Expert Systems With Applications 184 (2021) 115357

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Gaussian process regression-based learning rate optimization in convolutional neural networks for medical images classification

Yuanyuan Li, Qianqian Zhang, Sang Won Yoon *

Department of Systems Science and Industrial Engineering, State University of New York at Binghamton, Binghamton, NY 13902, United States

ARTICLE INFO

Keywords:
Learning rate optimization
Convolutional neural networks
Medical images classification
Gaussian process regression
Stimulated Raman Scattering images

ABSTRACT

This research proposes a series of novel learning rate optimization algorithms with two versions for Adaptive Moment Estimation (Adam), which is a common optimizer in Convolutional Neural Networks (CNNs). Optimizer that is used to control the training efficiency and prediction accuracy by controlling the convergence progress plays an important role in CNNs. However, optimizers such as Adam are usually not hyperparameter-free and very sensitive to the hyperparameters embedded in CNNs. For example, the learning rate is a hyperparameter that represents a step size in the calculations. The learning rate has the most significant influence on prediction accuracy, so optimizing the learning rate is the best way to improve accuracy. In this research, a series of Gaussian Process Regression (GPR)-based learning rate optimization (*GLRO*) algorithms are proposed to increase the classification accuracy. To be specific, the relationship between the learning rate and corresponding accuracy is studied and the potential learning rate is predicted by the GPR model which is built with previous learning rates and corresponding accuracies. Also, two strategies of the algorithm to select the input learning rate are tested separately. AlexNet, which is a state-of-the-art CNN, is used as a framework to evaluate the proposed algorithms. AlexNet is widely used in the healthcare system as medical imaging classification framework. The Stimulated Raman Scattering (SRS) images of human brain tumors are used to classify cells and non-cells in this research. The proposed *GLRO* are compared to the conventional learning rate annealing algorithm and the constant learning rate algorithm. The algorithms' classifications of SRS images are evaluated in terms of accuracy, sensitivity, specificity, and precision. To further validate *GLRO*, multiple benchmark medical images and CNN frameworks are tested. The experimental results illustrate that the proposed *GLRO* algorithms outperform other algorithms by showing a 96% classification accuracy on SRS images and achieve promising classification results on the other datasets and CNN frameworks.

1. Introduction

Machine learning has been studied by many researchers and applied in many different domains in recent years; for example, in speech recognition, natural language processing, and object detection (Abdel-Hamid, Mohamed, Jiang, & Penn, 2012; Yin, Kann, Yu, & Schütze, 2017). Classification is one of the most popular machine learning applications in all domains. Many approaches are used to solve the classification problem, such as the Nearest Neighbor, Random Forest, etc. The most common way to classify a big dataset is to use Convolutional Neural Networks (CNNs) (Lee & Kwon, 2017). Whether a CNN has a good classification result is mainly the result of prediction accuracy. Optimizer plays an important role in CNNs to control convergence for a better classification accuracy (Yang & Yang, 2018). Currently, Stochastic Gradient Descent (SGD) and Adaptive Moment Estimation (Adam) are two optimizers that are widely used by the researchers. The main rule of SGD is to update the parameter based on the direction with the negative gradient of the loss function (Reddi, Kale, & Kumar, 2019). Adam applies more information such as momentum into the updating process and is more widely used than SGD because of its better efficiency. Normally, there are some hyperparameters in each optimizer to control the training process. For instance, Adam has four hyperparameters. Among the four hyperparameters, the learning rate donated as α has the most significant influence on the classification results (Kingma & Ba, 2014). Much research has been done to tune the learning rate, manual and automatic tunings are two popular methods to obtain the best learning rate. Manual tuning needs more professional knowledge and takes much time, such that automatic tuning became the main

* Corresponding author.
E-mail addresses: yli352@binghamton.edu (Y. Li), qzhang40@binghamton.edu (Q. Zhang), yoons@binghamton.edu (S.W. Yoon).

https://doi.org/10.1016/j.eswa.2021.115357
Received 2 August 2020; Received in revised form 1 April 2021; Accepted 4 June 2021
Available online 30 June 2021

trend because of promising results and high-efficiency (Wu et al., 2019). Grid Search (GS) is widely used to search the best learning rate by exhaustively searching all possible candidate values (Bergstra, Bardenet, Bengio, & Kégl, 2011). Thus, GS usually guarantees the solution but the computation time is expensive. To overcome such drawback, Random Search (RS) is proposed to provide a chance to identify the best learning rate with a smaller calculation burden, but it could be unreliable for some complex models (Bergstra & Bengio, 2012). In pursuit of efficiency and accuracy, many advanced optimization methods are used to select the learning rate (Wu et al., 2019; Park, Yi, & Ji, 2020; Dubey et al., 2019). For instance, the Bayesian Optimization (BO) is applied to optimize the learning rate based on the model performance and proved to be an efficient method (Wu et al., 2019). Some other numerical optimization models are developed based on the cost function and successfully optimized the learning rate by showing high accuracies (Park et al., 2020; Dubey et al., 2019). Because of the importance of the learning rate, an algorithm with a novel optimization rule is proposed in this research to improve the classification performance. Optimizers cannot be tested in isolation, most are tested on CNNs, where they usually operate. Several state-of-the-art models are widely used in CNNs. Specifically, AlexNet was first developed in the 2012 ImageNet competition and won first place because of several innovative points, such as the development of a new activation function, a dropout layer, etc (Krizhevsky, Sutskever, & Hinton, 2012). Hence, the novel learning rate optimization rule and generic optimizers are tested mainly on the AlexNet in this research.

Image classification is one of the most popular domains and has many applications. Different medical images have been studied by many researchers who obtained superior prediction results (Kumar, Kim, Lyndon, Fulham, & Feng, 2016; Zhang, Wang, Lu, Won, & Yoon, 2018). In this research, images of human brain tumors taken by Stimulated Raman Scattering (SRS) spectroscopy comprise the dataset. In general, SRS produces label-free images that can map lipids and proteins to new and potentially cancerous cells without destroying function. This can help enhance the diagnosis and surgery processes for doctors and radiologists (Lu et al., 2015). There are two classes of images, but they are difficult to distinguish because the color features of each class are not clear so that the preprocessing step improves the classification.

To achieve better classification results on SRS images, the learning rate is optimized. Other than the traditional optimization models, a tree-based adaptive learning rate optimization model is developed and obtaines a higher accuracy than the constant learning rate (Takase, Oyama, & Kurihara, 2018). A gradient descent method is also applied to adaptively update the learning rate value for every epoch and provides a new direction for the learning rate study (Baydin, Cornish, Rubio, Schmidt, & Wood, 2017). A Reinforcement learning model is proposed to study the relationship between the learning rate and primary model performance, such that the best learning rate, which leads to the highest accuracy, will be identified during each step running (Xu, Qin, Wang, & Liu, 2017). Moreover, Gaussian Process Regression (GPR) is combined with Evolutionary strategy to optimize the learning rate, which demonstrated as a powerful method to optimize the learning rate and improve the classification accuracy (Zhang, Li, Lyu, Ling, & Su, 2019). Inspired by these methods, a series of GPR-based learning rate optimization (*GLRO*) algorithms are proposed in this research. To be specific, GPR is applied to explore the relationship between the learning rate and classification accuracy, and a probabilistic model is developed based on the known data and the relationship between the variables. In this way, the best learning rate is determined by the Gaussian surrogate model. The prediction results are compared with constant learning rate and simple learning rate-controlling algorithms in terms of accuracy, sensitivity, specificity, and precision.

The structure of the paper is organized as: the literature review and some background introduction are shown in Section 2. Section 3 introduces Adam and the proposed algorithm. In Section 4, experiments and results will be discussed. Finally, the conclusion, limitations, and future work are listed in Section 5.

2. Literature review

Optimizers that are important for CNNs are not hyperparameter-free. The learning rate that is necessary and plays the most significant role is studied. For example, in a conventional SGD, the learning rate is a constant value that is decided at the beginning of the learning process. However, it has limited training speed, low convergence rate, and a local minimum in some sparse data (Dong, Xu, Xu, & Zhuang, 2019). To overcome such disadvantages, the adaptive learning rate strategy is applied. For instance, a novel algorithm named ESGD uses the adaptive learning rate. In the algorithm, the learning rate is obtained by exploiting the Hessian metrics in the equilibration preconditioner which is an extension of the Jacobi preconditioner. ESGD achieves better overall performance by improving convergence speed (Dauphin, De Vries, & Bengio, 2015).

The adaptive learning rate strategy uses momentum as it is applied in the optimizers. AdaDelta is a well-known adaptive optimizer that was first introduced in 2012. In its process, the learning rate decays based on the first-order derivative and the number of the epoch. The basic concept is to make the learning speed fast at the beginning, then slow down when near the local minimum to avoid loss oscillations. There are several benefits such as better performance on loss and less computational resources (Zeiler, 2012). Also, some researchers discussed another algorithm named cyclical learning rates. In their algorithm, the learning rate was not a fixed number or needed to be tuned manually. It was based on the fact that a large learning rate will lead to a negative influence for a short time but could bring a positive influence after running for a long term. There was one maximum bound and one minimum bound; the learning rate could vary between the bounds. In this way, the CNN framework could save the learning rate information and processing time. The test results for standard datasets show a great benefit from the algorithm (Smith, 2017). A new gradient-based optimizer is designed to improve the Adam with the adaptive learning rate. A diffGrad friction coefficient (DFC) is defined based on the short-term gradient performance and can be represented by a nonlinear sigmoid function. DFC is introduced in the proposed diffGrad optimizer to control the learning rate and its value is adaptively updates based on the current gradient. When the gradient changes turn to slow, DFC imposes more friction and vice versa. It is shown that the diffGrad outperforms other state-of-the-art methods on the benchmark dataset (Dubey et al., 2019). A more direct method that designs a hypergradient descent update rule for learning rate is studied. The historical gradient will be stored in the memory and combined with the current gradient to calculate the new learning rate. By applying this chain rule, the learning rate adaptively updates over each epoch, the high-efficiency and time-saving are achieved (Baydin et al., 2017).

Others use global strategies, such as GS and RS, that exploit the entire dataset to adjust the learning rate. However, GS and RS suffer from some deficiencies such as computaional time consumption and inaccurate predictions (Bergstra et al., 2011; Bergstra & Bengio, 2012; LeCun, Bottou, Orr, & Müller, 2012). In recent years, the optimization method was adopted to overcome such drawbacks. The learning rate tuning is generally a black-box optimization problem (Wu et al., 2019). Different from the gradient-based methods, the optimization model can address more information by exploring the relationship between the learning rate and objective function. With a clear objective function, the learning rate tuning efficiency will be improved. The model accuracy can be used as an objective function. A specific neural network called Actor Network is designed to predict the best learning rate. The network is developed based on the performance of both the primary model and the current model. It is compared to other widely-used optimizers with the popular datasets. The high convergence efficiency is proved according to the training and test losses (Xu et al., 2017). To solve the learning rate optimization problem, a BO model is developed to update the posterior

of the optimization function based on the given prior function and data information. BO usually guarantees the optimal through only a small number of samples, such that it is reasonable to apply for the learning rate optimization problem. Multiple commonly-used acquisition functions are tested and compared in their research to explore the maximum value of the model accuracy, and the Expected improvement is selected because of the simplicity and good performance. Several popular machine learning algorithms are tested and obtained promising prediction accuracy (Wu et al., 2019). Training loss is another reliable objective function, and a tree-based Adaptable Learning Rate Tree algorithm (ALR) is proposed. In each epoch, the parent node will generate several branches with their coefficient, and one of the nodes with the smallest loss will be selected as a new parent to create new branches. During the learning process, the number of beam sizes needs to be defined to stop the algorithm and pick the best learning rate. This flexible and simple method figures out the good solutions, which proves its strong optimization ability (Takase et al., 2018).

Recently, one study proposes an algorithm named Hyper-parameter Optimization with sUrrogated-aSsisted Evolutionary Strategy (HOUSES), which uses GPR to update the learning rate. HOUSES studied the relationship between the error function and hyperparameters based on an evolutionary strategy and BO to build the probabilistic model. The Gaussian surrogate model based on a non-stationary kernel was built in the algorithm. HOUSES helps improve the convergence and achieves better classification accuracy (Zhang et al., 2019). However, the significance of accuracy is not included in the research. Thus, HOUSES gives insight into the proposed algorithm in this research to study the relationship between the accuracy and learning rate.

GPR is a popular method to study the relationship between the variables and perform the sensitivity analysis especially when the test data is non-linear, and the training data is limited (Zhang, Wang, Chen, & Zhang, 2019). Because of the great efficiency and good performance of GPR, more and more researchers apply it in their work. For example, GPR was applied to address the relationship between the validation data and error to help adjust the parameters in the research work on the wireless indoor localization with CNNs. Their results illustrate that GPR successfully improves the localization accuracy (Zhang et al., 2019). Also, GPR model was introduced in the dynamical nonlinear experiments. The model learned the response surface information and could help to decide the next data point by building the model based on the parameter space and historical data points. The model can guarantee the maximum of potential data points and successfully address the geometry of the curvature (Renson, Sieber, Barton, Shaw, & Neild, 2019).

3. Methodology

3.1. Adaptive moment estimation

Adam is a stochastic optimization method that has been popularly used in recent years. The objective function for Adam is $f(\theta)$ and the objective is to obtain the minimum expected value of $f(\theta)$. Some parameters and components update during the process, which changes the value of $f(\theta)$. The first important component is g_t, which is the first derivative of the $f(\theta)$. Another two considerable pieces are the biased estimations for the first and second order derivatives. They can be regarded as the approximations of the expected values of g_t and g_t^2. Next, m_t and v_t are introduced to represent the unbiased estimation. Hyperparameters β_1 and β_2 are used to control the decay rates of m_t and v_t, respectively. After that, the θ value is updated based on the value of the previous θ, m_t, v_t, learning rate α, and another hyperparameter ϵ (Kingma & Ba, 2014).

3.2. Conventional Learning Rate Annealing

Conventional learning rate annealing (CLRA) algorithm will be discussed and compared with constant learning rate and the proposed algorithms in this research. CLRA is built on the foundation of Adam, which was presented in the previous section. The only difference from the original algorithm is the rule to update the learning rate. A large learning rate at the beginning stage speeds the learning process while a small learning rate avoids oscillations in the loss function. CLRA automatically decreases the learning rate without considering information, such as the gradient value, current accuracy, or current loss function. The learning rate is annealed based on the number of epochs. Because of the simplicity of this algorithm, fewer parameters and small amounts of memory are needed, such that the calculation speed is fast. To be specific, in this algorithm, the learning rate will be decreased by γ times when the process runs for every δ epochs. The parameters settings are discussed in Section 4. Decreasing the learning rate over the epochs is proposed due to the time-based learning rate schedule which was studied in (Park et al., 2020). The detailed flows are shown in Algorithm 1.

Algorithm 1: Adam with CLRA

Require: α_t: Step size
Require: $\beta_1, \beta_2 \in [0, 1)$: Exponential decay rates for the momentum estimates
Require: δ, γ: Hyper-parameters to control the decay rates for learning rate
Require: $f(\theta)$: Stochastic objective function with parameters θ
Require: θ_0: Initial parameter vector; $t = 0$ Initialize timestep
$m_0 = 0$ Initialize 1^{st} moment vector; $v_0 = 0$ Initialize 2^{nd} moment vector
1: **while** θ_t not converged **do**
2: $t = t + 1$
3: $g_t = \nabla_\theta f_t(\theta_{t-1})$
4: $m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t$, $v_t = \beta_2 v_{t-1} + (1-\beta_2) g_t^2$
5: $\widehat{m}_t = m_t / (1-\beta_1^t)$, $\widehat{v}_t = v_t / (1-\beta_2^t)$
6: $\theta_t = \theta_{t-1} - \alpha_t \cdot \widehat{m}_t / (\sqrt{\widehat{v}_t} + \epsilon)$
7: **if** t *mod* $\delta = 0$ **then**
8: $\alpha_t = \alpha_{t-1} / \gamma$
9: **end if**
10: **end while**

3.3. Gaussian process regression-based learning rate optimization

A novel learning rate optimization algorithm is proposed in this research. It is based on the annealing method and GPR. GPR is a non-parametric model and has been demonstrated to be a strong method to study the relationship among different datasets. It is used to do the sensitivity analysis for input data even if the dataset is small and non-linear. In this section, GPR is briefly introduced.

Start with GP which is a stochastic process defined as a collection of random variables and any finite number has consistent joint Gaussian distributions (Quiñonero-Candela & Rasmussen, 2005). It is specified by the mean function and covariance function. The equations are listed below

$$\begin{cases} \mu(l) = E[f(l)] \\ k(l, l') = E[(f(l) - \mu(l))(f(l') - \mu(l'))]. \end{cases} \tag{1}$$

And the process function $f(l)$ can be written as

$$f(l) \sim GP(\mu(l), k(l, l')). \tag{2}$$

Now, it is assumed that the learning rate annealing model is defined by

$$y = f(l) + \epsilon, \tag{3}$$

where $f(l) \sim GP(\mu(l), k(l, l'))$ and $\epsilon \sim (0, \sigma^2)$. Then, the prior distribution is described as

$$p(f(l)|l) = N(m(l), k(l, l)), \tag{4}$$

where, $m(l)$ refers to the mean function of the learning rate annealing model and $k(l, l)$ represents the covariance matrix to describe the relationship between each pair of training data. Both training and testing data follow the Gaussian distribution, and training and test data follow a

joint Gaussian distribution. Then, according to prior distribution, the posterior joint distribution between the prediction value and training value can be written as

$$\begin{bmatrix} y \\ f^* \end{bmatrix} \sim N\left(\begin{bmatrix} \mu_T(L) \\ \mu_T(L^*) \end{bmatrix}, \begin{bmatrix} K(L,L)+\sigma_n^2 I & K(L^*,L)^T \\ K(L^*,L) & K(L^*,L^*) \end{bmatrix} \right), \tag{5}$$

where, f^* represents the prediction value, L is the training data, L^* is the testing data, σ_n is the noise term, and K means the calculation of the covariance. Usually, in the calculation space, the mean function is adjusted to zero, because it is believed that the random number is set at location O. Thus, the predictive distribution can be generated as

$$p(f^*|L,y,L^*) \sim N(\widehat{\mu}_T, \widehat{\Sigma}). \tag{6}$$

$\widehat{\mu}_T, \widehat{\Sigma}$ can be defined as

$$\begin{cases} \widehat{\mu}_T = K(L^*,L)^T(K(L,L)+\sigma_n^2 I)^{-1} y \\ \widehat{\Sigma} = K(L^*,L^*) - K(L^*,L)^T(K(L,L)+\sigma_n^2 I)^{-1} K(L^*,L). \end{cases} \tag{7}$$

And the term $\widehat{\mu}_T$ is the value of prediction. More details about the GPR can be found in the literature of Rasmussen and Williams (Rasmussen & Williams, 2006). The proposed Adam with $GLRO_1$ algorithm is described in Algorithm 2.

Algorithm 2: Adam with $GLRO_1$

Require: α_t: Step size
Require: $\beta_1, \beta_2 \in [0,1)$: Exponential decay rates for the moment estimates
Require: λ: Number of epoch before applying Gaussian surrogate model
Require: $lr_1, lr_2 \in (0,1)$: The starting step size and the last step size before applying Gaussian surrogate model
Require: $f(\theta)$: Stochastic objective function with parameters θ
Require: θ_0: Initial parameter vector;
$m_0 = 0$ Initialize 1^{st} moment vector
$v_0 = 0$ Initialize 2^{nd} moment vector; $t = 0$ Initialize timestep
while θ_t not converged **do**
  $t = t + 1$
  $g_t = \nabla_\theta f_t(\theta_{t-1})$
  $m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t$, $v_t = \beta_2 v_{t-1} + (1-\beta_2) g_t^2$
  $\widehat{m}_t = m_t/(1-\beta_1^t)$, $\widehat{v}_t = v_t/(1-\beta_2^t)$
  $\theta_t = \theta_{t-1} - \alpha_t \cdot \widehat{m}_t/(\sqrt{\widehat{v}_t} + \epsilon)$
  if $t \leqslant \lambda$ **then**
    $\alpha_t = \alpha_{t-1} - \dfrac{lr_1 - lr_2}{\lambda - 1}$
  else
    Use all the α_t to fit or update the Gaussian surrogate model f^* according Eq. 6
    Calculate $(L_i, f^*)_{i=1}^n$ for n new observations which obtain by Gaussian surrogate model and acquisition function in Eq. 6 and Eq. 7
    Set $L_t^* = argmax\ f_{i=1}^{*n}$
    Update $\alpha_t = L_t^*$
  end if
end while

However, when applying *GLRO* algorithm, the value of the learning rate near the optimal point of the Gaussian surrogate model is very small especially compared to the first learning rate input to the model. To eliminate the potential influence of the large gap between input datasets, the large learning rate introduced earlier will be removed. Based on the knowledge mentioned before, the specific flow for the proposed algorithm can be seen in Algorithm 3.

Algorithm 3: Adam with $GLRO_2$

Require: α_t: Step size
Require: $\beta_1, \beta_2 \in [0,1)$: Exponential decay rates for the moment estimates
Require: ω: Number of input learning rates in the Gaussian surrogate model

(*continued on next column*)

(*continued*)

Algorithm 3: Adam with $GLRO_2$

Require: $lr_1, lr_2 \in (0,1)$: The starting step size and the last step size before applying Gaussian surrogate model
Require: $f(\theta)$: Stochastic objective function with parameters θ
Require: θ_0: Initial parameter vector; $t = 0$ Initialize timestep
$m_0 = 0$ Initialize 1^{st} moment vector; $v_0 = 0$ Initialize 2^{nd} moment vector
while θ_t not converged
  $t = t + 1$
  $g_t = \nabla_\theta f_t(\theta_{t-1})$
  $m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t$, $v_t = \beta_2 v_{t-1} + (1-\beta_2) g_t^2$
  $\widehat{m}_t = m_t/(1-\beta_1^t)$, $\widehat{v}_t = v_t/(1-\beta_2^t)$
  $\theta_t = \theta_{t-1} - \alpha_t \cdot \widehat{m}_t/(\sqrt{\widehat{v}_t} + \epsilon)$
  if $t < \omega$ **then**
    $\alpha_t = \alpha_{t-1} - \dfrac{lr_1 - lr_2}{\omega - 1}$
  else
    if $t = \omega$ **then**
      $\alpha_t = \alpha_{t-1} - \dfrac{lr_1 - lr_2}{\omega - 1}$
      Use all the α_t to fit or update the Gaussian surrogate model f^* according Eq. 2
    else
      Remove $\alpha_{t-(t-1)}, \alpha_{t-(t-2)} \ldots \alpha_{t-10}$, use the left α_t to fit or update the Gaussian surrogate model f^* according Eq. 2
    end if
    Use all the α_t to fit or update the Gaussian surrogate model f^* according Eq. 2
    Calculate $(L_i, f^*)_{i=1}^n$ for n new observations which obtain by Gaussian surrogate model and acquisition function in Eq. 6 and Eq. 7
    Set $L_t^* = argmax\ f_{i=1}^{*n}$
    Update $\alpha_t = L_t^*$
  end if
end while

4. Experimental results and analysis

4.1. Data description

As mentioned before, SRS is a label-free image with significant merit that could help doctors and radiologists implement real-time detection. It is efficient in finding human brain tumors and mouse skin cells in ways that could keep the cells functioning in fresh specimens (Lu et al., 2015; Lu et al., 2016). SRS images are used as input data to AlexNet to test the proposed algorithm on Adam. A raw image example is shown in Fig. 1.

As shown by the colors in Fig. 1, there are two classes of cells in an overall SRS image. The preprocessing of the raw image is necessary to generate the input data and the details will be shown in this subsection.

The first step is to detect the cell in the raw image. As shown in Fig. 1, the image consists of many small pixels, and each pixel has three

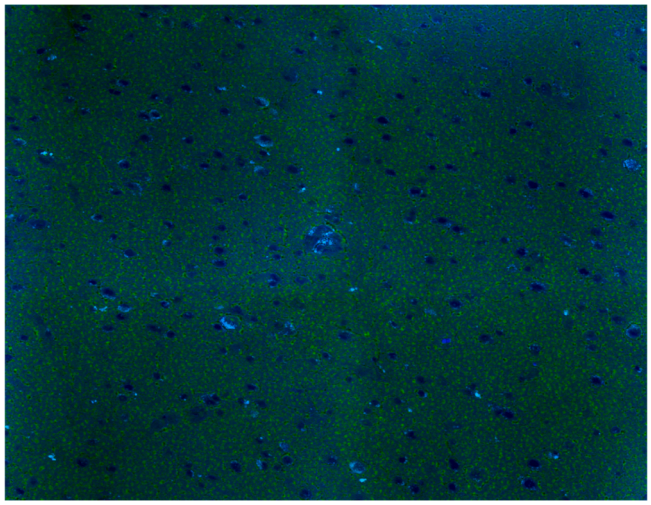


Fig. 1. SRS image example.

channels, which are known as RGB. RGB represents red, green, and blue. The range for each channel is (0, 255) so that for each pixel, there could be 16,777,216 possible colors. A study of the raw image concludes there is a high probability that a pixel represents a part of a cell if the value of the third channel is less than 39. So, all pixels whose third channel is less than 39 is marked to white. The manual marking process is added due to the poor quality of the automatic marking results. The full image after manual marking is presented in Fig. 2. Then, based on the edge of marked white points, the cells are extracted. Fig. 3(a) shows some cell examples, and each cell is an individual image for classification.

Non-cell can be extracted by selecting a random area that does not include any white points. There is no specific requirement on the size of non-cells. All the non-cells are generated based on the marked white points in Fig. 2. Some non-cell examples are shown in Fig. 3(b).

After the image preprocessing, two classes are recognized and extracted from the raw image. The following experiments are all based on the input data generated in the data preparation process.

4.2. Parameter testing

As illustrated in Section 3, there are some parameters in the proposed algorithms and CNNs. The parameters testing and defining will be shown in this section.

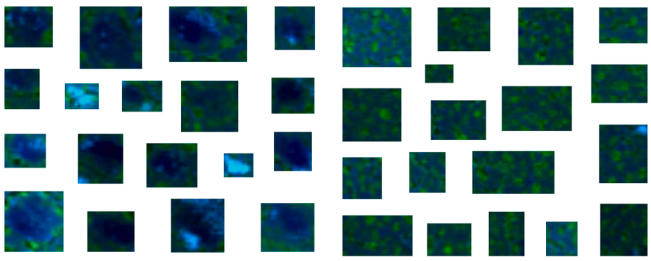
4.2.1. Conventional learning rate annealing parameter testing

In the CLRA algorithm, there are two parameters, δ and γ, which control the rate used to decrease learning rate. To decide the value of those two parameters, a group of validation data that is made of 400 cell images and 400 non-cell images is tested twice within a different combination of two parameters. The range of δ and γ is the integers from 2 to 5 so that 16 different combinations of the two parameters are generated. The small value of the learning rate is preferred because it can avoid the local optimal (Baydin et al., 2017). In the learning rate testing-related research, such as (Baydin et al., 2017 & Ismail et al., Ismail, Ahmad, Soh, Hassan, & Harith, 2019), the learning rate is set from 10^{-1} to 10^{-6}, and the decay rate is set as 0.25, 0.5, and 0.75. So, in this research, hence, γ and λ are set from 2 to 5 to make the learning rate test from 10^{-2} to 10^{-13}. Then, the accuracy will be calculated to evaluate the performance of each combination of parameters. The test results of the final accuracy are summarized in Table 1.

Fig. 4 summarizes the accuracy based on the different epochs. When AlexNet starts with a fast learning rate, our test results show, it produces the most accurate results when the slowing of the rate is controlled by $\delta = 3$ and $\gamma = 3$. To further validate the parameters settings, another brain tumor dataset is tested with all potential parameter combinations. 200 tumor and 200 normal images are used to train and test 16 different combinations of the parameters. The experimental results are summarized in Table 2. The parameter setting ($\delta = 3, \gamma = 4$) obtains the highest accuracy. Combined with the SRS classification results, two parameters settings ($\delta = 3, \gamma = 3$, and $\delta = 3, \gamma = 4$) are tested in the following experiments.

4.2.2. Experiment parameters setting

Table 3 summarizes the default values of other parameters for AlexNet that were used for training and testing data in the experiment. In particular, λ represents the number of epochs before applying the Gaussian surrogate model in $GLRO_1$ and ω is the number of input learning rates to Gaussian surrogate model in $GLRO_2$; i.e., $\lambda = 10$ and $\omega = 10$ is based on the previous experimental results. If the number of input learning rates is less than 10, the Gaussian surrogate model is fail to fit the input data. Thus, 10 is the minimum epoch to apply the Gaussian surrogate model in $GLRO_1$. In terms of $GLRO_2$, which only stores the recent learning rate information, the minimum number of input learning rates is set as 10 to guarantee the



(a) Cells. (b) Non-cells.

Fig. 3. Cells and non-cells examples.

Table 1
Classification accuracy for parameter settings with SRS images.

$\gamma \backslash \delta$	2	3	4	5
2	93.1%	94%	90.4%	92.5%
3	92%	**94.3%**	92.7%	92.2%
4	91.5%	**93.7%**	89.4%	92.5%
5	87.3%	92.5%	89.3%	92.1%

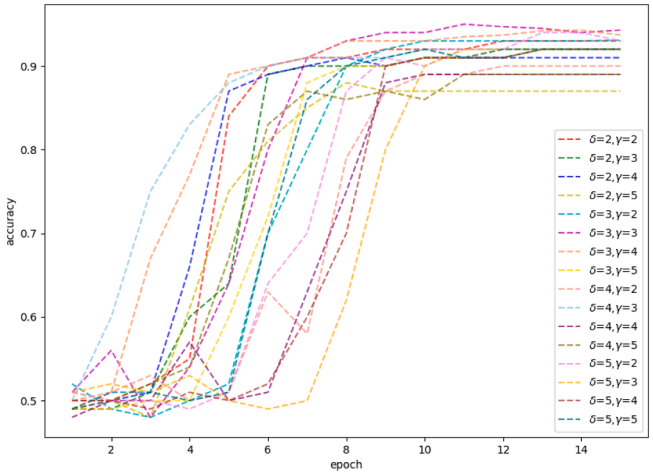


Fig. 4. Parameters testing results for δ and γ.

Table 2
Classification accuracy for parameter settings with brain tumor images.

$\gamma \backslash \delta$	2	3	4	5
2	92.7%	91.0%	91.4%	90.9%
3	92.3%	**93.5%**	**93.5%**	90.2%
4	91.1%	**93.9%**	92.4%	91.2%
5	89.2%	91.9%	90.5%	91.0%

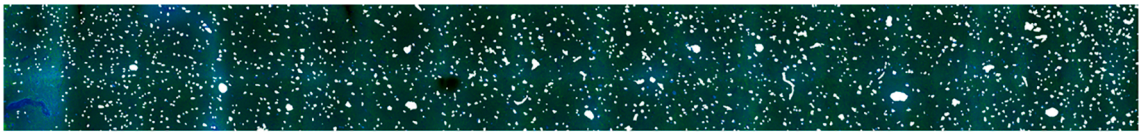


Fig. 2. A full SRS image after manually marking the cells to white.

Table 3
Parameters of experiment.

Batch size	32
Number of epoch	30
Number of cell samples (train)	1,600
Number of non-cell samples (train)	1,600
Number of cell samples (test)	400
Number of non-cell samples (test)	400
λ & ω	10
lr_1 & lr_2	0.01 & 0.001

Gaussian surrogate model performance. lr_1 refers the initial learning rate setting in $GLRO_1$ and $GLRO_2$. It is reported that the learning rate should be a large number at the beginning stage to increase the convergence speed (Dubey et al., 2019; Takase et al., 2018). Thus, 0.01 is adopted from (Hoseini, Shahbahrami, & Bayat, 2019) when setting the initial value for the learning rate decay process in this research. Then, lr_2 is 0.001 which can be calculated by the initial learning rate divided by the number of epochs before applying the GPR model.

4.3. Experimental results

Five tests compared different settings for the learning rate. In the first case (LR = 0.001), the constant learning rate with the value of 0.001 is tested. Then, the learning rate will change sometime during the process, according to Algorithm 1 in Section 3. Based on the parameter testing in the previous section, δ and γ are set as 3 and 3 (CLRA(3,3)) in the second test and defined as 3 and 4 (CLRA(3,4)) in the third test. Next, in the fourth case ($GLRO_1$), the learning rate will be updated dynamically based on the Algorithm 2. And the λ is defined as 10, lr_1 and lr_2, which are two hyperparameters are set as 0.01 and 0.001 respectively. In the last case ($GLRO_2$), only a certain number of the learning rate will input to the surrogate model. The ω is defined as 10, lr_1 and lr_2 are the same as $GLRO_1$. All experiments are implemented under the computational specification of 64-bit Windows 10, with Intel i9 processor (3.60 GHz), 32 GB random-access memory (RAM), and NVIDIA GeForce RTX 2080.

4.3.1. Classification result

Table 4 summarizes the mean and standard deviation (SD) regarding the performance of different classification models. From the experimental results, it is noted that the proposed $GLRO_1$ and $GLRO_2$ obtain a higher value in terms of accuracy, sensitivity, specificity, and precision. Also, $GLRO_2$ achieves the lowest SD of accuracy and sensitivity and CLRA(3,4) has the lowest SD of specificity and precision. Fig. 5 displays the box-plot for five tests. The accuracy of LR = 0.001 is not as stable as the other algorithms and has the lowest average accuracy. On the other hand, $GLRO_2$ is more stable and achieves the largest average accuracy. The highest accuracy could reach 98% in $GLRO_1$.

The learning rate update rule is the only difference between the five tests, Fig. 6 presents the average learning rate during the calculation process. The final learning rate for CLRA(3,3) is 5.08053E-08, and for CLRA(3,4) is 3.8147E-09. For the last two tests, the learning rates update according to the relationship between the previous learning and accuracy, the final learning rates are 6.3886E-10 and 7.0376E-09, respectively.

Table 4
Performance results for each classification model.

Classification model	Accuracy (SD)	Sensitivity (SD)	Specificity (SD)	Precision (SD)
LR = 0.001	89.96%	92.18%	88.01%	87.33%
	1.96%	1.68%	2.51%	2.89%
CLRA(3,3)	92.18%	93.08%	91.33%	91.13%
	2.11%	2.00%	2.42%	2.58%
CLRA(3,4)	92.77%	93.97%	91.64%	91.40%
	0.84%	0.80%	**1.11%**	**1.22%**
$GLRO_1$	95.98%	96.41%	95.56%	95.52%
	1.48%	1.40%	1.59%	1.63%
$GLRO_2$	**96.35%**	**96.89%**	**95.82%**	**95.86%**
	0.77%	**0.62%**	1.14%	1.24%

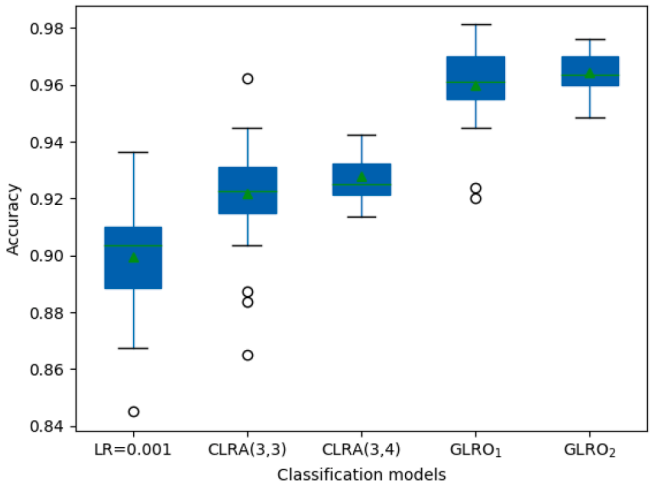


Fig. 5. Box-plot for each classification model.

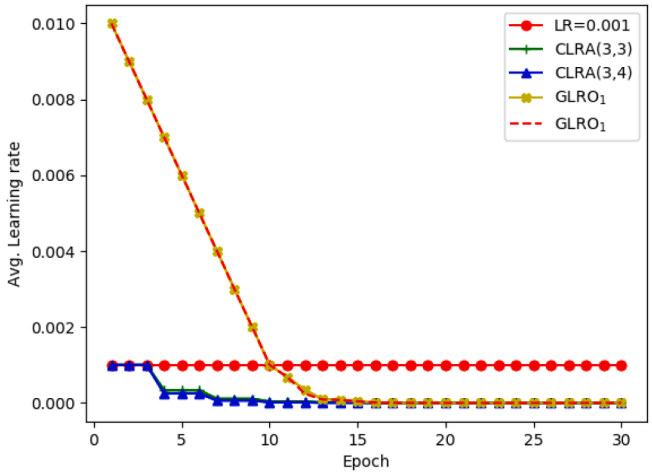


Fig. 6. Average learning rate for each classification model.

As shown in Fig. 7, the average accuracy among the calculation

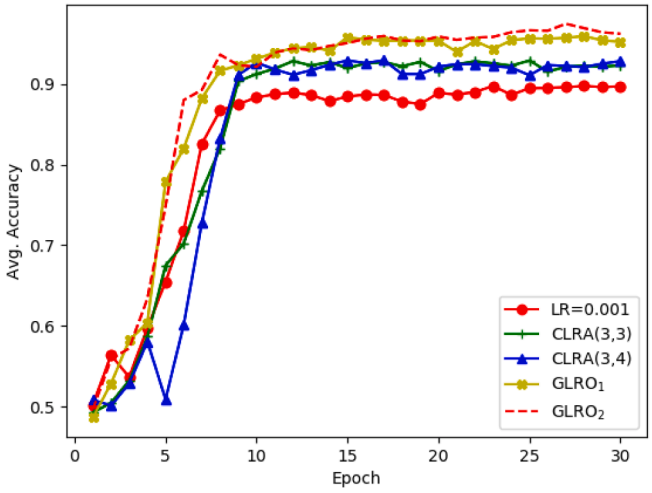


Fig. 7. Average accuracy value for each classification model.

process is demonstrated. $GLRO_2$ achieves the highest accuracy and CLRA (3,3) and CLRA(3,4) obtain similar results but less than $GLRO_1$ and $GLRO_2$. LR = 0.001 has quickly grown at the beginning, but increases slower after about 10 epochs and gets the lowest value finally.

4.3.2. Hypothesis test

After getting the classification accuracy results, hypothesis tests assessed the performance of the proposed algorithms. These tests were applied against the one factor that are studying, which is the learning rate. The tests considered five different update rules that were applied to the learning rate. Analysis of Variance (ANOVA) was used to perform the tests. The null hypothesis and alternative hypothesis are:

- Null hypothesis: two testing learning rate optimization cases have no significant differences in terms of the accuracy of the testing results
- Alternative hypothesis: two testing learning rate optimization cases have significant differences in terms of the accuracy of the testing results

Tables 5 and 6 summarize the p-values and F-values, respectively. From the test results, $GLRO_1$ and $GLRO_2$ have significant differences with other three tests. Combined with the box-plot of accuracy, the proposed $GLRO_1$ and $GLRO_2$ outperform the other three algorithms. Moreover, two CLRA algorithms and two *GLRO* algorithms have no significant differences. The table with the F value also indicates the same conclusion.

4.3.3. Comparison with other medical images and CNN frameworks

The proposed *GLRO* has shown as an effective method when classifying the cells and non-cells in SRS images and outperforms other simple learning rate annealing methods according to the hypothesis tests. To test the generalization ability of the proposed algorithms, three benchmark medical images (i.e., binary chest X-ray (Melendez et al., 2014), brain tumor image dataset (Menze et al., 2014), and Breast Cancer Histopathological Image (BreaKHis) (Spanhol, Oliveira, Petitjean, & Heutte, 2015)) are tested. These three medical datasets are widely used in many machine learning models to validate various proposed models, especially for classification tasks. More research and usages of the three datasets can be found in (Stephen, Sain, Maduh, & Jeong, 2019; Sharma, Jain, Bansal, & Gupta, 2020; Seetha & Raja, 2018; Pathak et al., 2019; Wei, Han, He, & Yin, 2017; Jiang, Chen, Zhang, & Xiao, 2019). The number of epochs for these experiments is set as 100, and the batch size is defined as 32. Other parameters settings are the same as the SRS image classification model. The classification accuracy results are summarized in Table 7.

As shown in Table 7, the proposed $GLRO_1$ and $GLRO_2$ outperform other three learning rate control methods in terms of their accuracies. Especially, $GLRO_2$ achieves the highest accuracy for Chest X-ray, brain tumor, and part of the BreaKHis datasets. Compared to the generic Adam (i.e., constant learning rate as 0.001), $GLRO_1$ and $GLRO_2$ achieve a significant improvement on three popular medical image classifications, which can imply the generalization ability of the proposed algorithms.

To further validate the reliability of the proposed algorithms, several popular CNN frameworks (i.e., VGG16 (Simonyan & Zisserman, 2014), VGG19 (Simonyan & Zisserman, 2014), ResNet-50 (He, Zhang, Ren, & Sun, 2016), DenseNet-121 (Huang, Liu, Van Der Maaten, & Weinberger, 2017), and MobileNet (Howard et al., 2017)) are applied to classify the cells and non-cells of SRS images. The generic Adam (i.e., constant learning rate as 0.001) is compared to *GLRO*-based Adam with these CNN frameworks in terms of the classification accuracy. The classification results are summarized in Table 8.

Based on the classification results in Table 8, it is noted that the proposed $GLRO_1$ and $GLRO_2$ outperform the generic Adam, which shows that the proposed algorithms obtain the promising classification results on the AlexNet and can be widely applied on the other commonly-used CNN models.

Table 5
p-values of the ANOVA test.

	LR = 0.001	CLRA(3,3)	CLRA(3,4)	$GLRO_1$	$GLRO_2$
LR = 0.001	-	0	0	0	0
CLRA(3,3)	0	-	0.2	0	0
CLRA(3,4)	0	0.2	-	0	0
$GLRO_1$	0	0	0	-	0.165
$GLRO_2$	0	0	0	0.165	-

Table 6
F-values of the ANOVA test.

	LR = 0.001	SLRA(3,3)	SLRA(3,4)	$GLRO_1$	$GLRO_2$
LR = 0.001	-	14.81	43.38	150.13	244.71
CLRA(3,3)	14.81	-	1.69	54.37	92.59
CLRA(3,4)	43.38	1.69	-	88.75	289.48
$GLRO_1$	150.13	54.37	88.75	-	1.98
$GLRO_2$	244.71	92.59	289.48	1.98	-

Table 7
Classification accuracy of benchmark medical images.

	Chest X-ray	Brain tumor	BreaKHis			
			40 ×	100 ×	200 ×	400 ×
LR = 0.001	87.35%	89.94%	85.23%	83.21%	81.44%	80.92%
CLRA (3,3)	91.17%	93.89%	87.40%	86.33%	83.95%	81.64%
CLRA (3,4)	91.24%	93.62%	87.25%	85.76%	83.67%	82.38%
$GLRO_1$	92.18%	96.05%	91.78%	91.51%	**91.44%**	90.29%
$GLRO_2$	**92.56%**	**96.27%**	**92.32%**	**91.77%**	91.17%	**90.43%**

5. Conclusions and future works

In summary, this research proposes two novel GPR-based learning rate optimization algorithms and compares them with constant learning rate and conventional learning rate annealing algorithms. Some conclusions can be obtained. $GLRO_2$ obtained the best results in terms of the value of four indicators in the confusion matrix and shows significant improvement of constant learning rate and conventional learning annealing algorithm. $GLRO_1$ and $GLRO_2$ achieve similar results. Moreover, the annealing learning rate algorithms are more stable and have fewer oscillations of loss value during the learning process than the constant learning rate. On the other hand, all of the five cases show a better ability to classify the cell than classifying the non-cell, which implies the characteristics of the experimental data. Moreover, the validation results of three medical images and five CNN frameworks show that the proposed *GLRO* algorithms have great generalization ability and reliability, so that they could be widely applied in other image data and CNN models to improve the model performance. However, some work can be done such as studying the threshold of the quantity of the learning rate of the surrogate model could be dynamical and changed in a different situation. Future works can be also focused on the explainability of the proposed algorithms. For instance, the features

Table 8
SRS image classification accuracy based on popular CNN frameworks.

	VGG16	VGG19	ResNet-50	DenseNet-121	MobileNet
LR = 0.001	89.33%	87.42%	87.90%	91.77%	90.87%
$GLRO_1$	**95.79%**	93.29%	93.75%	94.93%	95.06%
$GLRO_2$	95.52%	**93.81%**	**93.97%**	**96.75%**	**95.13%**

used to determine the cell classification results can be marked in the image to increase the model traceability. At the same time, to enhance the interpretability of the classification results, the inference process of the model can be visualized. As mentioned in (Holzinger, 2018), more knowledge and logic approaches can be applied to make the classification results explainable and re-traceable to exploit the results obtained from the CNN models.

CRediT authorship contribution statement

Yuanyuan Li: Methodology, Software, Writing - original draft, Writing - review & editing. **Qianqian Zhang:** Data curation, Validation, Visualization. **Sang Won Yoon:** Conceptualization, Validation, Supervision.

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.

References

Abdel-Hamid, O., Mohamed, A.-R., Jiang, H., & Penn, G. (2012). Applying convolutional neural networks concepts to hybrid nn-hmm model for speech recognition. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 4277–4280). IEEE.

Baydin, A.G., Cornish, R., Rubio, D.M., Schmidt, M., & Wood, F. (2017). Online learning rate adaptation with hypergradient descent. arXiv preprint arXiv:1703.04782,.

Bergstra, J., Bardenet, R., Bengio, Y., & Kégl, B. (2011). Algorithms for hyper-parameter optimization. In *25th Annual Conference on Neural Information Processing Systems (NIPS 2011). Neural Information Processing Systems Foundation volume 24*.

Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of Machine Learning Research, 13*, 281–305.

Dauphin, Y., De Vries, H., & Bengio, Y. (2015). Equilibrated adaptive learning rates for non-convex optimization. In *Advances in Neural Information Processing Systems* (pp. 1504–1512).

Dong, Z., Xu, W., Xu, J., & Zhuang, H. (2019). Application of adam-bp neural network in leveling fitting. In *IOP Conference Series: Earth and Environmental Science* (p. 022036). IOP Publishing volume 310.

Dubey, S. R., Chakraborty, S., Roy, S. K., Mukherjee, S., Singh, S. K., & Chaudhuri, B. B. (2019). diffgrad: an optimization method for convolutional neural networks. *IEEE Transactions on Neural Networks and Learning Systems, 31*, 4500–4511.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 770–778).

Holzinger, A. (2018). From machine learning to explainable ai. In *2018 World Symposium on Digital Intelligence for Systems and Machines (DISA)* (pp. 55–66). IEEE.

Hoseini, F., Shahbahrami, A., & Bayat, P. (2019). Adaptahead optimization algorithm for learning deep cnn applied to mri segmentation. *Journal of Digital Imaging, 32*, 105–115.

Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., & Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861.

Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 4700–4708).

Ismail, A., Ahmad, S. A., Soh, A. C., Hassan, K., & Harith, H. H. (2019). Improving convolutional neural network (cnn) architecture (minivggnet) with batch normalization and learning rate decay factor for image classification. *International Journal of Integrated Engineering, 11*.

Jiang, Y., Chen, L., Zhang, H., & Xiao, X. (2019). Breast cancer histopathological image classification using convolutional neural networks with small se-resnet module. *PloS One, 14*, Article e0214587.

Kingma, D.P., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980,.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems* (pp. 1097–1105).

Kumar, A., Kim, J., Lyndon, D., Fulham, M., & Feng, D. (2016). An ensemble of fine-tuned convolutional neural networks for medical image classification. *IEEE Journal of Biomedical and Health Informatics, 21*, 31–40.

LeCun, Y. A., Bottou, L., Orr, G. B., & Müller, K.-R. (2012). Efficient backprop. In *Neural Networks: Tricks of the Trade* (pp. 9–48). Springer.

Lee, H., & Kwon, H. (2017). Going deeper with contextual cnn for hyperspectral image classification. *IEEE Transactions on Image Processing, 26*, 4843–4855.

Lu, F.-K., Basu, S., Igras, V., Hoang, M. P., Ji, M., Fu, D., Holtom, G. R., Neel, V. A., Freudiger, C. W., Fisher, D. E., et al. (2015). Label-free dna imaging in vivo with stimulated raman scattering microscopy. *Proceedings of the National Academy of Sciences, 112*, 11624–11629.

Lu, F.-K., Calligaris, D., Olubiyi, O. I., Norton, I., Yang, W., Santagata, S., Xie, X. S., Golby, A. J., & Agar, N. Y. (2016). Label-free neurosurgical pathology with stimulated raman imaging. *Cancer Research, 76*, 3451–3462.

Melendez, J., van Ginneken, B., Maduskar, P., Philipsen, R. H., Reither, K., Breuninger, M., Adetifa, I. M., Maane, R., Ayles, H., & Sánchez, C. I. (2014). A novel multiple-instance learning-based approach to computer-aided detection of tuberculosis on chest x-rays. *IEEE Transactions on Medical Imaging, 34*, 179–192.

Menze, B. H., Jakab, A., Bauer, S., Kalpathy-Cramer, J., Farahani, K., Kirby, J., Burren, Y., Porz, N., Slotboom, J., Wiest, R., et al. (2014). The multimodal brain tumor image segmentation benchmark (brats). *IEEE Transactions on Medical Imaging, 34*, 1993–2024.

Park, J., Yi, D., & Ji, S. (2020). A novel learning rate schedule in optimization for neural networks and it's convergence. *Symmetry, 12*, 660.

Pathak, K., Pavthawala, M., Patel, N., Malek, D., Shah, V., & Vaidya, B. (2019). Classification of brain tumor using convolutional neural network. In *2019 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA)* (pp. 128–132). IEEE.

Quiñonero-Candela, J., & Rasmussen, C. E. (2005). A unifying view of sparse approximate gaussian process regression. *Journal of Machine Learning Research, 6*, 1939–1959.

Rasmussen, C., & Williams, C. (2006). Gaussian processes for machine learning, adaptive computation and machine learning.

Reddi, S.J., Kale, S., & Kumar, S. (2019). On the convergence of adam and beyond. arXiv preprint arXiv:1904.09237.

Renson, L., Sieber, J., Barton, D., Shaw, A., & Neild, S. (2019). Numerical continuation in nonlinear experiments using local gaussian process regression. arXiv preprint arXiv:1901.06970.

Seetha, J., & Raja, S. S. (2018). Brain tumor classification using convolutional neural networks. *Biomedical & Pharmacology Journal, 11*, 1457.

Sharma, H., Jain, J. S., Bansal, P., & Gupta, S. (2020). Feature extraction and classification of chest x-ray images using cnn to detect pneumonia. In *2020 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence)* (pp. 227–231). IEEE.

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

Smith, L. N. (2017). Cyclical learning rates for training neural networks. In *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)* (pp. 464–472). IEEE.

Spanhol, F. A., Oliveira, L. S., Petitjean, C., & Heutte, L. (2015). A dataset for breast cancer histopathological image classification. *IEEE Transactions on Biomedical Engineering, 63*, 1455–1462.

Stephen, O., Sain, M., Maduh, U.J., & Jeong, D.-U. (2019). An efficient deep learning approach to pneumonia classification in healthcare. Journal of Healthcare Engineering, 2019.

Takase, T., Oyama, S., & Kurihara, M. (2018). Effective neural network training with adaptive learning rate based on training loss. *Neural Networks, 101*, 68–78.

Wei, B., Han, Z., He, X., & Yin, Y. (2017). Deep learning model based breast cancer histopathological image classification. In *2017 IEEE 2nd International Conference on Cloud Computing and Big Data Analysis (ICCCBDA)* (pp. 348–353). IEEE.

Wu, J., Chen, X.-Y., Zhang, H., Xiong, L.-D., Lei, H., & Deng, S.-H. (2019). Hyperparameter optimization for machine learning models based on bayesian optimization. *Journal of Electronic Science and Technology, 17*, 26–40.

Xu, C., Qin, T., Wang, G., & Liu, T.-Y. (2017). *Reinforcement learning for learning rate control*. arXiv preprint arXiv:1705.11159.

Yang, J., & Yang, G. (2018). Modified convolutional neural network based on dropout and the stochastic gradient descent optimizer. *Algorithms, 11*, 28.

Yin, W., Kann, K., Yu, M., & Schütze, H. (2017). Comparative study of cnn and rnn for natural language processing. arXiv preprint arXiv:1702.01923.

Zeiler, M.D. (2012). Adadelta: an adaptive learning rate method. arXiv preprint arXiv:1212.5701.

Zhang, G., Wang, P., Chen, H., & Zhang, L. (2019). Wireless indoor localization using convolutional neural network and gaussian process regression. *Sensors, 19*, 2508.

Zhang, M., Li, H., Lyu, J., Ling, S.H., & Su, S. (2019). Multi-level cnn for lung nodule classification with gaussian process assisted hyperparameter optimization. arXiv preprint arXiv:1901.00276.

Zhang, Q., Wang, H., Lu, H., Won, D., & Yoon, S. W. (2018). Medical image synthesis with generative adversarial networks for tissue recognition. In *2018 IEEE International Conference on Healthcare Informatics (ICHI)* (pp. 199–207). IEEE.