Localized Squeeze and Excitation Block

Muhammad Bilal Shaukat
Air University

Muhammad Imran Farid (drimranfarid.au@gmail.com)
Air University

Research Article

Keywords: Deep Learning, Smaller Neural Networks, Classification, Residual Network, Breast Cancer.

Posted Date: August 28th, 2023

DOI: https://doi.org/10.21203/rs.3.rs-3287743/v1

License: ☬ This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.
Localized Squeeze and Excitation Block

Muhammad Bilal Shaukat\(^1\) and Muhammad Imran Farid\(^{1*\dagger}\)

\(^1\)Electrical and Computer Engineering, Air University, E-9, Islamabad, 44000, Pakistan.

*Corresponding author(s). E-mail(s): drimranfarid.au@gmail.com; malik.bilal4426@gmail.com;
†These authors contributed equally to this work.

Abstract
In this article a new technique called Localized Squeeze and Excitation (LSE) Block is proposed. Using proposed LSE-block one can extract core features from the images in the early stage, which results in smaller deep neural network with much less number of parameters; results in less computation time. By using our proposed technique one can reduce the size of any large deep neural network model. For experimentation the model named ResNet-50 is selected and it is shown that how ResNet-50 can be converted into a smaller neural network by using our proposed technique with better accuracy and less processing time. This work is evaluated on the Breast Cancer Histopathological Image Dataset, both for binary and multi-class problems.

Keywords: Deep Learning, Smaller Neural Networks, Classification, Residual Network, Breast Cancer.

1 Introduction

Deep Neural networks (DNN) are very effective for modeling complex function but recent studies have emphasise that it is over-parameterised \cite{1}, \cite{2}. Smaller and shallow Neural Networks are the topics of the community of Artificial Intelligence from the last decade \cite{3}. One key approach the researchers used is the L1 Regularization on weight
matrices [4], but this approach does not perform well in industrial applications, where focus is more on the speed and power consumption. To overcome this, the recent work is focusing more on neurons rather than weights [5], [6], but these techniques do not target the floating point operations (FLOPs). Researchers are also reducing the size of the large DNN architecture by trial-and-error [7], [8]. These architectures require months to get a single architecture and are more expensive as the data becomes more complex.

The proposed technique LSE-block is an auxiliary block, which can be embedded into any large DNN for extracting core features at an early stage. After inserting LSE-block the rest of blocks in the pipeline of ResNet-50 are discarded; resulting in less parameters with better performance. The large DNN that we have used for our analysis is pre-trained ResNet-50 model. Why ResNet-50? because it gives the best result compared to the other pre-trained models when Breast Cancer Dataset named BreakHis is used [9].

Breast cancer is the most common type of cancer found in women, annually increasing by 12.5% all over the world [10]. Every year the rate of breast cancer patients is increasing by 0.5% [11] with the second highest mortality rate after lung cancer [11]. In US only, the breast cancer cases are increased by 12% during the year 2021 and are expected to increase by 30% for the year 2022 [11]. In Pakistan 1.38 million new cases of breast cancer are reported annually, which is highest among all the Asian countries. This higher mortality rate of incurable breast cancer is due to the late diagnosis [12]. Almost 99 percent of breast cancer are treatable, if diagnosed at early stage. So, early diagnosis of breast cancer is crucial for women.

Deep learning plays a crucial role in breast cancer detection by leveraging its capabilities in pattern recognition and analysis [13–15]. It also provides easy and reliable diagnostics of cancer by extracting features from Histopathological images [16], learning the model, and predicting whether the samples are malignant or benign [17, 18]. In
the past few years, a lot of work is done for the binary classification of cancer, which determines whether the sample is malignant or benign but less focused on exact type of cancer [19]. This Research paper proposed a LSE-block [20, 21] to achieve better accuracy at low computational cost. BreakHis dataset is evaluated for our model performance [22]. For binary classification LSE-block is compared with the state of the art ResNet-50 and SE-ResNet-50 architecture while for multi-class classification [19] pre-trained ResNet-50 model is employed for evaluation of LSE-block.

2 Related work

2.1 Deep Convolution Neural Networks (DCNNs)

DCNNs are the architectures of deep learning, have achieved a lot of advancements in computer vision-related tasks like image & video recognition, image classification, media recreation, and natural language processing. Since 1989 to date a lot of improvements have been made for more smarter and precise architectures [23]. With increasing number of convolutional layers in DCNNs, the training error decreases initially but then increases up to the worst condition [9]. This is due to the vanishing gradient problem (VGP). This problem has been addressed by adding residual into the networks [24–26], so that very deep networks can give better performance [9].

2.2 Deep Residual Learning

Deep Residual Learning, commonly known as ResNet, is a ground breaking deep learning architecture introduced by Kaiming He [27]. The main challenge in training very deep neural networks was the issue of vanishing gradient problem (VGP), where the gradients diminish as they are back propagated through the network, leading to ineffectively training of deep architectures [28, 29]. Residual network addressed this problem through the use of shortcut connections in which one or more layers can be skipped. In residual networks the shortcut connection is simple identity mapping,
which skipped the one or more convolved layer, added to the outputs of the convolved layer. Identity shortcut connections add neither extra parameter nor computational complexity. The ResNet-50 is a residual based deep network with 16 residual building blocks, which is a deeper networks without VGP, this results in improved performance and higher accuracy in challenging image recognition tasks.

2.3 Squeeze and Excitation Block (SE-block)

In DCNNs, kernel filters collect spatial correlation between features and skip channel-wise correlation. Recent research shows that by exploiting channel-wise information, one can emphasize informative features and suppress less useful features [23]. SE-block is a computational unit, introduced for channel relationship [30], can be constructed for any given transformation function $F_{tr}: X \rightarrow U$, where $F_{tr}$ is a convolution operation. $X \in \mathbb{R}^{H' \times W' \times C'}$ is the input image of dimensions $H' \times W' \times C'$ and $U \in \mathbb{R}^{H \times W \times C}$ is the output feature map of dimensions $H \times W \times C$. In squeeze operation, the SE-block squeezes global spatial information of all channels into an array of channel descriptor by averaging feature maps across spatial dimensions [31–33]. Excitation operation capture channel wise inter dependencies, re-calibrate array of channel descriptors and excite the respective channels accordingly. Fig. 1 describes complete picture of SE-block. SE-block is an auxiliary computational block which can be used in any network for better performance of deeper network without adding extra parameter and computational complexity in the network.

![Fig. 1: SE-block.](image-url)
2.4 Concurrent Spatial and Channel Squeeze & Excitation in Fully Convolutional Networks

In SE-block, feature map was squeezed across the feature map and excited along the channels. This work presented an another block in which a feature map is squeezed along the channels and excited across the feature map, which is named as spatial excitation SE-block (sSE), while SE-block is named as channel excitation SE-block (cSE). The combination sSE and cSE formed another block named as spatial and channel squeeze & excitation Block (scSE) in which input image $U$ simultaneously undergoes from both blocks and the outputs from both blocks are summed up to give final feature map [34]. This work is integrated and evaluated in fast convolutional neural networks.

2.5 Competitive Inner-Imaging Squeeze and Excitation for Residual Network(CMPE-SE)

For CMPE-SE, SE-block is inserted into the residual networks. In this technique, instead of identity mapping as residual, the residual squeezing is used as an additional useful information for basic squeeze operation [27]. The CMPE-SE squeezes both residual and input feature maps, concatenates the output squeezed signals, excites the concatenated signal and re-calibrate the channels of input feature map accordingly [34]. This work presented four different versions with their comparisons.

3 Localized Squeeze and Excitation Block

The proposed method is inspired from the basic SE-block. The drawback observed in the basic SE-block and the other two versions is that the channel information as well as spatial information are not exploited simultaneously and resulting a loss of useful information. In depth, when we do the step of squeezing in SE-block we averaged the pixels of the image of each channel i.e. reducing all pixels information into a single
scalar. Which is not good, as this scalar does not reflect the pixels of the image. Its just a garbage value. Latter in the excitation step, these scalars are used for generating weights that will scaled every image of the channel according to the weights. This is shown in Fig. 1. So, the core idea is to localized the average value of the image of each channel with a fix window or kernel size. Which latter be fed into excitation step and then follow up to scale the localized pixel of the image of a channel. This core idea is describe in depth in Fig. 2.

This drawback is addressed in our proposed block which is named by localized squeeze and excitation block (LSE-block). For a transformation function \( F_{tr} : X \rightarrow U \), \( F_{tr} \) is a convolution operation, \( X \in \mathbb{R}^{H' \times W' \times C'} \) is the input image of dimensions \( H' \times W' \times C' \) and \( U \in \mathbb{R}^{H \times W \times C} \) is the output feature map of dimensions \( H \times W \times C \). In the set of convolutional kernel filters \( V = [v_1, v_2, v_3, ..., v_c], v_i, i = 1, ..., C \), refers to the parameters of the \( i \)-th filter. After convolving \( X \) by using \( F_{tr} \), the \( c \)-th output is:

\[
    u_c = v_c * X = \sum_{s=1}^{C'} (v_c^s * x^s) 
\]

where \( U = [u_1, u_2, u_3, ..., u_c] \)

and \( X = [x_{c1}, x_{c2}, x_{c3}, ..., x_{cC}] \)

and \( v_c = [v_{c1}, v_{c2}, v_{c3}, ..., v_{cC}] \)

After transformation \( F_{tr} \), our proposed LSE-block is applied, in which the SE-block is applied locally on the feature map [21], shown in Fig. 2. We split the input feature...
map into small patches $u \in R^{w \times w \times C}$ of dimension $w \times w \times C$, which are obtained by sliding kernel window over the feature map, where $w$ is the size of the kernel window for patch extraction. The stride rate for the kernel window is equal to the dimension of required patch (i.e. $w$) [20, 35, 36].

### 3.1 Squeeze: Local Information Embedding

For input $U$, we selected a patch $u \in R^{w \times w \times C}$, which can be seen by grey shaded region in Fig. 2. Here the squeeze operation is applied [30], which is mathematically expressed in Eq. 2, by applying average pooling with pooling size equal to the window size $w$.

The aggregated squeezed arrays against all the patches will be matrix $z$ with reduced dimensions $(W/w \times H/w \times C)$. So the $ij$-th element of $z$ for a patch $u \in R^{w \times w \times C}$ is calculated by:

$$z_c = F_{sq}(u_c) = \frac{1}{(w \times w)} \sum_{i=1}^{w} \sum_{j=1}^{w} u_c(i, j)$$  \hspace{1cm} (2)

The resultant squeezed array against the patch can be seen by shaded cube in the matrix $z$ which is now an array of channel descriptors for respective patch.

### 3.2 Excitation: Local Adaptive Re-calibration

Excitation operation captures channel wise inter-dependencies for a given array of channel descriptors against a patch $u \in R^{w \times w \times C}$. After re-configuration of channel descriptors, parent patch is scaled by its re-calibrated channel descriptor array.

Excitation operation is mathematically expressed as [30]:

$$s = F_{ex}(z, W) = \sigma(W_2(\delta(W_1z)))$$  \hspace{1cm} (3)

where $W_1 \in R^{(c \times c)}$ and $W_2 \in R^{(c \times c)}$

where $W_1$ and $W_2$ are weights of intermediate fully connected layers, $\delta$ refers to the ReLU[37] activation function after dense layers with weights $W_1$ and $\sigma$ refers to the
sigmoid activation function after dense layers with weights $W_2$. After excitation, patch $u$ will be scaled by the matrix of excited channel descriptors $s$ as follows:

$$\tilde{x}_c = F_{\text{scale}}(u_c, s_c) = s_c \cdot u_c$$  \hspace{1cm} (4)$$

where $\tilde{X} = [\tilde{x}_1, \tilde{x}_2, ..., \tilde{x}_c]$ and $F_{\text{scale}}(u_c, s_c)$ refers to channel-wise multiplication between the patch $u_c \in \mathbb{R}^{w \times w}$ and the scalar $s_c$. For all the arrays in matrix $z$ along the channels, results the $S$ excited matrix where each patch $u \in \mathbb{R}^{w \times w \times C}$ of the input feature-map $U$ is scaled by respective array of excited channel descriptors $s$, which is mathematically expressed in Eq. 4. $\tilde{X}$ is our final output for LSE-block in which channel information as well as spatial information are exploited simultaneously.

### 3.3 Exemplars: LSE-ResNet

As discussed earlier, for feature maps exploitation, blocks can be used anywhere in the intermediate layers of any DCNNs [24, 30]. In this work, we are using our proposed LSE-block in the residual block. We are taking residual, before applying LSE-block, adding back to the excited feature-map, shown in Fig. 3. Here, the dimension of residual is equal to the dimension of excited feature-map. So our final network would be localized squeeze and excitation residual network (LSE-ResNet), shown in Fig. 3.
4 Experiments and Results

4.1 Data Distribution and Augmentation

For experimentation, a very challenging dataset named BreakHis is used, which is composed of 9,109 breast cancer histopathology images comprised of 2,480 benign and 5,429 malignant images which are further sub-divided into total eight unique classes [38]. Fig. 4 (a) shows a sample of a benign cancer image named Adenosis and Fig. 4 (b) shows a malignant cancer image named Ductal Carcinoma. BreakHis dataset has highly unbalanced classes and the number of samples are also very low, which can be seen from Table 1. This makes BreakHis a challenging dataset to work on. In this work class balancing is only done for binary classification by up-sampling the minority classes while data augmentation is applied to both binary and multi-class classifications. Data augmentation includes random horizontal flipping, rotation, brightness, contrast, crop and blur. The data set is split into a fixed number of validation and test data, which is 40 from each class, and the rest is the training data. Data augmentation is applied to both training and validating data.

Fig. 4: (a) Benign class Adenosis with 40x magnification (b) Malignant class Ductal Carcinoma with 40x magnification.
Table 1: BreakHis dataset statistics.

<table>
<thead>
<tr>
<th>Types of Tumor</th>
<th>40x</th>
<th>100x</th>
<th>200x</th>
<th>400x</th>
<th>All</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adenosis(^1)</td>
<td>144</td>
<td>113</td>
<td>111</td>
<td>106</td>
<td>444</td>
</tr>
<tr>
<td>Fibroadenoma</td>
<td>253</td>
<td>260</td>
<td>264</td>
<td>207</td>
<td>1014</td>
</tr>
<tr>
<td>Phyllodes tumor</td>
<td>109</td>
<td>121</td>
<td>108</td>
<td>115</td>
<td>453</td>
</tr>
<tr>
<td>Tubular adenoma</td>
<td>149</td>
<td>150</td>
<td>140</td>
<td>130</td>
<td>569</td>
</tr>
<tr>
<td>Ductal carcinoma(^2)</td>
<td>864</td>
<td>903</td>
<td>896</td>
<td>788</td>
<td>3451</td>
</tr>
<tr>
<td>Lobular carcinoma</td>
<td>156</td>
<td>170</td>
<td>163</td>
<td>137</td>
<td>626</td>
</tr>
<tr>
<td>Mucinous carcinoma</td>
<td>205</td>
<td>222</td>
<td>196</td>
<td>169</td>
<td>792</td>
</tr>
<tr>
<td>Papillary carcinomas</td>
<td>145</td>
<td>142</td>
<td>135</td>
<td>138</td>
<td>560</td>
</tr>
</tbody>
</table>

\(^1\)Class with lowest number of images.
\(^2\)Class with highest number of images.

4.2 Experimentation

For binary classification we implemented a LSE-ResNet which is composed of a single residual network having identical internal structure to the first building block of ResNet-50, as shown in Fig. 5(a). For multi-class classification, due to highly unbalanced classes and a very few number of samples per class in the dataset as shown in Table 1, it is impossible to properly trained the model from scratch and evaluate its performance. To encounter this issue one can use transfer learning for model evaluation as done in previous works. Due to the lack of resources, instead of using transfer learning, we evaluated our proposed LSE-ReNet by assisting pre-trained ResNet-50 which was pre-trained on ImageNet dataset. We inserted SE-block as well as our proposed LSE-ResNet in ResNet-50 by skipping the layers of original network at three different depths as shown in Fig. 5(b) and compared the results with original ResNet-50.

4.3 Experimentation Environment

The proposed model is implemented in google colab with tensor-flow environment. For binary classification Google colab pro, GPU: NVIDIA A100-SXM4-40GB, RAM 84GB is used for getting results. For multi-classification using pre-trained ResNet-50 model, google colab pro, GPU: NVIDIA A100-SXM4-40GB, RAM 84GB is used.
Fig. 5: (a) Network Architecture of Localized SE-block in a Residual Network. (b) Network Architecture of Localized SE-block, assisting Pre-trained ResNet-50 at three different depths with fine-tuning.
4.4 Results

The experimental work is divided into two parts i.e. Binary classification and Multi-class classification.

4.4.1 Binary Classification

The implemented models are ResNet-50, SE-ResNet-50 and LSE-ResNet, trained using stochastic gradient descent optimizer with learning rate of 0.01, 0.02 and 0.1 (with decay=0.01) respectively with momentum 0.9. Models are trained for 20 Epochs with the batch size of 64. Results for binary classification in terms of accuracy and training time are shown in Table 2 and Table 3 respectively while Table 4 shows the number of parameters. The proposed technique of LSE-ResNet performs the best in terms of classification accuracy. Due to very less number of parameters as compare to the ResNet-50 and SE-ResNet-50, the processing time of the proposed technique LSE-ResNet is very low. For binary classification, the computational cost of the LSE-ResNet is 99.37% less than ResNet-50 and 99.42% less than SE-ResNet-50. In terms of the training time, LSE-ResNet took 35.64% less time than ResNet-50 and 38.14% less than SE-ResNet-50.

<table>
<thead>
<tr>
<th>S. No</th>
<th>Magnification</th>
<th>ResNet-50</th>
<th>SE-ResNet-50</th>
<th>LSE-ResNet</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>40X</td>
<td>85.50</td>
<td>83.00</td>
<td>87.50</td>
</tr>
<tr>
<td>2</td>
<td>100X</td>
<td>83.33</td>
<td>81.66</td>
<td>84.58</td>
</tr>
<tr>
<td>3</td>
<td>200X</td>
<td>86.25</td>
<td>84.16</td>
<td>89.16</td>
</tr>
<tr>
<td>4</td>
<td>400X</td>
<td>84.16</td>
<td>81.66</td>
<td>85.83</td>
</tr>
<tr>
<td>5</td>
<td>MI(^1)</td>
<td>85.00</td>
<td>82.91</td>
<td>90.00</td>
</tr>
</tbody>
</table>

\(^1\)MI stands for Magnification Independent.
Table 3: Binary classification training time in seconds (s).

<table>
<thead>
<tr>
<th>S. No</th>
<th>Magnification</th>
<th>ResNet-50</th>
<th>SE-ResNet-50</th>
<th>LSE-ResNet</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>40X</td>
<td>1241</td>
<td>1331</td>
<td>837</td>
</tr>
<tr>
<td>2</td>
<td>100X</td>
<td>1190</td>
<td>1341</td>
<td>852</td>
</tr>
<tr>
<td>3</td>
<td>200X</td>
<td>1282</td>
<td>1322</td>
<td>804</td>
</tr>
<tr>
<td>4</td>
<td>400X</td>
<td>1230</td>
<td>1369</td>
<td>771</td>
</tr>
<tr>
<td>5</td>
<td>MI(^1)</td>
<td>4957</td>
<td>5157</td>
<td>3190</td>
</tr>
</tbody>
</table>

\(^1\)MI stands for Magnification Independent.

Table 4: Number of parameters of models.

<table>
<thead>
<tr>
<th>S. No</th>
<th>Model</th>
<th>No. of Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ResNet-50</td>
<td>25,585,594</td>
</tr>
<tr>
<td>2</td>
<td>SE-ResNet-50</td>
<td>28,116,586</td>
</tr>
<tr>
<td>3</td>
<td>LSE-ResNet-50</td>
<td>160,530</td>
</tr>
</tbody>
</table>

4.4.2 Multi-class Classification

For multi-class classification the models that are implemented are pretrained ResNet-50; ResNet-50 with SE-block after 3-blocks, 7-blocks and 13-blocks; ResNet-50 with LSE-block after 3-blocks, 7-blocks and 13-blocks. Note that when for example the model ResNet-50 with LSE-block after 3-blocks is implemented the rest of the blocks after 3-blocks of ResNet-50 are discarded. This is true for rest of the models as well. The Models are fine tuned using stochastic gradient descent optimizer with learning rate of 0.1, decay of 0.01 and momentum of 0.9. Models are trained for 30 Epochs with the batch size of 64. Results for Multi-class classification accuracy and time are shown in Table 5 and in Table 6 respectively.

Table 5, shows the results for multi-class classification using SE-block with pretrained model at different depths. When SE-block is used after 3-blocks the accuracy decreased by 17.81\% compared to the simple ResNet-50; which gradually increased to 70.93\% and 83.43\% when SE-block is added after 7-blocks and 13-blocks respectively.

Table 6, shows the results for multi-class classification using the LSE-block with pretrained model at different depths. When LSE-block is used after 3-blocks the
accuracy decreased by 15% than ResNet-50; which increases to 90.31% when LSE-block is added after 7-blocks but decreased to 87.18% when LSE-block is added after 13-blocks. The reason for the decrease in accuracy is small spatial dimension of the feature map (i.e. $14 \times 14$). Overall, our proposed LSE-block performed better than simple SE-block with low computational cost and better accuracy.

Table 5: Eight-class classification using SE-block with pre-trained model

<table>
<thead>
<tr>
<th>S. No</th>
<th>Pre-Trained Model</th>
<th>Accuracy (%)</th>
<th>Training Time (s)</th>
<th>No. of Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ResNet-50</td>
<td>80.93</td>
<td>450</td>
<td>2,106,376</td>
</tr>
<tr>
<td>2</td>
<td>ResNet-50 with 3-blocks</td>
<td>63.12</td>
<td>515</td>
<td>492,184</td>
</tr>
<tr>
<td>3</td>
<td>ResNet-50 with 7-blocks</td>
<td>70.93</td>
<td>275</td>
<td>1,409,320</td>
</tr>
<tr>
<td>4</td>
<td>ResNet-50 with 13-blocks</td>
<td>83.43</td>
<td>257</td>
<td>4,546,120</td>
</tr>
</tbody>
</table>

Table 6: Eight-class classification using LSE-ResNet with pre-trained model

<table>
<thead>
<tr>
<th>S. No</th>
<th>Pre-Trained Model</th>
<th>Accuracy (%)</th>
<th>Training Time (s)</th>
<th>No. of Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ResNet-50</td>
<td>80.93</td>
<td>450</td>
<td>2,106,376</td>
</tr>
<tr>
<td>2</td>
<td>ResNet-50 with 3-blocks</td>
<td>65.93</td>
<td>378</td>
<td>492,184</td>
</tr>
<tr>
<td>3</td>
<td>ResNet-50 with 7-blocks</td>
<td>90.31</td>
<td>678</td>
<td>1,409,320</td>
</tr>
<tr>
<td>4</td>
<td>ResNet-50 with 13-blocks</td>
<td>84.37</td>
<td>383</td>
<td>4,546,120</td>
</tr>
</tbody>
</table>

5 Conclusion

The community of Artificial Intelligence is moving towards the Smaller Neural Networks (SNNs) for fast and reliable output, as large DNNs takes a lot of time to train. If SNNs can do better job than DNNs, then what’s the purpose of using the time consuming DNNs. In this study we have proposed a new technique called LSE-block that can be embedded into any DNN in order to change it into a SNN. For experimentation we have converted a DNN model named ResNet-50 into a SNN and proved that by embedding our proposed technique of LSE-block into ResNet-50 we can extract the core features of the data at the early stage, which results in better
classification accuracy as compared to using the ResNet-50 alone or when ResNet-50 is used with SE-block. The highly unbalanced and challenging dataset named BreakHis, a histopathology images of Breast Cancer having eight classes, is used for analysis. In both binary and multi-classification, our proposed technique outperforms the DNN technique of ResNet-50, not only in accuracy but also in time.

References


