Journal of Engineering Sciences

Vol 11, Issue 4 , April/2020
ISSN NO:0377-9254

SHIP DETECTION USING SQUEEZE AND EXCITATION

B. Sai Divya[1], G. Keerthi Anusha[2] , G. Gayatri[3] , M. Jyothi Bangari[4],V.Sasikala[5]

[1,2,3,4]IV B.Tech, Department of Information Technology, Vignan's Nirula Institute of Technology & Science for Women, Peda Palakaluru, Guntur-522009, Andhra Pradesh, India.

[5]Asst.Professor, Department of Information Technology, Vignan's Nirula Institute of Technology & Science for Women, Peda Palakaluru, Guntur-522009, Andhra Pradesh, India.

kala.sasiv88@gmail.com

ABSTRACT

The development in computer vision, Deep Learning has been used for ship detection in SAR images such as the faster region-based convolution neural network (R-CNN). SAR ship detection is an important part of marine monitoring. It has a much better detection performance than Traditional methods on nearshore areas. This is because traditional methods need sea-land segmentation before detection, and inaccurate mask decreases the performance. Though SAR ship detection methods still have many false detections inland areas. So, a new network architecture based on R-CNN is proposed to further improve the detection performance by using the Squeeze and Excitation mechanism.

Keywords: R-CNN, Deep Learning, Synthetic-aperture radar, SER Block, Squeeze and Excitation Metod.

1. INTRODUCTION

Deep learning is an artificial intelligence function that imitates the workings of the human brain in processing data and creating patterns for use in decision making[1]. Deep learning is a subset of Machine Learning in Artificial Intelligence (AI) that has networks capable of learning unsupervised from data that is unstructured or unlabeled so-known as a deep neural network. Deep learning has evolved hand-in-hand with the digital era, which has brought about an explosion of data in all forms and from every region of the world[2]. This data, known simply as big data, is drawn from sources like social media, internet search engines, e-commerce platforms, and online cinemas, among others[3]. This enormous amount of data is readily accessible and can be shared through applications like cloud computing[41][42].

Synthetic-aperture radar (**SAR**) is a form of radar that is used to create two-dimensional images or three-dimensional reconstructions of objects, such as landscapes. SAR is typically mounted on a moving platform [43] [44], such as an aircraft or spacecraft, and has its origins in an advanced form of side-looking airborne radar (SLAR)[4]. SAR is capable of high-resolution remote sensing, independent of flight altitude, and independent of weather, as SAR can select frequencies to avoid weather-caused signal attenuation. SAR has day and night imaging capability as illumination is provided by the SAR [45] [46].

Convolution Neural Networks indicates that the network employs a mathematical operation called convolution[5]. Convolution is a specialized kind of linear operation. Convolution networks are simply neural networks that use convolution in place of general matrix multiplication in at least one of their layers[47][48].

Faster R-CNN:

R-CNN is the first step for Faster R-CNN. It uses search selective to find out the regions of interest and passes them to a ConvNet[6]. It tries to find out the areas that might be an object by combining similar pixels and textures into several rectangular boxes. The regression between predicted bounding boxes (Bboxes) and ground-truth bounding boxes are computed[7]. Fast R-CNN moves one step

forward. Instead of applying 2,000 times CNN to proposed areas, it only passes the original image to a pre-trained CNN model once.

2. LITERATURE SURVEY

With the development of remote sensing technology, the re- search on remote sensing images has attracted increasing attention due to its broad applications, such as maritime surveillance [8][9], environmental monitoring [10], vegetation estimation [11], etc. As one of the most critical applications of remote sensing images, ship detection is always a hot issue, since the task is essential in ship search and maritime traffic [12] and is still a challenge with the existence of disruptors. Although the acquisition of high spatial resolution remote sensing images becomes easy, the sizes of the remote sensing images files grow very large with the resolutions being higher, which will bring out the great computational cost. Therefore, the research for a more accurate detection algorithm for small-scaled ships is necessary.

Many investigations on ship detection in remote sensing images have been proposed [13] used a multi-scale and hierarchical manner visual attention mechanism to select candidate regions. However, these methods often perform poorly on the complex environments, such as the nearshore areas.

Nowadays, deep learning has been extremely popular because of its overwhelming performance on some typical datasets, such as PASCAL VOC and COCO [14] for object detection. All methods can be roughly classified into one-stage and two-stage methods. The one-stage method divides the image into several regions and estimates objects in each region, which enable real-time detection. You Only Look Once (YOLO) [15][16] and Single Shot MultiBox Detector (SSD) [17] is the most popular one-stage methods.

YOLO solves the object detection task as a regression problem, which directly outputs the position and class of the object. SSD discreteness the output space of bounding boxes into a set of default boxes over different aspect ratios and scales, and generates scores for the presence of each object category in each default box and produces adjustments to the box at prediction time. However, the one-stage methods show poor performance in detecting small targets. RCNN utilizes convolution neural networks as a feature extractor to obtain the feature map of a particular stage and applies bounding boxes with specific scales and ratios box to get the candidate regions.

3. PROPOSED METHOD

In this section, the proposed SER faster T-CNN is introduced[18]. First, the SE mechanism and our rank modification are introduced. Then, we show the architecture of SER faster R-CNN. Finally, the training details are presented.

A. Squeeze and Excitation Mechanism

To improve the representational power of the neural network, the squeeze excitation module, which can selectively emphasize active features and suppress less useful ones, is combined into the neural network[19][20]. The left box is a layer that consists of C feature maps and each feature map has W ·H pixels.

The feature maps are passed through a squeeze operation, consisting of global average pooling and full connection to aggregate the feature maps across spatial dimensions and obtain a channel descriptor with size C. The descriptor embeds the global distribution of channel-wise feature responses, which governs the excitation of each channel.

The final feature maps are reweighted by channel-wise scale[21]. In the region proposal stage, since the model adopts the pre- trained residual convolution neural network as the backbone, the SE module is only added at the connection, which is also beneficial to control the model complexity.

Assume that input feature maps of the SE block X have shape W×H×C, where W is the width, H is height, and C is the channels of feature maps[22]. The feature maps are first pooled by a global average pooling layer to shape 1×1 ×C. Then, the pooled vector is encoded to shape 1×1×(C/r) and decoded back to 1 × 1 × C with two fully connected layers to get the excitation vector V. The output Y of SE block is defined as follows:

$$Y = \sum_i Xi * Vi . \quad (1)$$

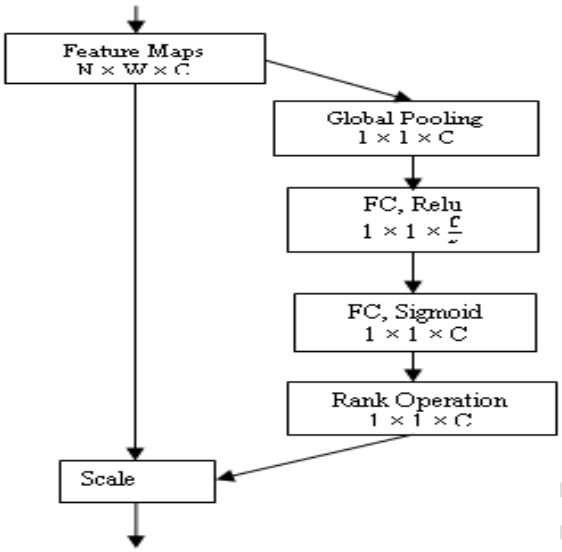


Fig. 1. Architecture of SER block.

In (1), each channel Xi in input feature maps X is multiplied with a corresponding excitation weight Vi in the excitation vector V. The target of SE block is to learn a weight for each feature maps and then suppress redundant feature maps. The faster R-CNN based on this mechanism is called SE faster R-CNN[23].

B. Rank Modification

When using the SE block to improve the performance of RPN, we observe that the value of each Vi is around 0.5. Although the performance has improved, the redundant feature maps are not fully suppressed[24]. In this letter, to further improve the performance, the rank operation is proposed. The values in excitation vector V are ranked and only top K values will be preserved. Other Vi will be set to 0 to strengthen the suppress function. So, in the output Y, the feature maps related to Vi = 0 will be completely suppressed. In the experiments, this can help further improve the performance and we call this SER block[25].

C. Squeeze and Excitation

To improve the representational power of the neural network, the squeeze excitation module, which can selectively emphasize active features and suppress less useful ones, are combined into the neural network[26]. The left box is a layer that consists of C feature maps and each feature map has W ·H pixels. The feature maps are passed through a squeeze operation, consisting of global average pooling and full connection to aggregate the feature maps across spatial dimensions and obtain a channel descriptor with size C. The descriptor embeds the global distribution of channel-wise feature responses, which governs the excitation of each channel. The final feature maps are reweighted by channel-wise scale[27]. In the region proposal stage, since the model adopts the pre-trained residual convolutional neural network as the backbone, the SE module is only added at the connection, which is also beneficial to control the model complexity. The diagram of connections in the last bottom-up path is shown in Fig. 4. The feature map Pi −1 first goes through a 3 ×3 convolutional layer and a max-pooling layer with stride 2 to get the same spatial size with Convi. Then the output is sent to a SE module.

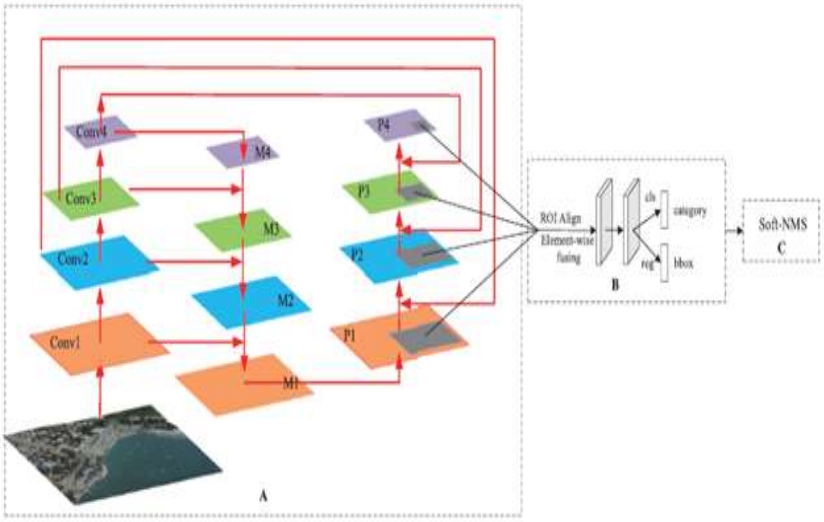


Fig 2: Architecture of Squeeze.

The processed feature map is laterally connected with another layer from the first bottom-up path by concatenation[28]. The fused feature map is then processed by another 3 ×3 convolution layer and SE module to generate Pi for the following operations. Because of the difference in the direction of the path, 3 ×3 convolution and max pooling operations are replaced by dilated convolution [39] in the structure of lateral connection in the top-down path. Dilated convolutions inflate the kernel by inserting spaces between the kernel elements, which can enlarge the size of the feature maps as a method of up sampling[29]. Thus the size of the result of M i is equal to the size of Con vi−1, which enables the concatenation of them.

D. Training Details

1) Concatenation of Multiple Layers: In current CNN architecture, the max-pooling layers are used to extract high-level feature representations[30]. So, one pixel in feature maps corresponds to the information of an area in the input image. In order to obtain multiscale information, the feature maps from last convolutional layers before max-pooling layers are concatenated. In this letter, we concatenate the last convolutional layer of VGG network Conv1_x, Conv2_x, and Conv3_x to obtain multiscale feature representations. The layer from Conv1_x is upsampled by a deconvolution layer to the same size of the layer from Conv2_x. At the same time, the layer from Conv3_x is downsampled by a max-pooling layer to the same size of the layer from Conv2_x. Then, these layers are concatenated along the channel axis as [31] and [32].

2) l2 Normalization: In general, with the depth of the network increases, the scale and norm of feature maps have a tendency to decrease. When trying to concatenate layers from different scales, if simply concatenate layers together, the resultant feature will not be discriminative, and heavy parameter tuning will be required to achieve sufficient accuracy because of different norms between layers [33]. By offering l2 normalization to each layer before concatenation, the training can be more stable and the performance can be improved. For a layer with N-dimensional input X = (x1 . . . xN), the output of l2 normalization ˆX can be denoted as follows:

$$\|X\|_2 = \left(\sum_{i=1}^{N} |x_i\right)^{1/2} \tag{2}$$

$$\hat{X} = \frac{X}{\|X\|_2}. \tag{3}$$

In order to accelerate the training, a scaling parameter γi is introduced for each channel, which scales the normalized value by yi = γi xˆi [34]. In this letter, γi is set to 40. This parameter will not influence performance.

3) Hyper parameter Setting: The learning rate of network is set to 1 × 10−4 initially, and maximal iteration is 20 000. The parameters of layers in SER block is initialized from the
Gaussian distribution with zero mean and a standard deviation of 0.01[35][36]. The first-stage feature stride is 2 when concatenating Conv1_x, Conv2_x, and Conv3_x as feature maps. The RPN
Anchor scales are 1, 2, 4, and 8 and aspect ratios are 0.125, 0.25, 0.5, and 1.

E. Modules

It consists of two modules

1) **Generate Faster RCNN Model:** In this module a train RCNN model will be generated using Squeeze and Excitation[37][38]. Input data to this module is given from 'VGGImageNet.h5py' file. RCNN internally uses CNN pooling technique to build mode. Below code describe model generation for train images and use five convolution layers[39].

2) **Upload Test Image & Detect Ship:** In this module we will upload test image and then application extract features from this test image and then apply RCNN train model on that test image to detect ships[40].

Program Code:

```
def FastRCNN():
   global model
   if os.path.exists('VGGImageNet.h5py'):
      model =
tf.keras.models.load_model('VGGImageNet.h5py')
      model.summary()
      text.insert(END,"model output can bee seen in
black console\n");
   else:
      f = open(r'dataset/shipsnet.json')
      dataset = json.load(f)
      f.close()

      input_data =
np.array(dataset['data']).astype('uint8')
      output_data =
np.array(dataset['labels']).astype('uint8')

      n_spectrum = 3 # color chanel RGB
      weight = 80
      height = 80
      X = input_data.reshape([-1, n_spectrum, weight,
height])
      pic = X[3]
      y = np_utils.to_categorical(output_data, 2)
      indexes = np.arange(4000)
      np.random.shuffle(indexes)
      X_train = X[indexes].transpose([0,2,3,1])
      y_train = y[indexes]
      X_train = X_train / 255
      np.random.seed(42)
      model = Sequential()
      model.add(Conv2D(32, (3, 3), padding='same',
input_shape=(80, 80, 3), activation='relu'))
      model.add(MaxPooling2D(pool_size=(2, 2)))
#40x40
      model.add(Dropout(0.25))
      model.add(Conv2D(32, (3, 3), padding='same',
activation='relu'))
      model.add(MaxPooling2D(pool_size=(2, 2)))
#20x20
      model.add(Dropout(0.25))
      model.add(Conv2D(32, (3, 3), padding='same',
activation='relu'))
      model.add(MaxPooling2D(pool_size=(2, 2)))
#10x10
      model.add(Dropout(0.25))
      model.add(Conv2D(32, (10, 10),
padding='same', activation='relu'))
      model.add(MaxPooling2D(pool_size=(2, 2)))
#5x5
      model.add(Dropout(0.25))
      model.add(Flatten())
      model.add(Dense(512, activation='relu'))
      model.add(Dropout(0.5))
      model.add(Dense(2, activation='softmax'))
      sgd = SGD(lr=0.01, momentum=0.9,
nesterov=True)
      model.compile(loss='categorical_crossentropy',
optimizer=sgd, metrics=['accuracy'])
      # training
      model.fit(X_train, y_train, batch_size=32,
epochs=18, validation_split=0.2, shuffle=True,
verbose=2)
      model.save("VGGImageNet.h5py1")
      text.insert(END,"model output can bee seen in
black console\n");
```

4. EXPERIMENTAL RESULTS

In this section, first, the experimental data set and settings are introduced. Next, an experiment is designed to explore which layers should be concatenated to get better feature representations for the first stage and the second stage. Then, the performance of SE faster R-CNN is proposed, and the values of excitation vector are investigated to show that the redundant feature maps are not fully suppressed. Finally, the proposed SER faster R-CNN is evaluated and compared with the SE faster R-CNN and state-of-the-art methods. In addition, the computational cost of models is presented.

A. Experimental Data Set and Settings

The data set used in this letter is from Sentinel-1A, which was collected in interferometric wide swath mode. The pixel spacing is 10 m. Twenty images with 6484 ships are used for training, and two extra images which reannotated by experts with 1348 ships are used for testing. The training data are not divided into extra validation data like other object detection literature in computer vision [8]. Some AIS data are used to help manually annotation [23]. For training data set, the labeled images were cut into 512 × 512 sized patches without overlap. The patches with ships are fed into the network for training. The testing images are processed in the same way.

All experiments are executed on a PC with an Intel single Core i7 CPU, NVIDIA GTX-1070 GPU (8-GB video memory), and 64-GB RAM. The PC operating system was Ubuntu 14.04.

In order to evaluate the detection performance of the network, the target detection probability, false detection probability which refers to the probability that one target in detection results to be a false detection, and F1 score [12] are defined as

$$P_d = \frac{N_{td}}{N_{\text{ground\_truth}}} \tag{4}$$

$$P_f = \frac{N_{fd}}{N_{\text{total\_target}}} \tag{5}$$

$$F1 = 2 \times \frac{P_d \times (1 - P_f)}{P_d + (1 - P_f)} \tag{6}$$

TABLE I

DETECTION PERFORMANCE OF DIFFERENT CONCATENATION STRATEGIES

Layers	P_d (%)	P_f (%)	$F1$
5	61.7	30	0.656
1+2+3	71	18.5	0.759
1+3+5	70.8	18.7	0.757
3+4+5	69	16.5	0.756
2+3+4	72	23.4	0.742

TABLE II

DETECTION PERFORMANCE OF A DIFFERENT FASTER R-CNN BEFORE AND AFTER USING THE SE BLOCK

Layers	P_d (%)	P_f (%)	$F1$	r
Faster R-CNN 1+2+3	71	18.5	0.759	-
SE faster R-CNN	72.5	24.7	0.739	1
SE faster R-CNN	76.5	16.5	0.799	2
SE faster R-CNN	78.7	17.8	0.804	4
SE faster R-CNN	77.8	18.1	0.798	8

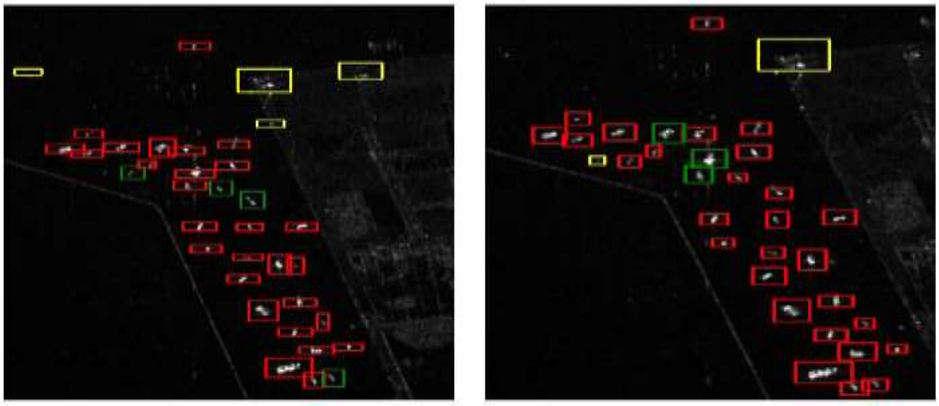
B. Multilayer Concatenation Strategies

Multiscale information obtained by concatenating different feature maps is effective for region proposal generation and detection, mainly because of its richness and appropriate resolution.

In this section, different models are trained and tested to show that concatenate Conv1_x, Conv2_x, and Conv3_x is the best strategy. The basic model uses feature maps from Conv_5 layer for the first-stage RPN and the second-stage classification. The second model combines layers 1, 2, and 3. The third model combines layers 1, 3, and 5, the fourth model combines layers 3, 4, and 5, and the final model combines layers 2, 3, and 4. The results are shown in Table I. As shown in Table I, the F1 performance of different concatenation strategies are close and all outperform basic model. However, the model which combines layers 1, 2, and 3 outperforms other models on F1 performance under experimental conditions in this letter. So, this strategy is used to extract shared feature maps for the first-stage RPN and the second-stage classification.

COMPARISON OF DETECTION PERFORMANCE BETWEEN SER FASTER R-CNN AND THE STATE-OF-THE-ART METHODS

Layers	P_d (%)	P_f (%)	$F1$	r
SER faster R-CNN	**81.1**	**13.8**	**0.836**	1
SER faster R-CNN	80.5	24.4	0.779	2
SER faster R-CNN	76.6	19.1	0.787	4
SER faster R-CNN	79.5	17.3	0.811	8
Contextual faster R-CNN [12]	72.7	20	0.762	-

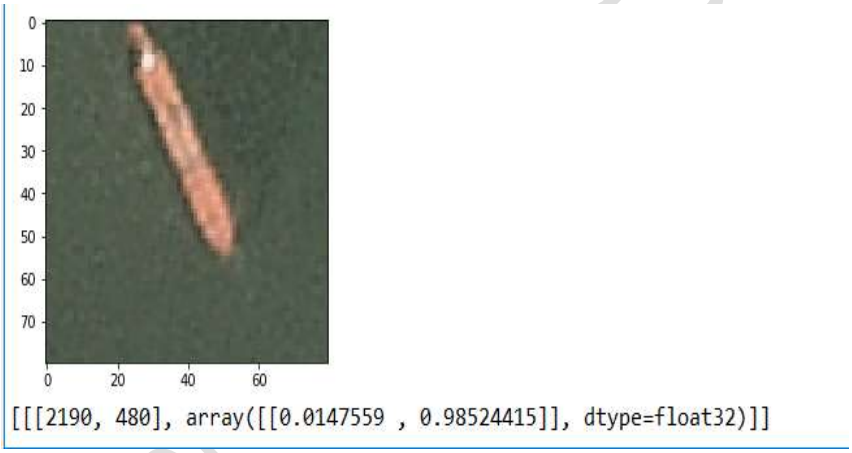


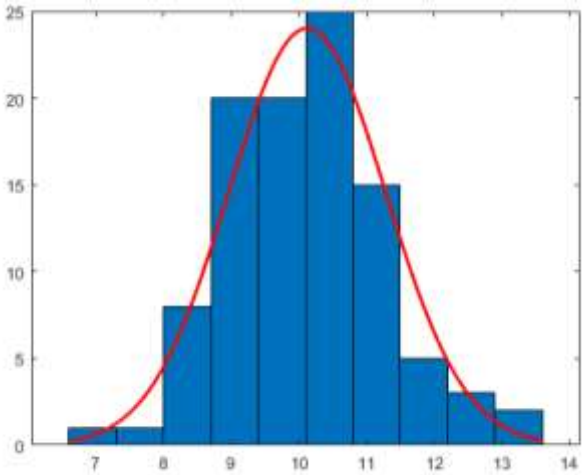
(a) (b)

C. SER Faster R-CNN

In this section, first, the performance of SE faster R-CNN with a different hyperparameter r which determines the length of encoding vector is presented. The multiscale feature map concatenation strategy combines layers 1, 2, and 3. The performance of SE faster R-CNN models is compared with the model which only uses multiscale feature map concatenation strategy and is shown in Table II. As shown in Table II, most SE faster R-CNN models can have better F1 performance.

The results show that the SE block is effective. However, when observing the values in excitation vector as shown in Fig. 3, most values are around 0.5 and the redundant feature maps are not completely suppressed. In order to further improve performance, the SER block is proposed to completely suppress redundant feature maps. The performance of proposed SER faster R-CNN with K = 256 and a different hyperparameter r is shown in Table III. At the same time, the proposed method is compared with the previous research.





The performance of the proposed network is 9.7% better than that of the state of the art [13]. Network gets best performance when r = 1. This may because a large encode layer helps network to better learn to

separate important feature maps, which will be preserved after SER block of network. An example of detection result on nearshore area is shown in Fig. 4. As shown in Fig. 4, the proposed SER faster R-CNN can successfully detect nearshore ships which hard to be detected by traditional methods.

5. CONCLUSION

The multiscale feature map concatenation strategy is used to improve the quality of shared feature maps extracted by a CNN. The SER mechanism is used to improve the first-stage RPN performance of faster R-CNN. The experimental results present how we design such high-performance object detector. The results show that our SER faster R-CNN outperforms the state-of-the-art method and much faster.

REFERENCES

[1]. Lakshman Narayana Vejendla and A Peda Gopi, (2019),” Avoiding Interoperability and Delay in Healthcare Monitoring System Using Block Chain Technology”, Revue d'Intelligence Artificielle , Vol. 33, No. 1, 2019,pp.45-48.

[2]. Gopi, A.P., Jyothi, R.N.S., Narayana, V.L. et al. (2020), “Classification of tweets data based on polarity using improved RBF kernel of SVM” . Int. j. inf. tecnol. (2020). https://doi.org/10.1007/s41870-019-00409-4.

[3]. A Peda Gopi and Lakshman Narayana Vejendla, (2019),” Certified Node Frequency in Social Network Using Parallel Diffusion Methods”, Ingénierie des Systèmes d' Information, Vol. 24, No. 1, 2019,pp.113-117.. DOI: 10.18280/isi.240117

[4]. Lakshman Narayana Vejendla and Bharathi C R ,(2018),“Multi-mode Routing Algorithm with Cryptographic Techniques and Reduction of Packet Drop using 2ACK scheme in MANETs”, Smart Intelligent Computing and Applications, Vo1.1, pp.649-658. DOI: 10.1007/978-981-13-1921-1_63 DOI: 10.1007/978-981-13-1921-1_63

[5]. Lakshman Narayana Vejendla and Bharathi C R, (2018), “Effective multi-mode routing mechanism with master-slave technique and reduction of packet droppings using 2-ACK scheme in MANETS”, Modelling, Measurement and Control A, Vol.91, Issue.2, pp.73-76. DOI: 10.18280/mmc_a.910207

[6]. Lakshman Narayana Vejendla , A Peda Gopi and N.Ashok Kumar,(2018),“ Different techniques for hiding the text information using text steganography techniques: A survey”, Ingénierie des Systèmes d'Information, Vol.23, Issue.6,pp.115-125.DOI: 10.3166/ISI.23.6.115-125

[7]. A Peda Gopi and Lakshman Narayana Vejendla (2018), “Dynamic load balancing for client server assignment in distributed system using genetic algorithm”, Ingénierie des Systèmes d'Information, Vol.23, Issue.6, pp. 87-98. DOI: 10.3166/ISI.23.6.87-98

[8]. Lakshman Narayana Vejendla and Bharathi C R,(2017),“Using customized Active Resource Routing and Tenable Association using Licentious Method Algorithm for secured mobile ad hoc network Management”, Advances in Modeling and Analysis B, Vol.60, Issue.1, pp.270-282. DOI: 10.18280/ama_b.600117

[9]. Lakshman Narayana Vejendla and Bharathi C R,(2017),“Identity Based Cryptography for Mobile ad hoc Networks”, Journal of Theoretical and Applied Information Technology, Vol.95, Issue.5, pp.1173-1181. EID: 2-s2.0-85015373447

[10]. Lakshman Narayana Vejendla and A Peda Gopi, (2017),” Visual cryptography for gray scale images with enhanced security mechanisms”, Traitement du Signal,Vol.35, No.3-4,pp.197-208. DOI: 10.3166/ts.34.197-208

[11]. A Peda Gopi and Lakshman Narayana Vejendla, (2017),” Protected strength approach for image steganography”, Traitement du Signal, Vol.35, No.3-4,pp.175-181. DOI: 10.3166/TS.34.175-181

[12]. Lakshman Narayana Vejendla and A Peda Gopi, (2020),” Design and Analysis of CMOS LNA with Extended Bandwidth For RF Applications”, Journal of Xi'an University of Architecture & Technology, Vol. 12, Issue. 3,pp.3759-3765. https://doi.org/10.37896/JXAT12.03/319.

[13]. Chaitanya, K., and S. Venkateswarlu,(2016),"DETECTION OF

BLACKHOLE & GREYHOLE ATTACKS IN MANETs BASED ON ACKNOWLEDGEMENT BASED APPROACH." Journal of Theoretical and Applied Information Technology 89.1: 228.

[14]. Patibandla R.S.M.L., Kurra S.S., Mundukur N.B. (2012), "A Study on Scalability of Services and Privacy Issues in Cloud Computing". In: Ramanujam R., Ramaswamy S. (eds) Distributed Computing and Internet Technology. ICDCIT 2012. Lecture Notes in Computer Science, vol 7154. Springer, Berlin, Heidelberg

[15]. Patibandla R.S.M.L., Veeranjaneyulu N. (2018), "Survey on Clustering Algorithms for Unstructured Data". In: Bhateja V., Coello Coello C., Satapathy S., Pattnaik P. (eds) Intelligent Engineering Informatics. Advances in Intelligent Systems and Computing, vol 695. Springer, Singapore

[16]. Patibandla, R.S.M.L., Veeranjaneyulu, N. (2018), "Performance Analysis of Partition and Evolutionary Clustering Methods on Various Cluster Validation Criteria", Arab J Sci Eng ,Vol.43, pp.4379–4390.

[17]. R S M Lakshmi Patibandla, Santhi Sri Kurra and N.Veeranjaneyulu, (2015), "A Study on Real-Time Business Intelligence and Big Data",Information Engineering, Vol.4,pp.1-6.

[18]. K. Santhisri and P.R.S.M. Lakshmi,(2015), " Comparative Study on Various Security Algorithms in Cloud Computing", Recent Trends in Programming Languages ,Vol.2,No.1,pp.1-6.

[19]. K.Santhi Sri and PRSM Lakshmi,(2017), "DDoS Attacks, Detection Parameters and Mitigation in Cloud Environment", IJMTST,Vol.3,No.1,pp.79-82.

[20]. P.R.S.M.Lakshmi,K.Santhi Sri and Dr.N. Veeranjaneyulu,(2017), "A Study on Deployment of Web Applications Require Strong Consistency using Multiple Clouds", IJMTST,Vol.3,No.1,pp.14-17.

[21]. P.R.S.M.Lakshmi,K.Santhi Sri and M.V.Bhujanga Ra0,(2017), "Workload Management through Load Balancing Algorithm in Scalable Cloud", IJASTEMS,Vol.3,No.1,pp.239-242.

[22]. K.Santhi Sri, P.R.S.M.Lakshmi, and M.V.Bhujanga Ra0,(2017), "A Study of Security and Privacy Attacks in Cloud Computing Environment", IJASTEMS,Vol.3,No.1,pp. 235-238.

[23]. R S M Lakshmi Patibandla and N. Veeranjaneyulu, (2018), "Explanatory & Complex Analysis of Structured Data to Enrich Data in Analytical Appliance", International Journal for Modern Trends in Science and Technology, Vol. 04, Special Issue 01, pp. 147-151.

[24]. R S M Lakshmi Patibandla, Santhi Sri Kurra, Ande Prasad and N.Veeranjaneyulu, (2015), "Unstructured Data: Qualitative Analysis", J. of Computation In Biosciences And Engineering, Vol. 2,No.3,pp.1-4.

[25]. R S M Lakshmi Patibandla, Santhi Sri Kurra and H.-J. Kim,(2014), "Electronic resource management using cloud computing for libraries", International Journal of Applied Engineering Research, Vol.9,pp. 18141-18147.

[26]. Ms.R.S.M.Lakshmi Patibandla Dr.Ande Prasad and Mr.Y.R.P.Shankar,(2013), "SECURE ZONE IN CLOUD", International Journal of Advances in Computer Networks and its Security, Vol.3,No.2,pp.153-157.

[27]. Patibandla, R. S. M. Lakshmi et al., (2016), "Significance of Embedded Systems to IoT.", International Journal of Computer Science and Business Informatics, Vol.16,No.2,pp.15-23.

[28]. AnveshiniDumala and S. PallamSetty. (2020),"LANMAR routing protocol to support real-time communications in MANETs using Soft computing technique", 3rd International Conference on Data Engineering and Communication Technology (ICDECT-2019), Springer, Vol. 1079, pp. 231-243.

[29]. AnveshiniDumala and S. PallamSetty. (2019),"Investigating the Impact of Network Size on LANMAR Routing Protocol in a Multi-Hop Ad hoc Network", i-manager's Journal on Wireless Communication Networks (JWCN), Volume 7, No. 4, pp.19-26.

[30]. AnveshiniDumala and S. PallamSetty. (2019),"Performance analysis of LANMAR routing protocol in SANET and MANET", International Journal of Computer Science and

Engineering (IJCSE) – Vol. 7,No. 5, pp.1237-1242.

[31]. AnveshiniDumala and S. PallamSetty. (2018), "A Comparative Study of Various Mobility Speeds of Nodes on the Performance of LANMAR in Mobile Ad hoc Network", International Journal of Computer Science and Engineering (IJCSE) – Vol. 6, No. 9, pp. 192-198.

[32]. AnveshiniDumala and S. PallamSetty. (2018),"Investigating the Impact of IEEE 802.11 Power Saving Mode on the Performance of LANMAR Routing Protocol in MANETs", International Journal of Scientific Research in Computer Science and Management Studies (IJSRCSMS) – Vol.7, No. 4.

[33]. AnveshiniDumala and S. PallamSetty. (2016),"Analyzing the steady state behavior of RIP and OSPF routing protocols in the context of link failure and link recovery in Wide Area Network", International Journal of Computer Science Organization Trends (IJCOT) – Vol. 34 No 2, pp.19-22.

[34]. AnveshiniDumala and S. PallamSetty. (2016),"Investigating the Impact of Simulation Time on Convergence Activity & Duration of EIGRP, OSPF Routing Protocols under Link Failure and Link Recovery in WAN Using OPNET Modeler", International Journal of Computer Science Trends and Technology (IJCST) – Vol. 4 No. 5, pp. 38-42.

[35]. VellalacheruvuPavani and I. Ramesh Babu (2019) ,"Three Level Cloud Storage Scheme for Providing Privacy Preserving using Edge Computing",International Journal of Advanced Science and Technology Vol. 28, No. 16, pp. 1929 – 1940.

[36]. VellalacheruvuPavani and I. Ramesh Babu,"A Novel Method to Optimize the Computation Overhead in Cloud Computing by Using Linear Programming" ,International Journal of Research and Analytical Reviews May 2019, Volume 6, Issue 2,PP.820-830..

[37]. Anusha Papasani and Nagaraju Devarakonda,(2016),"Improvement of Aomdv Routing Protocol in Manet and Performance Analysis of Security Attacks", International Journal Of Research in Computer Science & Engineering ,Vol.6,No.5, pp.4674-4685.

[38]. Sk.Reshmi Khadherbhi,K.Suresh Babu , Big Data Search Space Reduction Based On User Perspective Using Map Reduce ,International Journal of Advanced Technology and Innovative Research Volume.07, IssueNo.18, December-2015, Pages: 3642-3647

[39]. B.V.Suresh kumar,Sk.Reshmi Khadherbhi ,BIG-IOT Framework Applications and Challenges: A Survey Volume 7, Issue VII, JULY/2018 pg.no 1257-1264

[40]. P.Sandhya Krishna,Sk.Reshmi Khadherbhi,V.Pavani, Unsupervised or Supervised Feature Finding For Study of Products Sentiment ,International Journal of Advanced Science and Technology, Vol 28 No 16 (2019).

[41]. K.Santhi Sri, Dr.Ande Prasad (2013), "A Review of Cloud Computing and Security Issues at Different Levels in Cloud Computing" , International Journal on Advanced Computer Theory and Engineering Vol. 2,pp 67-73.

[42]. K.Santhi Sri, N.Veeranjaneyulu(2018), "A Novel Key Management Using Elliptic and Diffie-Hellman for Managing users in Cloud Environment", Advances in Modelling and Analysis B,Vol.61,No.2,pp 106-112.

[43]. K.Santhi Sri, N.Veeranjaneyulu(2019), "Decentralized Key Management Using Alternating Multilinear Forms for Cloud Data Sharing with Dynamic Multiprivileged Groups", Mathematical Modelling of Engineering Problems,Vol.6,No.4,pp511-518.

[44]. S.Sasikala, P.Sudhakar, "interpolation of CFA color Images with Hybrid image denoising", 2014 Sixth International Conference on Computational Intelligence and Communication Networks, DOI 10.1109/.53 193 DOI 10.1109/CICN.2014.53, pp. 193-197.

[45]. Me. Jakeera Begum and M.Venkata Rao, (2015), "Collaborative Tagging Using CAPTCHA" International Journal of Innovative Technology And Research, Volume No.3, Issue No.5,pp,2436 – 2439.

[46]. L.Jagajeevan Rao, M. Venkata Rao, T.Vijaya Saradhi (2016), "How The Smartcard Makes the Certification Verification Easy" Journal of

Theoretical and Applied Information Technology, Vol.83. No.2, pp. 180-186.

[47]. Venkata Rao Maddumala, R. Arunkumar, and S. Arivalagan (2018)"An Empirical Review on Data Feature Selection and Big Data Clustering" Asian Journal of Computer Science and Technology Vol.7 No.S1, pp. 96-100.

[48]. Singamaneni Kranthi Kumar, Pallela Dileep Kumar Reddy, Gajula Ramesh, Venkata Rao Maddumala, (2019), "Image Transformation Technique Using Steganography Methods Using LWT Technique" ,Traitement du Signalvol 36, No 3, pp. 233-237.