

SENext: Squeeze-and-ExcitationNext for Single Image Super-Resolution

Wazir Muhammada, Supavadee Aramvithb,[[1]](#footnote-1), and Takao Onoyec

aDepartment of Electrical Engineering, Faculty of Engineering, Chlalongkorn University, Bangkok, Thailand

b,\*Multimedia Data Analytics and Processing Research Unit, Department of Electrical Engineering, Faculty of

Engineering, Chulalongkorn University, Bangkok, Thailand

cGraduate School of Information Science and Technology, Osaka University, 1-5 Yamada-Oka, Suita, 565-0871 Japan.

ABSTRACT

Recent works on deep convolutional neural networks (CNNs) have demonstrated significant advancement in the field of single image super-resolution (SISR) to reconstruct the visually pleasing high-quality output image from a low-quality input image. However, most existing deep learning-based convolutional neural network approaches for SISR often use a deeper and wider network architecture, that requires a large number of network parameters, risk of overfitting, increases the computational complexity, and more memory consumption, as well as the high processing time for real-world applications. In this paper, we have proposed an effective Squeeze-and-ExcitationNext for Single Image Super-Resolution known as SENext. Our proposed network architecture used squeeze-and-excitation block (SEB) to adaptively recalibrate channel-wise feature mappings. Furthermore, short skip connections between each SEB are employed for enabling the feature reusability and stabilizing training convergence smoothly. Extensive quantitative and qualitative experiments on publicly available benchmark datasets, such as Set5, Set14, BSDS100, Urban100, and Manga109 demonstrate the superiority of the proposed method over state-of-the-art methods in terms of PSNR/SSIM, Number of parameters, processing speed, Number of FLOPs and visually pleasing effect.

© 2019 2 Published by Elsevier Ltd. Selection and/or peer-review under responsibility of Global Science and Technology Forum Pte Ltd

*Keywords:* Single image super-resolution; squeeze-and-excitation block, LReLU.

1. Introduction

Single image super-resolution (SISR) is a very hot research area in the field of image and computer vision tasks. The main task of SISR is to reconstruct the visually pleasing high-resolution (HR) image from the low-resolution (LR) input image. However, SISR is still a challenging task and is considered an ill-posed inverse problem due to the high-level information being lost during the image downsampling process. To solve this problem, earlier many algorithms [1-5] have been proposed, but performance is not satisfactory and has more computational complexity. Recently, deep convolutional neural networks (CNNs) captured the market for image super-resolution, and the research community shifted from the old hand-designed approach to a newly deep CNN-based approach. For the first time, Dong et al. proposed a shallow type Super-Resolution Convolutional Neural Network (SRCNN) [6] architecture to reconstruct a better HR image from the interpolated version of the LR input image [6]. Compared with the earlier conventional approaches, SRCNN can improve the performance through its shallow network architecture when reconstructing the HR image. The network architecture of SRCNN consists of basic three types of CNN layers which are called patch extraction, non-linear mapping, and reconstruction layers. Apart from the success of SRCNN in the image super-resolution, it has many shortcomings, including slow training speed, poor real-time reconstruction, bicubic interpolation as a pre-processing step, and large convolution kernels are used during the model design. In response to these problems, the same author has proposed the revised version of SRCNN and replaced the bicubic interpolation with a learnable upsampling (transpose convolution) layer to accomplish post-upsampling SR named Fast Super-Resolution CNN (FSRCNN) [7]. Furthermore, larger kernel sizes of SRCNN are replaced with small convolution kernels to optimize the efficiency of training and reconstruction. FSRCNN has lower computational complexity and better performance as compared to SRCNN but it has a limited network capacity. Inspired by the Visual Geometry Group network (VGG-net) [8] that was used for ImageNet classification, Kim et al. proposed very deep super-resolution (VDSR) [9], which pushed up the network and stacking multiple layers side-by-side up to 20 layers. The performance of the VDSR model significantly improved over previous models. This method suggested that deeper model architecture is the better architecture to increase the perceptual quality of the reconstructed HR image. Shi et al. [10] first time proposed an Efficient Sub-pixel Convolutional Neural Network (ESPCN) to reduce the computational cost and revised the upscaling process. In this approach, authors are replacing the pre-defined upsampling operator with a sub-pixel convolution layer and features are extracted from the low-dimensional space to reduce the training as well as the testing time of the model. Kim et al. proposed a Deeply Recursive Convolutional Network for image super-resolution (DRCN) [11] and uses the convolution layers multiple times. The key advantage of DRCN [11] is to fix the number of training parameters, although there are more recursions, and the main deficiency is to slow the training process. The authors similarly used the skip connection recursively to optimize model performance. Additionally, to obtain better reconstruction performance the SR models, using the concept of a deeper model and stacking the side of the layer by side, and many models with a depth exceeding 100 layers have appeared [12]. Increasing the depth of the model spatially can improve the performance of super-resolution quality, but it will bring a huge amount of computational cost and memory consumption. To reduce the computational cost and improve the processing speed of super-resolution networks, inspired by the SENet [13] and SESR [14], we are proposed a Squeeze-and-ExcitationNext for a single image super-resolution named SENext. In our SENext method, squeeze-and-excitation block (SEB) is used to develop the interdependencies between respective channels and reweighted the new features. Furthermore, state-of-the-art approaches such as SRCNN [6], VDR [9], DRCN [11], and FSRCNN [7] as shown in Figure 1, are used pre and post-processing bicubic/deconvolution layer as an upscaling factor to reconstruct the HR image. The main issue with these approaches having a more computational cost and reconstructing the HR image introduces blurry and ringing jagged artifacts. To resolve these issues we replaced both upscaling techniques with sub-pixel layers to reconstruct the visually pleasing HR image.

\\\\\\\\\\\

of smartphone and Internet, image has become an important information carrier in daily life. Image is a record and expression of the objective world, the resolution of which directly reflects the

clarity of the image and measures the amount of image information. Single image super-resolution (SISR) is an image processing technique and its aims to recover an accurate visually pleasing high-quality (also called high-resolution, HR) image from a degraded low-quality or low-resolution (LR) counterpart [[1](#_ENREF_1), [2](#_ENREF_2)]. **APPLICATIONS** Since the reconstructed visually pleasing high quality HR images preserve prominent texture details for image processing applications, because SISR is widely used in various field such as video surveillance[[3](#_ENREF_3)], medical images [[4](#_ENREF_4)], object recognition [[5](#_ENREF_5)], satellite imaging [[6](#_ENREF_6)] and etc. SISR problem usually assumes the observed LR image is a noninvertible low-pass filtered, downsampled and noisy version of HR image. Due to the loss of high-frequency information during the degradation of HR images, SISR is a highly ill-posed problem. **OLD METHODS** In the past two decades, many SR methods have been developed by computer vision community to handle the ill-posed problem, such as , including dictionary learning [[7](#_ENREF_7), [8](#_ENREF_8)], local linear regression [[9](#_ENREF_9), [10](#_ENREF_10)], random forest [[11](#_ENREF_11)] and sparse representation [[12](#_ENREF_12), [13](#_ENREF_13)]. **PROBLEMS WITH OLD METHODS** Although these SR methods made tremendous effort to promote the performance, but they still suffered from some limitations: previous research tended to boost the super-resolution performance via complex optimization methods. That may result in low execution speed. Another side effect is that these methods referred to manual setting parameters to obtain better results of SISR. Thereby, the more important thing is that a tool with power self-learning ability is critical to recover the HR image. To further increase the performance of image SR, deep learning-based approaches are best option for research community.

**SRCNN** Dong et al. [[14](#_ENREF_14)] first time proposed a shallow type network architecture with three CNN layers known as Super-Resolution Convolutional Neural Network (SRCNN). As a first CNN based approach for image SR, SRCNN has many shortcomings, including slow training speed, poor real-time reconstruction, bicubic interpolation as a pre-processing step, large convolution kernels are used during the model design, and inefficient feature inference. **FSRCNN** In response to these problems, same author has been proposed improved version of SRCNN and replace the bicubic interpolation with learnable upsampling (transpose convolution) layer to accomplish post-upsampling SR named as Fast Super-Resolution CNN [[15](#_ENREF_15)] . Furthermore, large kernels are replaced with small convolution kernels to optimize the efficiency of training and reconstruction. Several algorithms accelerate the SRCNN by extracting features directly from the input LR images (Fig. 1b) and replacing the pre-defined upsampling operator with sub-pixel convolution. **VDSR** Inspired by VGG model [[16](#_ENREF_16)] that used for ImageNet classification, Kim et al. proposed VDSR [[17](#_ENREF_17)], which pushed up the network and stacking multiple layers side-by-side up to 20 layers. The performance of VDSR model significantly improved over previous methods by a large margin. This indicates a deeper model is instructive to enhance the quality of generated images. **ESPCN** To avoid the computational complexities of feature extraction network and upscale process, Shi et al. [[18](#_ENREF_18)] first time proposed an Efficient Sub-pixel Convolutional Neural Network (ESPCN). In this approach authors are replacing the pre-defined upsampling operator with sub-pixel convolution. Features are extracted from the low-dimensional space to reduce the training as well as testing time of the model. Therefore, following the strategy of up-sampling layer, **SRResNet** Ledig et al. [[19](#_ENREF_19)] further proposed a SRResNet with a very deep ResNet [[20](#_ENREF_20)] architecture. **LapSRN** Lai et al. [[21](#_ENREF_21)] proposed the LapSRN, which use learned kernel as up-sampling unit to direct produced SR images.

In spite of great success achieved in the above architectures, the main issue that how to model mapping from LR to HR images better in a fast and flexible way remained unsolved. In this paper, we have proposed a Super-Resolution Fire Network (SrFireNet) architecture of SISR. The concept of Fire (Squeeze-and-Excitation) block [[22](#_ENREF_22)] is employed to better modeling interdependencies between channels. Short connections from input to each Fire block are used to remedy information lost. The proposed method is evaluated on some popular publicly available benchmarks. Extensive experiments show that our proposed model can achieves competitive accuracy in a more accurate and flexible way. It can greatly reduce model’s complexity by using less layers and allow designing more flexible applications.

**PROBLEMS WITH DEEP CNN MODELS** Dong et al. [[15](#_ENREF_15)] show that the convolutional neural networks (CNN) can be used to image SR and obtain an excellent performance. After that, CNN-based SR methods have drawn considerable attention due to the simple architecture and the impressive performance. However, CNN-based SR methods also exhibit limitations in architecture optimality. First, the network model of these methods is a fully convolutional neural network, which is limited to exploit the differentiated contextual information over global image region. Although the methods in literatures [16–18] have improved reconstruction quality by stacking the more convolution layers to exploit contextual information over a larger image region, they also increase the computation cost and memory usage. Thus, they exhibit limitations in terms of balancing the reconstruction accuracy and efficiency. Furthermore, these methods usually use convolution as the reconstruction layer to obtain the final HR image, which is limited to utilize the extracted feature information differentially to reconstruct the desired HR images because of the weight sharing of convolution in the height and width extent. Recently, the more network architectures [19– 29] and Datasets [30] have been used to solve SR reconstruction problems. Methods in literatures [19],[20] try to use transposed convolution or sub-pixel convolution to reconstruct the final HR images. Shocher et al. [29] propose "Zero-Shot" SR using deep internal earning, which does not rely on prior training. However, these limitations still exist in the use of the extracted features to reconstruct HR images. Second, most existing SR algorithms [15],[17],[19],[20] optimize the network models with L2 loss and thus inevitably generate blurred edges and textures in the reconstructed HR images. Several algorithms [21],[31],[33] have focused on improving the loss function to achieve the impressive measures and make the reconstructed HR images close to human visual perception on natural images. However, the blurring problem of sharp edges and texture structures still exists in reconstructed HR images.

To address the above-mentioned drawbacks, we propose a new image SR method based on the deep neural networks. Our method takes an LR image as input and trains a cascade of convolutional blocks inspired by deep Residual Networks used for ImageNet classification [36] to extract features in the LR space. Then, we use a fully connected layer which learns an array of upsampling weights to predict residual image (the differences between the upsampled image by bicubic interpolation and the ground truth HR image) from the extracted LR features. Finally, the desired HR image is obtained by adding the predicted residual image to the upsampled image using the bicubic interpolation. In addition, considering that L2 loss function used for most SR methods always leads to the blurring of texture details and edge structures, we introduce an edge difference constraint into the loss function of our proposed network to preserve edges and texture structures.

Overall, the contributions of this paper are mainly in three aspects:

1. We have introduced an effective super-resolution network with Fire block. It performs dynamic channel-wise feature recalibration to provide a new powerful architecture to improve the representational ability of information extraction part from low-resolution images.
2. We have set up a new state-of-the-art super-resolution method with fast running speed and accurate result in the measurement of PSNR and SSIM without increasing the complexity of the network, especially in case of large upscale rate.
3. By optimizing a fully connected upsampling layer to differentially exploit the contextual information over the global image region, our network can reduce the undesired visual artifacts effectively and obtain promising performance in computation time and memory usage.
4. Since all convolution layers can be shared by the networks of the different upscaling factors, our method could facilitate fast training and testing across the different upscaling factors.
5. We propose a new loss function with an edge difference constraint to optimize our proposed networks for making the reconstructed HR images with sharp edges and textures.
6. Related works
   1. Classical super-resolution methods

Numerous research has been taken on image super resolution problem. Early SR algorithms are based on interpolation, such as nearest, bicubic and Lanczos. Although these algorithms are widely used nowadays due to their fast speed, their outputs are usually too smooth and lack high-frequency information, which is not satisfying. Other approaches are based on reconstruction, which assume mapping between LR space and HR space, such as A+ [[9](#_ENREF_9)] based on anchored neighborhood regression, SelfExSR [[23](#_ENREF_23)] based on transformed self-exemplars.

* 1. Deep Learning Based Image Super-resolution

Since the success of AlexNet [[24](#_ENREF_24)] in ImageNet classification task [[25](#_ENREF_25)], deep convolutional neural networks are widely used in various computer vision tasks. Dong et al. proposed SRCNN [[14](#_ENREF_14)], which is the first deep learning model to solve super-resolution task. With a three-layer convolutional network, SRCNN outperformed most of traditional algorithms. Following work improves image super-solution model based on deep learning from two kinds of approaches. On one approach, some researchers propose

* 1. Efficient Deep CNN architectures

For some computer vision tasks, deeper model does not perform much better than the shallower one, which means explosive growth of the size of network brings limited improvement in the final performance. Thus rising attention has been paid to build small and efficient neural network. Iandola et al. proposed SqueezeNet [[26](#_ENREF_26)] with comparable performance and 50× fewer parameter compared with AlexNet. Mobilenet [[27](#_ENREF_27)] and MobileNetV2 [[28](#_ENREF_28)] build an efficient network with depthwise separable convolution, inverted residual and linear bottlenecks. Han et al. [[29](#_ENREF_29)] proposed deep compressing techniques to reduce the size of pretrained network, including pruning, vector quantization and Huffman coding.

* 1. SqueezeNet

SqueezeNet [[26](#_ENREF_26)] is the state-of-the-art CNN model which only uses 3 × 3 and 1 × 1 convolutional kernels. Using 1 × 1 filters reduces depth, hence, it reduces the computation of the 3 × 3 filters. It achieves the same accuracy as AlexNet does for ImageNet with 50× fewer parameters which make it suitable for the embedded systems. The distinct feature of SqueezeNet is a lack of fully connected layers. SqueezeNet uses an average pooling layer to calculate classification scores using small convolution kernels instead of using a fully connected layer which have immensely reduced computation and memory demand. This feature makes SqueezeNet best suited for the embedded platform with three key design strategies employed: (1) decrease the number of 3×3 filters, (2) decrease the number of input channels to 3 × 3 filters, and (3) downsample late in the network. This macro architecture is composed of fire modules that possess an incredibly small model size. Then, SqueezeNet v1.1 is introduced, where the number of filters as well as the filter sizes are further reduced, resulting in 2.4× less computation than the original SqueezeNet without sacrificing model accuracy. Inspired by the incredibly small macro architecture of SqueezeNet, insights are gained from this and some modifications are made in the proposed architecture. The SqueezeNet architecture is shown on the left side of Fig. 1.

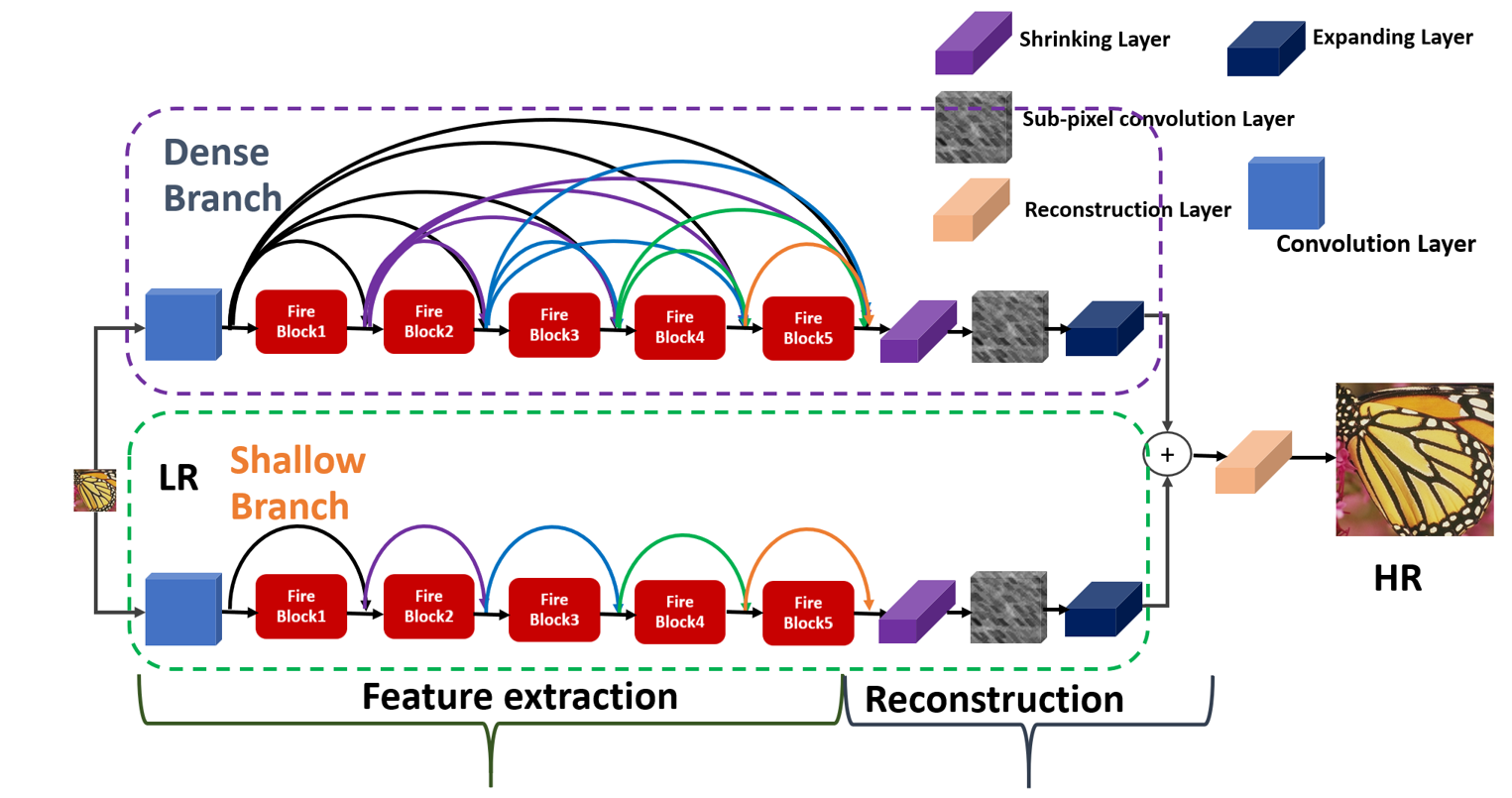
Diagram

Description automatically generated

Fig. 1: Illustration of SqueezeNet’s fire module (left) and SqueezeNext’s bottleneck module (right).

* 1. SqueezeNext Architecture

SqueezeNext [[30](#_ENREF_30)] uses SqueezeNet architecture as a baseline architecture. It consists of the following key strategies (1) A more aggressive channel reduction by incorporating a two-stage squeeze module, significantly reducing the total number of parameters used with the 3 × 3 convolutions. (2) Separable 3 × 3 convolutions to further reduce the model size, and remove the additional 1 × 1 branch after the squeeze module. (3) An element-wise addition skip connection similar to that of ResNet architecture. SqueezeNext baseline architecture comprises of bottleneck modules with four stage implementation, batch normalization layers, Relu and Relu(in-place) nonlinear activations, max, and average pool layers, Xavier uniform initialization, spatial resolution layer, and lastly, a fully connected layer is used with (1,2,8,1) four stage block configuration. The bottleneck module, shown in Fig. 1 (right hand side), is the backbone of the SqueezeNext architecture as it significantly reduces the number of parameters without reducing the model accuracy. The SqueezeNext baseline architecture achieves better model accuracy and size in comparison to SqueezeNet baseline architecture because of the use of bottleneck modules and the width multiplier.



The overall proposed network architecture of FireNet for single image super-resolution.

1. Proposed Network Architecture

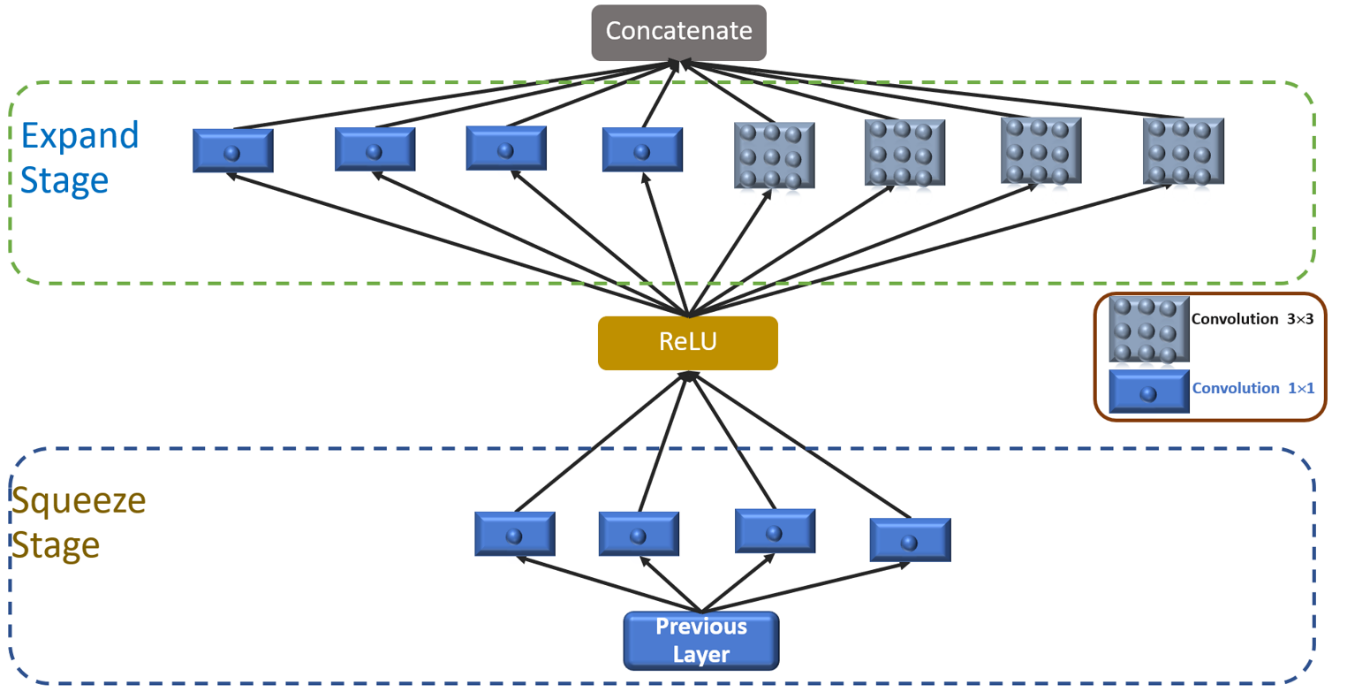
The proposed network architecture as shown in Figure. 1, consists of Dense branch and Shallow branch for feature extraction and reconstruction. The Dense branch used five Fire blocks with a dense skip connection. Similarly, a Shallow branch used same number of Fire blocks with local skip connection. Initial features are extracted from the original input LR image using one 3 × 3 CNN layer followed by PReLU activation function. The sequence of Fire blocks used in parallel branches to take the full benefit of the sparsity and channel-wise learning. In the reconstruction stage we used the bottleneck layer before and after in sub-pixel convolution layer to reduce the computational cost. Finally, the resultant high-resolution output image was the element-wise-sum of learned image through reconstruction layer. The detailed explanation of proposed architecture is as under:

* 1. Convolution Layer

asasasaa

* 1. Proposed Fire Block

**Original Fire Block**

Figure: Original Fire Block

* 1. Shrinking Layer

Aaa

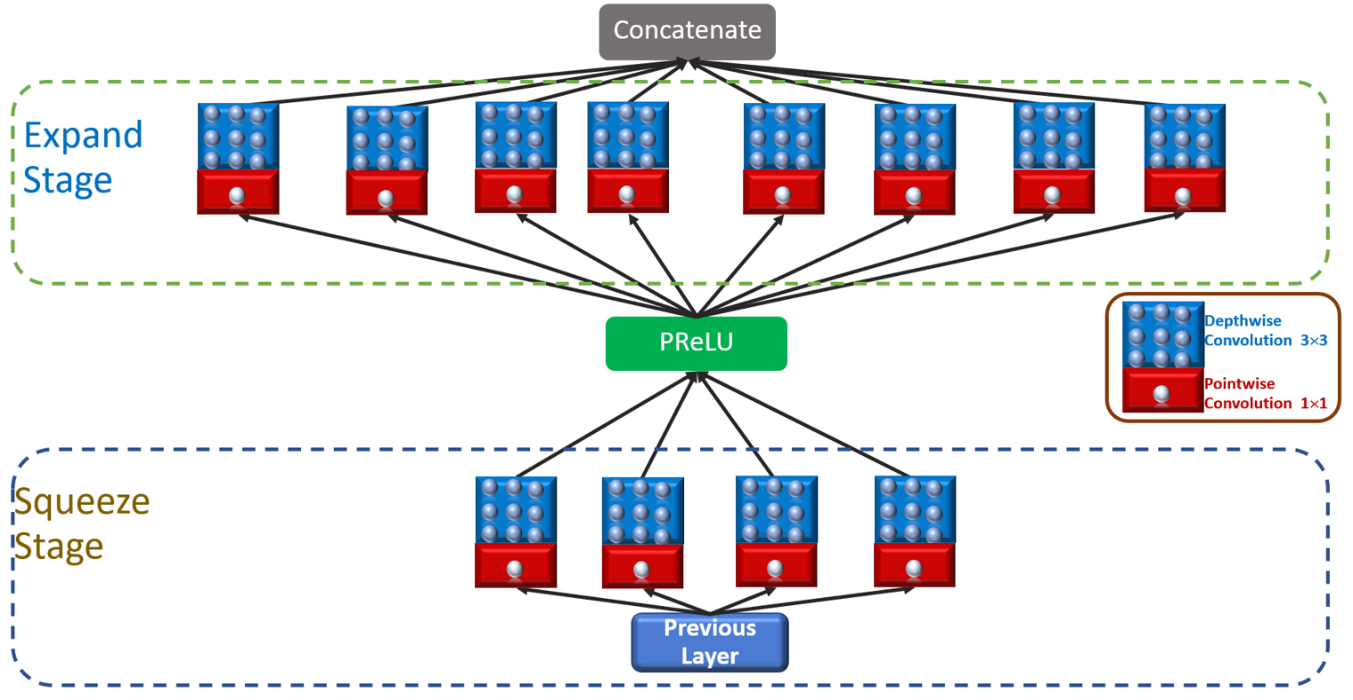
* 1. Sub-pixel Convolution Layer

Instead of using hand-designed interpolation methods we used sub-pixel convolution layer to upscale the original LR image into HR image. Extracted features from the different fire blocks followed by shrinking layer are added to sub-pixel convolution layer to upscale the original LR image to the target HR image. In addition two shrinking and expanding layer are inserted before and after the expensive sub-pixel-convolution layer to further reduce the number of model parameters.

* 1. Expanding Layer

Aaa

* 1. Reconstruction Layer.
  2. Dropout Layer



**Proposed Fire Block (DWCF Fire Module) Depthwise separable Convolutional Fire Module)**

Dropout is a technique used to improve over fit on neural networks. It is a regularization method that approximates training a large number of neural networks with different parallel architectures. Large neural nets trained on relatively small datasets can over fit the training data. This has the effect of the model learning the

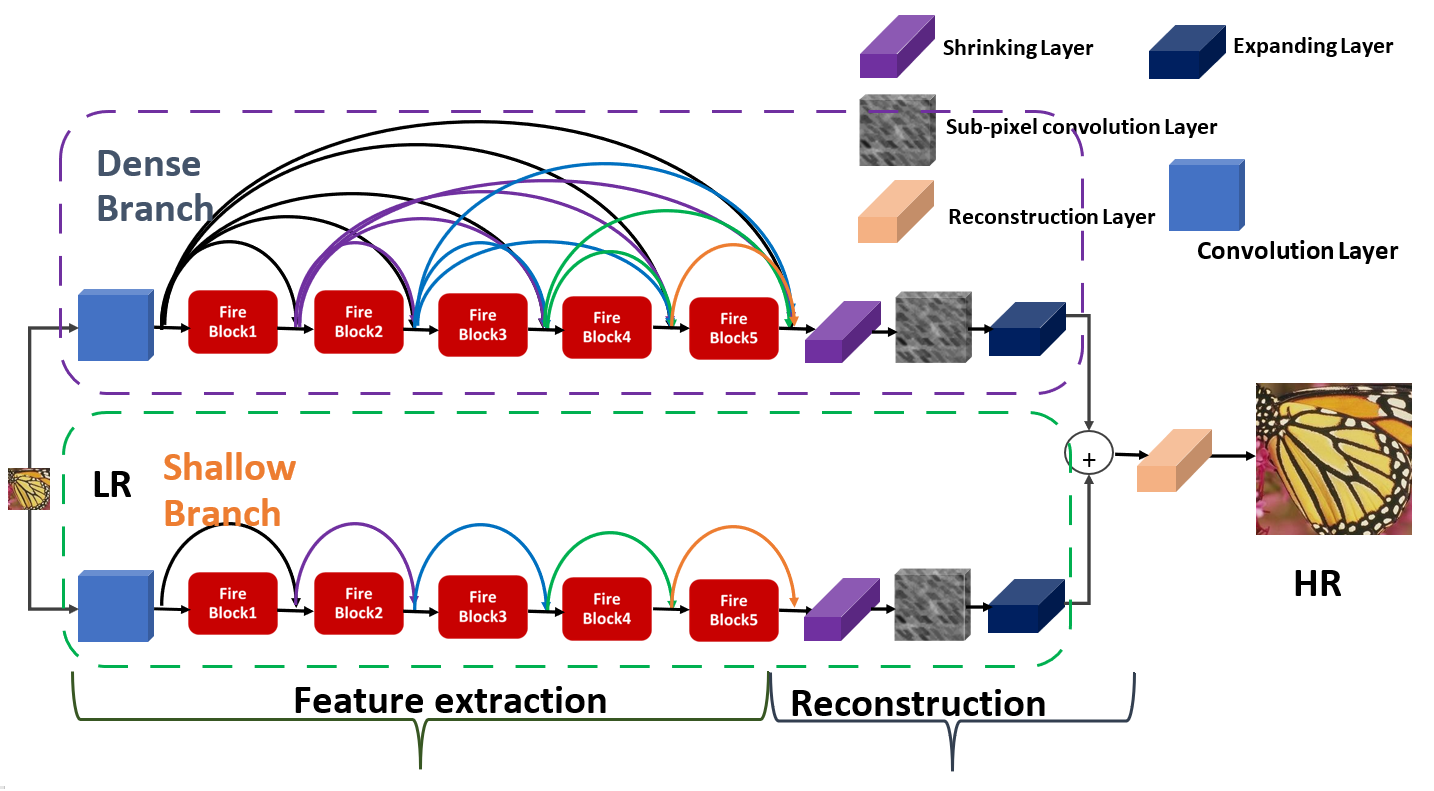
statistical noise in the training data, which results in poor performance and increase generalization errors due to over fitting. The approach to reduce over fitting is to fit all possible different neural networks on the same dataset

and to average the predictions from each model. In the proposed architecture, the dropout layer is used before the spatial resolution layer followed by the average pooling layer. It is observed during the experiments

conducted that the dropout layer performs better than an additional batch normalization layer.

sum of

Shallow SqueezeNext architecture is a CNN architecture. It is inspired from SqueezeNext [[30](#_ENREF_30)], SqueezeNet [[26](#_ENREF_26)] and Mobilenet [[27](#_ENREF_27)] architectures. It is based on the SqueezeNext architecture and a shallower architecture. It comprises of bottleneck modules [[26](#_ENREF_26)] which are further made up of basic blocks arranged in a four stage configuration followed by a spatial resolution layer, average pooling layer and a fully connected layer. The architecture implements SGD optimizer with momentum, decay and nestrov terms are used for the optimizer. It also makes use of a step decay with exponential based learning rate schedule with four LR update that first LR change after 60 epochs, second after 120 epochs, third after 150 epochs and fourth after 180 epochs. Further, the bottleneck module, shown in Fig. 2, comprises of a 1 × 1 convolution, second 1 × 1 convolution, 3 × 1 convolution, 1 × 3 convolution and then a 1 × 1 convolution. These convolutions are basic block (Fig 4) consists of a convolution layer, ELU in place, and batch normalization layer. These basic blocks form convolutions within bottleneck modules which further, are put together and arranged in the four stage implementation configuration along with a spatial layer, dropout layer, average pooling and a FC layer are shown in Fig. 3. The spatial layer (green block) can be removed in the proposed shallower versions or proposed architecture’s small sized models to reduce the parameter count with the CNN. The trained checkpoint file is saved using the model stat dictionary method of Pytorch avoiding the optimizer state dictionary or other parameters to again reduce the model size and improve the model speed.



Diagram

Description automatically generated

:

Fig. 2: Illustration of Shallow SqueezeNext’s bottleneck module

Diagram

Description automatically generated

Fig. 3: Illustration of Basic Block (left) and Shallow SqueezeNext architectures.

Diagram

Description automatically generated

Fig. 4: SqueezeNext baseline block and Proposed Shallow SqueezeNext Basic Block.

It is concluded with the descriptions of the two model shrinking hyper parameters such as the width multiplier and resolution multiplier in the following subsections. The right side of Fig. 4 illustrates the proposed architecture with (1,2,8,1) four stage configuration. Fig. 4 illustrates the Shallow SqueezeNext bottleneck module comprising of Shallow SqueezeNext basic blocks.

Diagram

Description automatically generated

* 1. Combining Dense and Shallow Branch

The proposed network architecture both branches finally combined to reconstruct the HR image. Although both dense and shallow branches are used the global as well as local skip connections to alleviate the vanishing gradient problems during the training. The dense branch takes more training time as compared shallow branch. In terms of comparison point of view shallow fire branch is able to reconstruct the low frequency content information easily but fails to capture high-frequency content information. To combine both Fire branches to facilitate the faster convergence and reconstruct the high-frequency details.

The shallow branch of consists of one CNN layer with 5 fire blocks adopted a local skip connection followed by PReLU activation function to extract the short as well as long features information simultaneously. In the reconstruction stage of shallow branch are added sub-pixel convolution layer with before and after a shrinking and expanding layer to reduce the computational burden on the model. In the dense branch authors are added same number of fires blocks but used global as well as local dense skip connection to reconstruct the low as well as high-frequency features information. The dense fire branch is not shared weights with shallow fire branch and both extract the features information parallelly taking as a same original low-resolution image. The final reconstructed perceptually high-quality HR image is calculated by

where denotes the original input LR image; and indicate the HR output of deep and shallow networks parametrized by and , respectively; is the final predicted HR image. reconstructed perceptually

and put a nomenclature if necessary, in a box with the same font size as the rest of the paper. The paragraphs continue from here and are only separated by headings, subheadings, images and formulae. The section headings are arranged by numbers, bold and 10 pt. Here follows further instructions for authors.

### Nomenclature

A radius of

B position of

### C further nomenclature continues down the page inside the text box

* 1. Structure

For this Procedia the files must be in MS Word format only and should be formatted for direct printing. Figures and tables should be embedded and not supplied separately. Please make sure that you use as much as possible normal fonts in your documents. Special fonts, such as fonts used in the Far East (Japanese, Chinese, Korean, etc.) may cause problems during processing. To avoid unnecessary errors you are strongly advised to use the ‘spellchecker’ function of MS Word. Follow this order when typing manuscripts: Title, Authors, Affiliations, Abstract, Keywords, Main text (including figures and tables), Acknowledgements, References, Appendix. Collate acknowledgements in a separate section at the end of the article and do not include them on the title page, as a footnote to the title or otherwise.

Bulleted lists may be included and should look like this:

* First point
* Second point
* And so on

Ensure that you return to the ‘Els-body-text’ style, the style that you will mainly be using for large blocks of text, when you have completed your bulleted list.

Please do not alter the formatting and style layouts which have been set up in this template document. As indicated in the template, papers should be prepared in single column format suitable for direct printing onto A4 paper (192mm x 262 mm). Do not number pages on the front, as page numbers will be added separately for the preprints and the Proceedings. Leave a line clear between paragraphs. All the required style templates are provided in this document with the appropriate name supplied, e.g. choose 1. Els1st-order-head for your first order heading text, els-abstract-text for the abstract text etc.

* 1. Tables

All tables should be numbered with Arabic numerals. Headings should be placed above tables, left justified. Leave one line space between the heading and the table. Only horizontal lines should be used within a table, to distinguish the column headings from the body of the table, and immediately above and below the table. Tables must be embedded into the text and not supplied separately. Below is an example which authors may find useful.

Table 1. An example of a table

|  |  |  |
| --- | --- | --- |
| An example of a column heading | Column A (*t*) | Column B (*T*) |
| And an entry | 1 | 2 |
| And another entry | 3 | 4 |
| And another entry | 5 | 6 |

* 1. Construction of references

References should be added at the end of the paper, and its corresponding citation will be added in the order of their appearance in the text. Authors should ensure that every reference in the text appears in the list of references and vice versa. Indicate references by Clark et al., 1962 or Deal and Grove, 2009 or Fachinger, 2006 in the text. The actual authors can be referred to, but the reference citation(s) must always be given.

Some examples of how your references should be listed are given at the end of this template in the ‘References’ section, which will allow you to assemble your reference list according to the correct format and font size.

* 1. Section headings

Section headings should be left justified, with the first letter capitalized and numbered consecutively, starting with the Introduction. Sub-section headings should be in capital and lower-case italic letters, numbered 1.1, 1.2, etc, and left justified, with second and subsequent lines indented. You may need to insert a page break to keep a heading with its text.

* 1. General guidelines for the preparation of your text

Avoid hyphenation at the end of a line. Symbols denoting vectors and matrices should be indicated in bold type. Scalar variable names should normally be expressed using italics. Weights and measures should be expressed in SI units. Please title your files in this order conferenceacrynom\_authorslastname.pdf

* 1. Footnotes

Footnotes should be avoided if possible. Necessary footnotes should be denoted in the text by consecutive superscript letters. The footnotes should be typed single spaced, and in smaller type size (8pt), at the foot of the page in which they are mentioned, and separated from the main text by a short line extending at the foot of the column. The ‘Els-footnote’ style is available in this template for the text of the footnote.

1. Author Artwork

All figures should be numbered with Arabic numerals (1,2,...n). All photographs, schemas, graphs and diagrams are to be referred to as figures. Line drawings should be good quality scans or true electronic output. Low-quality scans are not acceptable. Figures must be embedded into the text and not supplied separately. Lettering and symbols should be clearly defined either in the caption or in a legend provided as part of the figure. Figures should be placed at the top or bottom of a page wherever possible, as close as possible to the first reference to them in the paper.

The figure number and caption should be typed below the illustration in 8pt and left justified. For more guidelines and information to help you submit high quality artwork please visit: <http://www.elsevier.com/wps/find/authorsview.authors/authorartworkinstructions>. Artwork has no text along the side of it in the main body of the text. However, if two images fit next to each other, these may be placed next to each other to save space, see Fig 1. They must be numbered consecutively, all figures, and all tables respectively.

BS00559_

Fig. 1. (a) first picture; (b) second picture

Equations and formulae should be typed and numbered consecutively with Arabic numerals in parentheses on the right hand side of the page (if referred to explicitly in the text),

 (1)

They should also be separated from the surrounding text by one space.

Acknowledgements

These and the Reference headings are in bold but have no numbers. Text below continues as normal.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

CRediT authorship contribution statement

**Wazir Muhammad**: Conceptualization, Methodology, Software, Validation, Writing - original draft. **Supavadee Aramvith**: Methodology, Supervision, Writing - review & editing. **Takao Onoye**: Writing - review & editing.

Acknowledgments This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the “ICT Consilience Creative Program” (IITP2018-2017-0-01015) supervised by the IITP (Institute for Information & communications Technology Promotion). In addition, this work was also supported by the IITP grant funded by the Korea government (MSIT) (R7124-16-0004, Development of Intelligent Interaction Technology Based on Context Awareness and Human Intention Understanding)

References

1. Anwar, S., S. Khan, and N.J.A.C.S. Barnes, *A deep journey into super-resolution: A survey.* 2020. **53**(3): p. 1-34.

2. Wang, Z., J. Chen, and S.C.H. Hoi, *Deep Learning for Image Super-resolution: A Survey.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020: p. 1-1.

3. Zhang, L., et al., *A super-resolution reconstruction algorithm for surveillance images.* Signal Processing, 2010. **90**(3): p. 848-859.

4. Feng, C.-M., et al., *Brain MRI super-resolution using coupled-projection residual network.* Neurocomputing, 2021. **456**: p. 190-199.

5. Jiang, D., et al., *Semantic segmentation for multiscale target based on object recognition using the improved Faster-RCNN model.* Future Generation Computer Systems, 2021. **123**: p. 94-104.

6. Moustafa, M.S. and S.A. Sayed, *Satellite Imagery Super-Resolution Using Squeeze-and-Excitation-Based GAN.* International Journal of Aeronautical and Space Sciences, 2021.

7. Jianchao, Y., et al. *Image super-resolution as sparse representation of raw image patches*. in *2008 IEEE Conference on Computer Vision and Pattern Recognition*. 2008.

8. Yang, J., et al., *Image Super-Resolution Via Sparse Representation.* IEEE Transactions on Image Processing, 2010. **19**(11): p. 2861-2873.

9. Timofte, R., V. De Smet, and L. Van Gool. *A+: Adjusted Anchored Neighborhood Regression for Fast Super-Resolution*. in *Computer Vision -- ACCV 2014*. 2015. Cham: Springer International Publishing.

10. Yang, C. and M. Yang. *Fast Direct Super-Resolution by Simple Functions*. in *2013 IEEE International Conference on Computer Vision*. 2013.

11. Schulter, S., C. Leistner, and H. Bischof. *Fast and accurate image upscaling with super-resolution forests*. in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015.

12. Wright, J., et al., *Sparse Representation for Computer Vision and Pattern Recognition.* Proceedings of the IEEE, 2010. **98**(6): p. 1031-1044.

13. Alvarez-Ramos, V., V. Ponomaryov, and R. Reyes-Reyes, *Image super-resolution via two coupled dictionaries and sparse representation.* Multimedia Tools and Applications, 2018. **77**(11): p. 13487-13511.

14. Lin, G., et al. *Deep Convolutional Networks-Based Image Super-Resolution*. in *Intelligent Computing Theories and Application*. 2017. Cham: Springer International Publishing.

15. Dong, C., C.C. Loy, and X. Tang. *Accelerating the Super-Resolution Convolutional Neural Network*. in *Computer Vision – ECCV 2016*. 2016. Cham: Springer International Publishing.

16. Simonyan, K. and A. Zisserman, *Very Deep Convolutional Networks for Large-Scale Image Recognition*. 2015.

17. Kim, J., J.K. Lee, and K.M. Lee. *Accurate Image Super-Resolution Using Very Deep Convolutional Networks*. in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016.

18. Shi, W., et al. *Real-Time Single Image and Video Super-Resolution Using an Efficient Sub-Pixel Convolutional Neural Network*. in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016.

19. Ledig, C., et al. *Photo-realistic single image super-resolution using a generative adversarial network*. in *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017.

20. He, K., et al. *Deep residual learning for image recognition*. in *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.

21. Lai, W.-S., et al., *Deep Laplacian Pyramid Networks for Fast and Accurate Super-Resolution.* 2017.

22. Hu, J., L. Shen, and G. Sun. *Squeeze-and-Excitation Networks*. in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2018.

23. Huang, J., A. Singh, and N. Ahuja. *Single image super-resolution from transformed self-exemplars*. in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015.

24. Krizhevsky, A., I. Sutskever, and G.E. Hinton, *ImageNet classification with deep convolutional neural networks.* Communications of the ACM, 2012. **60**: p. 84 - 90.

25. Russakovsky, O., et al., *ImageNet Large Scale Visual Recognition Challenge.* International Journal of Computer Vision, 2015. **115**(3): p. 211-252.

26. Iandola, F.N., et al., *SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <1MB model size.* ArXiv, 2016. **abs/1602.07360**.

27. Howard, A., et al., *MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications.* 2017.

28. Sandler, M., et al. *MobileNetV2: Inverted Residuals and Linear Bottlenecks*. in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2018.

29. Han, S., H. Mao, and W. Dally, *Deep Compression: Compressing Deep Neural Network with Pruning, Trained Quantization and Huffman Coding.* arXiv: Computer Vision and Pattern Recognition, 2016.

30. Gholami, A., et al., *SqueezeNext: Hardware-Aware Neural Network Design.* 2018.

1. \* Corresponding author. Tel.: +6622186911.

   *E-mail address:* supavadee.a@chula.ac.th. [↑](#footnote-ref-1)