

LiNet: A Lightweight Network for Image Super Resolution

Armin Mehri
*Computer Vision Center,
Edifici O, Campus UAB,
08193, Bellaterra,
Barcelona, Spain
Email: amehri@cvc.uab.es

Parichehr B.Ardakani
*Computer Vision Center,
Edifici O, Campus UAB,
08193, Bellaterra,
Barcelona, Spain
Email: pbehjati@cvc.uab.es

Angel D. Sappa
[†]ESPOL Polytechnic University,
Guayaquil, Ecuador
*Computer Vision Center,
08193, Bellaterra, Barcelona, Spain
Email: sappa@ieee.org

Abstract—This paper proposes a new lightweight network, LiNet, that enhancing technical efficiency in lightweight super resolution and operating approximately like very large and costly networks in terms of number of network parameters and operations. The proposed architecture allows the network to learn more abstract properties by avoiding low-level information via multiple links. LiNet introduces a Compact Dense Module, which contains set of inner and outer blocks, to efficiently extract meaningful information, to better leverage multi-level representations before upsampling stage, and to allow an efficient information and gradient flow within the network. Experiments on benchmark datasets show that the proposed LiNet achieves favorable performance against lightweight state-of-the-art methods.

I. INTRODUCTION

Single Image Super-Resolution (SISR) attempts to restore a high-resolution (HR) image from its low-resolution (LR) one, with better visual quality and improved in details such as texture and edges. The SISR is still an important and demanding topic for research due to its complicated nature and high functional interest in improving image quality and texture. SR is also important for other devices such as HD TVs, computer monitors and handheld devices such as cameras, laptops, tables, and many more devices. In addition to that, SR helps to achieve and improve in a divers range of computer vision tasks, for instance, object detection [1], security and surveillance imaging [2], face recognition [3], and many other domains [4], [5], [6]. Single Image Super Resolution is challenging because: *i*) SR is an ill-posed inverse problem. In other word, there are multiple solutions to the same low-resolution image instead of a single unique solution; *ii*) By increasing the scale factor, difficulty of the problem increases as well [7]. Because, the recovery of lost scene data becomes much more complicated by greater factors, which also contribute to the creation of wrong information; and *iii*) Quality evaluation of output is not straightforward—i.e., quantitative metrics (e.g. PSNR, SSIM [8]) only loosely correlate to human perception [9].

In recent years, due to the exponential progress and relentless growth of deep learning approaches, there has been an enormous proliferation of CNN models to perform the SISR. The performance of SR approaches has been continuously en-

hanced by developing new architecture or by adding/or introducing new techniques or loss functions. Although substantial improvements have been made, most of the SR works have been focusing to increase the PSNR with the construction of a very deep and costly network, which resulting in an increase of the number of computational operations. Regrettably, these SISR approaches are not practical for low-capacity devices neither for real-world applications.

In this paper, a novel lightweight architecture called LiNet is presented to be a practical network for real-world applications and resolve the aforementioned problems by focusing on learning high-level information and adaptively learn the most useful features[†] and suppress the worthless ones. Furthermore, in order to achieve a better balance between efficiency and applicability, a new module is introduced, called the Compact Dense Module (CDM), which take advantage of different learning connections (global and local). CDM includes Efficient Dense Blocks (EDB) that are connected to each other by global skip-connections; and each EDB contains a set of inner Efficient Residual block (ERB) connected by local skip-connection together. These connections help our network to propagate information across LiNet network and make use of multi-level learning connections. Therefore, network has access to both intermediate and high frequency information by connecting different blocks with local and global connections followed by 1×1 conv layer. As a result the network produces high quality reconstruction results.

In summary, the main contributions of this paper are three-fold:

- A novel and efficient lightweight network is introduced, which achieves best performance compared to lightweight SOTA approaches and obtains competitive results compare to computational expensive models.
- A Compact Dense Module is introduced to effectively enhance the outcomes through multi-level representation and multiple residual learning.
- An efficient Residual Block is proposed, which consists of a multi-path residual learning to extract the features efficiently at a negligible computational cost.

II. RELATED WORKS

Due to the space limitation just some of recent state of the art deep learning (DL) based SR approaches are detailed in section II-A. Then, SR lightweight models, which focus on compressing the number of model's parameters and operations in a nutshell are reviewed in section II-B.

A. Single Image Super-Resolution Networks

The story of Single Image Super-Resolution in deep learning started with introducing SRCNN [7] by using CNN to address SR task. SRCNN achieves superior results compared with traditional approaches but due to using an upsampled image as input, the network became computationally expensive. Later on, to overcome this drawback, FSRCNN [10] and ESPCN [11] have been proposed to reduce the high cost of computing and running time by feeding with a LR image as input to the network, and then upsample the features close to output of the network. This technique leads to design low-memory strategies as opposed to SRCNN. However, the overall output could be decreased if not enough layers are available after the upsampling phase. In addition to that, these approaches could not handle multi-scale training process because the input image has different size for each scale factors.

The power of deep learning is coming from deep layers, the aforementioned approaches are known as shallow network due to the number of layers and training difficulties in their networks. For this reason, Kim et al. [12] use residual learning to alleviate training difficulties and expand the complexity of their network by adding 20 convolutional layers. Afterwards, Kim et al. introduce DRCN by using recursive learning to share the weights between layers in order to handles deep network. Later, Tai et al. [13] present DRRN by following the work of [14]. DRRN improves the DRCN approach by combining residual blocks and recursive technique. Then, Tai et al. introduce a new approach, MemNet [15] to overcome the problem of long-term dependency. Tai et al, used memory block in MemNet for deeper network with 84 layers. Therefore, CNN methods show a better reconstruction results for a deeper network with different types of residual connections. As a results, EDSR [16] proposed by Lim et al. remove Batch Normalization from residual block to improve and increase the network. Then, RDN [17] introduced by Zhang et al., employs residual and dense skip connections to completely take advantage of hierarchical features inside the network. Even though the very deep networks achieve a high reconstruction results, these networks cannot be used in real-world applications because these networks are computationally expensive (high number of network parameters and number of operations) and leads to high risk of over-fitting.

B. Lightweight Image Super-Resolution Networks

Over the last years, researchers try to overcome these drawbacks by designing a powerful and yet efficient neural networks or compressing pretrained network. For example, SqueezeNet [18] expands on the concept of AlexNet with

much fewer network parameters than AlexNet and achieves comparable results. Later no, MobileNet, [19] introduced by Howard et al., is an efficient network with applying depth-wise and point wise convolutional layers instead of normal ones to obtain comparable results with less computation.

Thus, various lightweight networks proposed into Image SR task are reviewed. Ahn et al. [20] introduce CARN, an efficient network, which is suitable for low-capacity devices. CARN built upon the cascade mechanism and residual network to design a lightweight network and improve the final results. Later, Chu et al. introduce MoreMNA [21] and FALSr [22] family by applying a neural architecture search strategy in SR task to design an efficient network. Despite the fact that both networks perform well with less computational, the performance of these methods is restricted, due to NAS search space constraints. All these works suggest that the lightweight SR networks can keep a good trade-off between reconstruction quality and parameters.

III. LiNET: A LIGHTWEIGHT NETWORK

This section presents the main characteristics of the proposed network. Firstly, the network structure is detailed. Then, the architecture of residual blocks is presented.

A. Network Structure

The proposed LiNet is based on the MobileNet [19] architecture to be efficient and practical for real-world applications, but with several differences. The key difference between LiNet and MobileNet is on the additional learning pathway and different operations in ERB, presence of inner and outer dense block, and different learning connections between them, which make inner block almost identical to a outer one. The proposed LiNet architecture is depicted in Fig 1. LiNet built upon the three different modules, namely, initial Feature Extraction (inFE) to extract the shallow features by using a 3×3 conv layer; Compact Dense Module (CDM), which contains three Efficient Dense Blocks (EDB) with three inner Efficient Residual Blocks (ERB) to focus on extracting mid- and high-level feature maps; the last module of LiNet is the upsampling module. Let us assume I_{LR} is the input image to the network and I_{SR} is the reconstructed image. Thus, initial feature extraction can be defined as:

$$\mathbf{H}_{inFE} = f_{inFE}(\mathbf{I}_{LR}; W_c), \quad (1)$$

where $f(\cdot)$ and W_c indicates convolutional layer and its weights respectively. $f_{inFE}(\cdot)$ is initial feature extractor, which applied on LR input image. H_{inFE} is the output of initial operation that later on is used as the input to Long Residual Learning and Compact Dense Module. Assume $\mathbf{H}_{CDM}^{i,j}$ to be the output of CDM, which contains i -th Efficient Dense Blocks that have j -th inner Efficient Residual Block. Therefore, CDM can be formalized as follow:

$$\mathbf{H}_{CDM} = f([\mathbf{H}_{inFE}, \dots, \mathbf{H}_{EDB}^{i-1}(\mathbf{H}_{ERB}^{j-1,R}; W_c^j), \mathbf{H}_{EDB}^i]; W_c^i), \quad (2)$$

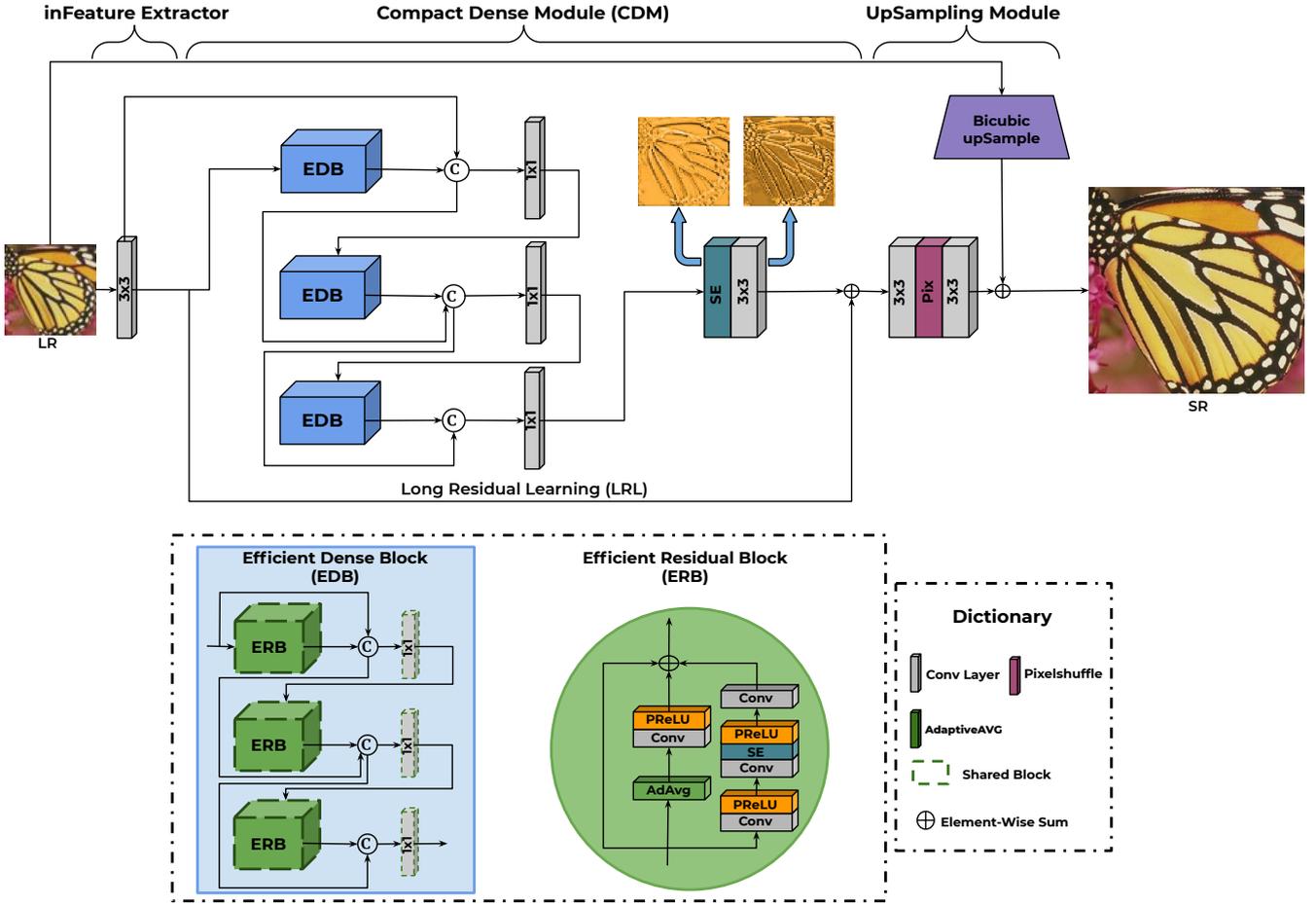


Fig. 1: Illustration of proposed network architecture (LiNet).

where H_{CDM} denotes the output of the Compact Dense Module. Also, it's worth mentioning that the proposed CDM, which contains i -th Efficient Dense Blocks with j -th inner Efficient Residual Blocks, takes advantage of multi-level connections to: *i*) help the proposed approach to learn multi-level representations and *ii*) help the network to disperse information rapidly from lower to higher layers and vice versa for back-propagation. In this case, the network shares all the information between all the blocks and results in better reconstruction results. Therefore, Efficient Dense Blocks with inner Efficient Residual Blocks can be defined as:

$$H_{EDB}^i = f([H_{ERB}^{j,R}, \dots, H_{ERB}^{j-1,R}(H^{i-1}; W_c^i)]; W_c^j). \quad (3)$$

The feature maps obtained from EDB blocks feed into a SE attention module followed by a 3×3 convolutional layer to first re-calibrate the feature maps and then extract more abstract features. Finally, they are accumulated with Long Residual Learning to effectively reduce the problems of vanishing/exploding gradient and ease the training process. So, the whole process of the CDM module can be defined as follows:

$$H_{CDM} = f(M_{SE}(H_{EDB}(H_{ERB}; W_c); W_c); W_c) + H_{LRL}, \quad (4)$$

where M_{SE} is the SE attention module and H_{LRL} is Long Residual Learning.

The final step after extracting all the informative information is to upsample the extracted feature maps through upsampling modules. Two different upsample modules have been used in order to achieve better reconstruction results: *i*) a 3×3 conv layer followed by a pixel shuffle layer and another 3×3 conv to upsample the extracted features and *ii*) a Bicubic interpolation to upsample the input image. Thus, upsample modules are defined as:

$$H_{UP} = f(f_{Pix}^\uparrow(H_{CDM}); W_c) + f_{Bic}^\uparrow(I_{LR}), \quad (5)$$

where $f_{Pix}(\cdot)$ denotes the pixel shuffle upsample module and $f_{Bic}(\cdot)$ is bicubic interpolation. So, we can formulate all the LiNet processes as follows:

$$I_{SR} = H_{LiNet}(I_{LR}) \quad (6)$$

In the next section, details of the Efficient Residual Block are given since this is another key contribution of the proposed architecture.

B. Efficient Residual Block

The Efficient Residual Block (ERB) is designed with a minor computational cost by introducing multi-learning pathways to make ERB structure more effective in the SR task. Each of the proposed learning pathways have different duties, which will be explain in details.

i) The first learning pathway consists of 1×1 pointWise Conv layer, a 3×3 depthWise Conv layer with SE attention module, and a linear 1×1 pointWise Conv layer. Since our aim was to have a lightweight network, PointWise and DepthWise convolutional layers have been used to replace a full convoluted operator with a factorized version by breaking the transition into two different layers to conduct light-weight filtering by adding a single convoluted filter per input path channel. *ii)* The second learning pathway consists of an adaptive average pooling followed by a linear 1×1 PointWise Conv layer. The intuition behind of using the mentioned operations are: *a)* to help the network deal better with the noisy inputs, since LR images includes a lot of noise and artifacts; and *b)* to mitigate the dimensionality of each feature maps but preserves essential details. *iii)* The third pathways is a local residual connection inside the Efficient Residual Block to link beginning and end of the block. The reasons of using local residual connection are: *a)* to ease the training process; and *b)* to allow the network to have access to earlier information that was not originally updated.

Empirically, we found that each component of the ERB is critical to achieve superior SR results so the proposed Efficient Residual Block can be define as:

$$\mathbf{H}_{ERB} = f_p(\sigma(f_d(f_p(\mathbf{H}^{i-1}; W_c^{i,1}); W_c^{i,2})); W_c^{i,3}) + f_p(\mathbf{P}_{avg}(\mathbf{H}^{i-1}; W_c^{i,1}); W_c^{i,2}) + \mathbf{R}_{local}, \quad (7)$$

where \mathbf{H}_{ERB} is the aggregation of all Learning pathways. \mathbf{P}_{avg} denotes avarage pooling layer. The subscript p and d denote the pointWise and depthWise convolutional operations respectively.

IV. SYSTEM SETUP

This section presents details on the dataset used for training the proposed architecture together with the evaluation metrics used to measure its performance.

A. Datasets & Evaluation Metrics

There are various single image super-resolution datasets available in the literature, such as the Berkeley Segmentation Dataset, Set291, among others. However, due to lack of training images for a deep neural network, recent SR approaches use DIV2K dataset [23] for training and validating their models due to including high quality ($2K$ resolution) and divers images. DIV2K contains 1000 images in total, which split up into 800 images for training, 100 images for validation, and 100 images for testing. The proposed model trained with all training images and four standard benchmark datasets has been considered for testing and benchmarking, namely, *Set5* [24], *Set14* [25], *B100* [26], *Urban100* [27]. In order to

measure the performance of the proposed model, PSNR and SSIM [8] are used to compute the differences between the obtained images and the corresponding ground truths. For a fair comparison, both quantitative metrics are computed on the Y channel of the $YCbCr$ color space.

B. Training Details

In the training phase, RGB input patches from each of the randomly selected 64 LR training images are used with a scale of 64×64 . Random horizontal flips and rotation of 90 degrees applied on the randomly selected patches are used as the augmentation methods. ADAM optimizer was employed with setting $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-8}$. The learning rate starts with 0.001 and halved the learning rate every 4×10^5 steps. To optimize our model $L1$ loss function has been used. The LiNet is implemented in the PyTorch framework.

V. EXPERIMENTAL RESULTS

In this section, comparisons with state-of-the-art (SOTA) SR approaches are provided. In addition, for more fair comparisons, the proposed LiNet is compared to other SOTA in terms of the number of network parameters along with the number of operations (Table I) and the inference time (Table II). The proposed model is compared with the recent lightweight SOTA methods and, it can also to be compared with deep and expensive computations models due to superior achievements; even in those scenarios, the proposed lightweight LiNet approach can gain higher or more competitive results between all of them. This implies that the proposed approach has a well balance between the number of parameters/operations and the restoration quality.

A. Quantities and Qualitative Analysis

In Table I, comparisons of the proposed LiNet and 15 recent SOTA models are presented. These quantitative comparisons, with lightweight and expensive methods, on benchmark datasets illustrate the good performance of the proposed approach. On all benchmark datasets, the proposed approach (LiNet) is able to outperforms all lightweight(less than 1000K number of parameters) SOTA approaches on all scale factors. Additionally, comparison with deep and costly SOTA methods (up to 6000K parameters) are presented in Table I. As an example, it can be mentioned that the comparison of the proposed LiNet, with only 509K parameters and MSRN with 6078K parameters and 160 layers, which is about $12 \times$ heavier computation; the proposed LiNet can achieve better reconstruction results in most the benchmark datasets when compared with MSRN.

A couple of visual results are presented in Fig 2. In general, the proposed LiNet will yield to more accurate reconstruction results. As can be seen in Fig 2, the orientation of texture on the reconstructed super resolution images from all comparative methods are absolutely incorrect. However, the results of the LiNet, use the abstract properties entirely and reliably to restore images close to the texture of the ground truth.

Model	Scale	Params	Flops	Set5	Set14	B100	Urban100
SRCNN	$\times 2$	57K	52.7G	36.66/0.9542	32.42/0.9063	31.36/0.8879	29.50/0.8946
	$\times 3$			32.75/0.9090	29.28/0.8209	28.41/0.7863	26.24/0.7989
	$\times 4$			30.48/0.8628	27.49/0.7503	26.90/0.7101	24.52/0.7221
FSRCNN	$\times 2$	12K	6.0G	37.00/0.9558	32.63/0.9088	31.53/0.8920	29.88/0.9020
	$\times 3$			33.16/0.9140	29.43/0.8242	28.53/0.7910	26.43/0.8080
	$\times 4$			30.71/0.8657	27.59/0.7535	26.98/0.7150	24.62/0.7280
VDSR	$\times 2$	665K	612.6G	37.53/0.9587	33.03/0.9124	31.90/0.8960	30.76/0.9140
	$\times 3$			33.66/0.9213	29.77/0.8314	28.82/0.7976	27.14/0.8279
	$\times 4$			31.35/0.8838	28.01/0.7674	27.29/0.7251	25.18/0.7524
LapSRN	$\times 2$	813K	29.9G	37.52/0.9590	33.08/0.9130	31.80/0.8950	30.41/0.9100
	$\times 4$		149.4G	31.54/0.8850	28.19/0.7720	27.32/0.7280	25.21/0.7560
DRRN	$\times 2$	297K	6796.6G	37.74/0.9591	33.23/0.9136	32.05/0.8973	31.23/0.9188
	$\times 3$			34.03/0.9244	29.96/0.8349	28.95/0.8004	27.53/0.8378
	$\times 4$			31.68/0.8888	28.21/0.7720	27.38/0.7284	25.44/0.7638
MemNet	$\times 2$	667K	2662.4G	37.78/0.9597	33.28/0.9142	32.08/0.8978	31.31/0.9195
	$\times 3$			34.09/0.9248	30.00/0.8350	28.96/0.8001	27.56/0.8376
	$\times 4$			31.74/0.8893	28.26/0.7723	27.40/0.7281	25.50/0.7630
CARN-M	$\times 2$	412K	91.2G	37.53/0.9583	33.26/0.9141	31.92/0.8960	31.23/0.9193
	$\times 4$		46.1G	33.99/0.9236	30.08/0.8367	28.91/0.8000	27.55/0.8385
	$\times 4$		32.5G	31.92/0.8903	28.42/0.7762	27.44/0.7304	25.62/0.7694
SRFBN-S	$\times 2$	483K	119G	37.78/0.9597	33.35/0.9156	32.00/0.8970	31.41/0.9207
	$\times 3$			34.20/0.9255	30.10/0.8350	28.96/0.8010	27.66/0.8415
	$\times 4$			31.98/0.9594	28.45/0.7779	27.44/0.7313	25.71/0.7719
DRCN	$\times 2$	1774K	222.8G	37.63/0.9588	33.04/0.9118	31.85/0.8942	30.75/0.9133
	$\times 3$		118.8G	33.82/0.9226	29.76/0.8311	28.80/0.7963	27.15/0.8276
	$\times 4$		90.9G	31.53/0.8854	28.02/0.7670	27.23/0.7233	25.14/0.7510
CARN	$\times 2$	1592K	222.8G	37.76/0.9590	33.52/0.9166	32.09/0.8978	31.92/0.9256
	$\times 3$		118.8G	34.29/0.9255	30.29/0.8407	29.06/0.8434	28.06/0.8493
	$\times 4$		90.9G	32.13/0.8937	28.60/0.7806	27.58/0.7349	26.07/0.7837
SRMDNF	$\times 2$	1513K	347.7G	37.79/0.9600	33.32/0.9150	32.05/0.8980	31.33/0.9200
	$\times 3$	1530K	156.3G	34.12/0.9250	30.04/0.8370	28.97/0.8030	27.57/0.8400
	$\times 4$	1555K	89.3G	31.96/0.8930	28.35/0.7770	27.49/0.7340	25.68/0.7730
OISR-RK-S	$\times 2$	1370K	316.2G	37.98/0.9604	33.58/0.9172	32.18/0.8996	32.21/0.9290
	$\times 3$	1370K	160.1G	34.39/0.9273	30.33/0.8420	29.10/0.8083	28.03/0.8544
	$\times 4$	1520K	114.2G	32.21/0.8950	28.63/0.7822	27.58/0.7364	26.14/0.7888
MSRN	$\times 2$	5930K	1365.4G	38.08/0.9605	33.74/0.9170	32.23/0.9013	32.22/0.9326
	$\times 3$	6008K	621.2G	34.38/0.9262	30.34/0.8395	29.08/0.8041	28.08/0.8554
	$\times 4$	6078K	349.8G	32.07/0.8903	28.60/0.7751	27.52/0.7273	26.04/0.7896
LiNet [Ours]	$\times 2$	509K	106.0G	38.03/0.9610	33.63/0.9176	32.22/0.9099	32.19/0.9330
	$\times 3$		66.2G	34.40/0.9285	30.33/0.8419	29.13/0.8175	28.07/0.8534
	$\times 4$		35.0G	32.28/0.9034	28.62/0.7810	27.60/0.7373	26.15/0.7956

TABLE I: Comparison with light computational methods on scale factors $[\times 2, \times 3, \times 4]$. Best results are **highlighted** and second best results are underlined.

Model	Params.	Inference Time	PSNR
MemNet	667K	0.481	25.54
SRFBN_S	483K	0.006	25.71
D-DBPN	10426K	0.015	26.38
RDN	23000K	1.268	26.61
EDSR	43000K	1.218	26.64
Ours	509K	0.005	26.15

TABLE II: Inference time comparison with other SOTA models on Urban100 with scale factor $\times 4$.

B. Inference Time

In Table II, the proposed LiNet is compared with five other state of the art approaches in term of running time speed on

Urban100 with scale factor $\times 4$, namely MemNet [15], SRFBN [28], D-DBPN [29], RDN [17], and EDSR [16]. The inference time of each approach is evaluated using their official code on the same environment with a NVIDIA 1080Ti graphic card. The LiNet has the fastest inference time compared to the other networks, which reflect the efficiency of the proposed method.

C. Memory Complexity Analysis

In order to illustrate the efficiency of the proposed model, comparison between LiNet and SOTA approaches, with respect to number of parameters/operation vs reconstruction results, are presented in Fig 3. The proposed LiNet produces the best performance across all lightweight networks and

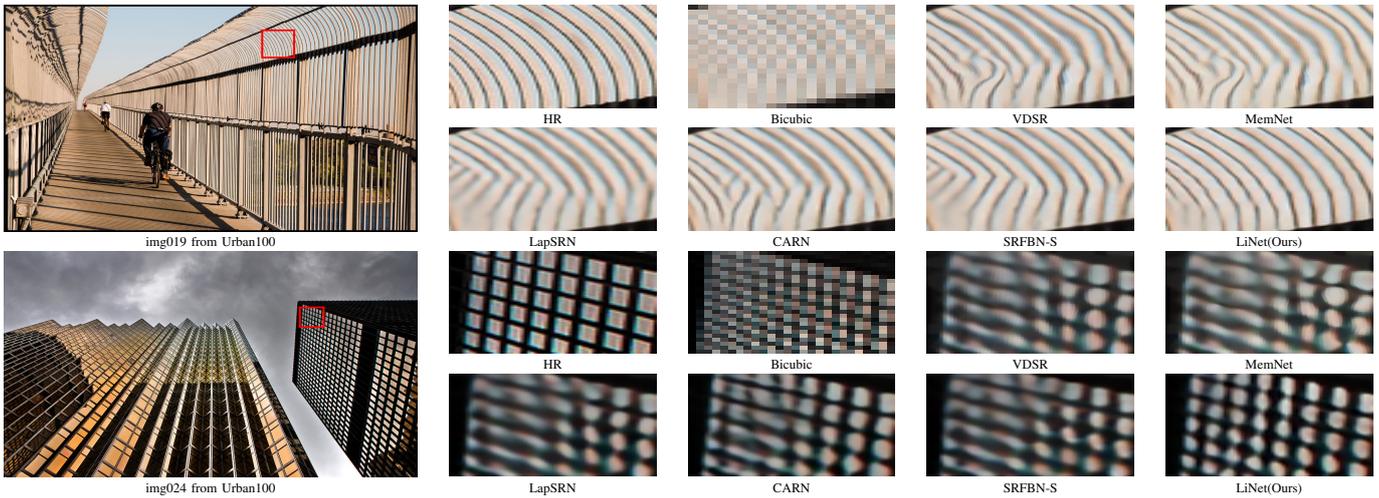


Fig. 2: Qualitative results on BI degradation dataset with scale factor $\times 4$.

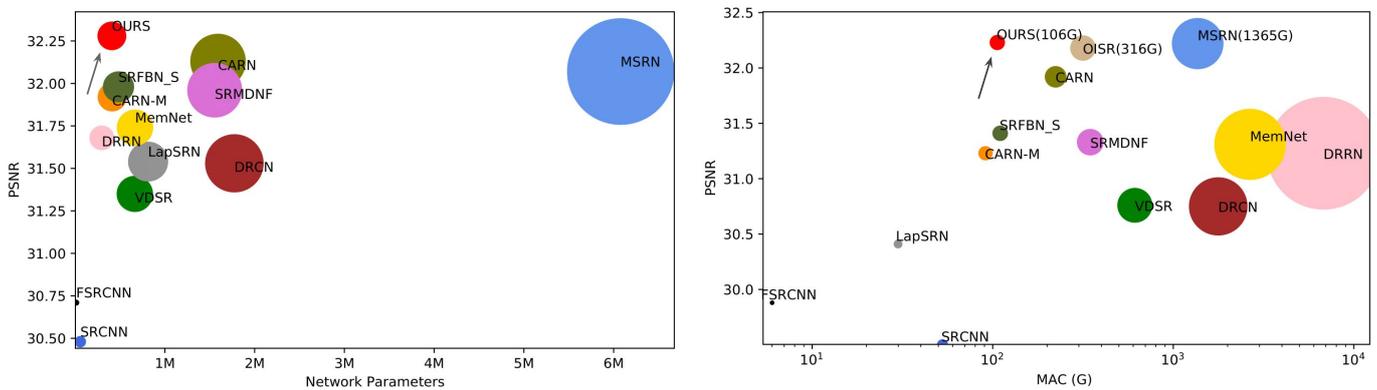


Fig. 3: **Left:** PSNR vs Network Parameters on Set5 with scale factor $\times 4$. **Right:** PSNR vs MAC on Urban100 with scale factor $\times 2$. **Red Circle** corresponds to the proposed model. Circle’s size is related with the number of parameters (left side) and number of operations (right side). Smaller circle size indicates less number of parameters/operations.

obtain efficient outcomes relative to expensive methods of computation. For instance, our LiNet (509K – 106G) with only needed of 8% of MSRN (5930K – 1365.4G) in terms of number of parameters and operations can produce better reconstruction results. By comparing these factors, it can verify that LiNet is well-balanced and more accurate to existing lightweight SOTA.

VI. CONCLUSION

This work focuses on introducing an efficient and practical lightweight network unlike most of the existed approaches, which try to design very deep and heavy network to achieve high performance. The proposed LiNet can overcome all the lightweight SOTA super resolution models. The key concept behind of this work was to build an advanced network for edge devices that could produce virtually the same results as heavy computational networks. To do so, a novel Compact Dense Module is proposed to focus and learn high-level feature maps. Also, the proposed module has the benefit of dense and residual skip connections to suppress the low-level information

and allows the LiNet to achieve rich feature-maps through the multiple learning pathways. As a result, LiNet achieves competitive results when compared to costly and very deep networks.

ACKNOWLEDGMENT

This work has been partially supported by the Spanish Government under Project TIN2017-89723-P; the “CERCA Programme / Generalitat de Catalunya”; and the ESPOL project CIDIS-56-2020.

REFERENCES

- [1] H. Greenspan, “Super-resolution in medical imaging,” *The Computer Journal*, vol. 52, no. 1, pp. 43–63, 2008.
- [2] W. W. Zou and P. C. Yuen, “Very low resolution face recognition problem,” *IEEE Transactions on image processing*, vol. 21, no. 1, pp. 327–340, 2011.
- [3] S. P. Mudunuri and S. Biswas, “Low resolution face recognition across variations in pose and illumination,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 5, pp. 1034–1040, 2015.

- [4] Z. Wang, S. Chang, Y. Yang, D. Liu, and T. S. Huang, "Studying very low resolution recognition using deep networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4792–4800.
- [5] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Generative image inpainting with contextual attention," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 5505–5514.
- [6] D. Liu, Z. Wang, Y. Fan, X. Liu, Z. Wang, S. Chang, and T. Huang, "Robust video super-resolution with learned temporal dynamics," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2507–2515.
- [7] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in *European conference on computer vision*. Springer, 2014, pp. 184–199.
- [8] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli *et al.*, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.
- [9] S. Anwar, S. Khan, and N. Barnes, "A deep journey into super-resolution: A survey," *arXiv preprint arXiv:1904.07523*, 2019.
- [10] C. Dong, C. C. Loy, and X. Tang, "Accelerating the super-resolution convolutional neural network," in *European conference on computer vision*. Springer, 2016, pp. 391–407.
- [11] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1874–1883.
- [12] J. Kim, J. Kwon Lee, and K. Mu Lee, "Accurate image super-resolution using very deep convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1646–1654.
- [13] Y. Tai, J. Yang, and X. Liu, "Image super-resolution via deep recursive residual network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 3147–3155.
- [14] J. Kim, J. Kwon Lee, and K. Mu Lee, "Deeply-recursive convolutional network for image super-resolution," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1637–1645.
- [15] Y. Tai, J. Yang, X. Liu, and C. Xu, "Memnet: A persistent memory network for image restoration," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 4539–4547.
- [16] B. Lim, S. Son, H. Kim, S. Nah, and K. Mu Lee, "Enhanced deep residual networks for single image super-resolution," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2017, pp. 136–144.
- [17] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, "Residual dense network for image super-resolution," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 2472–2481.
- [18] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "Squeezenet: Alexnet-level accuracy with 50x fewer parameters and 0.5 mb model size," *arXiv preprint arXiv:1602.07360*, 2016.
- [19] A. Howard, M. Sandler, G. Chu, L.-C. Chen, B. Chen, M. Tan, W. Wang, Y. Zhu, R. Pang, V. Vasudevan *et al.*, "Searching for mobilenetv3," *arXiv preprint arXiv:1905.02244*, 2019.
- [20] N. Ahn, B. Kang, and K.-A. Sohn, "Fast, accurate, and lightweight super-resolution with cascading residual network," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 252–268.
- [21] X. Chu, B. Zhang, R. Xu, and H. Ma, "Multi-objective reinforced evolution in mobile neural architecture search," *arXiv preprint arXiv:1901.01074*, 2019.
- [22] X. Chu, B. Zhang, H. Ma, R. Xu, J. Li, and Q. Li, "Fast, accurate and lightweight super-resolution with neural architecture search," *arXiv preprint arXiv:1901.07261*, 2019.
- [23] R. Timofte, E. Agustsson, L. Van Gool, M.-H. Yang, and L. Zhang, "Ntire 2017 challenge on single image super-resolution: Methods and results," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2017, pp. 114–125.
- [24] M. Bevilacqua, A. Roumy, C. Guillemot, and M. L. Alberi-Morel, "Low-complexity single-image super-resolution based on nonnegative neighbor embedding," 2012.
- [25] R. Zeyde, M. Elad, and M. Protter, "On single image scale-up using sparse-representations," in *International conference on curves and surfaces*. Springer, 2010, pp. 711–730.
- [26] D. Martin, C. Fowlkes, D. Tal, J. Malik *et al.*, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics." *Iccv Vancouver*, 2001.
- [27] J.-B. Huang, A. Singh, and N. Ahuja, "Single image super-resolution from transformed self-exemplars," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 5197–5206.
- [28] Z. Li, J. Yang, Z. Liu, X. Yang, G. Jeon, and W. Wu, "Feedback network for image super-resolution," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3867–3876.
- [29] M. Haris, G. Shakhnarovich, and N. Ukita, "Deep back-projection networks for super-resolution," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.