad (4)$$

$$v_j = ||I|| I \quad for\ I\ is\ short \quad (5)$$

$$v_j = \frac{I}{||I||} \quad for\ I\ is\ long \quad (6)$$

Hinton has given a routing algorithm between capsules which iterate for all the capsules in layer ℓ to all other capsules in layer ℓ+1. The squashing function is used to keep the value of the magnitude of the vector sum within the range of 0 to 1 with the pose matrix, as per (5) and (6). However, marginal loss is also calculated based on the length of instantiation vector, to represent the probability that a capsule entity exists. For a face detection using capsule network, it will look for all the features like eyes, ears, mouth, nose etc. and based on their spatial orientation it will figure out the decision whether it is face or not.

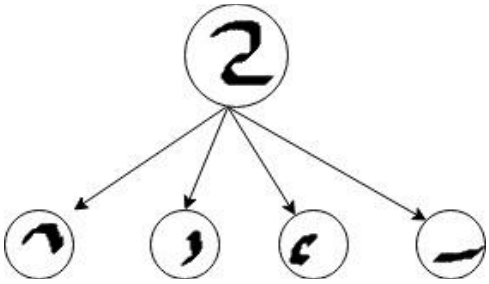

Fig.1 Numerical digit from MNIST dataset

In the MNIST dataset, as we can see in Fig 1, we have multiple hand written scripts in which, if we pass one digit to a capsule, it analyzes its representation and state out features like edges, curves, extended edge and spatial orientation. Human brain is really good at interpreting these images when shown, so if there is a mirror image of any alphabet, human brain first convert it to its original orientation and then compare it with the default image and make an interpretation that the letter is belong to a special set of class. In CNN, there is an issue with max pooling, it cannot handle the translational variance if the image is slightly twisted, it can still detect the digit but the problem is that it throws away all other information, but the max feature from the set of pixel and feed them to feed forward network, which is use for object detection, but as we can see and stated in the paper by Hinton, capsule network is capable of analyzing the magnitude as well as the orientation of the image.

Fig 2. Handwritten MNIST dataset

However, as the capsule networks are currently state of the art, in image representation and feature detection, it also has some limitations which are stated in a different paper on capsules network, “Matrix capsules with EM Routing” et al. [12]. According to this paper, the length of prediction vector, less than 1 prevent them from a sensible objective function that is minimized using the iterative routing procedure. As per EM Routing, cosine is used to represent the angle between pose matrices, which is not a good in distinguishing good and very good agreements, unlike Gaussian cluster. So they suggested a method of using the expectation maximization routing mechanism. The objective of this approach is to group capsules to form a part-whole relationship with a clustering

technique. EM Routing, is used to cluster data point into the different Gaussian distribution.

II. RELAT ED WORK

For object detection and feature recognitions in an image, there are multiple recent attempts which gives an underlying affine manifold on gaining viewpoint invariant to equivariant. Spatial transformer networks by Jaderberg et al. (2015) [1], gives the ideas of viewpoint invariant by changing the CNN according to the selection in the affine transformation, this theory was extended by De Brabandere et al. (2016) [2], for spatial transformer networks where the filter are learned during the change depending on input. This generates different filters for each spatial transformer in a feature map instead of applying a similar transformation to the all the filter in the network. Standard convolutional neural network by Dai and Xiong et al. (2017) [3], also talks about STNs, which generalize the simple method of filters among the layers which make it more flexible.

Due to the high success rate of CNN, many researches have extended to translational equivariance, which built the CNN using rotational equivariance according to Cohen and Welling et al. (2016) [4], Dieleman et al. (2016) [5]. A recent approach in harmonic networks by Worrall et al. (2017) [6], show how circular harmonic filters can be used to achieve rotation equivariance which return both the maximal response and complex number orientation. This gives us the basic representational idea of capsules, if we assume that there are only one instance of an entity of at a specific location, we can represent it properties by using several different numbers, they fixed the number of streams of rotation order by patch wise rotation equivariance.

As per the idea from Fasel and Gatica-Perez et al. (2006) [7], this approach is more efficient with respect to parameters than data augmenting approach. Symmetry network given by Gens and Domingos et al.(2014) [8], explained the Lucas Kanade optimization to find poses which are supported by lower level feature in an image, the only drawback of this approach is that the iterative algorithm always start at the same poses instead of bottom up votes mean. Lenc and Vedaldi et al. (2016) [9], propose a feature detection mechanism DetNet, which is equivariant to affin transformation. It is designed to detect same viewpoint in an image from different spatial viewpoint. This can be used to implement the pre-rendering of images, initially which activates the primary layer capsules.

According to the model given by Chang and Chen et al. (2015) [11], for deep convolutional network, this is an extended work on the similar architecture. In a recent research paper given by Hinton and team et al. (2017) for Matrix capsule with EM routing, it shows the limitation of capsule network and gives the ideas of maximizing routing, which means the competition is between the higher level capsule to lower level, to which it sends its vote based on which the capsule in the upper layer is decided and which is best suited for feature detection.

III.EXPERIMENT AND RESULT

In this paper, we propose a mechanism to improve the accuracy of capsule network by introducing inter capsule routing protocol (ICRP), which uses activity vector output from the lower layer capsule and selects capsule from upper layer for a specific feature in an image, once the feature is detected and the gained knowledge is stored in a (n x m) feature detection matrix (FDM), shown in Fig (3), where n is the total number of capsules in the lower layer, and m is the total number of capsules in the upper layer.

$$\begin{bmatrix} u_1\ v_1 & \cdots & v_n \\ \vdots & f_i & \vdots \\ u_n & \cdots & u_n\ v_n \end{bmatrix}$$

Fig 3. Frequency Distribution Matrix FDM [u,v]

In order to detect images with different orientations of the same object, capsule network gives outstanding performance, but it needs to be trained first on multiple datasets and images to select a specific vector in the upper layer with respect to that activity vector. So using FDM (Fig 3), we can speed up the process by iterating for only those features which are not present in the FDM, as per Hinton, he has used MNIST dataset for dynamic routing.

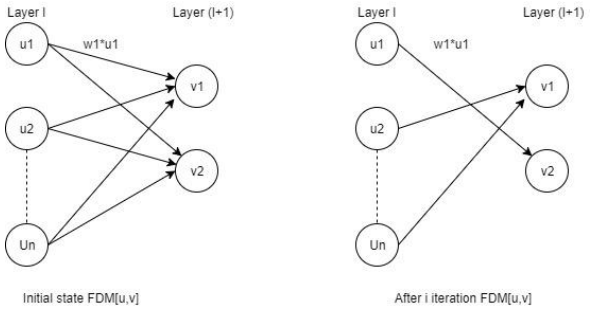

Fig 4. Capsules in lower and upper layer using FDM

However, in Fig 4, we can see how lower layer capsule interact with upper layer capsule using FDM both before and after. We have much complex images in the real world and an image can have multiple object with different features or different objects with same features. At the start, it takes time to populate FDM, but as the system start learning, FDM will be smarter, and informative, which will, gradually decrease the processing time. In routing algorithm from Hinton, we can see an extra loop in the routing algorithm, which executes every time we need to map the lower level capsule to the upper

level capsule. For all the lower layer capsule u in the layer ℓ and capsule v in the upper layer (ℓ +1), connect with each other to identify the initial state and identify the lower level features, like edges and corners, these feature detections are also important for image representation.

Our routing protocol can re-used in the second iteration as well. Once a capsule is capable of identifying a specific set of features and knows which upper layer capsule has the highest probability for detecting similar processed feature, it will return from FDM without reprocessing. So by using this mechanism we can reduce the an external loop calculation up to 30% of the time and which will allow the system to learn fast with limited training labels.

Also, in a capsule network, since the pose and magnitude in activity vector, represents the 2D feature results for an image, it will not be an easy task to determine the image showing same feature multiple time, or defining an overlapping, this is what LeCun's has stated in his paper.

The purpose of introducing FDM, is to minimize the overhead of recalculation. This also helps in identifying multiple instantiation of a single object with different pose in a single image. This leads to an idea of imaginary cell, which can predict features based on the related information it gathers in FDM.

Here we show, modified architecture of Hinton's capsule network with FDM.

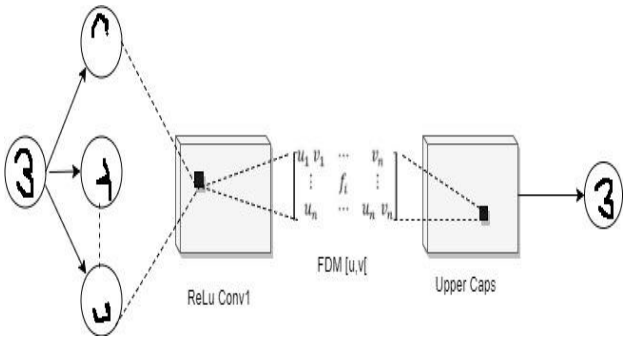

Fig 5. Architecture using Conv1 and FDM [u,v]

In Fig 5, for each capsule u in layer ℓ, and v in layer (ℓ+1) we create a correlation of probabilities between two layers and update FDM.

As per the Hinton's paper this is only implemented over the MNIST dataset, only handwritten digits. However, this concept can be extended to more complex images, having objects like cat, dogs, bicycle, house, etc. In comparison with CNN, capsule network needs to learn much more and be more flexible in terms of identifying the in depth feature understanding and related correlations in an image.

The algorithm for capsule network using FDM is given below:

Procedure 1 Inter Capsule Routing Protocol

1. **procedure** ICRP(u_i, f_i, v_i)
2. for all the u_i in layer l, with feature f_i, searching for the suitable v_i in layer (l+1)
3. Create a u_i x v_i matrix, FDM[u_i, v_i], where u_i is the layer capsule and v_i is upper layer capsule
4. For i iteration do
5. if (FDM[u_i, v_i]<> NULL)
6. return FDM[u_i, v_i]
7. else
8. $\forall\ u_i$ in layer l, c_i = softmax(b_i)
9. $\forall\ v_i$ in layer (l+1), $s_i = \sum_{i=1}^{n} c_i \, . \, v_i$
10. $\forall\ v_i$ in layer (l+1), v_i = squash(s_i)
11. Using hinton's dynamic routing algorithm find next layer capsule, v_i for feature f_i
12. Add f_i to FDM[u_i, v_i]
13. **return** FDM[u_i, v_i]

Using ICRP we can reduce the processing time for dynamic routing protocol up to approx. 20%. We have used MNIST dataset for identifying the handwritten digits. During processing, FDM will be updated at every iteration, this is one of the limitation of this approach. However, after n iterations, where n can be small or large based on the complexity of an image, this approach is much more efficient that executing the dynamic routing protocol.

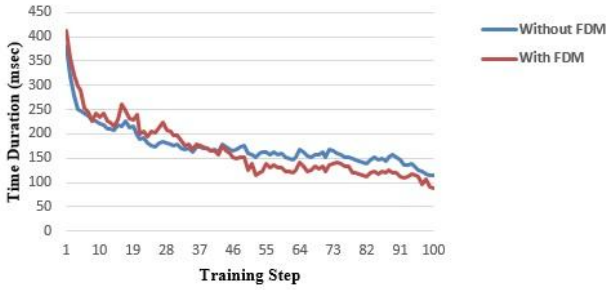

Fig 6. Comparing result with and without FDM [u,v]

In Fig 6, we present the comparison the result before and after using FDM on capsule network, we can see that at the start using FDM is more profound, but as soon we increase the training step it gives us better results. We have used the results from 100 training steps and their respective processing time which and compared the result using line graph. We have used 100 training steps and the accuracy of the system 94.5%. Based on this analysis, we believe that if we execute this on a large dataset we can see a substantial gain in performance of a capsule network.

The limitation of this algorithm as already discussed in the above section is updating FDM at every iteration. In this paper, we are proposing an approach to minimize the image processing time. This could also lead us to process more complex feature detection, but we have not tested this system on such images. Hypothetically, the idea of implementing a virtual capsule network (VCN), could possibly solve this problem but we are working on this concept, which we will continue to explore in our future work.

IV. CONCLUSION

CNN has become the most effective approach for feature detection and speech recognition and has outperformed other models like recurrent neural network or Markov model with Gaussian mixture. However, it is an open question to explore whether there is an exponential inefficiencies which can be the potential issue with this model. As per Hinton's paper et al. (2011) [13], capsules avoid these exponential inefficiencies by converting pixel intensities into a vector of instantiation parameters of recognized fragments. For large and complex visual entities, agreement between poses can be better predicted by lower layer capsules. Capsules also makes very strong representational assumptions and are very good in dealing with segmentations, by using routing by agreement.

Object recognition in an image requires powerful computation and large processing time. Using ICRP, we explained how we can reduce the processing time for capsule network by introducing FDM in this paper, which stores the gained knowledge in a matrix. Instead of encoding the instantiation parameters by activating high dimensional point in the grid, distributed representation is exponentially more efficient, and this spatial relationship can be modelled better by matrix multiplication. The concept of ICRP is inherited by dynamic programming, where each new computation done by network contributes to the information gain in the entire layer. Research in ICRP is still worth exploring, but it require lot more insight before it can enhance the performance of a capsule network by a considerable factor.
Our future scope is to extend this idea for more complex images and to study the impact and efficiency of ICRP on capsule network. To optimize the efficiency of capsule network model is still in progress, but we believe using this concept, we can see a considerable amount of improvement in processing time for object recognition, which gives us the confidence to continue our research in this domain.

V. REFERENCES

[1] Max Jaderberg, Karen Simonyan, Andrew Zisserman, and Koray Kavukcuoglu. Spatial transformer networks. In *Advances in Neural Information Processing Systems*, pages 2017–2025, 2015.

[2] Bert De Brabandere, Xu Jia, Tinne Tuytelaars, and Luc Van Gool. Dynamic filter networks. In *Neural Information Processing Systems (NIPS)*, 2016.

[3] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. *arXiv preprint arXiv:1703.06211*, 2017.

[4] Taco Cohen and Max Welling. Group equivariant convolutional networks. In *International Conference on Machine Learning*, pp. 2990–2999, 2016.

[5] Sander Dieleman, Jeffrey De Fauw, and Koray Kavukcuoglu. Exploiting cyclic symmetry in convo-lutional neural networks. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ICML'16, pp. 1889–1898. JMLR.org, 2016.

[6] Daniel E. Worrall, Stephan J. Garbin, Daniyar Turmukhambetov, and Gabriel J. Brostow. Harmonic networks: Deep translation and rotation equivariance. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[7] Beat Fasel and Daniel Gatica-Perez. Rotation-invariant neoperceptron. In *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, volume 3, pp. 336–339. IEEE, 2006.

[8] Robert Gens and Pedro M Domingos. Deep symmetry networks. In *Advances in neural information processing systems*, pp. 2537–2545, 2014.

[9] Karel Lenc and Andrea Vedaldi. Learning covariant feature detectors. In *Computer Vision–ECCV 2016 Workshops*, pp. 100–117. Springer, 2016.

[10] Geoffrey E. Hinton, Sara Sabour, Nicholas Fross, Dynamic Routing Between Capsules, Google Brain, arXiv:1710.09829v2 [cs.CV] 7 Nov 2017

[11] Jia-Ren Chang and Yong-Sheng Chen. Batch-normalized maxout network in network. arXiv preprint arXiv:1511.02583, 2015.

[12] Matrix Capsules With EM Routing, Anonymous authors, Under review as a conference paper at ICLR 2018

[13] Geoffrey E Hinton, Alex Krizhevsky, and Sida D Wang. Transforming auto-encoders. In International Conference on Artificial Neural Networks, pages 44–51. Springer, 2011.