International Journal on Emerging Technologies 10(4): 43-49(2019)

ISSN No. (Print): 0975-8364
ISSN No. (Online): 2249-3255

Image Inpainting through Textures Synthesis using Spiking Neural Networks

Vineet Kumar[1], A. K. Sinha[2] and A. K. Solanki[3]
[1]Noida Institute of Engineering & Technology, Gr. Noida, (U.P.), India.
[2]UST Software India Pvt Ltd., New Delhi, India.
[3]Department of Computer Science & Engineering, BIET, Jhansi, India.

(Corresponding author: Vineet Kumar)
(Received 27 September 2019, Revised 30 October 2019, Accepted 01 November 2019)
(Published by Research Trend, Website: www.researchtrend.net)

ABSTRACT: In this paper, we are showing how spiking neural networks are applied in image repainting, and its results are outstanding compared with other machine learning techniques. Spiking Neural Networks uses the shape of patterns and shifting distortion on images and positions to retrieve the original picture. Thus, Spiking Neural Networks is one of the advanced generations and third generation of machine learning techniques, and is an extension to the concept of Neural Networks and Convolutional Neural Networks. Spiking Neural Networks (SNN) is biologically plausible, computationally more powerful, and is considerably faster. The proposed algorithm is tested on the TUM-IID dataset of images which contains 17 different texture and complex structure. Filling the hole (missing parts) with maintaining the texture and structure of the image so that it looks like an original image is the main challenge of image inpainting. The performance of the algorithm is examined to find the PSNR, QF, and SSIM. The model has an effective and fast to complete the image by filling the gaps (holes).

Keywords: Biological Neuron Model, Image Inpainting, Machine Learning, Spiking Neural Networks, STDP rule, Response Entropy.

I. INTRODUCTION

Image completion refers to complete the image by filling the missing part of an image with the best plausible scene. In computer vision theory, Image completion also known as image inpainting. Many researchers have proposed a state of art techniques to fill the missing region of the image by finding the best pixel or patch near the missing region boundary. In the evolution of machine learning and deep learning algorithm, researchers gain more attention tothe fast completion of the image.

a) Damaged Image b) Inpainted Image.

Fig. 1.

In this paper, we proposed a Spiking Neural network for fast and efficient completion of the image.Spiking neural networks is a machine learning technique that has involved in all domains including optimal pathfinding and security domains [1]. From the past four years, SNN is purely involved in all aspects and solving many problems, as it has an excellent scope and powerful algorithm with spikes, the performance, and accuracy calculated using SNN is outstanding. Because of the close representation of the concept of the human brain, spiking neural networks is the most advanced version of neural networks [2]. In the workflow of SNN, it has got a Synaptic, neural state and included the idea of time. SNN is purely based on the activity of neurons.

In SNN's, neurons are active only if the electrical potential difference between two neurons exceeds the predetermined value [3]. Once the neurons are activated, then the generated spikes reach other neurons, and neurons potential gets changed in favour of received value, i.e., the potential value gets either increased or decreased. Depending on the interval between spikes and frequency of the spikes, the number of coding methods are used for interpreting the outgoing spike as a real-valued number or not.

The network consists of spiking neurons, and each neuron is randomly connected with each other with axonal conduction delays. The network has spike-timing-dependent plasticity (STDP) [4]. Synaptic connections among neurons have a fixed conduction delay, which is expected as a random integer that could vary between 1ms and 20ms [5].

The advantage of the spiking network is that the output can be represented in sparse in time. SNN conquers the energy consumption compared to the biological system for the spikes, which have high information [6]. In Spike Neural network, sparse codes are beneficial for restoring the missing pattern in an image using learning rules. It is useful to complete the complex curved structure and is effective in energy saving [7].

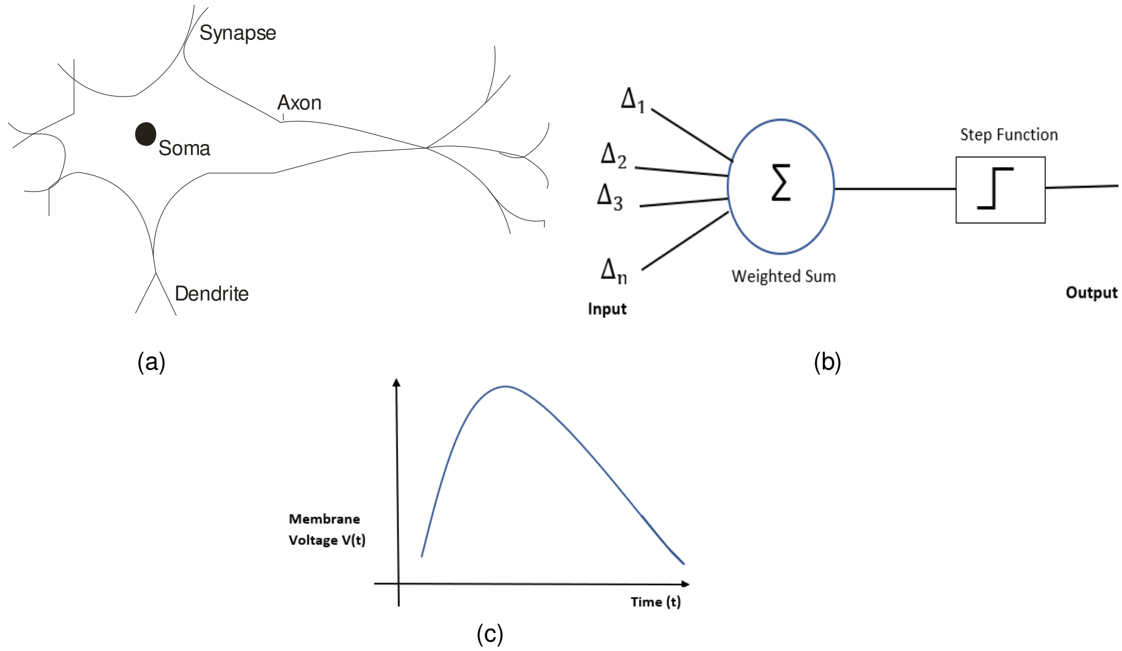

Fig. 2. (a) Biological representation of Neuron. (b) An Artificial Neuron Model. (c) Graphical Presentation of Spike in response to Inputs when a neuron is integrated and triggered.

II. LITERATURE SURVEY

It is important to classify the objects in an image so that the process solves many issues related to image processing and computer vision [8]. Generally, the image classification problems are mostly affected due to the factors like noise, poor quality of the image, and occlusion. Image inpainting or completion refers to filling the lost part of an image with the most likely plausible visual scene. Many researchers explored inpainting methods. Guillemot & Meur [3] classified the inpainting methods in to diffusion-based inpainting method [9-19] which propagate the local structure from the outside boundary of the inpainting region to the interior hole and exemplar-based inpainting methods [20-24]. Diffusion based inpainting methods used to diffuse the information in linear/nonlinear, isotropic/anisotropic direction in maintaining the curvature structure exist in the surroundings of the missing part.These models are not well suited to fill the large missing region and not able to maintain the textual properties of the image. On the other hand, exemplar inpainting techniques copy the patches (exemplar) nearby the holes or known and paste this patch into the inpainting region. These methods are suitable for reconstruction of the texture of the Inpainted area.

Recently computer vision problems gain much attention due to emerging of deep neural networks. Most of the image classification techniques are based on the features that are extracted from an existing image by training the artificial neural network in a supervised and unsupervised manner [25].

If the learning process of an image is designed to form a mapping from one set of data, say features to another set of data, which could be information classes, under human intervention is called supervised classification. Unsupervised classification is the same as supervised classification, but it does not require any human intervention [9]. For assigning pixels to informational classes, some of the supervised classifications techniques are Artificial Neural Networks (ANN), Support Vector Machine (SVM), Minimum Distance from Mean(MDM), Maximum Likelihood(ML) [26].

The Support Vector Machine (SVM) [27] applies to pattern recognition as well as regression. It is a new universal machine. Generally, in supervised classification, there is a human intervention due to which, if errors are made, those could be detected and corrected during training. But this classification deals with high costs and consumes more time. No prior information is required in unsupervised classification since it is free from human intervention. Using statistical methods such as clustering algorithms, it is possible to understand the structure of the data when reliable training data are absent.

K-means and ISODATA are popular clustering algorithms that are faster and errors free, and it is not necessary to have detailed prior knowledge. Major drawbacks are due to maximally separable clusters in this technique [28-29].

Many machine learning methods were proposed by the research, which is used in image classification, object detection, pattern recognition speech recognition, data analysis, etc. some of these methods are decision tree, feed-forward artificial neural network, Bayesian network, SVM [30, 31]. Thus, ANN has its key role everywhere because of its advantages, such as fault tolerance, working with incomplete data as it gets trained for once. But the main drawback is, it takes more time for training the sets which contain millions of patterns and many features. That is the reason; It is important to have limited important features from every layer, which can be processed by the next layer which makes the classification better. Vectorization, which played an important role in scaling up the neural network model is a process of transforming the actual data structure into a vector form. It has introduced deep learning, which is a part of machine learning, which generally works on various layers. Here, the output of every layer acts as an input of other layers. This is how; it helps both supervised and unsupervised learning. Some of the deep learning networks are CNN (Convolutional Neural Network) [32], SAE (Stacked Auto Encoder) [33], and RBM's (Restricted Boltzman Machines) [34-35] which helps in extracting useful data from digital images.

CNN uses several layers like input, output layers, and hidden layers like pooling, fully connected layer, convolution layer, and normalization layer. CNN is a hierarchical structured artificial neural network and was the first one, which is based on neural connectivity. It was found in the mid-1980. The algorithm which was proposed was a multi-layered network made up of neurons that deal with a problem with shifts in distortion in images and positions in shape of patterns.

It is important to fill the missing pixels of any image, which is known as image inpainting. The challenges of image inpainting are synthesizing semantically plausible and visually realistic pixels, where the areas of missing of pixels of an image occur. To solve the problem we have, SNN and generative adversarial networks (GAN) [36], which formulates inpainting as an image generation problem. GAN – based approaches and deep learning for image inpainting, has a process of training SNN for small denoising areas of the image.

A spiking neural network is used as an intention to replicate the human brain. It happens by implementing individual spike, and it includes spatial, temporal information in cross-layer connection. Neurons use pulse coding, which means they process individual pulses that allow image multiplexing.

In spiking models, spiking artificial neural networks have the internals, which is the same as a biological analogy. It receives information coming from many inputs and produces them as a single spike response. As the excitatory inputs increases, the probability of spike generation increases, and it decreases by inhibitory inputs. When neurons get activated by reaching the threshold value, spikes are generated as a response [37].

$$y_j = \begin{cases} 1, & if\ \sum_{i=1}^{n} w_{ji} x_i \geq \theta \\ 0, & if\ \sum_{i=1}^{n} w_{ji} x_i < \theta \end{cases} (1)$$

Where x_i is input, w_{ji} is the weight which denotes the synaptic energy, and y_j is a spike response.

In this paper,the spike neuron network is used to generate the feature of images and using those features, the missing regions of the image are completed. The proposed technique is based on creating a neural network which is presented in [38]. Paper is organized intofive sections. Section I refers to an introduction to image completion and a spike neural network. Section II deals with the literature of existing methods on image inpainting (completion) and approaches. Section III elaborateson the proposed model. The results are discussed in section IV. In this section, results are also compared with the nearest neighbour approach. Section V concludes the results. And finally, in section VI, we have presented the future scope.

III. PROPOSED METHODOLOGY

A. Spiking Neural Networks as a tool

The spiking neural network training process is generally divided into a few steps [39,40]. Since our model consists of many RBM's, they should be trained individually.

- An independent RBM determines the weight of the coefficients within the input and abstract layers.
- Later, supervised learning between the Associative and Label layer is established. But as input, previously trained RBM (Input and Abstract) are used in the Associative layer.
- Every RBM must be trained for predefined epoch times.

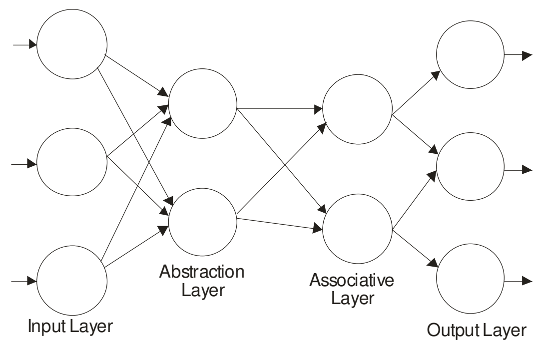

Fig 3. The architecture of Spike Neural Network.

B. Spike Timing Dependent Plasticity (STDP) Rule

STDP is a biological process that adjusts the strength of connections between neurons in the brain. The process adjusts the connection strengths based on the relative timing of a particular neuron's output and input action potentials (or spikes) [41].

Neurons fire stochastically as a function of membrane potential.

S (neuron j spikes at time t) = k$\boldsymbol{e}^{(\boldsymbol{\gamma i(t)-T})/\boldsymbol{m}}$(2)

Where k is constant, t is the time taken, and T is total time taken $\boldsymbol{\gamma i}$is neurotransmitter concentration.

Good idea to minimize response variability: -

Response entropy:

$$\text{H}(\Omega\text{i}) = -\int_{\Omega i}^{n} \boldsymbol{G}(£)\boldsymbol{log}\big(\boldsymbol{G}(£)\big)\boldsymbol{d}£ (3)$$

$G(£)$ is defect identified and $log\big(G(£)\big)d£$ gives the border of the defected image

Gradient:

$$\frac{\partial H(\Omega i)}{\partial \omega ij} = -\int_{\Omega i}^{n} \boldsymbol{G}(£) \frac{\partial log(G(£))}{\partial \omega ij} \left(\boldsymbol{log}\big(\boldsymbol{G}(£)\big) + \mathbf{1}\right) \boldsymbol{d}£ \quad (4)$$

This equation shows how the mask of the image is inpainting the defected image using train data [42,43].

The SNN algorithm [43] has been proposed based on the Learning algorithm for a one-layered Spiking Neural Network. In this algorithm, we apply the derivation of the learning rule. We used a gradient descent method to decrease the error landscape proportional to the derivative of the error. Parameter Optimization is an important tool used in the learning algorithm. At each layer of the SNN model, features are extracted for the reference and generate an image. Then errors(Є) between correlation matrices are calculated. We can use the L-BFGS algorithm [44] for computing results. For fixing patches, we use SNN.

Let X_p is the patch to be filled, then $Q=\{X_q\}$ where q=1 to N is the top N most similar patches with X_p, which are collected from source 'ω'.Let dataset be 'D'.

Algorithm:

1.Start

2.Preprocessing: segmentation of frames

3. Image inpainting:

3.1. Train image dataset

3.2. If D=small,

Then, fix all the weights using SNN (feature extractor) and

Retrain the only classifier

3.3. Swap the SoftMax layer at the end

3.4. If D=medium,

Use old weights for initialization in SNN,

a) Set the initial parameters.

b) Initialize v = 1 to n
Where v=output of neuron

c) Calculate the partial derivative of ω(source) with respect to I (output) in accordance with the Є(errors of the source)

d) Compute gradient(G) of potential.

e) If G<0.1,
Then G=0.1
End of the loop

f) else loop

g) Weights are initialized to any random values.

h) For u=1 to m
where u is the input of neuron.

i) Consider weights of the connections from input u to output v.

j) Calculate partial derivative networks all weights of the gradient, input and output neuron.

k) If there are any weight-changes, then calculate the delta weight of input neuron to output neuron with delay time k.

l) End for

3.5. Train some of the higher layers

3.6. Retrain bigger portion of the network usingSNN

4. End.

IV. RESULTS AND ANALYSIS

Spike Neural Network performance is tested by providing the input image with the same sized masked image. This mask image will overlap on the original image, and the outer line is going be framed once the neural network is trained by computing the feature. The network first detects the edges and hole exist in the images. These holes are filled by finding the best suitable pixel from the boundary of the hole. The best suitable pixel is selected based on the proposed model. The proposed model performance is evaluated on the three different parameters which are popular in computer vision. The parameters, PSNR to measuring the correctness of images after removal of the noise, Quality factor (QF) for measuring the correlation of pixel of different images, and for finding similarity in structure, structure similarity index (SSIM) are used.

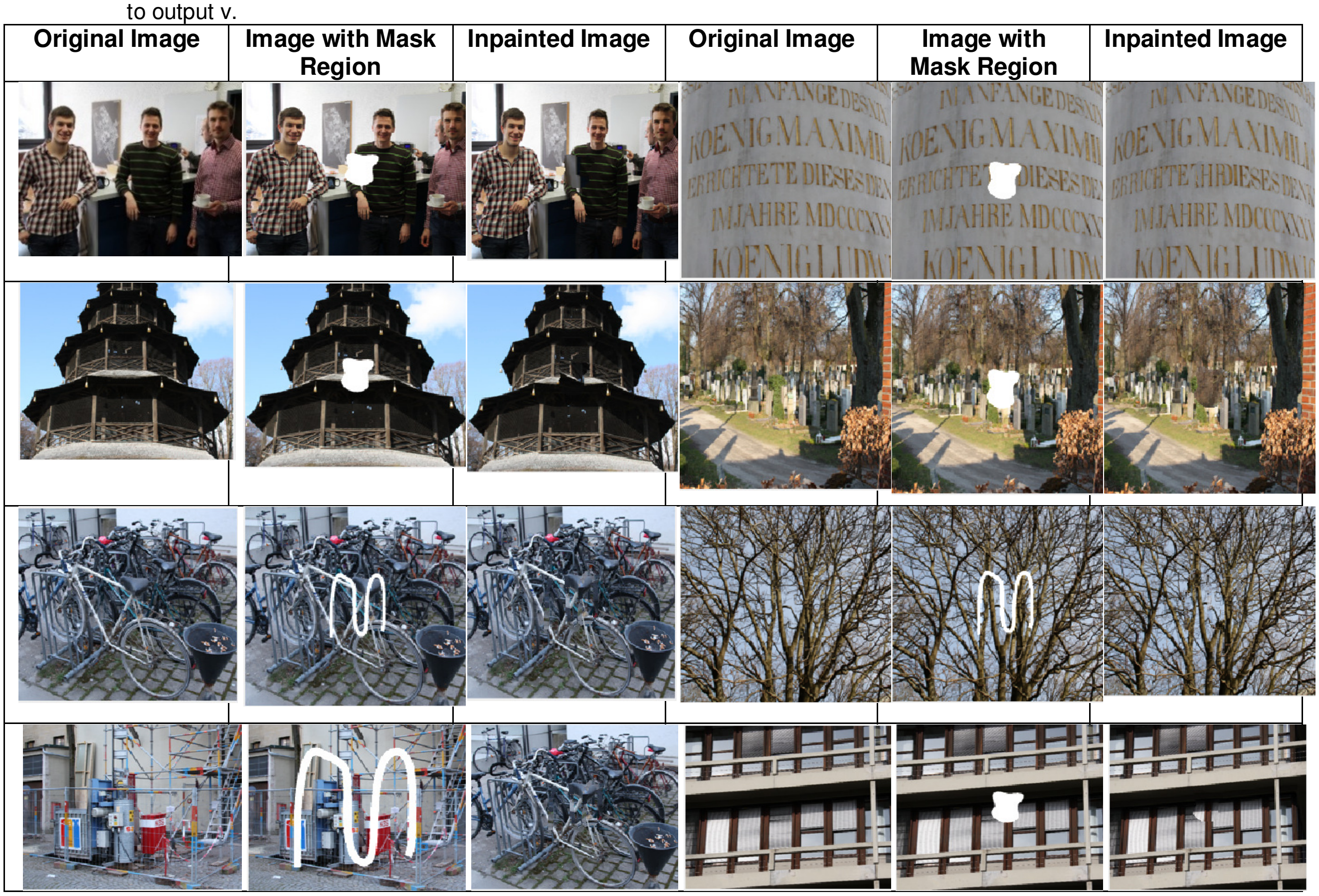

Fig. 4. Results observed with Spiking Neural Network on TUM-IID Dataset Images.

The model is tested on the TUM-IID dataset. To check the performance of the model, eight images of a complex structure out of 17 images are taken. The model performance is compared with existing nearest neighbour in painting techniques.

Let image I is an input image that is passed into the proposed model with masked image Imask, and we get InP, an inpainted image. The size of each image is M x N

$$MSE = \left(\frac{1}{MN}\right)\sum_{i=1}^{M}\sum_{j=1}^{N}(InP(i,j) - I(i,j))^2 \quad (5)$$

$$PSNR = 10 \times log_{10}\left(\frac{255^2}{MSE}\right)\text{dB} \quad (6)$$

Structural Similarity of the original input image and inpainted Image is verified using SSIM. SSIM is calculated on the statistical feature of the original and inpainted image as mean (μ), variance (σ2) and covariance (cov)

$$SSIM(x,y) = \frac{(2\mu_1\mu_2+c_1)(2\,COV_{12}+c_2)}{(u_1^2+u_2^2+c_1)(\sigma_1^2+\sigma_2^2+c_2)} \quad (7)$$

Where c1 and c2 are the constant used to keep away from vulnerability. The values of constants depend on image size.

The pixel-wise correlation between the images are measured by the quality factor (QF)

$$QF = CC * \frac{(2\mu_1\mu_2)(2\sigma_1+\sigma_2)}{(u_1^2+u_2^2)(\sigma_1^2+\sigma_2^2)} \quad (8)$$

Where CC denotes the correlation coefficient.

Table 1: Comparative analysis of existing Nearest Neighbor method with spiking neural networks along with % of improvements.

Image	Existing Nearest Neighbor Method			Proposed SNN Model			% of Improvements		
	PSNR	SSIM	QF	PSNR	SSIM	QF	PSNR	SSIM	QF
1	18.956	0.9917	0.9564	27.851	0.9981	0.9945	46.9244566	0.64535646	3.98368883
2	20.703	0.9975	0.9921	20.96	0.9977	0.9926	1.24136599	0.02005013	0.05039815
3	19.252	0.9967	0.9739	22.154	0.9983	0.9865	15.0737586	0.16052975	1.29376733
4	25.005	0.9918	0.9087	26.304	0.9939	0.9294	5.19496101	0.21173624	2.27797953
5	10.322	0.9996	0.9698	12.913	0.9997	0.9831	25.1017245	0.010004	1.37141679
6	20.018	0.9977	0.9726	21.478	0.9982	0.9804	7.29343591	0.05011527	0.80197409
7	20.828	0.997	0.9733	21.926	0.9977	0.9792	5.27174957	0.07021063	0.60618514
8	17.101	0.9947	0.9775	24.241	0.9991	0.9956	41.7519443	0.44234443	1.8516624

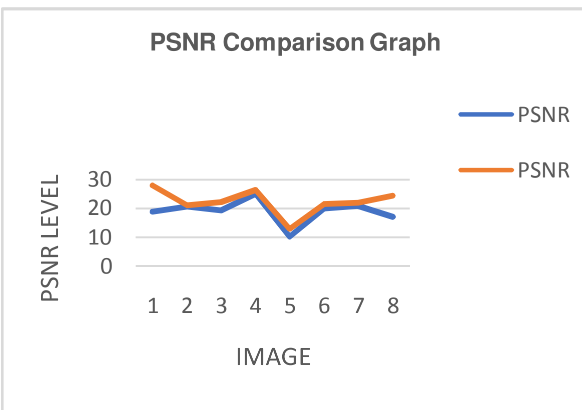

Fig. 5. PSNR Comparison Graph of NN & SNN Model.

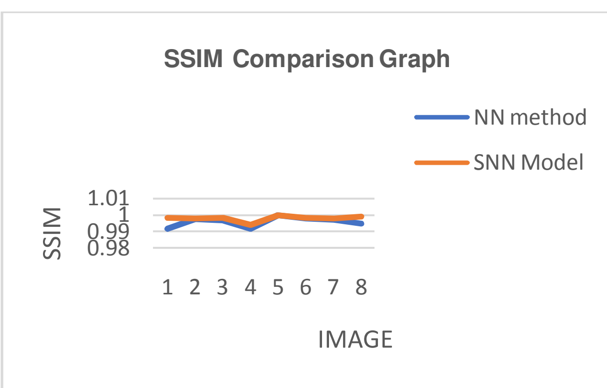

Fig. 6. SSIM Comparison Graph of NN & SNN Model.

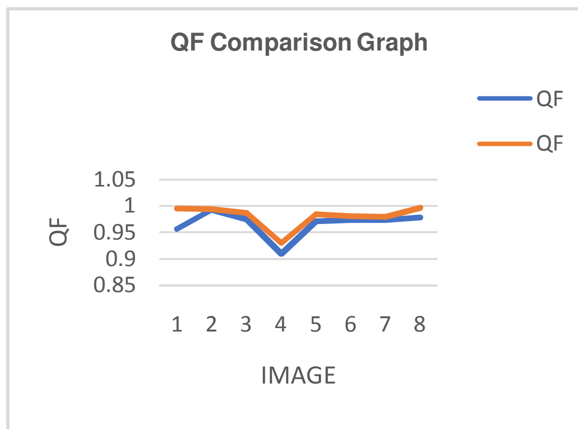

Fig. 7. QF Comparison Graph of NN & SNN Model.

V. CONCLUSION

We propose a Spiking neural network model that analyzes the feature of the whole image including the missing region. The model predicts the patch outside the missing region based on the calculated feature and synthesizes that patch in such a manner in the inpainting region that it looks like an original image. Our model producesa very sharp feature. The proposed algorithm is applied to this uncorrupted image to get the plausible original image. The results display significant enhancement in the estimation of PSNR, Quality factor, and SSIM over the current techniques. The proposed technique offers good results when missing regions of various sizes are present in the image. It is notable that when mask size is bigger in the complex structured image, and it is difficult to reproduce their texture and structural similarities, the proposed technique works well to retrieve the texture and structure of missing parts of the object or region. The algorithm offers sharp inpainting capabilities for commercial applications.

VI. FUTURE SCOPE

Further, our work can be extended with ensemble machine learning techniques or bio-inspired techniques to get more accurate results. So, we are suggesting the modern developers and researchers use this technique to inpaint the image with free form mask. The future scope is as we are unable to extend it for large resolution images with multiple hole.The proposed method takes much time to compute the whole image feature as well missing region features. We are unable to predict the features of large hole region. Proposed model has an scope of providing an enhanced method of structure completion efficiently. Exact image what we are expected, received blur image as an output during completion of large missing region.

REFERENCES

[1]. Bohte, S. M., & Kok, J. N. (2005). Applications of spiking neural networks. *Information Processing Letters*, *6*(95), 519-520.

[2]. Pavlidis, N. G., Tasoulis, O. K., Plagianakos, V. P., Nikiforidis, G., & Vrahatis, M. N. (2005, July). Spiking neural network training using evolutionary algorithms. In *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.* Vol. 4, pp. 2190-2194). IEEE.

[3]. Khan, G. M., Miller, J. F., & Halliday, D. M. (2011). Evolution of cartesian genetic programs for the development of learning neural architecture. *Evolutionary computation*, *19*(3), 469-523.

[4]. Kube, K., Herzog, A., Michaelis, B., de Lima, A. D., & Voigt, T. (2008). Spike-timing-dependent plasticity in small-world networks. *Neurocomputing*, *71*(7-9), 1694-1704.

[5]. Pavlidis, N. G., Tasoulis, O. K., Plagianakos, V. P., Nikiforidis, G., & Vrahatis, M. N. (2005, July). Spiking neural network training using evolutionary algorithms. In *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005*. (Vol. 4, pp. 2190-2194). IEEE.

[6]. Matsubara, T. (2017). Conduction delay learning model for unsupervised and supervised classification of Spatio-temporal spike patterns. *Frontiers in computational neuroscience*, *11*, 104.

[7]. Majumder, D. D. (1986). Pattern recognition, image processing and computer vision in fifth generationcomputer systems. *Sadhana*, *9*(2), 139-156.

[8]. Guillemot, C., & Le Meur, O. (2013). Image inpainting: Overview and recent advances. *IEEE signal processing magazine*, *31*(1), 127-144.

[9]. Perona, P., & Malik, J. (1990). Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on pattern analysis and machine intelligence*, *12*(7), 629-639.

[10]. Weickert, J. (1996). Theoretical foundations of anisotropic diffusion in image processing. In *Theoretical foundations of computer vision* (pp. 221-236). Springer, Vienna.

[11]. Weickert, J. (1999). Coherence-enhancing diffusion filtering. *International journal of computer vision*, *31*(2-3), 111-127.

[12]. Shen, J., & Chan, T. F. (2002). Mathematical models for local nontexture inpaintings. *SIAM Journal on Applied Mathematics*, *62*(3), 1019-1043.

[13]. Chan, T. F., & Shen, J. (2001). Nontexture inpainting by curvature-driven diffusions. *Journal of Visual Communication and Image Representation*, *12*(4), 436-449.

[14]. Casaca, W., Boaventura, M., De Almeida, M. P., &Nonato, L. G. (2014). Combining anisotropic diffusion, transport equation and texture synthesis for inpainting textured images. *Pattern Recognition Letters*, *36*, 36-45.

[15]. Bertalmio, M., Bertozzi, A. L., & Sapiro, G. (2001, December). Navier-stokes, fluid dynamics, and image and video inpainting. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001* (Vol. 1, pp. I-I). IEEE.

[16]. Telea, A. (2004). An image inpainting technique based on the fast marching method. *Journal of graphics tools*, *9*(1), 23-34.

[17]. Tschumperlé, D. (2006). Fast anisotropic smoothing of multi-valued images using curvature-preserving PDE's. *International Journal of Computer Vision*, *68*(1), 65-82.

[18]. Rudin, L. I., Osher, S., & Fatemi, E. (1992). Nonlinear total variation based noise removal algorithms. *Physica D: nonlinear phenomena*, *60*(1-4), 259-268.

[19]. Voci, F., Eiho, S., Sugimoto, N., & Sekibuchi, H. (2004). Estimating the gradient in the Perona-Malik equation. *IEEE Signal Processing Magazine*, *21*(3), 39-65.

[20]. Efros, A. A., & Leung, T. K. (1999, September). Texture synthesis by non-parametric sampling. In *Proceedings of the seventh IEEE international conference on computer vision* (Vol. 2, pp. 1033-1038). IEEE.

[21]. Wei, L. Y., & Levoy, M. (2000). Fast texture synthesis using tree-structured vector quantization. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques* (pp. 479-488). ACM Press/Addison-Wesley Publishing Co.

[22]. Ashikhmin, M. (2001). Synthesizing natural textures In: Proceedings of the Symposium on Interactive 3D Graphics, pp 217-226.

[23]. Liang, L., Liu, C., Xu, Y. Q., Guo, B., & Shum, H. Y. (2001). Real-time texture synthesis by patch-based sampling. *ACM Transactions on Graphics (ToG)*, *20*(3), 127-150.

[24]. Bugeau, A., Bertalmío, M., Caselles, V., &Sapiro, G. (2010). A comprehensive framework for image inpainting. *IEEE Transactions on Image Processing*, *19*(10), 2634-2645.

[25]. Lu, D., & Weng, Q. (2007). A survey of image classification methods and techniques for improving classification performance. *International journal of Remote sensing*, *28*(5), 823-870.

[26]. Sathya, R., & Abraham, A. (2013). Comparison of supervised and unsupervised learning algorithms for pattern classification. *International Journal of Advanced Research in Artificial Intelligence*, *2*(2), 34-38.
[27]. Zhang, H., Hou, D., & Zhou, Z. (2005, August). A novel lane detection algorithm based on support vector machine. In *Progress In Electromagnetics Research Symposium* (pp. 22-26).
[28]. Estivill-Castro, V., & Yang, J. (2000, August). Fast and robust general purpose clustering algorithms. In *Pacific Rim International Conference on Artificial Intelligence* (pp. 208-218). Springer, Berlin, Heidelberg.
[29]. Xu, R., & Wunsch, D. C. (2008). Recent advances in cluster analysis. *International Journal of Intelligent Computing and Cybernetics*, *1*(4), 484-508.
[30]. Kotsiantis, S. B., Zaharakis, I., &Pintelas, P. (2007). Supervised machine learning: A review of classification techniques. *Emerging artificial intelligence applications in computer engineering*, *160*, 3-24.
[31]. Ikonomakis, M., Kotsiantis, S., & Tampakas, V. (2005). Text classification using machine learning techniques. *WSEAS transactions on computers*, *4*(8), 966-974.
[32]. Albeahdili, H. M., Alwzwazy, H. A., & Islam, N. E. (2015). Robust Convolutional Neural Networks for Image Recognition. *International Journal of Advanced Computer Science and Applications*, *6*(11), 105-111.
[33]. Gehring, J., Miao, Y., Metze, F., & Waibel, A. (2013). Extracting deep bottleneck features using stacked auto-encoders. In *2013 IEEE international conference on acoustics, speech and signal processing* (pp. 3377-3381). IEEE.
[34]. Wang, Z., & Wu, Q. (2017). Shape Completion Using Deep Boltzmann Machine. *Computational Intelligence and Neuroscience*, *2017*.
[35]. Ping, W., & Ihler, A. (2017). Belief propagation in conditional RBMs for structured prediction. *arXiv preprint arXiv:1703.00986*.
[36]. Wang, T. C., Liu, M. Y., Zhu, J. Y., Tao, A., Kautz, J., & Catanzaro, B. (2018). High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 8798-8807).
[37]. Xin, J., & Embrechts, M. J. (2001). Supervised learning with spiking neural networks. In *IJCNN'01. International Joint Conference on Neural Networks. Proceedings (Cat. No. 01CH37222)* (Vol. 3, pp. 1772-1777). IEEE.
[38]. Natschläger, T., & Ruf, B. (1998). Spatial and temporal pattern analysis via spiking neurons. *Network: Computation in Neural Systems*, *9*(3), 319-332.
[39]. Gupta, A., & Long, L. N. (2007, August). Character recognition using spiking neural networks. In *2007 International Joint Conference on Neural Networks* (pp. 53-58). IEEE.
[40]. Tavanaei, A., Ghodrati, M., Kheradpisheh, S. R., Masquelier, T., & Maida, A. (2018). Deep learning in spiking neural networks. *Neural Networks*.
[41]. Dan, Y., & Poo, M. M. (2004). Spike timing-dependent plasticity of neural circuits. *Neuron*, *44*(1), 23-30.
[42]. Ourdighi, A., & Benyettou, A. (2016). An efficient spiking neural network approach based on spike response model for breast cancer diagnostic. *Int. Arab J. Inf. Technol.*, *13*(6B), 1032-1038.
[43]. Stromatias, E., & Marsland, J. S. (2015). Supervised learning in spiking neural networks with limited precision: Snn/lp. In *2015 International Joint Conference on Neural Networks (IJCNN)* (pp. 1-7). IEEE.
[44]. Moritz, P., Nishihara, R., & Jordan, M. (2016). A linearly-convergent stochastic L-BFGS algorithm. In *Artificial Intelligence and Statistics* (pp. 249-258).

How to cite this article: Kumar, V., Sinha, A. K. and Solanki, A. K. (2019). Image Inpainting through Textures Synthesis using Spiking Neural Networks. *International Journal on Emerging Technologies,* **10**(4): 43–49.