Comparative Analysis of ResNet and DenseNet for Differential Cryptanalysis of SPECK 32/64 Lightweight Block Cipher

Ayan Sajwan[1] and Girish Mishra[2]

[1] Delhi Technological University, Delhi - 110 042,India
ayansajwan2003@gmail.com,
[2] DRDO-Scientific Analysis Group, Delhi - 110 054, India
gmishratech28@gmail.com

Abstract. This research paper explores the vulnerabilities of the lightweight block cipher SPECK 32/64 through the application of differential analysis and deep learning techniques. The primary objectives of the study are to investigate the cipher's weaknesses and to compare the effectiveness of ResNet as used by Aron Gohr at Crypto2019 and DenseNet . The methodology involves conducting an analysis of differential characteristics to identify potential weaknesses in the cipher's structure. Experimental results and analysis demonstrate the efficacy of both approaches in compromising the security of SPECK 32/64.

Keywords: Differential Cryptanalysis, Deep Learning, Speck, ResNet, DenseNet

1 Introduction

Cryptography is the technique of converting data into an incomprehensible form known as cipher text. It is done by using mathematical principles and algorithms. This crucial sector ensures the security and privacy of modern digital communications and data storage. Throughout history, critical information ranging from military communications[1] to commercial transactions[2] have been protected via cryptographic processes.

Over the centuries, cryptography has been an art practised by many who have invented techniques to meet some of the information security requirements. The previous two decades have seen the field evolve from an art to a science[3].

Data secrecy,integrity and authenticity are the main goals of cryptography. Confidentiality ensures that only authorised personnel may access the information. Integrity ensures that the data is unchanged throughout transmission or storage. Authentication makes sure that only reliable sources are sharing information

With the aid of encryption algorithms[4], cryptography secures data and communication. This is achieved by converting plaintext into ciphertext, a form

that cannot be deciphered. On the other side, Cryptanalysis is the science of analysing cryptographic systems to find vulnerabilities. These weaknesses can be exploited to obtain the original plaintext or encryption keys. Lightweight block ciphers serve a critical role in cryptography in situations where computational resources are constrained. These ciphers' low computational and memory overhead makes them ideal for secure, effective encryption[5].

The study of decrypting cryptographic methods, or cryptanalysis, is a crucial area for maintaining the security of encryption systems. It involves investigating the mathematical features and design choices to find the flaws in the cipher . Understanding these flaws allows cryptanalysts to create more successful attacks. This helps in increasing the security of the cryptographic systems.

Numerous fields, including cryptanalysis[6], have seen the emergence of machine learning and deep learning as highly effective tools. These methods make use of the models' computational capabilities. They can automatically identify patterns, detect features, and generate predictions.Machine learning techniques can be used in the context of cryptanalysis as well[7]. It can be used to analyse and categorise cryptographic data, such as ciphertexts, plaintexts, or encryption keys.

Deep learning, a branch of machine learning, has achieved outstanding results in a number of fields. Ranging from speech recognition[8], computer vision[9] to natural language processing[10]. Deep neural networks are able to learn complex data representations and identify deep correlations.
In this paper, we investigate the use of deep learning methods, more specifically DenseNet and ResNet. They are used for differential cryptanalysis on the lightweight block cipher SPECK 32/64. We intend to compare the effectiveness of ResNet and DenseNet. It ultimately helps us in understanding of the security of the SPECK 32/64 cipher by utilising the expressive potential of deep neural networks.

2 SPECK 32/64 Overview

SPECK is a lightweight block cipher developed by the National Security Agency (NSA). It was a part of the Lightweight Cryptography Project of the NSA. Here, the term "lightweight" refers to cryptographic algorithms that are designed for efficient operation and low resource consumption.

This qualifies them for usage in environments with limited resources such as Internet of Things devices[11], wireless sensor networks[12], and embedded systems[13].

The SPECK family consists of a variety of block and key sizes. The block is made up of 2 words, and is of the form $2n$. Here, n is the size of the word which may be 16, 24, 32, 48 or 64 bits. The key size(k) is mn bits. The key contains 2,3 or 4 words depending on the variant. Hence, the SPECK family is of the form

SPECK 2n/mn and has ten variants. We use SPECK 32/64 in our work, which denotes 2 words of 16-bits each and 4 keys of 16-bits each as well.

The SPECK family cipher is made up of a Feistel network structure[14]. In this network, the input block is split into two equally sized halves. Encryption rounds are then performed on these halves using a subkey. A different subkey is derived for each round. The number of rounds differ in each variant of the family.

The round function of SPECK 32/64 uses a range of bitwise operations for its cryptographic operations. It includes rotation, XOR, and modular addition, to induce confusion and diffusion features, assuring the security of the cipher.

2.1 Round Function

Speck's round function is very simple.It is an ARX structure, which means it is made out of the fundamental functions of modular addition (mod 2^k), bitwise rotation, and bitwise addition. They are denoted by $\boxplus$, $\gg$ and $\oplus$ respectively. SPECK n/m represents Speck with n bit block size and m bit key size. It produces the next round state $(\mathbf{L_{i+1}}, \mathbf{R_{i+1}})$ with an input k-bit subkey $\mathbf{K}$ and the current cipher state consisting of two k-bit words $(\mathbf{L_i}, \mathbf{R_i})$. The algorithm is as follows:

$$\mathbf{L_{i+1}} = ((\mathbf{L_i} \gg \alpha) \boxplus \mathbf{R_i}) \oplus \mathbf{K}$$

$$\mathbf{R_{i+1}} = (\mathbf{R_i} \ll \beta) \oplus \mathbf{L_{i_i+1}}$$

The values of α and β are constant: $(\alpha = 7, \beta = 2)$ for Speck32/64 and $(\alpha = 8, \beta = 3)$ for other members of the Speck family. The cipher text output is generated by applying the round function on the plain text input for 22 rounds in the case of Speck 32/64. However, we refer to round reduced speck in this paper. The key used in each round is generated from a master key by applying a key schedule. The key schedule depends on the member of the Speck family, we refer to Beaulieu et al[15] in this paper for the key scheduling.

3 ResNet and DenseNet Architectures

3.1 ResNet

ResNet or Residual Network is a powerful deep learning architecture first published by Kaiming[16]. Resnet is intended to address the issue of disappearing gradients[17] in very deep neural networks. It does so, by incorporating residual connections[18] or skip connections, which enable the building of deeper and more precise models.

ResNet is composed of many residual blocks or towers that are layered on top of each other and contain a sequence of convolutions and a skip connection. The skip connection is added to a block's output and then passed on to the following block. This helps in reducing the vanishing gradient problem and allows for better model training. Figure 2 depicts the working of a skip connection.

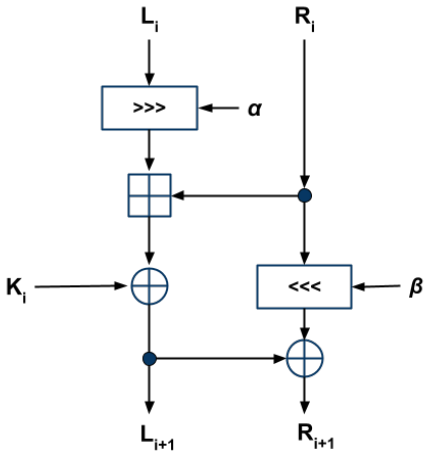


Fig. 1: General round of Speck

ResNet Structure The ResNet structure used in this work is the one used by Gohr in 2019[19]. It consists of a residual tower of depth ten, having a two layer convolutional network. The convolutional network has 32 filters.

First, Convolution is applied followed by a Batch normalization[20] for faster and stable training . It is followed by a Rectified Linear Unit layer[21] which introduces Non-linearity to the model. Then a skip/jump connection at the end adds the output of the final rectifier layer of the block to the convolutional block's input and forwards the result to the next block.

The initial layer is a bit-sliced 1 Dimensional Convolution with 32 output channels, which is followed by Batch normalization. Finally, a Rectified Linear Unit is applied to the preceding layer's output. The final result is a $32x16$ matrix that is fed into the depth-10 Residual Tower.

Finally, the data is flattened and transmitted to the prediction layer. This final layer consists of two densely linked hidden layers of 64 units each, followed by batch normalization, a Rectified Linear unit, and sigmoid activation for a single output head.

3.2 DenseNet

DenseNet is made up of Dense blocks and transition layers. DenseNet, which stands for "Densely Connected Networks" is a deep learning architecture designed by Gao Huang et al. originally published in their paper[22] in 2017.

DenseNet, like ResNet, aims to solve the vanishing gradient problem by maximising feature reuse. DenseNet introduces dense connections between layers and

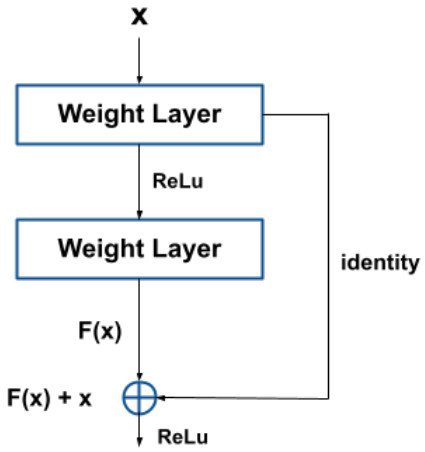


Fig. 2: Skip Connection

blocks as opposed to traditional neural networks, which connect layers sequentially. Unlike ResNet, which utilises an additive approach of adding previous layer output to subsequent layers, DenseNet uses all past outputs as input for future layers. As a result, each layer is directly linked to all the following layers.

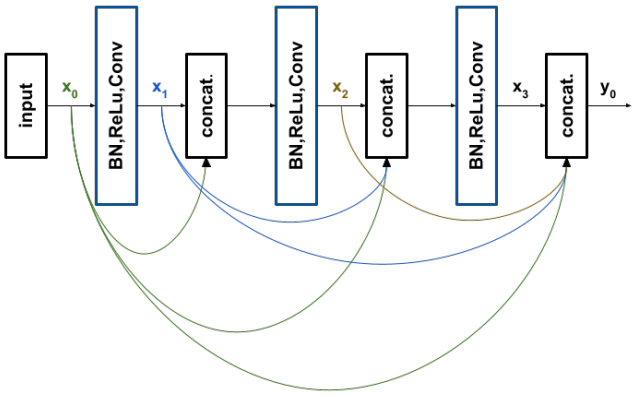


Fig. 3: Dense Connection

DenseNet Structure The first layer, like the one in the Resnet model, is a bit-sliced 1 Dimensional Convolution with 32 output channels.It is followed by

Batch normalisation, and lastly a Rectified Linear Unit is applied to the output of the preceding layer. The resulting 32x16 matrix is passed into the Dense Network.

DenseNet is made up of Dense blocks and transition blocks.The dense block has a depth-8 and is made up of two layers of 1D-convolution, Batch normalisation, and a Rectified linear unit layer. The convolution layer is made up of 64 filters and a kernel with a size of 3.Finally, the output is concatenated with the layer's input and handed on to the next layer.This occurs eight times since the depth is eight.

To restrict the amount of feature maps and minimise spatial dimensions, transition layers are inserted between dense blocks. The transition layer is made up of 1D convolution with 32 filters, batch normalisation, a ReLu layer, and 1D average pooling. The transition layer's output is subsequently passed on to the following dense block.

The dense block and transition layers are now merged with a depth of 2. This indicates alternating dense block and transition layer, followed by a final dense block. The final result is an overall structure of three dense blocks and two transition layers.

Finally, the data is flattened and transmitted to the prediction layer. This prediction layer consists of two dense hidden layers of 64 units each, followed by batch normalisation, Rectified linear unit, and sigmoid activation, similar to ResNet.

3.3 Input Data

Input data: Input consists of a pair of cipher texts (C_0, C_1). They are transformed into a 4x16 matrix with each row consisting of a word of the ciphertext. This way the data consists of four 16-bit words and therefore the input layer has 64 units. This input data is then passed into the ResNet and DenseNet architecture.

4 Experimental Setup and Methodology

4.1 Data Generation

The data generation methodology used is similar to the one used by Aron Gohr in 2019. A random number generator is used to create evenly distributed keys K_i and plain text pairings P_i with the input difference $\Delta = 0x0040/0000$, along with a vector of binary-valued real/random labels Y_i. If Y_i is set(=1), the plain text pair P_i is encrypted for $\mathbf{k}$ rounds to create training or validation data for k-round Speck, and if not, the second plain text in the pair is changed to a newly created random plain text. This way we have cipher texts belonging to 2 classes: Chosen Input difference ($\mathbf{Y} = 1$) and random input difference ($\mathbf{Y} = 0$). As a result we have 10^6 samples for our dataset for training and validation.

4.2 Training and Testing Procedure:

The data set of 10^6 samples is used for training in batches of 5000 and run for 20 epochs as opposed to 200 epochs by Gohr .
Mean Square Error (MSE)[23] loss is used with L2 weights regularization using the Adam algorithm[24] for optimization. This is a down scaled version of Gohr's experiment in which he used 10^7 samples for training for 200 epochs.

Testing data also contains a set of 10^6 samples with 2 classes. One of the chosen input difference and the other of random input difference. Table 1. provides the list of hyper-paramaters used in training with their values.

Hyper-parameters	**values**
Sample Size	10^6
Batch Size	5000
Epochs	20
Encryption Rounds	5,6,7,8
Optimizer	Adam
Loss function	MSE loss
Cyclic Learning Rate	0.002-0.0001

Table 1: Hyper-parameters for training of model

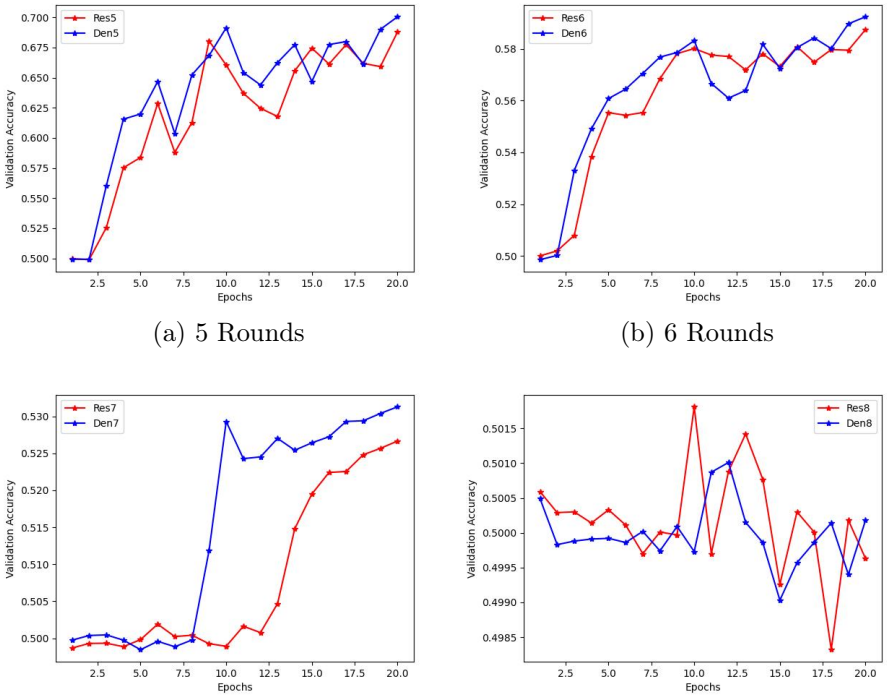
5 Results and Analysis:

In this section, we present the findings of our experiments on the differential cryptanalysis of the round reduced (rounds 5,6,7,8) SPECK 32/64 lightweight block cipher using the ResNet and DenseNet architectures. R5, R6, R7 & R8 refers to the ResNet architecture for rounds 5, 6, 7 and 8 respectively, and similarly D5, D6, D7 & D8 refers to the DenseNet architecture. Table 2 depicts the training and validation accuracy for both the models.

Rounds	**R (ResNet)**		**D (DenseNet)**	
	Training	Validation	Training	Validation
5	0.9332	0.6779	0.9309	**0.7005**
6	0.7952	0.5874	0.7917	**0.5923**
7	0.6096	0.5267	0.6053	**0.5313**
8	0.5012	0.4996	0.4998	0.5002

Table 2: Training and Validation accuracy for ResNet and DenseNet models

The DenseNet model achieved slightly better validation accuracy than the ResNet model for rounds 5, 6 and 7. For round 8, both the models failed to give a prominent result since the models could not learn an accurate pattern. Figure 4 below shows the comparison of accuracy for both the models.



(a) 5 Rounds

(b) 6 Rounds

(c) 7 Rounds

(d) 8 Rounds

Fig. 4: Validation Accuracy comparison

6 Conclusions

In this paper, we compared the ResNet and DenseNet architectures for differential cryptanalysis of the SPECK 32/64 lightweight block cipher. Our analysis attempted to evaluate their accuracy in deciphering the cipher's complicated differential behaviour.

According to our findings, the DenseNet architecture outperforms the ResNet architecture marginally. DenseNet achieved slightly higher predictions of ciphertext differences in the context of differential cryptanalysis. However, neither model produced a satisfactory result for an 8-round (or higher rounds) encryption cipher.

As of now, this work does not include a key retrieval approach. At last, our findings highlight the importance of architecture selection in differential crypt-

analysis. The observed accuracy improvements of DenseNet support its use in scenarios requiring lightweight block cipher analysis.

Acknowledgements

The Authors would like to show their sincere gratitude to the Scientific Analysis group (SAG),Defense Research and Development Organization (DRDO) for their invaluable support and collaboration throughout the course of this research. The authors would also like to thank Delhi Technological University (DTU),India for providing the opportunity to work in the field

References

1. Doukas, Nikolaos, and Nikolaos V. Karadimas. "A blind source separation based cryptography scheme for mobile military communication applications." WSEAS Trans. Commun 7.12 (2008): 1235-1245.
2. Lamprecht, C., et al. "Investigating the efficiency of cryptographic algorithms in online transactions." International Journal of Simulation: Systems, Science & Technology 7.2 (2006): 63-75.
3. Menezes, Alfred J., Paul C. Van Oorschot, and Scott A. Vanstone. Handbook of applied cryptography. CRC press, 2018.
4. Mahajan, Prerna, and Abhishek Sachdeva. "A study of encryption algorithms AES, DES and RSA for security." Global Journal of Computer Science and Technology 13.15 (2013): 15-22.
5. Hatzivasilis, G., Fysarakis, K., Papaefstathiou, I. et al. A review of lightweight block ciphers. J Cryptogr Eng 8, 141–184 (2018). https://doi.org/10.1007/s13389-017-0160-y
6. C. de Canniere, A. Biryukov and B. Preneel, "An introduction to Block Cipher Cryptanalysis," in Proceedings of the IEEE, vol. 94, no. 2, pp. 346-356, Feb. 2006, doi: 10.1109/JPROC.2005.862300.
7. Benamira, A., Gerault, D., Peyrin, T., Tan, Q.Q. (2021). A Deeper Look at Machine Learning-Based Cryptanalysis. In: Canteaut, A., Standaert, FX. (eds) Advances in Cryptology – EUROCRYPT 2021. EUROCRYPT 2021. Lecture Notes in Computer Science(), vol 12696. Springer, Cham. https://doi.org/10.1007/978-3-030-77870-5_28
8. L. Deng, G. Hinton and B. Kingsbury, "New types of deep neural network learning for speech recognition and related applications: an overview," 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, Vancouver, BC, Canada, 2013, pp. 8599-8603, doi:10.1109/ICASSP.2013.6639344.
9. Q. Wu, Y. Liu, Q. Li, S. Jin and F. Li, "The application of deep learning in computer vision," 2017 Chinese Automation Congress (CAC), Jinan, China, 2017, pp. 6522-6527, doi: 10.1109/CAC.2017.8243952
10. D. W. Otter, J. R. Medina and J. K. Kalita, "A Survey of the Usages of Deep Learning for Natural Language Processing," in IEEE Transactions on Neural Networks and Learning Systems, vol. 32, no. 2, pp. 604-624, Feb. 2021, doi: 10.1109/TNNLS.2020.2979670.
11. Dinu, D., Corre, Y.L., Khovratovich, D. et al. Triathlon of lightweight block ciphers for the Internet of things. J Cryptogr Eng 9, 283–302 (2019). https://doi.org/10.1007/s13389-018-0193-x

12. M. Cazorla, K. Marquet and M. Minier, "Survey and benchmark of lightweight block ciphers for wireless sensor networks," 2013 International Conference on Security and Cryptography (SECRYPT), Reykjavik, Iceland, 2013, pp. 1-6.
13. Manifavas, C., Hatzivasilis, G., Fysarakis, K., Rantos, K. (2014). Lightweight Cryptography for Embedded Systems – A Comparative Analysis. In: Garcia-Alfaro, J., Lioudakis, G., Cuppens-Boulahia, N., Foley, S., Fitzgerald, W. (eds) Data Privacy Management and Autonomous Spontaneous Security. DPM SETOP 2013 2013. Lecture Notes in Computer Science(), vol 8247. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-54568-9_21
14. Nyberg, K. (1996). Generalized Feistel networks. In: Kim, K., Matsumoto, T. (eds) Advances in Cryptology — ASIACRYPT '96. ASIACRYPT 1996. Lecture Notes in Computer Science, vol 1163. Springer, Berlin, Heidelberg. https://doi.org/10.1007/BFb0034838
15. Beaulieu, Ray, et al. "The SIMON and SPECK families of lightweight block ciphers." cryptology eprint archive (2013).
16. He, Kaiming, et al. "Deep residual learning for image recognition." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.
17. H. Li, J. Li, X. Guan, B. Liang, Y. Lai and X. Luo, "Research on Overfitting of Deep Learning," 2019 15th International Conference on Computational Intelligence and Security (CIS), Macao, China, 2019, pp. 78-81, doi: 10.1109/CIS.2019.00025.
18. Jastrzębski, S., Arpit, D., Ballas, N., Verma, V., Che, T., & Bengio, Y. (2017). Residual connections encourage iterative inference. arXiv preprint arXiv:1710.04773.
19. Gohr, Aron. "Improving attacks on round-reduced speck32/64 using deep learning." Advances in Cryptology–CRYPTO 2019: 39th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18–22, 2019, Proceedings, Part II 39. Springer International Publishing, 2019.
20. Ioffe, Sergey, and Christian Szegedy. "Batch normalization: Accelerating deep network training by reducing internal covariate shift." International conference on machine learning. pmlr, 2015.
21. Agarap, Abien Fred. "Deep learning using rectified linear units (relu)." arXiv preprint arXiv:1803.08375 (2018).
22. Huang, Gao, et al. "Densely connected convolutional networks." Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.
23. Toro-Vizcarrondo, Carlos, and T. Dudley Wallace. "A test of the mean square error criterion for restrictions in linear regression." Journal of the American Statistical Association 63.322 (1968): 558-572.
24. Kingma, Diederik P., and Jimmy Ba. "Adam: A method for stochastic optimization." arXiv preprint arXiv:1412.6980 (2014).