LRG-Net: Lightweight Residual Grid Network for Modeling Electrical Induction Motor Dynamics

Hao-Hsiang Yang
ASUS Intelligent Cloud Services
Taipei, Taiwan
danny1_yang@asus.com

Kuan-Chih Huang
ASUS Intelligent Cloud Services
Taipei, Taiwan
kuanchih_huang@asus.com

Wei-Ting Chen
Graduate Institute of Electronics Engineering
National Taiwan University
Taipei, Taiwan
jimmy3505090@gmail.com

Sy-Yen Kuo
Department of Electrical Engineering
National Taiwan University
Taipei, Taiwan
sykuo@ntu.edu.tw

***Abstract*—Modeling the dynamics of the induction motor is a crucial problem because induction motors are used widely in several scenarios. However, it is difficult to model the dynamics of the induction motor precisely, because the induction motor system is modeled as the complicated high order non-linear differential equation. To address this problem, we propose a novel residual grid network. The proposed grid connection effectively merges the various levels of feature information. Moreover, previous methods are usually based on complex network architecture with a mass of parameters. It may be infeasible for deploying this application on edge devices in real-world scenarios. Therefore, in the proposed method, we introduce the lightweight strategy with grid connection to reduce the number of parameters. Experimental results show that the proposed network contains fewer parameters but outperforms other existing models and achieves state-of-the-art performance on both simulated and real-world motor data.**

***Index Terms*—Motor dynamics, Residual blocks, Grid connection, Lightweight model**

I. Introduction

Modeling the dynamics of the induction motor is a crucial problem because induction motors have a wide range of applications. For example, electrical motors are used from more than 10^8 W in power plants to 10^{-6} W in electronic watches. Additionally, they cover a wide range of speed $> 10^5$ RPMs in centrifuge applications and torque $> 10^7$ Nm in mills [1]. For these services, it is required to understand the dynamical physical model of induction motors. The induction motor and its equivalent circuit are illustrated in Fig. 1 (a) and Fig. 1 (b). The inputs are voltages V_d, V_q, and rotor speed ω; outputs are currents I_d, I_q, and torque τ, as plotted in Fig. 1 (c) and Fig. 1 (d). The dynamical model of the induction motor is expressed:

$$[I_d, I_q, \tau] = F(t, V_d, V_q, \omega) \tag{1}$$

where F is the function of the induction motor model. The indices d and q denote three-phase quantities represented in a two-phase orthogonal rotating reference frame [4], and t means that the model is time-variant. According to [5], it is difficult to accurately model the dynamics of the induction motor because the induction motor system F is represented as the fifth-order nonlinear state-space model and some electrical parameters in the induction motor like resistances and inductances vary frequently with temperature.

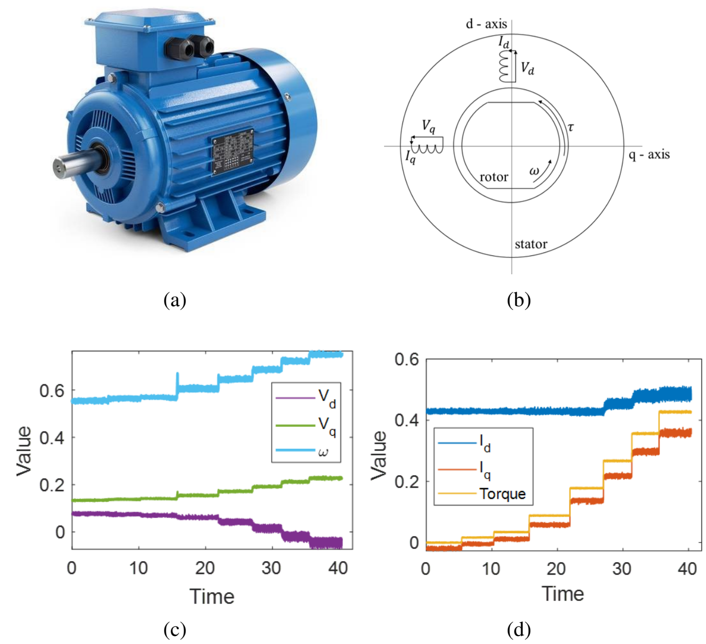


Fig. 1. The induction motor and its physical quantities: (a) The induction motor. (b) The equivalent circuit and corresponding physical quantities [1]. (c) The inputs are V_d, V_q and ω. (d) The outputs are I_d, I_q and τ. It is noted that all data are normalized between (-1, 1).

Recently, deep neural networks have obtained remarkable success in many applications like image processing [6]–[11], computational photography [12], [13], and complex physical simulation [14]. The classical encoder-decoder model [3], [15]

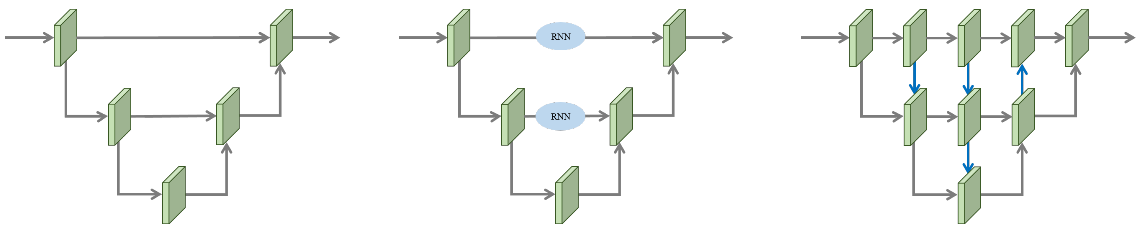


Fig. 2. Illustration of convolutional encoder-decoder networks. (a) Wave U-net [2]. (b) Wave U-net with recurrent skip connection [3]. (c) Proposed residual grid network. Different form (a) and (b), we integrate cross scale aggregation (blue arrows).

can be used to model the induction motor dynamics. Recent researches indicate that the convolutional neural network (CNN) architecture is more competitive than recurrent neural networks [16] (RNNs) and gated recurrent units (GRUs) [15] on many time series tasks. Furthermore, CNNs provide advantages of computational efficiency and modeling effectiveness due to the inherent parallelism [17]. Therefore, in this paper, we select the CNN architecture model as the backbone.

For the CNN based encoder-decoder models, they concatenate the stacks of convolution to extract different scale features and decode features to the output signal. The Wave U-net [2] utilizes skip connection to combine the identical size feature maps and reconstruct accurate signals. Recently, in [3], RNN layers are introduced to replace skip connections and extract temporal information. Furthermore, diagonalized recurrent skip connections are proposed to diagonalize weights in the RNN to decreases the number of parameters.

Although the previous methods can generate desired results, there still some issues which may degrade the performance and cause limitations in real-world scenarios. **First**, those models cannot fuse the features in different levels effectively, which may cause unsatisfactory predicting results. **Second**, previous methods [3], [17] usually consist of mass parameters, which may cause the infeasibility for deploying on edge device to monitor various electro-mechanical devices in real-time. To address the aforementioned problems, we propose a novel lightweight residual grid network which is shown in Fig. 2(c). The proposed model densely concatenates vertical and horizontal features in different scales. In addition, we leverage the residual grid units to substitute the vanilla convolutional units in the network because residual blocks improve gradient flow during the back-propagation, and prevents the exploding gradient problem [18]. With the effective and efficient feature extraction mechanism in the proposed architecture, our method can achieve better performance but with less usage of parameters.

We summarize the contributions in this paper as follows:

1) A novel end-to-end lightweight residual grid network (LRG-net) with grid topology is proposed. This architecture can aggregate different scale features for estimating the induction motor dynamics effectively.
2) Several experiments on simulated and real-world data show that the proposed method can achieve much better performance than previous methods.
3) With the lightweight network design, the proposed model can at least save **64%** of parameters but achieve the comparable performance compared with other methods. The proposed lightweight model is suitable for monitoring the status of motors in the real-world scenario.

II. Methodology

To design the lightweight models, many methods [19], [20] are proposed to adopt fewer parameters to approximate the vanilla convolutions. Furthermore others tasks adopt neural architecture search [21] to construct the lightweight model. Though these methods can mine efficient networks, their specific kernels are more difficult to be transformed into compatible models for edge devices than conventional convolutions [22] due to hardware limitation. On the other hands, other methods adopt the vanilla convolutions with the efficient feature fusion [23]–[25] to improve performance with fewer model parameters. For example, DenseNet [24] introduces direct connections between any two layers with the same feature-map size to improve the accuracy and save model parameters. Motivated by it, we propose the novel grid structure network that improves the feature propagation and reduces the number of parameters. In this section, We first describe the proposed LRG-net, and then provide model complexity analysis and loss function.

A. Residual Grid Network

The proposed LRG-net subsumes a backbone and two simple 1-D convolutional kernels for pre-processing and post-processing. The kernel size of all convolutions in our paper is set as 3. The pre-processing convolution C_i generates 32-channel feature maps from a given input motor signal and the post-processing convolution C_o transforms the feature maps into the final motor signal. The topology of the backbone is a grid of convolutional units, as shown in Fig. 3. In this paper, we choose a grid network with three rows and five columns, and it contains nine convolutional units. This model is divided into the encoder part (blue rectangle) and the decoder part (pink rectangle). The location of each convolutional unit in the network, C_{mn}, is specified by a height and length coordinate (m, n). The following notations are employed to describe in which section of the grid convolutional unit exists:

$$C_{mn} = \begin{cases} C_{en}, & m \leq 3 \\ C_{de}, & otherwise \end{cases} \tag{2}$$

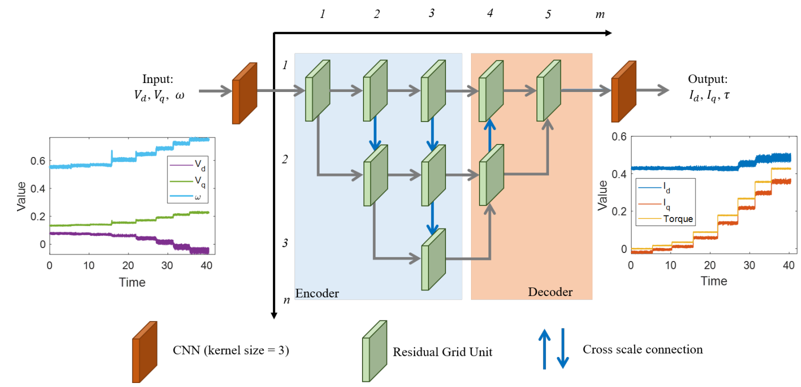


Fig. 3. The proposed lightweight residual grid network. Brown and green blocks are 1-D convolution and residual grid units.

where C_{en} and C_{de} means the encoder and decoder convolution units, respectively. Moreover, the x_{mn}, and y_{mn} denote the input and the output of the C_{mn}. We develop two kinds of grid connection to fuse the different scale convolutional units in encoder and decoder parts. In the encoder part, feature maps from the upper row are passed to the lower convolution units. Those top-down pathways effectively combine multi-scale features. For the decoder part, lower feature maps are passed to the upper convolutional unit. Those bottom-up paths are augmented to make low-layer information easier to be propagated. The formula of grid connection in the proposed network is expressed:

$$x_{mn}^{en} = \begin{cases} x_{11}, & m = n = 1 \\ y_{(m-1)n}, & m \neq 1, n = 1 \\ concat[\ (y_{(m-1)n},\ y_{m(n-1)}\ ], & otherwise \end{cases} \tag{3}$$

$$x_{mn}^{de} = concat[\ y_{(m-1)n},\ y_{(m-1)(n+1)}\ ] \tag{4}$$

where $concat$ is the concatenation operation. It is noted that both bottom-up and top-down pathways in our model fuse cross-scale features, which can learn the high order non-linear mapping for modeling motor dynamics [26].

To increase the accuracy and robustness, the convolutional units C_{mn} called residual grid unit is used in the LRG-net. This unit consists of a convolutional layer ($Conv$) and followed by three repeated residual layers [18]. Comparing to the conventional CNN, residual layers are intelligently learned residual functions with reference to layer inputs. This reformulation makes the training process effective, especially in the event of deeper networks [18]. The final output is a summation of two convolutions and it can be expressed as:

$$\begin{aligned} y_{mn} &= C_{mn} \otimes x_{mn} \\ &= f(f(f(Conv(x_{mn})))) + x_{mn} \end{aligned} \tag{5}$$

where $\otimes$ is the convolution operator, and f is the function of the residual layer. The illustration of the residual grid unit is depicted in Fig. 4.

B. Analysis of the Lightweight Model

Efficient grid connection encourages us to adjust a number of convolution channel for fewer model parameters. To strike a balance between the accuracy and the model complexity, we set the number of feature maps at all convolutions to 32. We find that in spite of increasing the number of feature maps, the LRG-net does not achieve significant improvement. According to [20], the authors indicate the redundancy in feature maps is an essential characteristic for those successful CNN-based methods but increases both model parameters and computational cost. Moreover, unlike high-level time series analysis tasks, such as sleeping staging and trading forecast, modeling motor dynamics does not require complex and high dimensional features. Therefore, with grid connection to extract multi-scale features, reducing the channel in the convolutions is feasible to design a lightweight model. Overall, details of LRG-Net are shown in Table I. The input dimension of C_{mn} is 100×64 if input tensors are concatenated. Otherwise, the input dimension is 100×32.

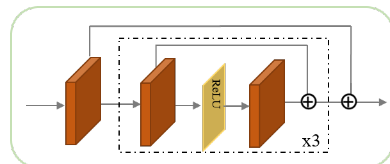


Fig. 4. The overall residual grid unit in the LRG-net. This unit contains one convolution layer and three residual layers.

TABLE I
THE COMPONENTS AND DETAIL OF THE LRG-NET.

Layers	Components	Input Size	Output Size
C_i	$Conv$	100×3	100×32
C_{mn}	$Conv \rightarrow 3 \times (Conv \rightarrow$ReLU$\rightarrow Conv)$	100×32 or 64	100×32
C_o	$Conv$	100×32	100×3
Params	228348		

C. Loss Function

To train our LRG-net, the Charbonnier loss [27] is applied and expressed as:

$$L_{Cha}(x, \hat{x}) = \frac{1}{T} \sum_{i}^{T} \sqrt{(x_i - \hat{x}_i)^2 + \epsilon^2} \tag{6}$$

where x means the predicted signal, $\hat{x}$ means the ground truth signal, and ϵ is a small constant (e.g., 10^{-3}). Instead of minimizing the mean square errors, this loss function is robust to handle outliers and more stable during training. It is noted when ϵ is 0, Eq.(6) is equivalent to L_1 loss.

III. Experimental Results

A. Datasets and Training Details

We train and evaluate the proposed network on the simulated and the real-world induction motor signal collected in [3]. The simulated dataset generated by Simulink [4] consists of simulations performed by the control law described in [5] and covers a wide range of operating conditions for generalization. The real-world dataset is recorded from a 4-kilowatt induction motor. Data from 10 different operating conditions are collected. The sampling rates of both data are 250Hz.

In our experiments, the data are split into three parts; training and validation parts subsume 70% and 30% of the simulation data, respectively. The real sensor data are tested from the model trained on the simulated data. During the training process, the input lengths are set as 100 and all data are normalized to (-1, 1). After sampling, we have 400k trained data. A large amount of data can avoid over-fitting. The SGD [29] is used as an optimization algorithm with a mini-batch of 512. The learning rate starts from 0.1 and is divided by ten after 50 epochs. The models are trained for 200 epochs. Mean absolute error (MAE), root mean square error (RMSE), root mean square logarithmic error (RMSLE), and symmetric mean absolute percentage error (SMAPE) [30] are chosen as objective metrics for quantitative evaluation. The four metrics are written as:

$$\mathrm{MAE}(x, \hat{x}) = \frac{1}{T}\sum_{i}^{T}|x - \hat{x}_i| \tag{7}$$

$$\mathrm{SMAPE}(x, \hat{x}) = \frac{100}{T}\sum_{i}^{T}\frac{|x - \hat{x}|}{|x| + |\hat{x}|} \tag{8}$$

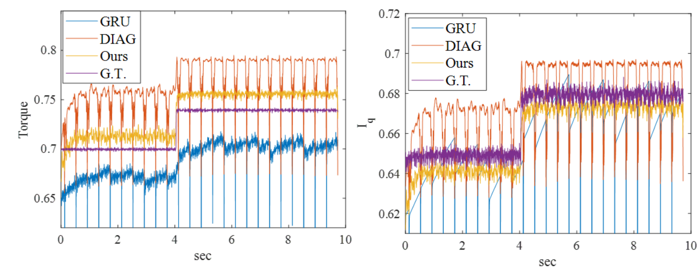
$$\mathrm{RMSE}(x, \hat{x}) = \sqrt{\frac{1}{T}\sum_{i}^{T}(x - \hat{x}_i)^2} \tag{9}$$

$$\mathrm{RMSLE}(x, \hat{x}) = \sqrt{\frac{1}{T}\sum_{i}^{T}(\log(x+1) - \log(\hat{x}_i + 1))^2} \tag{10}$$

B. Modeling Induction Motor Dynamics Results

We select five state-of-the-art methods as deep learning-based benchmarks to make fair comparisons with our method. The five methods are GRU [28], time convolution network (TCN) [17], Wave U-net [2], Wave U-net with RNN skip connection (RNN skip) [3], and Wave U-net with diagonal RNN skip connection (DIAG skip) [3]. All comparative methods are trained with an identical setting. Comparison results on the simulated and the real-world data, GFLOPs and parameters of the model are shown in Table II. The errors from GRU are larger than errors from other methods, which demonstrates that CNN can model dynamics better. The proposed method outperforms the state-of-the-art by a wide margin on two datasets. Furthermore, compared to the second-best performance method (i.e., DIAG skip [3]), our model can achieve better accuracy and at least save 64 % parameters with less computation. It can prove that the grid and cross-scale connection not only save parameters but effectively extract the features from different scales in the proposed network.

We also plot the modeling results of torque and I_q for one of the raw samples in Fig. 5. The results from other models have some offset in its prediction. On the other hand, the result from our model is the closest to the ground truth.



(a) (b)

Fig. 5. The visualization of predicted signals by GRU [15], DIAG skip [3], our model and the ground truth. (a) Torque τ. (b) I_q.

C. Ablation Study

To verify the effectiveness of the residual grid unit and the Chabonnier loss, we conduct the ablation studies. The first experiment uses our model without residual blocks and grid connection. The second experiment applies the grid structure model without residual blocks. We also train the LRG-net with different losses such as smooth L_1 , L_1 and L_2. We apply the simulated data and list MAE and SAMPE of various experiments in Table III. As shown in Table III, the combination of grid connection, residual blocks and Chabonnier loss contributes to the best performance.

IV. Conclusion

In this paper, we propose a new lightweight residual grid network (LRG-net) to model the induction motor dynamics. The model applies residual blocks and is stacked as a grid shape to effectively extract the feature of the input signal and predict the output signal. Several experimental results show our network outperforms other deep learning models. With effective grid connection, our model takes the least parameters, which is suitable for monitoring the status of motors in the real-world scenario.

In the future, we will extend this model to design the motor control raw. Moreover, LRG-net topology will be investigated for other applications like speech enhancement [31] and image denoising [6], [32]

V. Acknowledgement

We thank to National Center for High-performance Computing (NCHC) for providing computational and storage resources.

TABLE II

QUANTITATIVE COMPARISONS OF EXISTING FIVE METHODS ON SIMULATION AND THE REAL-WORLD INDUCTION MOTOR SIGNAL.

	#Model params	#Model flops	Simulation Data				Real-world Data			
			MAE	SMAPE	RMSE	RMSLE	MAE	SMAPE	RMSE	RMSLE
GRU [28]	0.23M	0.03G	0.0392	7.03	0.0666	0.0437	0.0499	9.20	0.0738	0.0498
TCN [17]	0.55M	0.01G	0.0324	5.69	0.0568	0.0368	0.0417	7.69	0.0570	0.0417
Wave U-net [2]	0.37M	0.02G	0.0202	3.62	0.0434	0.0292	0.0327	6.11	0.0495	0.0318
RNN skip [3]	0.64M	0.05G	0.0179	3.22	0.0431	0.0283	0.0310	5.75	0.0509	0.0322
DIAG skip [3]	0.62M	0.05G	0.0197	3.54	0.0448	0.0293	0.0300	5.86	0.0484	0.0318
LRG-net	**0.23M**	**0.02G**	**0.0154**	**2.78**	**0.0376**	**0.0248**	**0.0297**	**5.43**	**0.0427**	**0.0274**

TABLE III

THE ABLATION STUDY SHOWS THE EFFECTIVENESS OF THE RESIDUAL LAYER (RES), THE GRID CONNECTION (GRID) AND LOSS FUNCTION.

Res	Grid	Loss	MAE	SMAPE
		L_{Cha}	0.0221	3.68
	$\surd$	L_{Cha}	0.0199	3.60
$\surd$	$\surd$	Smooth L_1	0.0162	2.88
$\surd$	$\surd$	L_1	0.0176	2.94
$\surd$	$\surd$	L_2	0.0182	3.02
$\surd$	$\surd$	L_{Cha}	**0.0154**	**2.78**

REFERENCES

[1] Y. Dorjsuren, L. Tumenbayar, and J. Tsevegmid, "Three-axis dynamic modeling of induction motor," *International Journal of mathematical models and methods in applied sciences*, 2015.

[2] D. Stoller, S. Ewert, and S. Dixon, "Wave-u-net: A multi-scale neural network for end-to-end audio source separation," 2018.

[3] S. Verma, N. Henwood, M. Castella, F. Malrait, and J.-C. Pesquet, "Modeling electrical motor dynamics using encoder-decoder with recurrent skip connection." in *Association for the Advancement of Artificial Intelligence, AAAI*, 2020.

[4] H. Erdal and V. Halliyev, "A simulator for the course of electromechanical control techniques," in *Proceedings of the 10th WSEAS International Conference on Automatic Control, Modelling & Simulation*, 2008.

[5] F. Jadot, F. Malrait, J. Moreno-Valenzuela, and R. Sepulchre, "Adaptive regulation of vector-controlled induction motors," *IEEE Transactions on Control Systems Technology*, 2009.

[6] H.-H. Yang, C.-H. H. Yang, and Y.-C. J. Tsai, "Y-net: Multi-scale feature aggregation network with wavelet structure similarity loss function for single image dehazing," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020.

[7] W.-T. Chen, J.-J. Ding, and S.-Y. Kuo, "PMS-net: Robust haze removal based on patch map for single images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[8] W.-T. Chen, H.-Y. Fang, J.-J. Ding, and S.-Y. Kuo, "PMHLD: patch map-based hybrid learning dehazenet for single image haze removal," *IEEE Transactions on Image Processing*, 2020.

[9] W.-T. Chen, H.-Y. Fang, J.-J. Ding, C.-C. Tsai, and S.-Y. Kuo, "JSTASR: Joint size and transparency-aware snow removal algorithm based on modified partial convolution and veiling effect removal," in *European Conference on Computer Vision*, 2020.

[10] H.-H. Yang, K.-C. Huang, and W.-T. Chen, "LAFFNet: A lightweight adaptive feature fusion network for underwater image enhancement," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2021.

[11] H.-H. Yang, C.-H. H. Yang, and Y.-C. F. Wang, "Wavelet channel attention module with a fusion network for single image deraining," in *IEEE International Conference on Image Processing (ICIP)*, 2020.

[12] H.-H. Yang, W.-T. Chen, H.-L. Luo, and S.-Y. Kuo, "Multi-modal bifurcated network for depth guided image relighting," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2021.

[13] H.-H. Yang, W.-T. Chen, and S.-Y. Kuo, "S3Net: A single stream structure for depth guided image relighting," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2021.

[14] A. Sanchez-Gonzalez, J. Godwin, T. Pfaff, R. Ying, J. Leskovec, and P. W. Battaglia, "Learning to simulate complex physics with graph networks," *arXiv preprint arXiv:2002.09405*, 2020.

[15] K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder–decoder for statistical machine translation," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014.

[16] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[17] E. Aksan and O. Hilliges, "Stcn: Stochastic temporal convolutional networks," in *International Conference on Learning Representations*, 2018.

[18] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016.

[19] Z. Lu, K. Deb, and V. N. Boddeti, "Muxconv: Information multiplexing in convolutional neural networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

[20] K. Han, Y. Wang, Q. Tian, J. Guo, C. Xu, and C. Xu, "Ghostnet: More features from cheap operations," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

[21] Y. Li, X. Jin, J. Mei, X. Lian, L. Yang, C. Xie, Q. Yu, Y. Zhou, S. Bai, and A. L. Yuille, "Neural architecture search for lightweight non-local networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020.

[22] Y. Gorbachev, M. Fedorov, I. Slavutin, A. Tugarev, M. Fatekhov, and Y. Tarkan, "Openvino deep learning workbench: Comprehensive analysis and tuning of neural networks inference," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2019.

[23] C. Feichtenhofer, H. Fan, J. Malik, and K. He, "Slowfast networks for video recognition," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019.

[24] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017.

[25] Y. Chen, H. Fan, B. Xu, Z. Yan, Y. Kalantidis, M. Rohrbach, S. Yan, and J. Feng, "Drop an octave: Reducing spatial redundancy in convolutional neural networks with octave convolution," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019.

[26] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018.

[27] J. T. Barron, "A general and adaptive robust loss function," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[28] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

[29] L. Bottou, "Online learning and stochastic approximations," *On-line learning in neural networks*, vol. 17, no. 9, p. 142.

[30] P. Goodwin and R. Lawton, "On the asymmetry of the symmetric mape," *International journal of forecasting*, 1999.

[31] C.-H. H. Yang, J. Qi, P.-Y. Chen, Y. Ouyang, I.-T. D. Hung, C.-H. Lee, and X. Ma, "Enhanced adversarial strategically-timed attacks against deep reinforcement learning," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 3407–3411.

[32] H.-H. Yang and Y. Fu, "Wavelet u-net and the chromatic adaptation transform for single image dehazing," in *2019 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2019, pp. 2736–2740.