Expert Systems With Applications 315 (2026) 131696

Contents lists available at ScienceDirect

Expert Systems With Applications

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

Adaptive image enhancement for robust CNN classification under low illumination

José A. Rodríguez-Rodríguez [a], Ezequiel López-Rubio [b,c], Salvador Jiménez-Segura [a], Miguel A. Molina-Cabello [b,c,*]

[a] Department of Computer Languages and Computer Science, University of Málaga, Málaga, Spain
[b] ITIS Software, University of Málaga, Málaga, Spain
[c] Instituto de Investigación Biomédica de Málaga y Plataforma en Nanomedicina-IBIMA Plataforma BIONAND, Málaga, Spain

ARTICLE INFO

Keywords:
Image classification
Convolutional neural network
Image enhancement algorithm
Illumination conditions
Brightness

ABSTRACT

Convolutional Neural Networks (CNNs) are widely used in image classification tasks, but their performance can degrade significantly under poor illumination conditions. Although numerous image enhancement methods have been developed to mitigate this issue, identifying the most appropriate technique for a specific image remains challenging. This work aims to determine the most effective image enhancement algorithm for a potentially dimmed input image to enhance the classification prediction performance. To do this, a trained regressor, which evaluates different features of an image, ascertains the increase or decrease in classification prediction performance between the input image and the enhanced image. These predictions are then used to select the most suitable image enhancement algorithm to be applied to the input image for optimal CNN classification. Experimental results using various CNN architectures and enhancement techniques demonstrate that the proposed strategy consistently improves classification performance under challenging lighting conditions.

1. Introduction

In a few years, artificial intelligence (Russell & Norvig, 2009) has experienced significant growth, driven primarily by major advancements in the field of deep learning. This has contributed significantly to improving machines' ability to comprehend, learn, and perform complex functions, leading to significant breakthroughs in areas ranging from pattern recognition to natural language processing (Górriz et al., 2020).

The field that allows us to identify patterns in input data using algorithms that classify factors based on their degree of influence is Machine Learning (Russell & Norvig, 2009). This process continues iteratively, enabling learning and improvement over time, and the goal is to mimic human intelligence. One of the subsets in machine learning is Deep learning (Schulz & Behnke, 2012), where input data is fed into an artificial neural network (Drew & Monson, 2000), which contains multiple hidden layers responsible for processing data. Through this processing, we achieve learning by establishing connections and weights, progressively obtaining better results. This requires training to experiment with different data processing combinations until minimizing errors and improving predictions. Convolutional Neural Networks (CNNs) (CS231n, n.d.; Yamashita et al., 2018) are widely used for their performance in many computer vision tasks, but they are also applied in other domains such as speech recognition (Zhang et al., 2017) or natural language processing (Wang & Gang, 2018).

Image classification is a common application, and in many cases, CNNs outperform humans. Different methods have been developed to face image classification in general (Han et al., 2018) as well as a specific type of images such as hyperspectral images (Gao et al., 2025; Li et al., 2024; Wu et al., 2024a), satellite images (Shendy & Nalepa, 2024), soil images (Pandiri et al., 2024), remote sensing images (Chen et al., 2025), images with different marine species (Zhang et al., 2024), food images (Phiphitphatphaisit & Surinta, 2024), skin cancer images (Doğan et al., 2025) …CNNs are trained using these images, taking their pixels and labels as input, and learning with filters applied to each image. These filters enable the network to identify specific patterns by searching for parts of the image that match each filter. The network architecture is crucial for obtaining better results when making modifications to different layers, using activation functions, or learning algorithms. However, the quality of the image also plays a significant role in classification results as can be observed through experiments conducted with various classification models (Rodríguez-Rodríguez et al., 2024, 2021; Rodríguez-Rodríguez et al., 2021), and it is often overlooked. The performance of classification algorithms can be degraded due to the next physic effects: the presence of noise and low

* Corresponding author.
E-mail address: miguelangel@lcc.uma.es (M.A. Molina-Cabello).

https://doi.org/10.1016/j.eswa.2026.131696
Received 27 June 2025; Received in revised form 28 September 2025; Accepted 12 February 2026
Available online 20 February 2026

illumination. The first one, the noise, consist in the corruption of the pixel value and there are several types or sources including Gaussian, Speckle and Shot noise. To address this, several image denoising algorithms can be found in the literature to improve image quality (Elad et al., 2023) proposing from classic approaches like advanced Wiener filters (Bled & Pitié, 2023) to more advanced techniques like transformers (Choi et al., 2023) and the use of auxiliary CNNs (Tiantian et al., 2024; Wu et al., 2024b; Zhang et al., 2023), making an improvement in the CNN classification performance. Furthermore, it is important to highlight that the research of image synthesis like Generative Adversarial Networks (Goodfellow et al., 2020) (GANs) or diffusion models (Peebles & Xie, 2023) have contributed indirectly introducing some steps for reducing the noise.

The second physic effect is the low illumination that will reduce the constrast of the images and it will make more difficult the features extraction by the intermediates layers in the CNN. It is essential in this case to consider image preprocessing techniques for classification (Koo & Cha, 2017; Pal & Sudeep, 2016) to train an efficient Neural Network. Networks are typically trained and tested with high-quality image sets, but real-life scenarios may involve external factors affecting image quality. Improving the image involves enhancing the quality and informative content of the original data before processing. This procedure aims to amplify specific important features of the image for analysis, diagnosis, and visualization. Changes can be made to intensity, contrast, noise reduction, edge sharpening, filtering, etc.

Image improvement plays a crucial role in image processing applications. There are many image enhancement algorithms, given their subjective nature and dependence on the application. Four algorithms commonly used are Gamma Correction (GC) (Poynton, 2003), Histogram Equalization (HE) (Patel et al., 2013), Contrast Limited Adaptive Histogram Equalization (CLAHE) (Reza, 2004), and Logarithmic Transformation (LT) (Manikpuri & Yadav, 2014). All these techniques work on the pixels of an image to improve contrast, and they are traditional and well-known methods; however, many advances in low-light image enhancement have been reached in recent years, particularly from the irruption of deep learning (Liu & Fan, 2025). There are supervised-learning methods such as CNN-based models (Deng et al., 2022; Lin et al., 2023; Liu et al., 2023, 2024; Yao et al., 2024) or transformer-based models (She et al., 2024; Wen et al., 2025; Xu et al., 2022); unsupervised-learning methods such as Generative adversarial networks (GANs) (Guan et al., 2024; Wu et al., 2024c) or unsupervised domain adaptation methods (Gao et al., 2022); by using diffusion models (Du et al., 2024; Lv et al., 2024; Wang et al., 2024), etc.

Although many image enhancement techniques have been introduced to improve visual quality under challenging lighting conditions, and CNN-based classifiers perform well in controlled environments, a key limitation persists: the relationship between enhancement methods and classification accuracy is not thoroughly explored. Most existing approaches apply a single, fixed enhancement algorithm without considering whether it actually improves classification for a specific image. Additionally, there is no established method for adaptively choosing the most appropriate enhancement technique based on the image content. This limitation becomes particularly significant in real-world settings with unpredictable illumination. To address this, we propose a regression-based approach that evaluates the potential effect of multiple enhancement algorithms on classification performance and automatically selects the one most likely to yield optimal results.

In this work, given a possible dimmed input image to be classified using a CNN classifier, we try to identify which image enhancement algorithm is more suitable to be applied to that input image in order to improve the performance of the classification prediction. To achieve that, we will train a regressor (Freedman, 2009; Zhang & O'Donnell, 2020) to, given the characteristics of an image with reduced illumination, determine the loss or gain in performance in the classification of the input image and the image with the applied enhancement algorithm. Then, this information is used to select which image enhancement algorithm is, a priori, the best one to be applied to the input image in order to be better classified by the CNN.

The structure of this paper is as follows. The proposed methodology is depicted in Section 2. After that, the carried out experiments are provided in Section 3. Finally, Section 4 describes the conclusions.

2. Methodology

In this section, the proposed method to improve the performance of deep convolutional image classifiers by means of automated selection of image enhancement algorithms is specified. Section 2.1 establishes the theoretical framework of our proposal. Section 2.2 explains our proposed classification performance optimization procedure. Finally, Section 2.3 outlines the workflow of our approach.

2.1. Framework

This subsection establishes the theoretical framework of our approach. Let us assume that the set of image classes is:

$$C = \{C_1, \dots, C_K\} \tag{1}$$

where K is the number of classes. Given an input image $\mathbf{X}$, a deep convolutional image classifier $\mathcal{F}$ can be used to estimate the class y that $\mathbf{X}$ belongs to:

$$\mathbf{p} = \mathcal{F}(\mathbf{X}) \tag{2}$$

$$q = \arg \max_{i \in \{1,\dots,K\}} p_i \tag{3}$$

where $\mathbf{p} \in [0,1]^K$ is the vector of the estimated probabilities that image $\mathbf{X}$ belongs to each of the K classes, and $q \in \{1, \dots, K\}$ is the estimated class for image $\mathbf{X}$.

Next, let us consider that a corrupted image $\hat{\mathbf{X}}$ is given as input. Let us also assume that M enhancement algorithms are available such that $\tilde{\mathbf{X}}_j$ is the enhanced image obtained by applying the jth enhancement algorithm to $\hat{\mathbf{X}}$, with $j \in \{1, \dots, M\}$. For the sake of notational simplicity, it will be assumed that $j = 1$ corresponds to the no-operation enhancement algorithm, i.e. $\tilde{\mathbf{X}}_1 = \hat{\mathbf{X}}$.

The goal is to select an enhancement algorithm for $\hat{\mathbf{X}}$ such that the image classification performance is improved. The aim is to maximize the following goal function:

$$r\left(\hat{\mathbf{X}}, \tilde{\mathbf{X}}_j\right) = \mathcal{L}\left(\hat{\mathbf{X}}\right) - \mathcal{L}\left(\tilde{\mathbf{X}}_j\right) \tag{4}$$

where $\mathcal{L}$ stands for the cross-entropy loss function:

$$\mathcal{L}(\mathbf{X}) = -\sum_{i=1}^{K} \mathbb{I}\left(\mathbf{X} \in C_i\right) \log p_i \tag{5}$$

where $\mathbb{I}$ denotes the indicator function:

$$\mathbb{I}(z) = \begin{cases} 1 & \text{if } z \text{ is true} \\ 0 & \text{if } z \text{ is false} \end{cases} \tag{6}$$

The selection of the cross-entropy loss function is driven by its differentiability with respect to the vector of estimated probabilities $\mathbf{p}$. Such differentiability makes it more likely that the optimization landscape of the cross-entropy is smoother and better behaved than other possible choices for the loss function, such as accuracy gain, top-1 margin, or rank flips.

For a fixed value of $\hat{\mathbf{X}}$, the maximization of (4) is equivalent to the decision problem of choosing the enhancement algorithm that minimizes the cross-entropy loss function $\mathcal{L}\left(\tilde{\mathbf{X}}_j\right)$:

$$j^* = \arg \min_{j \in \{1,\dots,M\}} \mathcal{L}\left(\tilde{\mathbf{X}}_j\right) \tag{7}$$

This decision problem is related to meta-learning or algorithm selection, for the specific task to image enhancement. Previous related approaches can be found in Chen et al. (n.d.), Czako et al. (2021).

2.2. Classification performance optimization

Next, the proposed procedure to maximize the classification performance of the system is detailed. The information about the true class labels is not available at test time. Therefore, an estimation of r must be done that can be evaluated at test time, from a suitable vector of characteristic features $\mathbf{c}$ obtained from the input image $\mathbf{X}$ by the feature extraction function $\mathcal{H}$:

$$\mathbf{c} = \mathcal{H}(\mathbf{X}) \tag{8}$$

where $\mathbf{c} \in \mathbb{R}^H$ is the feature vector obtained from image $\mathbf{X}$, which contains H features. In our experiments we have $H = 6$:

$$\mathbf{c} = \left(c_1, c_2, c_3, c_4, c_5, c_6\right) \tag{9}$$

Please note that the definition of the features c_i, $i \in \{1, ., 6\}$, is given in Section 3.1

Consequently, an estimator $\tilde{r}$ of r must be learned such that:

$$\tilde{r}\left(\hat{\mathbf{X}}, \tilde{\mathbf{X}}_j\right) = \mathcal{G}\left(\hat{\mathbf{c}}, \tilde{\mathbf{c}}_j\right) \tag{10}$$

$$\hat{\mathbf{c}} = \mathcal{H}\left(\hat{\mathbf{X}}\right) \tag{11}$$

$$\tilde{\mathbf{c}}_j = \mathcal{H}\left(\tilde{\mathbf{X}}_j\right) \tag{12}$$

where $\mathcal{G}$ is a regressor trained to minimize the following regression error:

$$\mathcal{E} = \frac{1}{NM} \sum_{n=1}^{N} \sum_{j=1}^{M} \left(r\left(\hat{\mathbf{X}}_n, \tilde{\mathbf{X}}_{n,j}\right) - \tilde{r}\left(\hat{\mathbf{X}}_n, \tilde{\mathbf{X}}_{n,j}\right)\right)^2 \tag{13}$$

where N is the number of training images, $\hat{\mathbf{X}}_n$ is the nth training image, and $\tilde{\mathbf{X}}_{n,j}$ is the result of applying the jth image enhancement algorithm to $\hat{\mathbf{X}}_n$.

Let $\mathbf{z} \in \mathbb{R}^{2H}$ be the input for the regressor $\mathcal{G}$ in (10). The SVR (Support Vector Regressor) method is one of our choices for $\mathcal{G}$, so that the regression is given by:

$$\mathcal{G}_{SVR}(\mathbf{z}) = \langle \mathbf{w}, \phi(\mathbf{z}) \rangle + \beta \tag{14}$$

$$\begin{aligned} \min_{\mathbf{w},\beta,\xi,\xi^*} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + \lambda \sum_{i=1}^{Q} \left(\xi_i + \xi_i^*\right) \\ \text{s.t.} \quad & r_i - \langle \mathbf{w}, \phi(\mathbf{z}_i) \rangle - \beta \le \varepsilon + \xi_i, \quad i = 1, \ldots, Q, \\ & \langle \mathbf{w}, \phi(\mathbf{z}_i) \rangle + \beta - y_i \le \varepsilon + \xi_i^*, \quad i = 1, \ldots, Q, \\ & \xi_i, \xi_i^* \ge 0, \quad i = 1, \ldots, Q. \end{aligned}$$

where $Q = NM$ is the number of training samples for the SVR, λ is a regularization parameter, $\mathbf{w}$ is the weight vector, β is the independent term, ϵ is the margin, ξ_i, ξ_i^* are slack variables, and $\tilde{r}_i$ is the output ground truth for the ith training sample of the SVR.

The second method chosen for $\mathcal{G}$ is the Gradient Boosting Regressor (GBR). An iterative process is carried out as follows. An initial constant model ($m = 0$) is computed:

$$F_0(\mathbf{z}) = \frac{1}{Q} \sum_{i=1}^{Q} r_i \tag{15}$$

Residuals and leaf values are computed for the next mth model:

$$t_{im} = r_i - F_{m-1}(\mathbf{z}_i) \tag{16}$$

$$\gamma_{jm} = \frac{\sum_{z_i \in R_{jm}} t_{im}}{\#\left(z_i \in R_{jm}\right)} \tag{17}$$

where # stands for the set cardinal operator, and R_{jm} are the regions associated with the leaves of a regression tree fitted to the targets t_{im}. Then, the next mth model is obtained:

$$F_m(\mathbf{z}) = F_{m-1}(\mathbf{z}) + \nu \sum_{j=1}^{J_m} \gamma_{jm} \mathbb{I}\left(z \in R_{jm}\right) \tag{18}$$

where ν is the step size.

Finally, after a predefined number of iterations S, the process finishes with $\mathcal{G} = F_S$.

The third regression method that we have considered is XGBoost. Again, an iterative process is carried out. The mth regressor is given by:

$$\tilde{r}_i^{(m)} = \sum_{k=1}^{m} f_k(\mathbf{z}_i), \quad f_k \in \mathcal{F} \tag{19}$$

where f_k is a regression tree from the function space $\mathcal{F}$. Also, the initial regressor is $\tilde{r}_i^{(0)} = 0$. Then we have:

$$\tilde{r}_i^{(m)} = \tilde{r}_i^{(m-1)} + f_m(\mathbf{z}_i) \tag{20}$$

$$f_m = \arg\min_{f \in F} \sum_{i=1}^{Q} \left[g_i f(\mathbf{z}_i) + \frac{1}{2} h_i f(\mathbf{z}_i)^2 \right] + \Omega(f) \tag{21}$$

$$g_i = \frac{\partial \mathcal{E}(r_i, \tilde{r}_i^{(m-1)})}{\partial \tilde{r}_i^{(m-1)}} \tag{22}$$

$$h_i = \frac{\partial^2 \mathcal{E}(r_i, \tilde{r}_i^{(m-1)})}{\partial \tilde{r}_i^{(m-1)2}} \tag{23}$$

$$\Omega(f) = \gamma T + \frac{1}{2} \lambda \sum_{u=1}^{T} w_u^2 \tag{24}$$

with T the number of leaves in the regression tree, w_u the weight of the uth leaf, γ the complexity cost per leaf, and λ the L2 regularization on leaf weights.

Once the regressor $\mathcal{G}$ is properly trained, the classification of a new test image $\hat{\mathbf{X}}$ proceeds as follows. First the feature vector $\hat{\mathbf{c}}$ for $\hat{\mathbf{X}}$ is computed by (11). Then the M enhancement algorithms are applied to obtain $\tilde{\mathbf{X}}_j$ for $j \in \{1, \ldots, M\}$. Next, the feature vectors $\tilde{\mathbf{c}}_j$ are computed by (12) for $j \in \{1, \ldots, M\}$. After that, the estimated goal function values $\tilde{r}\left(\hat{\mathbf{X}}, \tilde{\mathbf{X}}_j\right)$ are computed by (10). The enhancement algorithm that maximizes the goal function r is estimated as:

$$s = \arg\max_{j \in \{1, \ldots, M\}} \tilde{r}\left(\hat{\mathbf{X}}, \tilde{\mathbf{X}}_j\right) \tag{25}$$

Finally, the vector $\hat{\mathbf{p}}$ of estimated class probabilities for the test image $\hat{\mathbf{X}}$ is computed from the enhanced image $\tilde{\mathbf{X}}_s$ associated to the sth enhancement algorithm determined by (25):

$$\hat{\mathbf{p}} = \mathcal{F}\left(\tilde{\mathbf{X}}_s\right) \tag{26}$$

2.3. Workflow

This subsection goes into the details of the proposed workflow. In the subsequent experiments, a brightness scaling transformation has been considered as the corruption process. That is, given an original image $\mathbf{X}$, the corrupted input image $\hat{\mathbf{X}}$ is obtained as follows:

$$\hat{\mathbf{X}} = \frac{1}{b}\mathbf{X} \tag{27}$$

where $b \in \mathbb{R}$ is the brightness scale factor, with $b \ge 1$. Note that $b = 1$ does not produce any change in the illumination conditions of the image.

In this work, we have considered $M = 5$ enhancement algorithms. In particular, they are the no-operation algorithm (as previously mentioned, it will be assumed that the first algorithm corresponds to the no-operation enhancement algorithm) and 4 image enhancement algorithms selected from the literature: Gamma Correction, Histogram Equalization, Contrast Limited Adaptive Histogram Equalization, and Logarithmic Transformation. In Fig. 1, a scheme of the operation of the developed method is shown. The workflow is as follows. First, we have an input image that has already undergone a decrease in illumination. Then, we apply each considered image enhancement algorithm to the

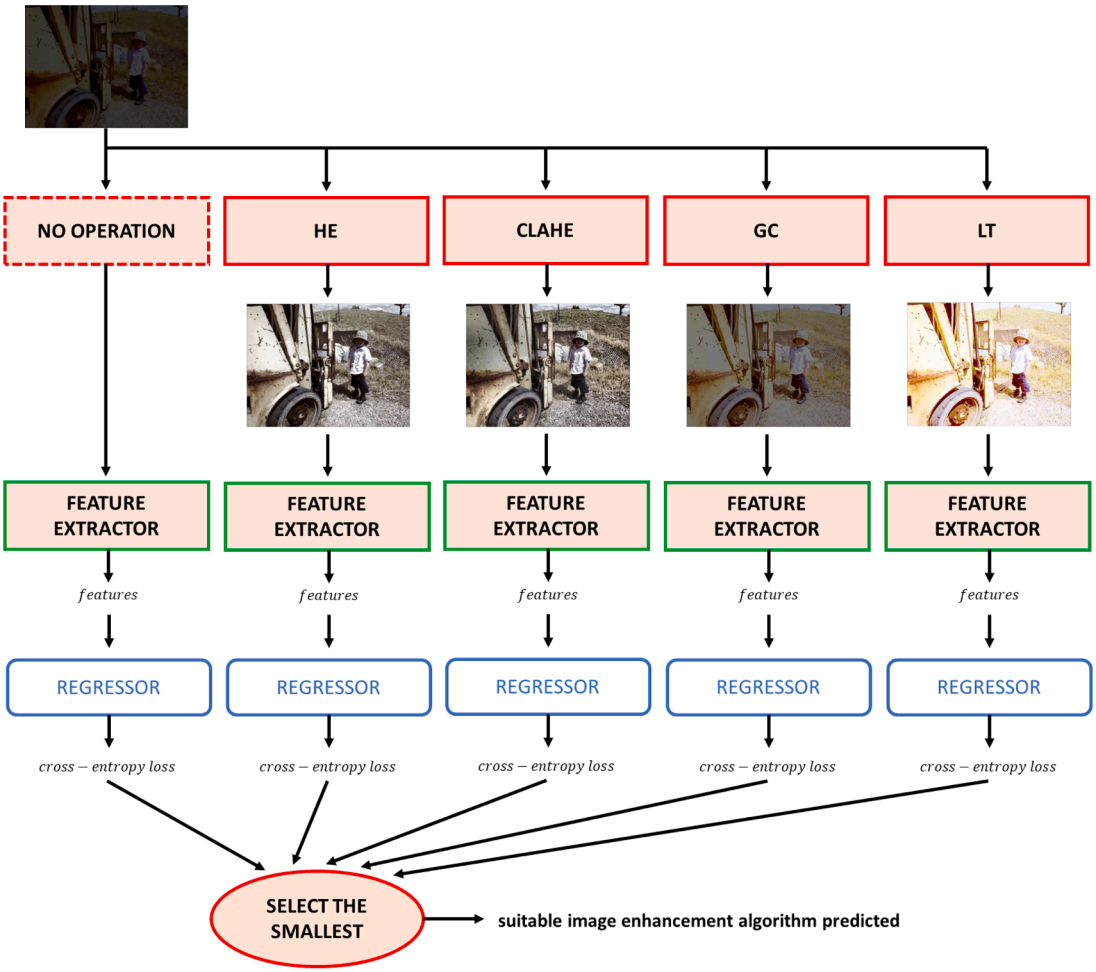

Fig. 1. Scheme of the operation of the proposed methodology to predict the suitability of applying an image enhancement algorithm. A dimmed image is processed by various selected image enhancement algorithms (no-operation algorithm and Histogram Equalization (HE), Contrast Limited Adaptive Histogram Equalization (CLAHE), Gamma Correction (GC), and Logarithmic Transformation (LT)). For each resulting image, we extract the considered features and use a regressor to compute the predicted cross-entropy loss (CEL). The method that yields the smallest value is predicted to be the most suitable for optimally enhancing the input image to be classified.

input image. Subsequently, we extract the features of each generated image and calculate the prediction of the cross-entropy loss (CEL in the image) using the regressor. The minimum of these values predicts which method is, a priori, the suitable one that enhances the input image in the best way.

3. Experiments

This section details the experiments we have carried out. First, different image enhancement algorithms are described. Next, the dataset we used to train and test our proposal is depicted. Then, it is reported the setup of these experiments. After that, the obtained results are reported.

3.1. Methods

It is crucial to emphasize that preprocessing techniques for image classification are essential for training an efficient neural network. In real-world applications, image quality may be lower due to external factors, unlike test datasets that contain high-quality images.

By employing image preprocessing techniques, we achieve an enhancement in both quality and informative content. To achieve these improvements, common practices include intensity and contrast manipulation, noise reduction, edge sharpening, filtering, and more.

The first image enhancement algorithm considered in this work is the Gamma Correction (GC) (Poynton, 2003), which adjusts the relationship between the input and output intensities of an image using a gamma power function. In the simplest cases, we can define gamma correction as follows:

$$V_{out} = AV_{in}^{\gamma}$$

where the input value V_{in} is raised to the power of γ and multiplied by the constant A to obtain the output value V_{out}.

Secondly, Histogram Equalization (HE) (Patel et al., 2013) is an image processing technique aimed at enhancing contrast by redistributing the most frequent intensity values present in the image effectively. This is achieved by expanding the range of available intensities. As a result, brighter areas become even brighter, and darker areas become darker, improving the overall contrast of the image and enhancing visibility and details. In Fig. 2, in the color histograms, it can be observed how the pixels are redistributed across the tones.

The third algorithm is Contrast Limited Adaptive Histogram Equalization (CLAHE) (Reza, 2004) is a variant of Adaptive Histogram Equalization (AHE), which performs calculations on sections of the image. Instead of applying a single histogram to the entire image as in the previous case, it calculates multiple histograms, each corresponding to a specific section of the image. It then utilizes these histograms to redistribute luminosity values in the image. In the case of CLAHE, the process

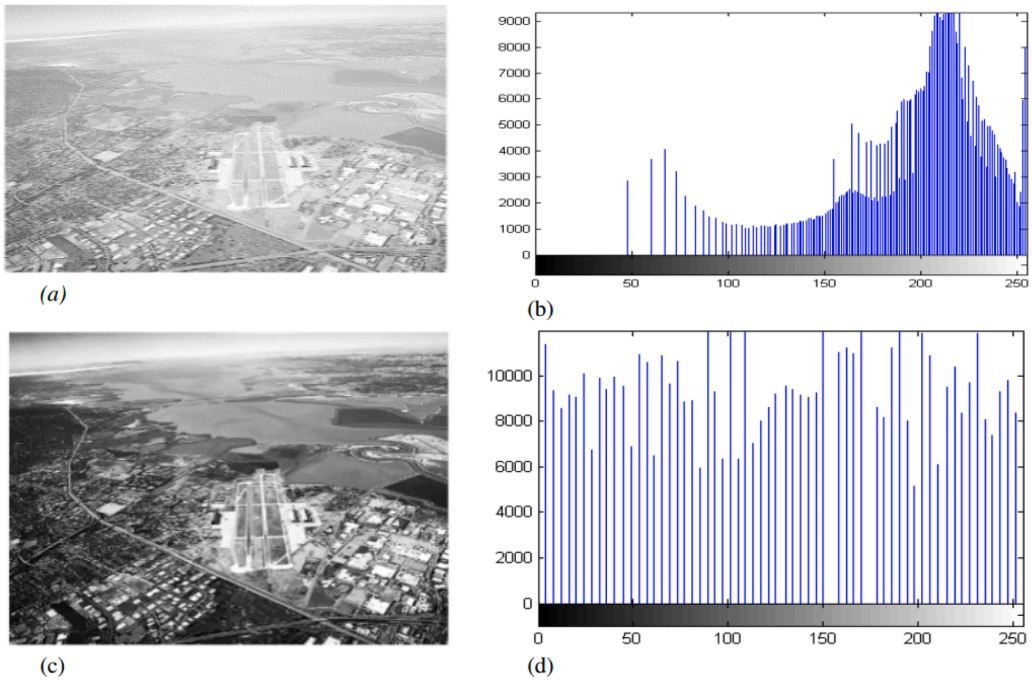

Fig. 2. (a) shows the input image with low contrast, (b) shows the histogram of the input image, (c) shows the histogram equalized image, and (d) shows the histogram of the processed image (Patel et al., 2013).

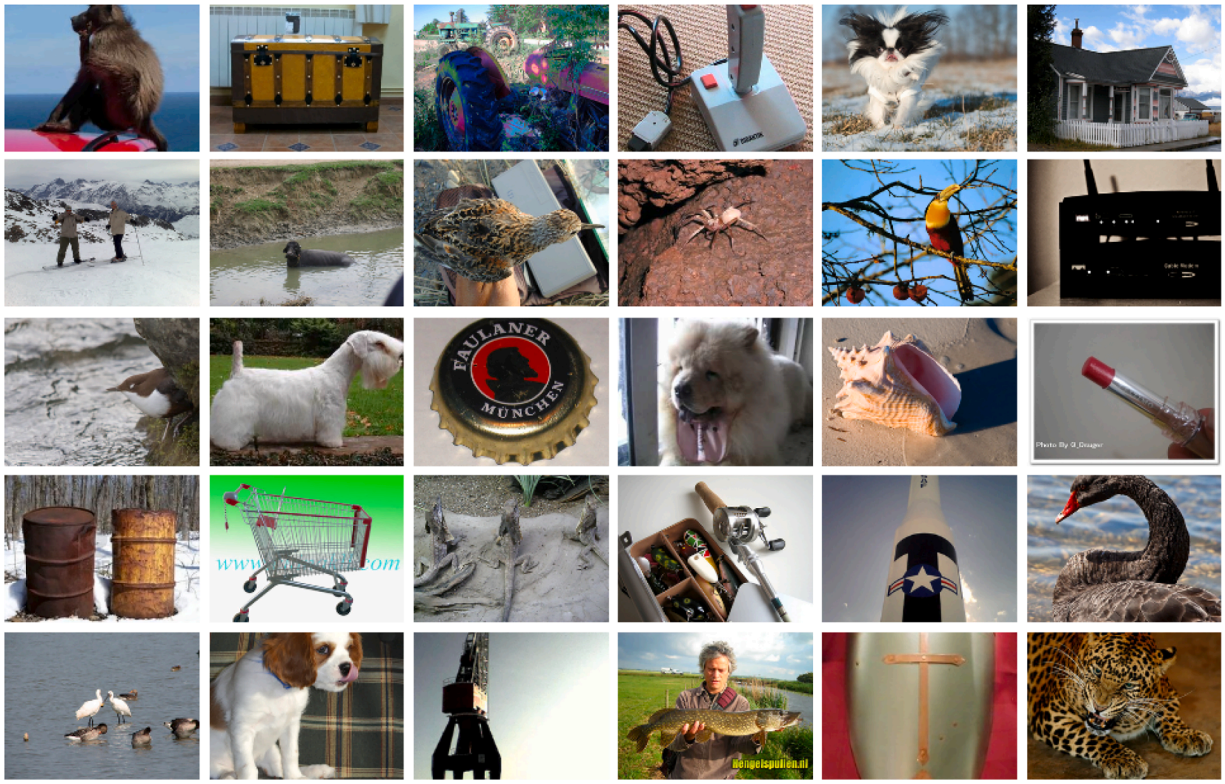

Fig. 3. Some images from ILSRVC2012 validation image set.

is the same, but it incorporates a contrast-limiting mechanism in each section of the image to prevent excessive amplification of noise.

The last selected algorithm is Logarithmic Transformation (LT) (Manikpuri & Yadav, 2014), which applies a logarithmic function to the pixels of the image. When this function is applied to the pixels, it results in an expansion of the dark pixels, highlighting details in darker areas of the image.

Regarding the CNN classifier, some well-known architectures have been selected. In particular, it has been used AlexNet (Krizhevsky et al., 2012) (8 layers), GoogLeNet (Szegedy et al., 2015) (22 layers), and

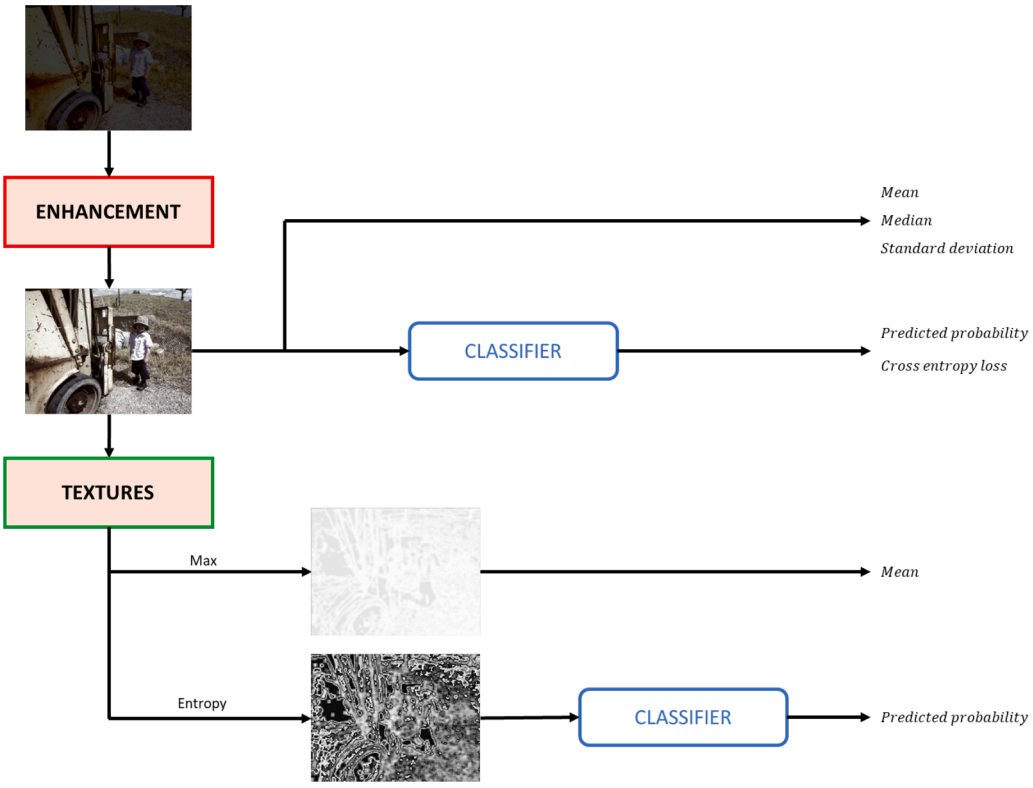

Fig. 4. Feature extractor pipeline.

ResNet (He et al., 2016) architecture (50 layers). We have used pre-trained models of these classifiers by using the PyTorch library (Paszke et al., 2019). We have chosen AlexNet, GoogLeNet, and ResNet as backbones because they have a low footprint, so that the suitability of our approach for low-cost, low-power devices is demonstrated.

In relation to the regressors, to perform their training, we have taken the following image features as independent variables for our problem:

- Mean: the mean of the image pixels.
- Median: the median of the image pixels.
- Standard deviation: the standard deviation of the image pixels
- Predicted class probability: when making a prediction with a model, it provides a list of classes and their associated probabilities, sorted from highest to lowest. Therefore, the first probability in the list is the predicted class probability for the image. The value of this probability is an independent variable for our regressor.
- Cross entropy loss: the cross entropy loss of the probability distribution predicted by the classifier model.

We have also used two texture variables obtained with the Gray-Level Co-occurrence Matrix (GLCM):

- Mean maximum texture: the maximum texture is the most dominant texture present in the image. These are areas or patterns that exhibit higher contrast in terms of gray levels. Then, the mean of this image pixels is computed.
- Predicted probability by obtaining entropy regarding texture: entropy in this context is a statistical metric to quantify the randomness or lack of repetitive patterns in the image. Higher entropy indicates a more complex or less predictable texture, while lower entropy suggests patterns or regularities in the image. After applying this, we classify the resulting image and obtain the class probability as mentioned before.

Fig. 4 exhibits the feature extractor pipeline of the proposed approach.

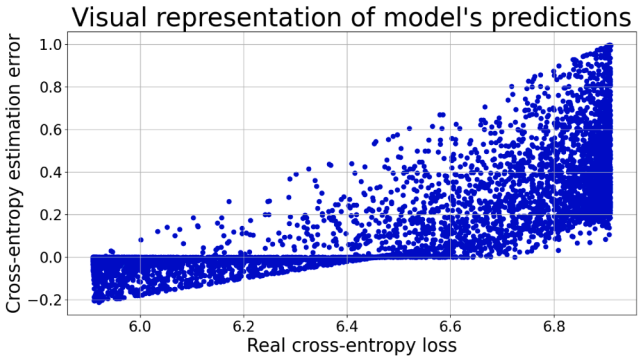

Fig. 5. Visual representation of model's predictions, where the vertical axis represents the cross-entropy estimation error and the horizontal axis represents the real cross-entropy loss.

3.2. Dataset

We have used a subset from the ILSRVC2012 validation image set [1] from IMAGENET (Russakovsky et al., 2015). IMAGENET is a database that contains a collection of labeled images organized according to the WordNet hierarchy. In particular, there are millions of labeled images spanning over a thousand classes, which causes computer vision and deep learning tasks to predominantly use it. Fig. 3 exhibits some examples of images from IMAGENET.

3.3. Setup

First of all, we have created a synthetic dataset from the ILSRVC2012 validation image set by applying different low illumination conditions.

[1] https://image-net.org/challenges/LSVRC/2012

We have conducted tests with different lighting reduction values according to several b values. These configurations of b have been chosen from the results of our previous work (Rodríguez-Rodríguez et al., 2021). The selected configurations are:

- $b = \{1, \ldots, 7\}$, meaning random values of b from 1 to 7.
- $b = \{3, 4, 5\}$, meaning random values of b from 3 to 5.
- $b = 3$.
- $b = 10$.

According to that paper Rodríguez-Rodríguez et al. (2021), $b = 3$ presents the minimum performance difference between the competitors, while $b = 10$ represents the image affected by hard low illumination conditions. Note that higher values of b produce practically black images, while $b = 1$ represents the image without low illumination conditions. Regarding the configuration $b = \{1, \ldots, 7\}$, it represents the usual random brightness conditions. About the configuration $b = \{3, 4, 5\}$, it emulates the typical low illumination conditions.

Once a configuration of b has been selected, for each image from the dataset, a dimmed image is created according to that configuration of b. Then, for each dimmed image, the enhancement algorithms are applied. After that, a classifier predicts the class of each image (the dimmed image and the images generated by the enhancement algorithms). This way, for each dimmed image, we have samples composed of 5 different predictions: the prediction associated with the input dimmed image, and the prediction associated with the image generated by each considered enhancement algorithm (in this work we have considered 4 enhancement algorithms: HE, CLAHE, GC, and LT). In order to have a varied and balanced dataset, we only considered those samples that, for a given dimmed image and its related generated images, had been correctly classified exactly in only one image, while the remaining images had not been classified correctly. By employing this process, we have taken 5000 samples, with 1000 samples for each method.

A 5-fold cross-validation (Refaeilzadeh et al., 2009) over the dataset has been used to test the performance of the considered regressor, by using the coefficient of determination (R^2) and the Root Mean Squared Error (RMSE) as metrics.

In order to determine which regressor among the existing literature would be used in this work, Support Vector Regression (SVR) (Platt, 1999) was chosen due to its good performance in general and its low number of hyperparameters: $kernel$, C, $epsilon$, and $gamma$. For simplicity, we have used the RBF kernel, which is usually the default option[2]. Regarding the remaining parameters, default values are also used in a preliminary way. In terms of performance, it obtained an R^2 of 0.96721 and an RMSE of 0.0395. Then, we used validation curves (Scikit-learn, n.d.) to adjust the hyperparameters. By employing this technique, parameters C and $epsilon$ were updated ($C = 1$ and $epsilon = 0.0001$), and after that, the performance of the regressor is an R^2 of 0.99998 and an RMSE of 0.001.

In Fig. 5, we can observe that the error between the real and predicted cross-entropy loss is very low, with only a few values having slightly higher errors (values more dispersed along the vertical axis).

In the learning curve (Scikit-learn, n.d.) at Fig. 6, we can observe how the model's performance evolves as more samples are incorporated during the training process. It provides insight into the statistical performance impact of the model, evaluated through training data and test data. We can see that the model has a good fit, as the curves corresponding to the training and test sets reach a stable point, and the distance between them is small. This indicates that the model not only performs well on the training data but also effectively generalizes to new data, demonstrating consistency and robustness in its performance.

[2] https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVR.html

Table 1
Results of the cross-entropy loss, probability, and predicted class by AlexNet for each of the proposed image enhancement competitors in dimmed image ILSVRC2012_val_00018473, whose real class is "forklift".

	Cross-entropy loss	Predicted Class	Probability	Correct prediction
No-operation	6.842	tractor	35.036%	No
HE	6.495	forklift	41.346%	Yes
CLAHE	6.745	tractor	27.348%	No
GC	6.902	tricycle	21.627%	No
LT	6.905	tricycle	15.35%	No
Randomly: HE	6.495	forklift	41.346%	Yes
Proposal SVR: HE	6.495	forklift	41.346%	Yes
Proposal GBR: HE	6.494	forklift	41.506%	Yes
Proposal XGBR: HE	6.498	forklift	41.506%	Yes

Table 2
Results of the cross-entropy loss, probability, and predicted class by AlexNet for each of the proposed image enhancement competitors in dimmed image ILSVRC2012_val_00005480, whose real class is "passenger car".

Method	Cross-entropy loss	Predicted Class	Probability	Correct prediction
No-operation	6.862	freight car	15.502%	No
HE	6.264	passenger car	64.477%	Yes
CLAHE	6.64	passenger car	26.863%	Yes
GC	6.82	streetcar	15.137%	No
LT	6.832	fire engine	41.292%	No
Proposal SVR: HE	6.264	passenger car	64.477%	Yes
Proposal GBR: HE	6.314	passenger car	58.995%	Yes
Proposal XGBR: HE	6.315	passenger car	58.995%	Yes

Gradient Boosting Regressor (GBR)[3] (Friedman, 2001) and XGB Regressor (XGBR)[4], an implementation of XGBoost (eXtreme Gradient Boosting) algorithm, Chen and Guestrin (2016) have also been added to the comparison because they are two well-known prominent regressors. Due to the number of parameters of these models, they have been fixed to default values.

3.4. Results

3.4.1. AlexNet

As mentioned in Section 3.3, we have conducted experiments with different lighting reductions in the images. Firstly, we create the training set associated with the lighting reduction we want to test. Subsequently, we perform the test, in which we will use the regression result to estimate which transformation of the input image is most promising, and verify that this estimation serves to improve the classification performance.

A qualitative result is shown in Fig. 7, where an image with decreased illumination and the application of each of the four enhancement algorithms can be observed. The first row details the input image, its predicted classification, and its real class (ground truth), while the remaining images correspond to those generated after the application of an image enhancement algorithm and exhibit their classification prediction. For this image, the results obtained considering the cross-entropy loss and the predictions of AlexNet can be seen in Table 1. As shown, our regressor has succeeded in choosing the most appropriate algorithm for this classification. Another example is depicted in Fig. 8, while their results are shown in Table 2.

In order to study the goodness of our proposal, we have analyzed its performance compared with that yielded by other competing alternatives. The competitors we have considered are as follows:

[3] https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingRegressor.html

[4] https://xgboost.readthedocs.io/en/latest/python/python_api.html#xgboost.XGBRegressor

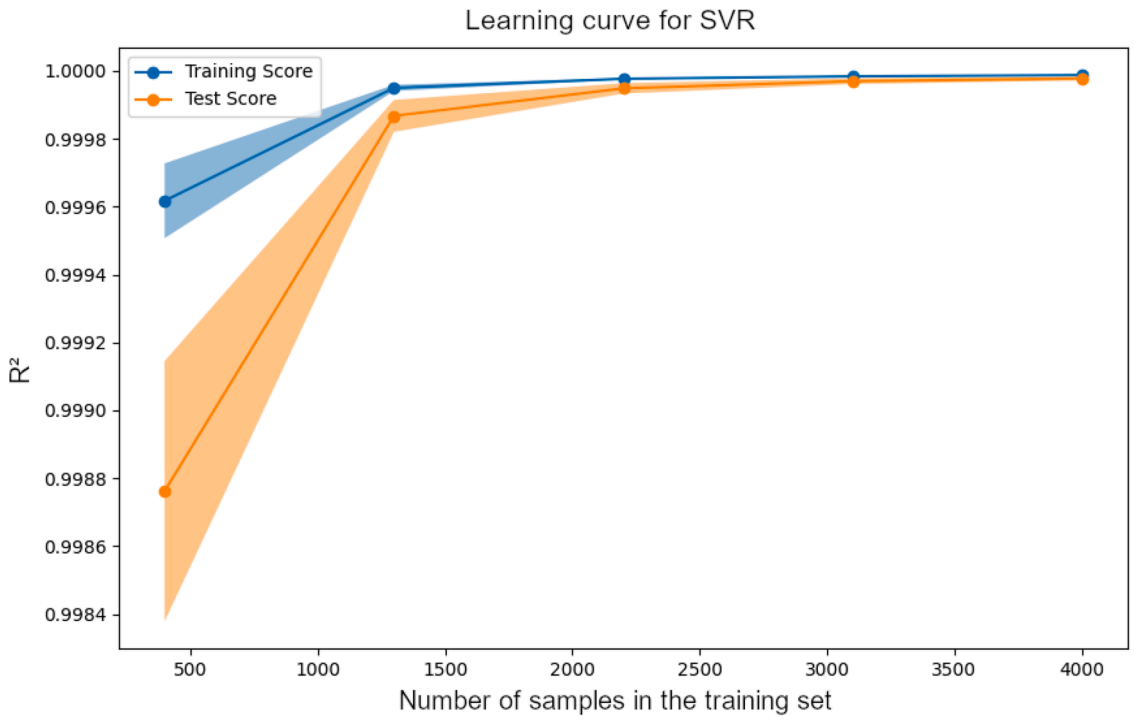

Fig. 6. Learning curve of the model, where how the curve stabilizes around 4000 samples can be observed, while the distance between the test score and the training score is very low.

Table 3
Classification accuracy of AlexNet by alternative, depending on the applied lighting reduction. The lighting reduction configuration is represented in each row, while the competitors are represented by the columns. For each row, the best result is highlighted in **bold** while the second best result is highlighted in *italics*.

Illumination conditions (b)	{1, ..., 7}	{3, 4, 5}	3	10
No operation	0.319 ± 0.0109	0.313 ± 0.0130	0.400 ± 0.0083	0.055 ± 0.0066
HE	0.327 ± 0.0131	0.310 ± 0.0063	0.342 ± 0.0164	0.263 ± 0.0132
CLAHE	0.341 ± 0.0071	0.330 ± 0.0083	0.360 ± 0.0129	0.235 ± 0.0132
GC	0.409 ± 0.01338	0.396 ± 0.0098	0.430 ± 0.0144	0.305 ± 0.0181
LT	0.330 ± 0.0189	0.346 ± 0.0182	0.317 ± 0.0097	0.418 ± 0.0166
Random	0.331 ± 0.0128	0.353 ± 0.0497	0.362 ± 0.0325	0.220 ± 0.0934
Proposal SVR	0.438 ± 0.0141	0.424 ± 0.0068	0.452 ± 0.0089	**0.420 ± 0.0101**
Proposal GBR	*0.440 ± 0.0113*	*0.424 ± 0.0066*	*0.454 ± 0.0202*	0.417 ± 0.0200
Proposal XGBR	**0.441 ± 0.0098**	**0.425 ± 0.0135**	**0.454 ± 0.0172**	*0.418 ± 0.0092*

- Use a CNN classifier with the input image.
- For each enhancement algorithm, use the same classifier with the transformed version of the input image.
- Use the same classifier with a randomly chosen transformed version of the input image.
- Use the same classifier with the transformed version of the input image that minimizes the cross-entropy loss estimated by our regressor.

With each of these competitors, we measure their performance by using well-known metrics, that will help us to understand the behavior of our regressor and establish a fair comparison between the performance yielded by the competitors. Table 3 shows the accuracy of each of the proposed competing alternatives for different lighting reduction configurations, and Table 4 exhibits the F1-score (also known F-measure). The random selection of one of the considered image enhancement algorithms has been also added as a competitor to the comparison.

According to the obtained results, using our regressor proposal to select an image enhancement algorithm to be applied to the input image improves the classification performance of that image in all considered illumination conditions scenarios. It is interesting to see how, when the lighting reduction is greater, the LT algorithm achieves the highest accuracy of the four considered, while the GC algorithm achieves the highest accuracy when the lighting reduction is lower. In most cases, the use of one of the considered enhancement algorithms is better compared with the use of the no-operation algorithm. Regarding the values of the metrics, the use of the GC algorithm offers a better performance than the no-operation method. However, the use of the proposed methodology increases that performance.

Additionally, we will perform the statistical Cochran's Q Test (Conover, 1999; Fahmy & Bellétoile, 2017; Patil, 1975) to verify that the observed differences in classification performance among the evaluated enhancement methods are statistically significant. This non-parametric statistical test allows us to study whether the proportion of successes is different for all the images and verify whether a given number of treatments have identical effects on a set of data. By carrying out Cochran's Q Test, a statistic and a p-value are obtained: if the p-value is less than the statistic, it means that there are differences in effectiveness among the treatments.

The results obtained when performing Cochran's Q test are detailed in Table 5. It can be observed that the statistic is greater than the p-value.

Table 4
Classification F1-score of AlexNet by alternative, depending on the applied lighting reduction. The lighting reduction configuration is represented in each row, while the competitors are represented by the columns. For each row, the best result is highlighted in bold while the second best result is highlighted in italics.

Illumination conditions (b)	{1, ..., 7}	{3, 4, 5}	3	10
No operation	0.319 ± 0.0137	0.313 ± 0.0171	0.400 ± 0.0081	0.055 ± 0.0034
HE	0.327 ± 0.0082	0.310 ± 0.0212	0.342 ± 0.0164	0.263 ± 0.0137
CLAHE	0.341 ± 0.0095	0.330 ± 0.0181	0.360 ± 0.0103	0.235 ± 0.0098
GC	0.409 ± 0.0131	0.396 ± 0.0222	0.430 ± 0.0068	0.305 ± 0.0077
LT	0.330 ± 0.0165	0.346 ± 0.0189	0.317 ± 0.0132	0.418 ± 0.0128
Random	0.328 ± 0.0084	0.349 ± 0.0522	0.354 ± 0.0259	0.167 ± 0.1604
Proposal SVR	*0.439 ± 0.0164*	0.424 ± 0.0165	0.452 ± 0.0085	**0.421 ± 0.0116**
Proposal GBR	0.439 ± 0.0176	*0.424 ± 0.0130*	*0.454 ± 0.0205*	0.417 ± 0.0088
Proposal XGBR	**0.441 ± 0.0099**	**0.426 ± 0.0128**	**0.454 ± 0.0153**	*0.418 ± 0.0113*

Table 5
Results of the statistic and the p-Value of the Cochran's Q test on AlexNet.

Illumination conditions (b)	Statistic Proposal SVR	Statistic Proposal GBR	Statistic Proposal XGBR	p-Value
{1, ..., 7}	2114.294 ± 74.7621	2110.165 ± 107.5590	2117.648 ± 39.6683	~ 0.0
{3, 4, 5}	2162.798 ± 39.3147	2155.694 ± 37.7328	2162.353 ± 65.2444	~ 0.0
3	2039.105 ± 32.6191	2043.656 ± 84.6751	2037.752 ± 62.5290	~ 0.0
10	2737.824 ± 36.8000	2673.720 ± 104.7350	2695.001 ± 84.5296	~ 0.0

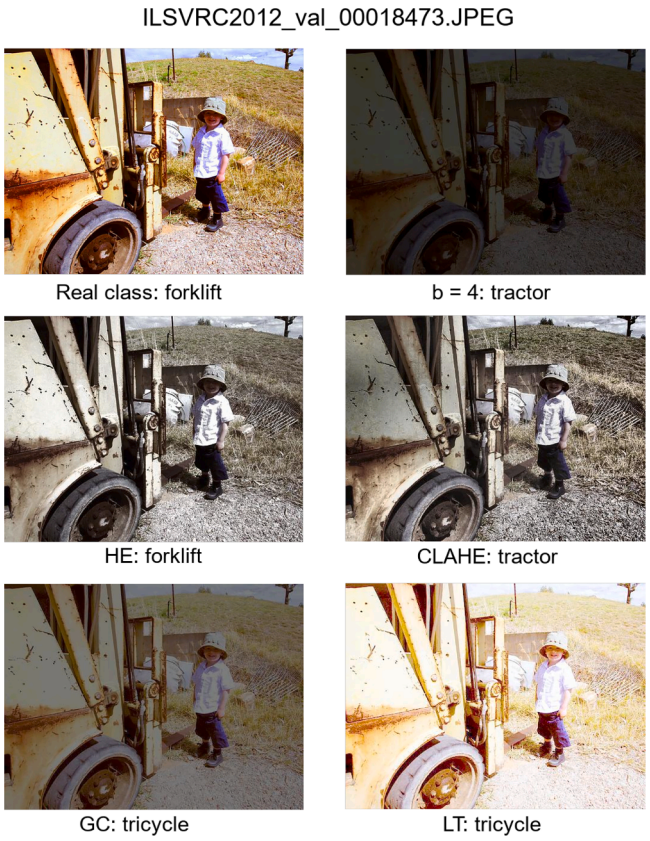

Fig. 7. Comparison between the image enhancement algorithms applied over the dimmed image ILSVRC2012_val_00018473 where the bright scale was set to $b = 4$, and their posterior classification predictions provided by AlexNet. The first row exhibits the raw image with its real class (ground truth) and the dimmed version of that image for the selected value of b and its prediction. The second and third rows exhibit the image after applying the selected image enhancement algorithm over the dimmed image and its prediction.

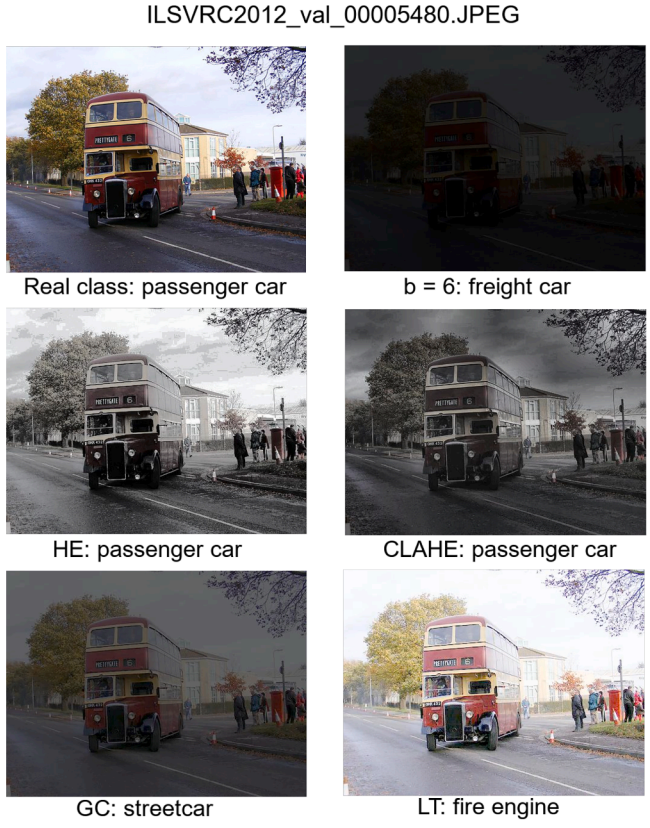

Fig. 8. Comparison between the image enhancement algorithms applied over the dimmed image ILSVRC2012_val_00005480 where bright scale was set to $b = 6$ and their posterior classification predictions provided by AlexNet. The first row exhibits the raw image with its real class (ground truth) and the dimmed version of that image for the selected value of b and its prediction. The second and third rows exhibit the image after applying the selected image enhancement algorithm over the dimmed image and its prediction.

Table 6
Classification accuracy of GoogLeNet by alternative, depending on the applied lighting reduction. The lighting reduction configuration is represented in each row, while the competitors are represented by the columns. For each row, the best result is highlighted in **bold** while the second best result is highlighted in *italics*.

Illumination conditions (b)	{1, ..., 7}	{3, 4, 5}	3	10
No operation	0.598 ± 0.0147	0.606 ± 0.0110	0.645 ± 0.0157	0.371 ± 0.0137
HE	0.636 ± 0.0212	0.631 ± 0.0146	0.342 ± 0.0164	0.570 ± 0.0155
CLAHE	0.639 ± 0.0194	0.630 ± 0.0197	0.648 ± 0.0119	0.572 ± 0.0076
GC	0.650 ± 0.0214	0.642 ± 0.0202	0.650 ± 0.0119	0.587 ± 0.01254
LT	0.608 ± 0.0165	0.606 ± 0.0225	0.594 ± 0.0089	**0.643 ± 0.0094**
Random	0.619 ± 0.0343	0.635 ± 0.0147	0.635 ± 0.0221	0.549 ± 0.1081
Proposal SVR	0.680 ± 0.0178	*0.667 ± 0.0235*	*0.675 ± 0.0156*	0.642 ± 0.0065
Proposal GBR	*0.681 ± 0.0108*	**0.667 ± 0.0175**	0.675 ± 0.0168	0.642 ± 0.0107
Proposal XGBR	**0.681 ± 0.0053**	0.666 ± 0.0096	**0.675 ± 0.0075**	*0.643 ± 0.0165*

Table 7
Results of the statistic and the p-Value of the Cochran's Q test on GoogLeNet.

Illumination conditions (b)	Statistic Proposal SVR	Statistic Proposal GBR	Statistic Proposal XGBR	p-Value
{1, ..., 7}	1131.826 ± 80.1311	1134.441 ± 38.9547	1134.804 ± 23.4257	~ 0.0
{3, 4, 5}	1170.206 ± 87.3804	1165.517 ± 40.3810	1165.614 ± 35.1732	~ 0.0
3	1161.431 ± 42.5365	1163.382 ± 60.4025	1163.750 ± 41.3265	~ 0.0
10	1537.832 ± 42.6984	1539.884 ± 43.4403	1539.799 ± 55.6194	~ 0.0

Table 8
Classification accuracy of ResNet by alternative, depending on the applied lighting reduction. The lighting reduction configuration is represented in each row, while the competitors are represented by the columns. For each row, the best result is highlighted in **bold** while the second best result is highlighted in *italics*.

Illumination conditions (b)	{1, ..., 7}	{3, 4, 5}	3	10
No operation	*0.784 ± 0.0164*	*0.781 ± 0.0127*	**0.797 ± 0.0137**	0.774 ± 0.0087
HE	0.772 ± 0.0160	0.761 ± 0.0120	0.783 ± 0.0065	0.746 ± 0.0127
CLAHE	0.762 ± 0.0185	0.758 ± 0.0121	0.768 ± 0.0116	0.743 ± 0.0100
GC	0.782 ± 0.0201	0.780 ± 0.0154	0.792 ± 0.0107	0.771 ± 0.0090
LT	0.759 ± 0.0143	0.760 ± 0.0113	0.775 ± 0.0106	0.763 ± 0.0092
Random	0.775 ± 0.0172	0.768 ± 0.0090	0.791 ± 0.0105	0.757 ± 0.0204
Proposal SVR	**0.784 ± 0.0149**	0.779 ± 0.0160	0.795 ± 0.0113	*0.775 ± 0.0085*
Proposal GBR	0.783 ± 0.0178	**0.782 ± 0.0104**	0.796 ± 0.0077	0.775 ± 0.0087
Proposal XGBR	0.783 ± 0.0227	0.781 ± 0.0167	*0.796 ± 0.0068*	**0.775 ± 0.012**

Table 9
Results of the statistic and the p-Value of the Cochran's Q test on ResNet.

Illumination conditions (b)	Statistic Proposal SVR	Statistic Proposal GBR	Statistic Proposal XGBR	p-Value
{1, ..., 7}	737.222 ± 61.4752	736.957 ± 51.0689	733.995 ± 89.3801	~ 0.0
{3, 4, 5}	760.273 ± 54.9314	757.523 ± 55.8848	758.908 ± 53.3188	~ 0.0
3	709.853 ± 52.9625	707.508 ± 28.6331	707.493 ± 33.2127	~ 0.0
10	760.002 ± 36.5668	765.687 ± 27.8512	763.027 ± 54.3129	~ 0.0

Therefore, it can be said that there are differences in effectiveness among the treatments. Additionally, the p-value is very low, which means that these differences are highly significant. Therefore, it is validated that the methodology we have proposed provides a statistically significant advantage.

Table 10
Average reading time of the image (in seconds).

Reading the Image
0.0026 ± 0.000144

3.4.2. GoogLeNet and ResNet

In addition, we have conducted experiments using both GoogLeNet and ResNet architectures to assess the generalization of the approach across different deep learning models. We have analyzed the classification accuracy for each alternative depending on the applied lighting reduction, as well as the statistic and p-value of Cochran's Q test for each model.

By conducting the experiments with GoogLeNet in Table 6, we observe that for the lighting conditions b = {1, ..., 7}, b = {3, 4, 5}, and b = 3, our regressor achieves higher accuracy. For b = 10, the LT algorithm performs slightly better, with an improvement of one hundredth. Looking at the metric values, the GC algorithm performs better than the no-operation method. Also, using the proposed method improves the results even more.

Similarly, for GoogLeNet, the statistic once again exceeds the p-value (Table 7), confirming the presence of differences in treatment effectiveness. The very low p-value reinforces the significance of these

Table 11
Average enhancement time using HE, CLAHE, GC, and LT (in seconds).

HE	CLAHE	GC	LT
0.0013 ± 0.000166	0.0025 ± 0.000355	0.0008 ± 0.000127	0.0046 ± 0.000453

Table 12
Average feature extraction time using AlexNet, GoogleNet, and ResNet (in seconds).

AlexNet	GoogleNet	ResNet
1.2707 ± 0.0224	0.9456 ± 0.196	1.3831 ± 0.146

Table 13
Average regression time using SVR, GBR, and XGBR (in seconds).

SVR	GBR	XGBR
$0.0014 \pm 8.01 \times 10^{-6}$	$0.0014 \pm 5.83 \times 10^{-5}$	0.0017 ± 0.000105

differences, supporting the conclusion that our proposed methodology provides a statistically significant improvement.

For ResNet, we can observe that without applying any operation, the neural network is able to achieve good accuracy under all lighting conditions, as reported in Table 8. In this case, our regressor obtains better results for b = 1, …, 7 and b = 10. Looking at the metric values, the no-operation method performs better than the other considered algorithms. Additionally, using the proposed method improves the results in half of the cases.

Although the statistic is lower than that of the other classifiers, it still exceeds the p-value, as shown in Table 9.

In order to evaluate the computational performance of the proposed methodology, we measured the average execution times required at different stages of the pipeline. Specifically, Table 10 reports the average reading time of the image, Table 11 presents the average enhancement times obtained with the four considered methods (Histogram Equalization (HE), Contrast Limited Adaptive Histogram Equalization (CLAHE), Gamma Correction (GC), and Linear Transformation (LT)), Table 12 shows the average feature extraction times when using the three selected CNNs (AlexNet, GoogleNet, and ResNet), and Table 13 provides the average regression times for three regression models (Support Vector Regression (SVR), Gradient Boosting Regression (GBR), and Extreme Gradient Boosting Regression (XGBR)).

3.5. Discussion

As expected, the worse the illumination conditions the worse the classification performance yielded by the classifiers. This effect is especially damaging in the most complex lighting reduction configuration for those architectures with a lower complexity, such as AlexNet or GoogLeNet, where the performance dramatically drops (AlexNet is more affected than GoogLeNet); however, the largest architectures, such as ResNet, practically achieve the same performance as when image brightness is usual. Therefore, under more challenging illumination conditions, it is suggested that the higher the CNN architecture's complexity, the lower the performance degradation.

In the majority of cases, applying one of the evaluated enhancement algorithms yields better results than using no enhancement at all. Therefore, in general, under typical illumination conditions, applying one of the considered enhancement algorithms is recommended, as it will likely outperform the case where no preprocessing is applied. In particular, the GC enhancement algorithm surpasses the remaining algorithms. Additionally, it is remarkable to observe how LT algorithm achieves the highest accuracy of the four considered enhancement algorithms under extremely lighting reductions; however, when the lighting reduction is lower, GC algorithm achieves the highest accuracy.

Although this performance improvement is noticeable, it is even greater when applying the proposed approach. In particular, using the proposal when employing Alexnet and GoogLeNet as classifiers outperforms the accuracy by more than 8 points in terms of accuracy under general conditions, and more than 27 points in images affected by hard low illumination conditions.

In the particular case of ResNet, it, without using an enhancement algorithm, performs similarly (even better) than using it. Thus, linked to the analysis of its behaviour under different illumination conditions, this suggests that the largest CNNs may decently deal with brightness without needing image preprocessing. Compared with the proposal's performance, both are analogous.

Regarding the restrictions of the proposal, the most critical constraint resides in its computational cost: it is directly influenced by the number of enhancement algorithms available to be applied. Therefore, higher values of algorithms result in greater computational time. Additionally, the complexity of each enhancement algorithm further contributes to the overall computational cost.

4. Conclusions

A methodology composed of several image enhancement algorithms and a trained regressor has been proposed in order to improve the classification prediction performance provided by pre-trained CNNs to a given input image, probably affected by low image conditions. The trained regressor analyzes different features of the images to determine how classification performance changes (either improving or declining) between the original image and the enhanced version of that image. These evaluations are then used to choose the most effective image enhancement algorithm for optimizing classification results by the CNN. Experiments were defined considering a wide range of usual normal or low illumination conditions for input images, four well-known image enhancement algorithms (Gamma Correction, Histogram Equalization, Contrast Limited Adaptive Histogram Equalization, and Logarithmic Transformation), and three distinguished pre-trained models (AlexNet, GoogLeNet, and ResNet). The obtained results have demonstrated the goodness of the proposal, and statistical tests have been employed to confirm that the results achieved have substantial importance: in most cases, the image enhancement algorithm indicated by the regressor offered a better classification performance than the remaining methods. This way, given an input image probably affected by low illumination conditions, the proposed methodology might improve the CNN performance by automatically applying the most suitable image enhancement algorithm to the input image.

Funding

This work is partially supported by the Autonomous Government of Andalusia (Spain) under project PPRO-TIC163-G-2023 (TIC163-G-FEDER); also by the Ministry of Science and Innovation of Spain, grant number PID2022-136764OA-I00, project name Automated Detection of Non Lesional Focal Epilepsy by Probabilistic Diffusion Deep Neural Models. It includes funds from the European Regional Development Fund (ERDF). It is also partially supported by the Fundación Unicaja under project PUNI-003_2023, project name Intelligent System to Help the Clinical Diagnosis of Non-Obstructive Coronary Artery Disease in Coronary Angiography, the Instituto de Investigación Biomédica de Málaga y Plataforma en Nanomedicina-IBIMA Plataforma BIONAND under project ATECH-25-02, and the Instituto de Salud Carlos III, project code PI25/02129 (co-financed by the European Union).

CRediT authorship contribution statement

José A. Rodríguez-Rodríguez: Conceptualization, Software, Validation, Investigation, Data curation, Writing – original draft; **Ezequiel López-Rubio:** Methodology, Formal analysis, Writing – review & editing, Visualization, Project administration, Funding acquisition; **Salvador Jiménez-Segura:** Conceptualization, Software, Data curation, Writing – original draft; **Miguel A. Molina-Cabello:** Formal analysis, Investigation, Resources, Writing – review & editing, Supervision, Funding acquisition.

Data availability

Publicly available datasets[5] were analyzed in this study. The source code developed for this work is openly available[6].

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.

Acknowledgments

The authors thankfully acknowledge the computer resources, technical expertise, and assistance provided by the SCBI (Supercomputing and Bioinformatics) Center of the University of Malaga. They also gratefully acknowledge the support of NVIDIA Corporation with the donation of an RTX A6000 GPU with 48Gb. Funding for open access charger: Universidad de Málaga/CBUA. The authors also thankfully acknowledge the grant of the Universidad de Málaga and the Instituto de Investigación Biomédica de Málaga y Plataforma en Nanomedicina-IBIMA Plataforma BIONAND.

References

Bled, C., & Pitié, F. (2023). Pushing the limits of the wiener filter in image denoising. In *2023 IEEE International conference on image processing (ICIP)* (pp. 2590–2594). https://doi.org/10.1109/ICIP49359.2023.10222826

Chen, S., Wang, X., Wei, X. et al. (2025). Deeply understanding features to achieve efficient remote sensing image classification. *Expert Systems with Applications*, (p. 128743). https://doi.org/10.1016/j.eswa.2025.128743

Chen, T., Chen, X., Chen, W., Wang, Z., Heaton, H., Liu, J., & Yin, W. (a). Learning to optimize: A primer and a benchmark. *Journal of Machine Learning Research*, *23* (1), 8562–8620.

Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 785–794). ACM. https://doi.org/10.1145/2939672.2939785

Choi, H., Na, C., Kim, J. et al. (2023). Exploration of lightweight single image denoising with transformers and truly fair training. In *Proceedings of the 2023 ACM international conference on multimedia retrieval* ICMR '23 (p. 452–461). New York, NY, USA: Association for Computing Machinery. https://doi.org/10.1145/3591106.3592265

Conover, W. J. W. J. (1999). Practical nonparametric statistics / W.J. Conover. Wiley series in probability and statistics. Applied probability and statistics section (Third edition. ed.). New York ;: Wiley.

CS231n . Convolutional neural networks for visual recognition. https://cs231n.github.io/convolutional-networks/.

Czako, Z., Sebestyen, G., & Hangan, A. (2021). AutomaticAI – A hybrid approach for automatic artificial intelligence algorithm selection and hyperparameter tuning. *Expert Systems with Applications*, *182*, 115225.

Deng, X., Wang, P., Lian, X. et al. (2022). Nightlab: A dual-level architecture with hardness detection for segmentation at night. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 16938–16948). https://doi.org/10.1109/CVPR52688.2022.01643

Doğan, N., Mayanja, A., & Taşdemir, Ş. (2025). Class-weighted reinforcement learning for skin cancer image classification. *Expert Systems with Applications*, (p. 128426). https://doi.org/10.1016/j.eswa.2025.128426

Drew, P. J., & Monson, J. R. T. (2000). Artificial neural networks. *Surgery*, *127* (1), 3–11. https://doi.org/10.1067/msy.2000.102173

Du, Z., Shi, M., & Deng, J. (2024). Boosting object detection with zero-shot day-night domain adaptation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 12666–12676). https://doi.org/10.1109/CVPR52733.2024.01204

Elad, M., Kawar, B., & Vaksman, G. (2023). Image denoising: The deep learning revolution and beyond–a survey paper. *SIAM Journal on Imaging Sciences*, *16* (3), 1594–1654. https://doi.org/10.1137/23M1545859

Fahmy, T., & Bellétoile, A. (2017). Algorithm 983: Fast computation of the non-asymptotic cochran's Q statistic for heterogeneity detection. *ACM Transactions on Mathematical Software. Association for Computing Machinery*, *44* (2). https://doi.org/10.1145/3095076

Freedman, D. A. (2009). Statistical models: Theory and practice. Cambridge University Press.

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, (pp. 1189–1232). https://doi.org/10.1214/aos/1013203451

Gao, H., Guo, J., Wang, G. et al. (2022). Cross-domain correlation distillation for unsupervised domain adaptation in nighttime semantic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 9913–9923). https://doi.org/10.1109/CVPR52688.2022.00968

Gao, J., Ji, X., Ye, F. et al. (2025). Invariant semantic domain generalization shuffle network for cross-scene hyperspectral image classification. *Expert Systems with Applications*, *273*, 126818. https://doi.org/10.1016/j.eswa.2025.126818

Goodfellow, I., Pouget-Abadie, J., Mirza, M. et al. (2020). Generative adversarial networks. *Communications of the ACM*, *63* (11), 139–144. https://doi.org/10.1145/3422622

Guan, J., Qi, N., Zhu, Q. et al. (2024). UTRCGAN: Uncertainty-driven cycle-consistent generative adversarial network for low-light image enhancement. In *2024 IEEE International conference on image processing (ICIP)* (pp. 1473–1479). IEEE. https://doi.org/10.1109/ICIP51287.2024.10648196

Górriz, J. M., Ramírez, J., Ortíz, A. et al. (2020). Artificial intelligence within the interplay between natural and artificial computation: Advances in data science, trends and applications. *Neurocomputing*, *410*, 237–270. https://doi.org/10.1016/j.neucom.2020.05.078

Han, D., Liu, Q., & Fan, W. (2018). A new image classification method using CNN transfer learning and web data augmentation. *Expert Systems with Applications*, *95*, 43–56. https://doi.org/10.1016/j.eswa.2017.11.028

He, K., Zhang, X., Ren, S. et al. (2016). Deep residual learning for image recognition. In *2016 IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 770–778). https://doi.org/10.1109/CVPR.2016.90

Koo, K.-M., & Cha, E.-Y. (2017). Image recognition performance enhancements using image normalization. *Human-Centric Computing and Information Sciences*, *7*. https://doi.org/10.1186/s13673-017-0114-5

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

Li, Y., Yang, X., Tang, D. et al. (2024). RDTN: Residual densely transformer network for hyperspectral image classification. *Expert Systems with Applications*, *250*, 123939. https://doi.org/10.1016/j.eswa.2024.123939

Lin, C.-Y., Haq, M. A., Chen, J.-H. et al. (2023). Efficient saliency map detection for low-light images based on image gradient. *IEEE Transactions on Circuits and Systems for Video Technology*, *34* (2), 852–865. https://doi.org/10.1109/TCSVT.2023.3292940

Liu, F., & Fan, L. (2025). A review of advancements in low-light image enhancement using deep learning. *Neurocomputing*, *652*, 131052. https://doi.org/10.1016/j.neucom.2025.131052

Liu, H., Jin, F., Zeng, H. et al. (2023). Image enhancement guided object detection in visually degraded scenes. *IEEE Transactions on Neural Networks and Learning Systems*, *35* (10), 14164–14177. https://doi.org/10.1109/TNNLS.2023.3274926

Liu, W., Cai, J., Li, Q. et al. (2024). Learning nighttime semantic segmentation the hard way. *ACM Transactions on Multimedia Computing, Communications and Applications*, *20* (7), 1–23. https://doi.org/10.1145/3650032

Lv, X., Zhang, S., Wang, C. et al. (2024). Fourier priors-guided diffusion for zero-shot joint low-light enhancement and deblurring. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 25378–25388). https://doi.org/10.1109/CVPR52733.2024.02398

Manikpuri, U., & Yadav, Y. (2014). Image enhancement through logarithmic transformation. *International Journal of Innovative Research in Advanced Engineering (IJIRAE)*, . https://www.academia.edu/download/36849730/SPEC1008852.pdf.

Pal, K. K., & Sudeep, K. S. (2016). Preprocessing for image classification by convolutional neural networks. In *2016 IEEE International conference on recent trends in electronics, information & communication technology (RTEICT)* (pp. 1778–1781). https://doi.org/10.1109/RTEICT.2016.7808140

Pandiri, D. N. K., Murugan, R., & Goel, T. (2024). Smart soil image classification system using lightweight convolutional neural network. *Expert Systems with Applications*, *238*, 122185. https://doi.org/10.1016/j.eswa.2023.122185

Paszke, A., Gross, S., Massa, F. et al. (2019). PyTorch: An imperative style, high-performance deep learning library. In *Advances in neural information processing systems 32*. Curran Associates, Inc.

Patel, O., Maravi, P., & Sharma, S. (2013). A comparative study of histogram equalization based image enhancement techniques for brightness preservation and contrast enhancement. *Signal & Image Processing : An International Journal*, *4*. https://doi.org/10.5121/sipij.2013.4502

Patil, K. D. (1975). Cochran's Q test: Exact distribution. *Journal of the American Statistical Association*, *70* (349), 186–189. https://doi.org/10.1080/01621459.1975.10480285

[5] https://image-net.org/challenges/LSVRC/2012

[6] https://github.com/icai-uma/Automated-selection-of-image-enhancement-algorithms-for-classification-with-deep-neural-networks

Peebles, W., & Xie, S. (2023). Scalable diffusion models with transformers. In *2023 IEEE/CVF International conference on computer vision (ICCV)* (pp. 4172–4182). https://doi.org/10.1109/ICCV51070.2023.00387

Phiphitphatphaisit, S., & Surinta, O. (2024). Multi-layer adaptive spatial-temporal feature fusion network for efficient food image recognition. *Expert Systems with Applications*, *255*, 124834. https://doi.org/10.1016/j.eswa.2024.124834

Platt, J. (1999). Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in Large Margin Classifiers*, *10* (3), 61–74.

Poynton, C. (2003). Digital Video and HD: Algorithms and interfaces. Electronics & Electrical. Elsevier Science.

Refaeilzadeh, P., Tang, L., & Liu, H. (2009). Cross-validation. In L. LIU, & M. T. ÖZSU (Eds.), *Encyclopedia of Database Systems* , pp. 532–538). Boston, MA: Springer US. https://doi.org/10.1007/978-0-387-39940-9_565

Reza, A. M. (2004). Realization of the contrast limited adaptive histogram equalization (CLAHE) for real-time image enhancement. *Journal of VLSI Signal Processing Systems for Signal, Image and Video Technology*, *38* (1), 35–44. https://doi.org/10.1023/B:VLSI.0000028532.53893.82

Rodríguez-Rodríguez, J. A., López-Rubio, E., Ángel-Ruiz, J. A., & Molina-Cabello, M. A. (2024). The impact of noise and brightness on object detection methods. *Sensors*, *24* (3), 821. https://doi.org/10.3390/s24030821

Rodríguez-Rodríguez, J. A., Molina-Cabello, M. A., Benítez-Rochel, R., & López-Rubio, E. (2021). The effect of noise and brightness on convolutional deep neural networks. In *International conference on pattern recognition* (pp. 639–654). Springer. https://doi.org/10.1007/978-3-030-68780-9_49

Rodríguez-Rodríguez, J. A., Molina-Cabello, M. A., Benítez-Rochel, R., & López-Rubio, E. (2021). The effect of image enhancement algorithms on convolutional neural networks. In *2020 25th International conference on pattern recognition (ICPR)* (pp. 3084–3089). https://doi.org/10.1109/ICPR48806.2021.9412110

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., & Fei-Fei, L. (2015). ImageNet large scale visual recognition challenge. *International Journal of Computer Vision (IJCV)*, *115* (3), 211–252. https://doi.org/10.1007/s11263-015-0816-y

Russell, S., & Norvig, P. (2009). Artificial intelligence: A modern approach. (3rd ed.). USA: Prentice Hall Press.

Schulz, H., & Behnke, S. (2012). Deep learning. *KI - Künstliche Intelligenz*, *26* (4), 357–363. https://doi.org/10.1007/s13218-012-0198-z

Scikit-learn . Validation curves: Plotting scores to evaluate models. https://scikit-learn.org/stable/modules/learning_curve.html.

She, C., Han, F., Wang, L., Duan, S., & Huang, T. (2024). MPC-Net: Multi-prior collaborative network for low-light image enhancement. *IEEE Transactions on Circuits and Systems for Video Technology*, *34* (10), 10385–10398. https://doi.org/10.1109/TCSVT.2024.3408007

Shendy, R., & Nalepa, J. (2024). Few-shot satellite image classification for bringing deep learning on board OPS-SAT. *Expert Systems with Applications*, *251*, 123984. https://doi.org/10.1016/j.eswa.2024.123984

Szegedy, C., Liu, W., Jia, Y. et al. (2015). Going deeper with convolutions. In *2015 IEEE Conference on computer vision and pattern recognition (CVPR)* (pp. 1–9). Los Alamitos, CA, USA: IEEE Computer Society. https://doi.org/10.1109/CVPR.2015.7298594

Tiantian, W., Xie, L., Zhang, J. et al. (2024). An efficient lightweight network for image denoising using progressive residual and attention mechanism fusion. *Scientific Reports*, *14*, 60139. https://doi.org/10.1038/s41598-024-60139-x

Wang, W., & Gang, J. (2018). Application of convolutional neural network in natural language processing. (pp. 64–70). https://doi.org/10.1109/ICISCAE.2018.8666928

Wang, W., Yang, H., Fu, J. et al. (2024). Zero-reference low-light enhancement via physical quadruple priors. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 26057–26066). https://doi.org/10.1109/CVPR52733.2024.02462

Wen, Y., Xu, P., Li, Z. et al. (2025). An illumination-guided dual attention vision transformer for low-light image enhancement. *Pattern Recognition*, *158*, 111033. https://doi.org/10.1016/j.patcog.2024.111033

Wu, G., Al-qaness, M. A. A., Al-Alimi, D. et al. (2024a). Hyperspectral image classification using graph convolutional network: A comprehensive review. *Expert Systems with Applications*, *257*, 125106. https://doi.org/10.1016/j.eswa.2024.125106

Wu, W., Liu, S., Xia, Y. et al. (2024b). Dual residual attention network for image denoising. *Pattern Recognition*, *149*, 110291. https://doi.org/10.1016/j.patcog.2024.110291

Wu, Y., Wang, G., Liu, S. et al. (2024c). Towards a flexible semantic guided model for single image enhancement and restoration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. https://doi.org/10.1109/TPAMI.2024.3432308

Xu, X., Wang, R., Fu, C.-W. et al. (2022). SNR-aware low-light image enhancement. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 17714–17724). https://doi.org/10.1109/CVPR52688.2022.01719

Yamashita, R., Nishio, M., Do, R. K. G. et al. (2018). Convolutional neural networks: An overview and application in radiology. *Insights into Imaging*, *9* (4), 611–629. https://doi.org/10.1007/s13244-018-0639-9

Yao, Z., Su, J.-N., Fan, G. et al. (2024). Gaca: A gradient-aware and contrastive-adaptive learning framework for low-light image enhancement. *IEEE Transactions on Instrumentation and Measurement*, *73*, 1–14. https://doi.org/10.1109/TIM.2024.3353285

Zhang, F., & O'Donnell, L. J. (2020). Chapter 7 - support vector regression. In A. Mechelli, & S. Vieira (Eds.), *Machine learning* (pp. 123–140). Academic Press. https://doi.org/10.1016/B978-0-12-815739-8.00007-9

Zhang, K., Li, Y., Liang, J. et al. (2023). Practical blind image denoising via swin-conv-UNet and data synthesis. *Machine Intelligence Research*, *20* (6), 822–836. https://doi.org/10.1007/s11633-023-1466-0

Zhang, W., Chen, G., Zhuang, P. et al. (2024). CatNet: Cascaded attention transformer network for marine species image classification. *Expert Systems with Applications*, *256*, 124932. https://doi.org/10.1016/j.eswa.2024.124932

Zhang, Y., Pezeshki, M., Brakel, P., Zhang, S., Bengio, C. L. Y., & Courville, A. (2017). Towards end-to-end speech recognition with deep convolutional neural networks. arXiv preprint arXiv:1701.02720.