A transfer learning approach to image-based intelligent detection of street lighting lamp types and wattages

1st Vitor Berger Bonella
Department of Informatics
Federal University of Espírito Santo
Vitória, Espirito Santo, Brazil
vitor.bonella@edu.ufes.br

2nd Igor Mattos dos Santos Varejão
Department of Informatics
Federal University of Espírito Santo
Vitória, Espirito Santo, Brazil
igor.varejao@edu.ufes.br

***Abstract*—Energy distribution companies in Brazil face the critical challenge of maintaining an accurate database for public lighting networks to ensure precise billing of city councils. Changes in this network often go unreported, underscoring the need for vigilant monitoring and regular updates to minimize commercial losses. This paper introduces new machine learning techniques to enhance the database update process, offering a cost-effective alternative to human-intensive methods for lamp classification. In a departure from previous approaches, the classification task exclusively utilizes image data, employing traditional and deep learning classifiers well-established in the literature. The implementation of nested cross-validation ensures unbiased and reliable experimental results. Notably, the investigation is extended by incorporating features extracted from deep learning architectures. The reported results demonstrate state-of-the-art performance in public lighting lamp classification, showcasing the efficacy of machine learning approaches in optimizing database management processes for energy distribution companies. In order to foster transparency and facilitate the exchange of knowledge within the research community, the code of this work is publicly available at Github**

***Index Terms*—public lighting, machine learning, image classification, nested cross-validation, deep learning, transfer learning, visual transformers**

I. Introduction

Electricity distribution companies in Brazil are grappling with the problem of energy losses occurring in Public Lighting (PL) systems. While these companies provide the power supply, the maintenance of the facilities, such as poles and lamps, is the responsibility of the city council. As there are no energy meters associated with the PL points, the energy consumption is estimated based on the available information in the company's database. However, inaccurate information in the database due to the local government's failure to provide updates on new PL points or changes in PL lamp capacity can lead to significant revenue losses for the companies.

The current inspection technique involves a human inspector personally visiting each street and visually verifying the light poles. This thorough procedure involves inspecting the lamps, registering the changes, and documenting these findings for updating the database. Regrettably, this approach has the inherent disadvantage of being expensive, time-consuming, and inefficient. The huge number of regions requiring inspection increases these difficulties, making an effective and timely examination of the street lamps increasingly difficult. As a result, to solve these restrictions and speed the inspection process, an alternate method that utilizes automation and machine learning techniques is required.

To simplify and speed up the inspectors' task, an electronic device attached to a car roof was developed by [33]. This device registers public lighting points using radiometric sensors and a digital camera. Based on a sampling of real lighting points, it was possible to train classifiers to automatically determine the type and capacity of a lamp using Machine Learning (ML) techniques.

However, radiometric sensors are designed to measure radiation, such as heat or light, emitted by objects. While they may seem like a viable solution for inspecting street lamps, several challenges hinder their practicality. These sensors require precise calibration and maintenance to ensure accurate measurements, which adds to the overall cost and complexity of the inspection process. Moreover, environmental factors like weather conditions and obstructions can interfere with the sensors' ability to accurately detect and measure the lamp's type and wattage. This limitation poses a significant obstacle to achieving reliable and consistent data collection. But, more important than this, the implementation of this device on a production scale proved to be economically unfeasible due to the specific nature of the task and the limited market for which it was intended.

Considering these challenges, alternative approaches that strike a balance between cost, efficiency, and reliability must be explored to address the shortcomings of the current inspection procedure. The implementation of the device would be greatly simplified, becoming feasible to apply at scale if only the digital camera were used. This work investigates whether machine learning techniques using only images would be effective in the task of classifying lamp types and wattages. A wide spectrum of machine learning techniques was used in this study, involving image processing techniques, robust classification techniques, deep learning, and feature selection methods. It focuses especially on the use of transfer learning techniques for image classification to accomplish this task. This is the main contribution of this work, considering the previous work only tried to classify the luminary types [2] or did not exclusively use the images to identify lamp type and wattages [1, 32, 33].

In summary, the primary contributions of this work include:

- Integrating signal processing techniques with robust traditional machine learning methods, relying solely on images of public lighting lamps for lamp type and power classification;
- Exploring the application of deep neural network transfer learning techniques for the same task;

The remainder of this paper is organized as follows: Section 2 provides a contextualization of the problem and some methods have already been applied to it. Section 3 explains the image descriptors extracted from the digital images of the public lighting dataset. Section 4 presents an in-depth look at the techniques used. Section 5 describes the experimental methodology, the dataset, the employed classifiers, and the feature selection process, and Section 6 shows the experiment results and their analysis. Section 7 presents the conclusions and future work.

II. Public Lighting Classification

Public lighting systems have recently become a focal point of research. Mavromatis et al. [24] utilized a large image dataset to monitor whether public lighting lamps correctly switch on and off in accordance with illumination needs. Liu et al. [23] employed multiple regression analysis to assess the perceived quality of street lighting from the perspective of pedestrians. Kanthi and Dilli [18] concentrated on minimizing power consumption in the implementation of a smart street lighting system. It's worth noting that these works are not directly related to the specific problem of public lamp classification.

Previous works have already addressed the problem of correct billing for public lighting. Methods were developed to collect, extract features, and classify data. Following these methods is detailed, and a critical view of their experimental methodologies is presented.

Soares et al. [33] show the problem of public lighting network databases in Brazilian energy distribution companies and the need for updating them. As there are no frequent changes reported to the public lighting provider, it becomes essential to develop a way to register these changes in the databases. A computational methodology was developed to extract information about public lighting points. The development of an electronic system, consisting of hardware and software elements, that can accurately identify the types and wattages of lamps installed on lighting poles was discussed. The system incorporates light sensors, data acquisition devices, positioning sensors, an accelerometer, a digital camera, and a global positioning system (GPS) to collect information about the lighting equipment. Radiometric measurements were performed to analyze the spectrum signature of each lamp and determine its type. The authors built a database, which consists of radiometric signals and images of the lamp taken by the digital camera. The overall setup is shown in Figure 1.

Broetto et al. [2] describe a specific component of the device that employs image processing and recognition methods to identify the luminary type where the public lamp is positioned. The authors anticipate that this approach would aid in the classification of lamp types and wattages. Therefore, the paper addresses the problem of detecting two types of luminaries, and the classification is based only on the image captured by the digital camera. Three sets of image descriptors are extracted from the images: the fourier descriptors [12] [7], the hu descriptors [16], and the haralick descriptors [13]. The evaluated classification techniques that have shown good results were Decision Tree [26], K Nearest Neighbors [6], and Multi Layer Perceptron [28].

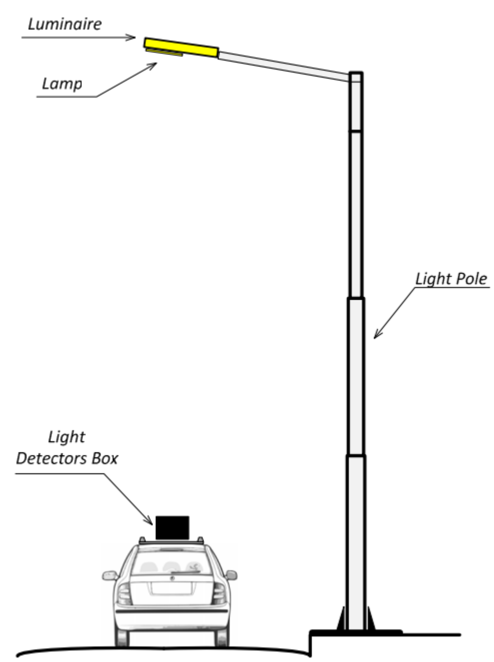

Fig. 1. Device Use Schema. A vertical-cut view of a typical acquisition scene, including transportation vehicles, device cases, and street lighting equipment, extracted from Soares et al. [33].

Feature selection is the process of identifying and retaining the most relevant and informative variables in a dataset. Soares et al. [31] used the image descriptors and radiometric signals together to improve the classifiers performance and observed that feature selection can help improve performance by removing redundant information.

Heterogeneous feature models and feature selection were applied by Broetto and Varejão [1] to improve the classification task. Heterogeneous feature models involve integrating data from diverse sources into the model. As much information as possible was used using radiometric signal data and visual data, which are the three kinds of image descriptors used in [2], plus Red-Green-Blue (RGB) descriptors and the Exchangeable Image File Format (EXIF) Descriptors. The paper showed good results when using feature selection algorithms such as the GRASP metaheuristic [27], sequential forward selection (SFS) [20], and sequential floating forward selection (SFFS) [25]. The classification problem was addressed by considering each class as the lamp type plus its wattage.

A common limitation across the mentioned previous works is the reliance on a single round of k-fold cross-validation. While cross-validation is often employed for assessing model performance, it does have certain limitations. One drawback of k-fold cross-validation is that it does not provide an unbiased estimate of the model's performance on unseen data. This is because the classification model hyperparameters are chosen considering the testing dataset, potentially leading to overoptimistic performance estimates.

To address the issue of overestimation, Silva et al. [29] conducted experiments utilizing repeated nested cross-validation on the same database employed in previous studies. This database comprises radiometric data and three types of image descriptors. In addition to the machine learning classifiers utilized in prior research, the authors incorporated more robust algorithms, such as Support Vector Machine, XGBoost, and Random Forest. Their findings indicate that models employing these robust algorithms can achieve a final mean accuracy of 0.86. They also applied a convolutional neural network classifier using the same data as the other classifiers but obtained an accuracy of 0.62, significantly lower than the other classifiers.

This work stands out significantly from previous studies as it exclusively relies on image data for the classification of public lighting lamp types and wattages. Moreover, the novel approach of employing pre-trained image classification deep neural networks for this specific task has not been explored in any prior research.

III. Image Descriptors

This section presents the theoretical basis for extracting numerical features from an image that represent it in a multidimensional flattened feature space, i.e., transforming data described in two dimensions into just one. There are various ways of making this transformation, and the ones used in this work will be presented in two approaches: mathematical descriptors and descriptors extracted from deep convolutional networks with transfer learning between domains.

A. Mathematical Descriptors

The most important features of an image may be captured by image descriptors, which are critical in computer vision and image processing applications. These descriptors work as a condensed summary of the essential traits and qualities of the visual data, facilitating quick analysis, comparison, and identification. Image descriptors enable a variety of applications, including object identification, image classification, content-based image retrieval, and image synthesis, by getting important information from pictures, such as shape, texture, color, or spatial connections. In this work, Fourier descriptors, Hu moments, and Haralick textural descriptors were used to extract features of the images of the public lighting dataset.

1) Fourier Descriptors: Fourier descriptors [12] are a set of shape descriptors based on the Fourier transform and are widely used in signal processing. They capture the shape information of an object by analyzing its boundary. The Fourier descriptors represent the object's boundary as a combination of sine and cosine functions of different frequencies. The low-frequency components capture the overall shape of the object, while the high-frequency components capture finer details.

In this work, 512 points were used to define an object's boundary. After applying the transform, the 10 first normalized Fourier coefficients were considered.

2) Hu Descriptors: Hu descriptors [16] are the only methods to extract image static moments. In a 2-D image, these descriptors are numerical values that provide a brief spatial distribution of the image points. Hu moments are particularly useful for image recognition and pattern-matching tasks.

3) Haralick Textural Descriptors: Haralick textural descriptors [13] are a set of statistical measures used to quantify the texture of an image. They capture information about the spatial distribution of pixel intensities and can be used to characterize various textures, such as smooth, rough, or textured surfaces. The Haralick descriptors are based on the gray-level co-occurrence matrix (GLCM) [35] [4], which calculates the joint probability distribution of pixel intensities at different spatial offsets.

The Haralick descriptors measure the local variations in pixel intensities. In this paper, six textural descriptors were used. Energy, Entropy , Constrast , Probabilistic Maximum (ProbMax), and Correlation.

By extracting and analyzing these Haralick descriptors, one can characterize and compare textures in images for various applications such as image classification, segmentation, and retrieval.

B. Deep Learning Descriptors

Deep learning has emerged as a groundbreaking paradigm in artificial intelligence (AI) research, revolutionizing the field of computer vision through its ability to extract high-level features from raw data. This section provides an introduction to deep learning, with a specific focus on Convolutional Neural Networks (CNNs) for extracting image features. CNNs have proven to be remarkably effective in modeling and understanding visual data, allowing for significant advancements in tasks such as image recognition, object detection, and scene understanding[21]. The inception of deep learning in computer vision has been a pivotal moment, as it has enabled the development of state-of-the-art image descriptors that outperform traditional feature extraction methods[22]. This section delves into the fundamentals of deep learning, highlights the key architectural components of CNNs, and discusses their application in the context of image feature extraction[14].

1) Deep Learning as a Feature Extractor: Fundamentally, deep learning leverages neural networks with multiple hidden layers to automatically learn hierarchical representations from data, making it exceptionally adept at modeling complex relationships within images. CNNs, a type of deep neural network, are particularly designed to excel in tasks related to image analysis. These networks consist of several essential components, including convolutional layers, pooling layers, and fully connected layers. Convolutional layers perform local feature extraction through the application of convolutional filters, capturing various patterns within an image[21]. Pooling layers, on the other hand, reduce spatial dimensions while preserving essential information, aiding in the creation of translation-invariant features. The fully connected layers at the end of the network facilitate high-level feature combinations and decision-making. This architecture, with its unique ability to capture both low-level and high-level features, serves as a powerful tool for image feature extraction.

In the context of image feature extraction, CNNs have demonstrated their prowess in numerous applications. They are widely employed in image classification tasks, where they automatically extract discriminative features from raw images and subsequently classify them into predefined categories.

The capability of CNNs to automatically learn and extract relevant image features has significantly outperformed traditional methods of feature extraction. Their versatility, adaptability, and ability to model intricate visual data have solidified their position as indispensable tools in computer vision tasks, underscoring the transformative impact of deep learning in the realm of image feature extraction.

When embarking on a task of image feature extraction, one fundamental decision is whether to create a new CNN architecture tailored to the specific requirements of the task or to leverage a well-known, pre-existing architecture. Both approaches have their merits and are valuable options for practitioners in the field of artificial intelligence.

Creating a new CNN architecture offers the advantage of tailored design. This approach allows researchers to design a network architecture from the ground up, aligning it precisely with the specific characteristics and requirements of the task at hand. Custom architectures provide the flexibility to experiment with novel architectural components and hyperparameters, potentially leading to a more efficient and task-specific solution. This flexibility is especially beneficial when working with unconventional data types or addressing unique challenges.

On the other hand, leveraging a well-known architecture, such as ResNet[14], Inception[34], VGG[30] or ViT[9], presents several compelling advantages. Firstly, well-known architectures have undergone extensive research and validation, making them highly reliable and robust. They have often proven their effectiveness in a wide range of tasks and are well optimized. Additionally, by using established architectures, practitioners can tap into the wealth of pre-trained models, which can significantly expedite the process of model development and improve performance. Pre-trained models save time and computational resources and often provide competitive results, making them a practical choice for many image feature extraction tasks.

Ultimately, the choice between creating a new CNN architecture and leveraging a well-known one depends on the specific project's requirements, resources, and objectives. While custom architectures allow for tailored solutions, the use of established architectures offers the advantages of reliability, efficiency, and rapid deployment, particularly in scenarios where resources are constrained or time is of the essence.

2) Transfer Learning: One of the remarkable attributes of convolutional neural networks (CNNs) is their ability to generalize and adapt to new data, even without having found it before. This capacity is leveraged through a concept known as transfer learning, a technique that has proven to be invaluable in the realm of image feature extraction.

Transfer learning in the context of CNNs is based on the premise that a pre-trained network, initially developed for a specific task, can be repurposed to tackle a different but related task effectively[36]. The process involves fine-tuning the network on a new dataset or task by adjusting its weights and parameters.

Training a pre-existing architecture from scratch involves reinitializing the entire network and fully training it with the new dataset. While this approach provides the flexibility to adapt the model to the specific task, it typically requires a significant amount of data and computational resources, often making it less practical compared to the advantages offered by pre-trained models. Nevertheless, in scenarios where the existing architectures do not align well with the task at hand, starting from scratch may be a viable option.

One transfer learning strategy involves utilizing the pretrained convolutional network without adjusting weights, solely to extract the set of features to be employed by classifiers. In this work, this strategy is referred to as "pretrained feature extraction."

An alternative strategy is to freeze the convolutional layers of a pre-trained model while retraining only the fully connected layers, in this work called 'partial fine-tuning". This approach is particularly beneficial when the pre-trained model has already learned generic, low-level features that are relevant to the new task. By keeping the convolutional layers fixed, practitioners save considerable computational resources and time while still benefiting from the established knowledge within the network. This method is often preferred when the lower-level features need not be modified significantly.

"Full fine-tuning" constitutes the ultimate strategy, involving the adaptation of both convolutional and fully connected layers to the new task while utilizing pre-trained weights. Despite its higher computational demands compared to freezing, fine-tuning provides practitioners with the flexibility to tailor the architecture more closely to the specific task requirements. This process is particularly advantageous when the lower-level features necessitate adjustment or when the task significantly diverges from the original purpose of the pre-trained model. However, it's crucial to note that fine-tuning also demands a substantial amount of data to effectively recalibrate the model's parameters and prevent overfitting, highlighting the importance of ample and diverse datasets for successful fine-tuning endeavors.

In the context of image feature extraction, transfer learning demonstrates the adaptability and efficiency of CNNs. It empowers practitioners to harness the collective knowledge of pre-trained networks[21], facilitating the extraction of meaningful image features with unparalleled precision.

IV. Image Based Classification of Public Lamps Types and Wattages

This section presents the different machine learning approaches used in this work as well as justifying the choices made for the subsequent experiments. The idea behind the use of various feature sources and their individual use is presented, along with the integration of features from multiple sources to optimize image feature extraction and classification.

Figure 2 illustrates the research investigation process. It begins by assessing the performance of mathematical image descriptors, including Hu moments, Haralick textures, and Fourier descriptors. These descriptors serve as the initial benchmark, providing a baseline result for subsequent evaluations.

To establish the efficacy of these mathematical descriptors, they are subjected to rigorous testing with various machine learning classifiers and feature selection techniques. The objective is to identify the most effective combination of classifiers and feature selection methods that yields the highest accuracy in image feature extraction.

Once the baseline has been established, the investigation proceeds to the second phase of the process, where the focus shifts towards leveraging deep learning techniques for feature extraction. Features are extracted from images using pre-trained deep learning architectures, which have been proven to excel in capturing intricate visual patterns.

To ensure a thorough and reliable evaluation, the best-performing classifiers identified in the initial phase are employed in conjunction with the deep learning features.

A. *Mathematical Image Descriptors with Traditional ML*

The utilization of mathematical image descriptors in combination with machine learning (ML) classifiers serves as a baseline in the evaluation of image feature extraction techniques. The main idea consists of employing a diverse set of mathematical image descriptors, including Hu moments, Haralick textures, and Fourier descriptors. These descriptors encapsulate distinct aspects of image content, such as shape, texture, and frequency domain information. By incorporating these diverse descriptors into ML classifiers, the goal is to assess how effectively the classifiers capture and model the underlying features of different image classes.

The next step involves applying feature selection to enhance the results obtained from the classifiers. Feature selection aims to identify the most discriminative and relevant features from the mathematical descriptors. This process optimizes the classifiers performance by reducing dimensionality and focusing on the most informative aspects of the data.

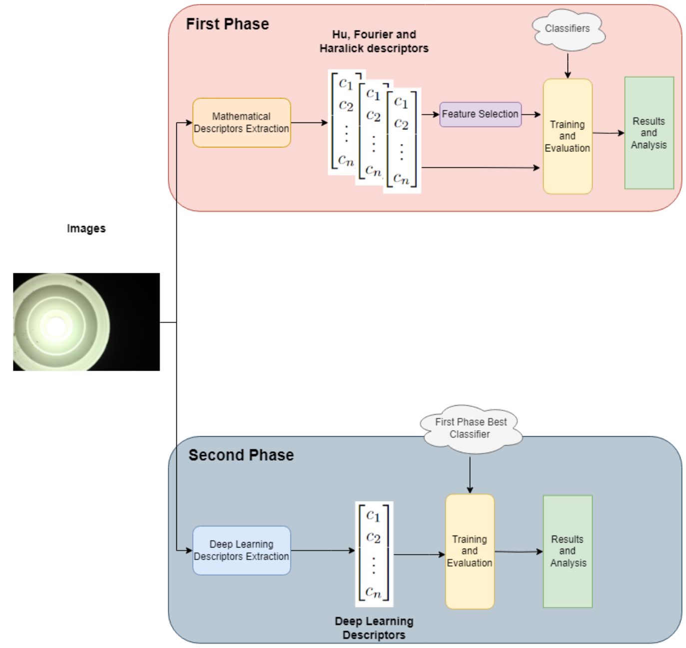

Fig. 2. Research and Investigation Process. The first phase investigates traditional approaches. The second phase investigates deep learning approaches. The third phase investigates the integration of both.

The following two subsections summarize the classifier models utilized and the employed feature selection methods.

1) Classifiers: Several classifiers were employed to perform the analysis and classification tasks on the experimental data. These classifiers included K-Nearest Neighbors (KNN) [6], Decision Tree (DT) [26], Random Forest (RF) [15], Support Vector Machine (SVM) [5], LightGBM (LGBM) [19], XGBoost (XGB) [3], and Multilayer Perceptron (MLP) [28].

KNN is a non-parametric classification algorithm that assigns a data point to the class most commonly found among its k nearest neighbors. It is based on the assumption that similar data points tend to belong to the same class.

The DT classifier builds a tree-like model by recursively splitting the data based on the feature values. Each internal node represents a feature test, and the leaf nodes correspond to the class labels. It is a powerful and interpretable algorithm for classification tasks.

RF is an ensemble method that combines multiple decision trees to create a more robust and accurate model. It works by training each tree on a random subset of the data and features and then averaging their predictions to make the final classification decision.

SVM is a popular algorithm for both classification and regression tasks. It constructs a hyperplane or set of hyperplanes in a high-dimensional space that maximally separates the different classes. SVM is particularly effective when dealing with complex data distributions and non-linear decision boundaries.

LGBM and XGB are gradient-boosting frameworks that have gained significant popularity in recent years. They are based on the boosting technique, where weak classifiers are iteratively combined to form a strong classifier. LGBM and XGB provide efficient implementations of gradient boosting and have shown superior performance in various machine learning tasks.

MLP is a type of artificial neural network with multiple layers of interconnected nodes, known as neurons. It can be used for classification tasks by adjusting the weights and biases of the neurons through a process called backpropagation. MLP is capable of modeling complex relationships between input features and class labels.

The selected classifiers are compared with the experimental data and assessed for their effectiveness in achieving accurate and reliable classification results. The choice of classifiers enables exploration of different approaches, leveraging their unique characteristics for the classification task.

2) Feature Selection: Different feature selection methods were applied in order to identify the most relevant subset of features for each classifier model. By lowering the computational complexity and removing redundant or noisy features, the classifier's performance may be significantly improved.

The first investigated feature selection method consists of manually dividing the image descriptors into subgroups. The three subgroups of descriptors were the Hu descriptors, the Haralick descriptors, and the Fourier descriptors. Each subgroup feature set was examined independently.

The second method is a filter approach using the statistical analysis of variance (ANOVA) [10]. ANOVA to estimate the correlation between each feature and the class labels. The filter consists of selecting the 10 best features that demonstrated a notable contrast between the classes by taking into account the F-value and p-value.

A feature selection technique based on wrappers called sequential forward selection (SFS) [20] is also investigated. It successively adds features taken from the feature set. It checks the classifier's performance using various feature combinations and chooses the subset that produces the best results. In order to choose the best feature subset, SFS considers the classifiers accuracy.

B. Deep Learning Descriptors

Given the success of deep neural networks in feature extraction and image classification, this work extends its exploration to deep learning descriptors, focusing on well-known and renowned architectures of deep neural networks. The selected architectures for study include AlexNet[21], ResNet50[14], ResNet101[14], DenseNet121[17], VGG19[30], Inception V3[34] and Vision Transformer[9]. In the case of the deep learning descriptors, the output layer of each of these networks is removed, and the neurons in the layer immediately before it are used as new image descriptor features.

AlexNet comprises eight layers, including five convolutional and three fully connected layers. AlexNet demonstrated the power of deep convolutional neural networks (CNNs) in image classification tasks. It introduced key innovations, such as the rectified linear unit (ReLU) activation function and local response normalization.

Addressing the challenge of vanishing gradients in deep networks, ResNet50 and ResNet101 are Residual Network (ResNet) variants with 50 and 101 layers, respectively. The breakthrough concept of residual learning involves using shortcut connections to skip one or more layers, enabling the network to learn residual functions. This architecture facilitates the training of extremely deep networks, leading to improved accuracy and convergence.

DenseNet diverges from traditional architectures by establishing dense connections between layers. In DenseNet, each layer receives input from all preceding layers, fostering feature reuse and enhancing model compactness. DenseNet121, with 121 layers, offers a robust solution that mitigates vanishing gradient issues and encourages feature propagation across the network.

The Visual Geometry Group (VGG) architecture is characterized by its simplicity and uniformity. VGG19, a variant of the VGG architecture, consists of 19 layers, including 16 convolutional layers and three fully connected layers. VGG's straightforward design, with small receptive fields and convolutional layers of the same size, facilitates an easy-to-understand and highly transferable network architecture.

Inception V3 represents the Inception architecture's third iteration. Known for its inception modules, which use filters of multiple sizes within the same layer, Inception V3 excels at capturing diverse and multi-scale features. The architecture employs factorized convolutions to optimize computational efficiency, resulting in a powerful and parameter-efficient model for image classification and feature extraction tasks.

Self-attention-based architectures, particularly Transformers, have become the preferred model in natural language processing (NLP). Inspired by the successful scaling of Transformers in NLP, the Vision Transformer (ViT) is developed by directly applying a standard transformer to images. The ViT demonstrates competitive or superior performance compared to the state-of-the-art (SOTA) on various image recognition benchmarks, especially when trained on extensive datasets ranging from 14 million to 300 million images.

From all the architectures chosen, the ViT model has reached the SOTA in various benchmarks. Given that, a greater effort was invested in this architecture. Despite its impressive performance, it excels mainly with larger datasets, yielding more modest accuracies with mid-sized datasets. This behavior mainly occurs because the Transformer architecture has made it possible to train models of unprecedented size. The ViT model has made the fewest alterations on the original transformer architecture, where its gist are focused on dividing the image into patches and use the sequence of linear embeddings of these patches as input for a Transformer as shown in Figure 3. In this process, image patches are treated similarly to tokens (words) in a natural language processing (NLP) application.

The default weights found in these pre-trained networks were determined using the ImageNet dataset [8]. ImageNet is an extensive and diverse dataset that has a vast array of object categories, encompassing over a million images. ImageNet's richness lies in its diversity, featuring images from various

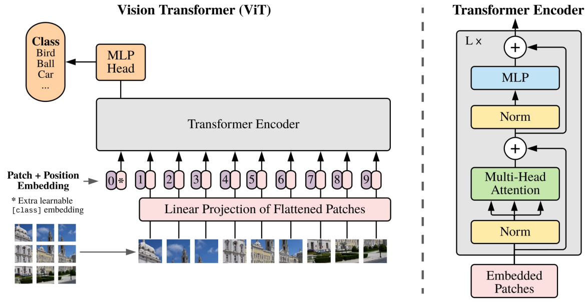

Fig. 3. The model overview of ViT involves splitting an image into fixed-size patches, linearly embedding each patch, adding position embeddings, and feeding the resulting sequence of vectors into a standard Transformer encoder. For classification purposes, the standard approach includes adding an extra learnable "classification token" to the sequence. Extracted from [9]

domains, scenes, and perspectives. The dataset's size and diversity enabled these deep learning architectures to learn intricate and generalized features, facilitating their success in subsequent image recognition tasks outside of the ImageNet domain.

Following the extraction of features from these pre-trained deep learning architectures, the evaluation is extended by employing the best-performing classifier identified in the investigation's first phase, where mathematical image descriptors were utilized.

To further refine and validate the results, this work also takes a step beyond feature extraction and delves into fine-tuning. The best-performing deep learning network, identified through initial evaluations, and the ViT architecture are fine-tuned to adapt to the specific characteristics of the target dataset. Fine-tuning enables the model to specialize in the nuances of the given data, potentially improving its performance and generalization capabilities.

However, it's worth noting that fine-tuning is a delicate process, particularly when confronted with limited data. The lack of sufficient data for fine-tuning may pose challenges, potentially leading to overfitting or suboptimal convergence. Consequently, this limitation is carefully considered, and its potential impact on the fine-tuning process is duly acknowledged.

In summary, the exploration of deep learning descriptors involves a meticulous evaluation of well-known architectures, rigorous comparisons with mathematical descriptors, fine-tuning for dataset adaptation.

V. Experimental Methodology

This section describes the dataset used in the experiments, the resampling strategy, the evaluation metrics and statistical tests employed, and specifies the conducted experiments.

A. Public Lighting Image Dataset

For the experiments, an image dataset of streetlights was collected and classified by the human inspectors. This dataset has 297 images. Each of them belongs to one of nine classes. Each class is a combination of lamp type (sodium, mercury, and metal) and wattage (values in the range of 70W to 400W).

Table I shows the class distribution of the data. One can observe that the base is almost balanced with respect to the classes. The less represented class has 21 instances, and the more represented class has 49 instances.

Figure 4 shows samples of digital images of the different lamp classes. One may note that the lamps may be placed in different kinds of luminaires, and the image of the lamp is not necessarily in the same direction. According to their type, they also have different colors (white and yellow).

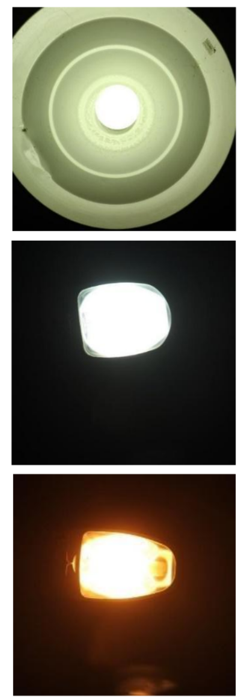

Fig. 4. Samples of Images of the PL Dataset.

TABLE I
PL Dataset Class Distribution.

Class (Lamp Type)	Instances	Distribution
Sodium70W	30	10.10%
Sodium100W	32	10.77%
Sodium150W	35	11.78%
Sodium250W	33	11.11%
Sodium400W	37	12.46%
Mercury125W	21	7.07%
Metal150W	23	7.74%
Metal250W	49	16.50%
Metal400W	37	12.46%

B. Resampling Strategy

In order to avoid the problem of over-optimistic performance estimation, this work adopts repeated nested cross-validation as a resampling strategy. This approach involves performing multiple rounds of nested training, validation, and testing where the data is split into different sets at each iteration. Nested cross-validation incorporates an inner loop for hyperparameter tuning, ensuring a more accurate evaluation of the model's performance. In this way, no aspect of the model training is influenced by the occasional testing dataset. Moreover, many rounds of cross-validation enable more appropriate application of statistical hypothesis tests. Figure 5 shows an example of division for nested cross-validation using 4 folds in the outer loop and 2 folds in the inner loop.

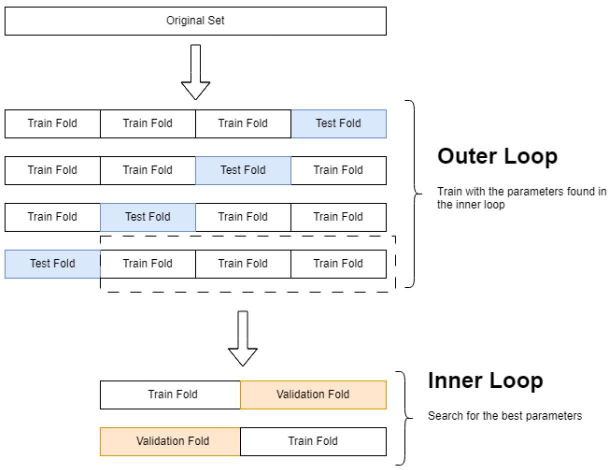

Fig. 5. Four Fold Nested Cross Validation Example.

Experiments were performed with ten folds in the outer loop. In each round, nine folds were used for training and one fold for testing in the outer loop. In the inner loop, the nine outer folds are used to perform a grid search with four folds to find the best hyperparameters for the classifiers. After the best hyperparameters for the classifiers have been found, the best set is used in the test set to measure the performance of the classifier. As a way to reduce the effects of an eventual ill-selected sample division in folds, this process is repeated three times, each time with a different division of the samples in the folds.

C. Evaluation Metric and Statistical Tests

Given that the dataset does not exhibit significant imbalances among its classes, the metric of choice for evaluating classifier performance is accuracy. This decision is grounded in the suitability of accuracy as a comprehensive measure in scenarios where class distribution disparities are not pronounced. By prioritizing accuracy, the evaluation process aims to provide an overall assessment of the classifiers' ability to correctly classify instances across all classes, offering a holistic perspective on their performance.

D. Experiments Description

Before effectively evaluating the methods, a preprocessing step is required to extract the mathematical and deep learning descriptors from the images, thus forming a new dataset composed of the descriptors described in Section III. Each image in the dataset is used to extract the values of the mathematical descriptors. These values are inserted into the corresponding entry of the new dataset. Similarly, each image in the dataset serves as input to each of the pretrained deep neural networks, generating the respective values in the image descriptor layer, which are then assigned to the corresponding entry in the descriptor dataset. This process is illustrated in figure 6.

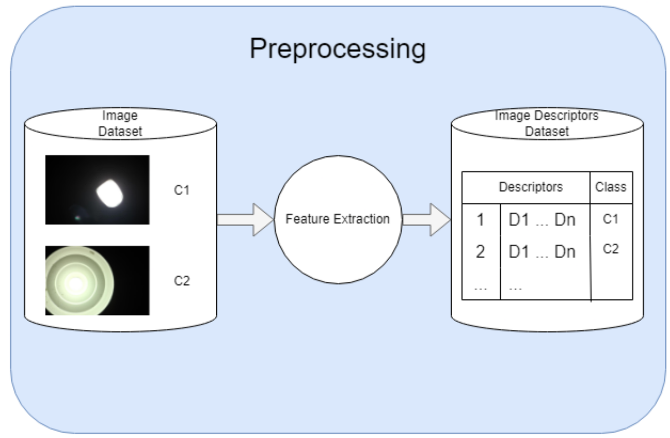

Fig. 6. Image Preprocessing. From each lamp image in the dataset, mathematical descriptors and deep learning descriptors are extracted.

Figure 7 shows an overview of the entire investigation process. Following the preprocessing step, the descriptor dataset becomes the input for feature selection. In this step, only the features necessary for the model being assessed are retained from the dataset. For instance, if the model employs Haralick mathematical descriptors with a traditional machine learning classifier, only the Haralick descriptors are utilized in the repeated nested cross-validation. Analogously, if evaluating AlexNet, only the deep learning descriptors obtained from this pretrained network are used in the repeated nested cross-validation. Upon completing the evaluation step, 30 results (3 repetitions of 10 test folds) are obtained and analyzed.

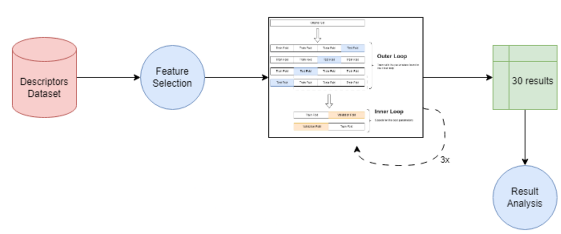

Fig. 7. Overview of the Investigation Process. For each machine learning model, there is a feature selection step followed by its experimental evaluation. The results obtained by each machine learning model are compared and analyzed.

1) First Phase: This experiment only uses mathematical descriptors. Three types of feature selection methods are employed. Each selected feature subset is used in the training and evaluation of all traditional machine learning classifiers indicated in section IV-A1. Therefore, results are obtained for each combination of feature subsets and machine learning classifiers.

Figure 8 illustrates the entire experimentation process of this phase. First, all classifiers are trained and evaluated using the union of the Fourier, Haralick, and Hu descriptors. Then, a handcrafted feature selection is performed. Each group of descriptors forms a feature subset by itself. All classifiers are trained and evaluated using each feature subset. Next, a filter feature selection method is used together with the ANOVA statistic. The 10 best-evaluated features compound the feature subset. Once again, all classifiers are evaluated with this feature subset. At last, a wrapper feature selection method is executed using the SFS algorithm and the KNN classifier. All classifiers are trained and evaluated with the selected feature subset. After all, the best-evaluated combination of feature subset and classifier is chosen for comparison with the next phase of experiments.

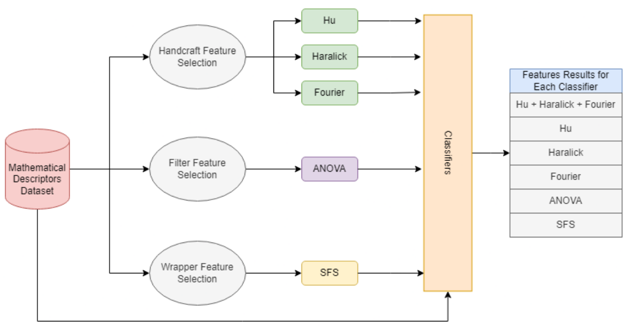

Fig. 8. First Phase Experiments Description. Three different feature selection methods and seven classifier methods were combined and evaluated.

2) Second Phase: In this phase, the focus shifted to evaluating features extracted from the deep learning architectures mentioned in Section IV-B. Three different approaches are investigated: pretrained feature extraction, partial fine-tuning, and full fine-tuning. Figure 9 illustrates the pretrained feature extraction experiment. To streamline the experiments, only the best-performing classifier from the first phase experiment was evaluated.

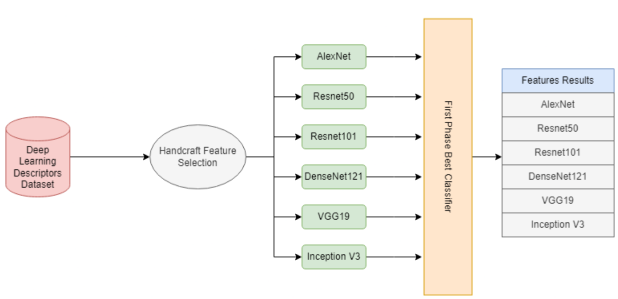

Fig. 9. Second Phase Experiment Description. Pre-trained feature extraction approach. Each deep neural network extracts features to compose the feature vector inputted to the best-performing classifier from the first phase.

Table II indicates the number of features extracted for each deep neural network. Unlike other networks, the feature vector extracted from the ViT is derived from the first output embedding. The decision to forego testing all features together is driven by practical considerations. The high computational cost and the challenges posed by the curse of dimensionality prompted a strategic choice to assess features separately based on their originating architectures. In addition, only two models underwent fine-tuning. The best-performing architecture was trained using both partial fine-tuning and full fine-tuning. However, due to hardware limitations, the ViT, selected to assess the nuances of the transformer architecture, underwent only partial fine-tuning.

TABLE II
ARCHITECTURES FEATURE VECTOR SIZE

Architecture	Feature Vector Size
AlexNet	4096
ResNet50	2048
ResNet101	2048
DenseNet121	1024
VGG19	4096
InceptionV3	2048
ViT	768

VI. Results and Analysis

The section presents and analyzes the results of the three experimental phases, concluding with a discussion about the main results.

A. First Phase

Table III presents the mean accuracy achieved by each classifier on each feature subset. The best result for each feature subset is highlighted in bold. The findings indicate that RF, SVM, and MLP consistently performed well on all feature subsets, with each achieving the highest mean accuracy in two of the feature subsets. It is noteworthy to note that the XGB and LGBM methods showed relatively poor results overall, comparable to the KNN and DT methods. Analyzing the feature subsets, one may see that Hu and Fourier descriptors had a bad performance regardless of the combined classifier. The feature set selected by the SFS algorithm did not perform as poorly, but the other three feature selection methods generally yielded better results. The combination of the Haralick subset with the SVM classifier produced the overall best mean accuracy.

TABLE III
CLASSIFIERS MEAN ACCURACY FOR EACH FEATURE SUBSET. BEST RESULTS ARE HIGHLIGHTED IN BOLD.

Feature Set	KNN	DT	RF	SVM	LGBM	XGB	MLP
Hu	0.39	0.37	**0.49**	0.46	0.37	0.47	0.48
Fourier	0.42	0.32	0.46	**0.49**	0.28	0.45	0.46
Haralick	0.55	0.53	0.60	**0.71**	0.49	0.60	0.65
All	0.55	0.53	**0.66**	0.65	0.52	0.65	0.66
ANOVA	0.62	0.54	0.64	0.65	0.51	0.62	**0.66**
SFS	0.52	0.47	0.58	0.56	0.48	0.58	**0.59**

Table IV displays the mean, standard deviation, and the lower and upper bounds of the confidence interval at a 95% significance level for the classifiers that attained the highest mean accuracy in each feature subset. The overall best mean accuracy, with its corresponding values, is highlighted in bold.

TABLE IV
BEST CLASSIFIER DETAILED RESULT FOR EACH FEATURE SUBSET. BEST RESULT IS HIGHLIGHTED IN BOLD. BEYOND HAVING THE HIGHEST MEAN ACCURACY, THE COMBINATION OF HARALICK FEATURES AND THE SVM CLASSIFIER HAS A SMALLER STANDARD DEVIATION VALUE AND THE HIGHEST LOWER AND UPPER BOUNDS OF THE CONFIDENCE INTERVAL.

Feature Subset	Best Classifier	Mean	Std	Lower	Upper
Hu	*Random Forest*	0.48	0.08	0.45	0.52
Fourier	*SVM*	0.49	0.09	0.45	0.52
Haralick	**SVM**	**0.71**	**0.08**	**0.68**	**0.74**
All	*Random Forest*	0.66	0.09	0.62	0.69
ANOVA	*MLP*	0.66	0.09	0.62	0.69
SFS	*MLP*	0.58	0.10	0.54	0.62

Figure 10 shows a boxplot comparing the results achieved by the best classifier using all features (RF), the overall best (SVM with Haralick), and the best with feature selection (MLP with ANOVA). The plot does not show a clear difference between them.

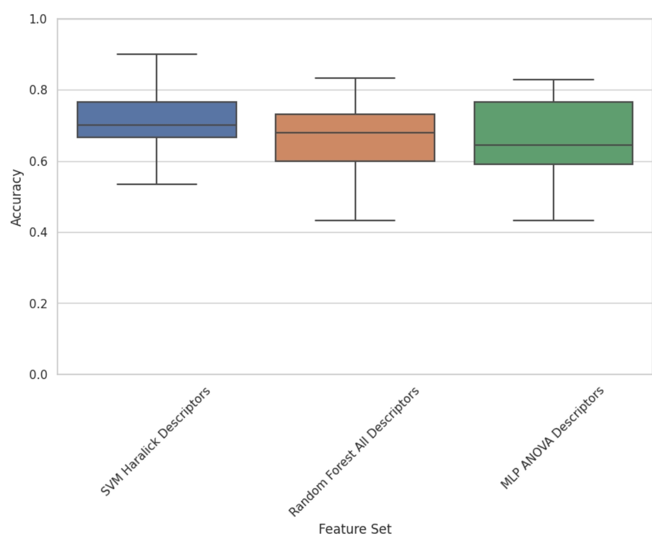

Fig. 10. Boxplot of the First Phase Best Classifiers. Visually, results of the SVM with Haralick combination appear to be better.

Considering that the Haralick with SVM combination was found to be statistically superior in the mean, the SVM classifier is chosen to perform the next experimental phase. An additional advantage of choosing SVM is that this classifier is faster to train than RF and MLP classifiers.

B. Second Phase

Table V presents the results of the second phase experiments. Feature sets were extracted by the pre-trained convolutional networks, and the classification was performed by the SVM classifier. The table provides the mean accuracy, standard deviation, and the lower and upper bounds of the confidence interval at a 95% significance level for each deep architecture with the SVM classifier. While ViT stands as one of the most contemporary technologies, one may see that DenseNet121 achieved the highest mean accuracy, the lowest standard deviation value, and the highest lower and upper bounds of the confidence interval.

TABLE V
RESULTS OF DEEP LEARNING FEATURE EXTRACTORS. BEST VALUES ARE HIGHLIGHTED IN BOLD.

Feature Set	Mean	Std	Lower Bound	Upper Bound
AlexNet	0.75	0.07	0.73	0.77
ResNet50	0.80	0.07	0.77	0.82
ResNet101	0.82	0.07	0.79	0.84
DenseNet121	**0.89**	**0.05**	**0.87**	**0.91**
VGG19	0.74	0.06	0.72	0.76
InceptionV3	0.67	0.08	0.64	0.70
ViT	0.83	0.06	0.81	0.86

The results are visually depicted in the boxplot in Figure 11, where DenseNet121 appears to be clearly superior to the other approaches.

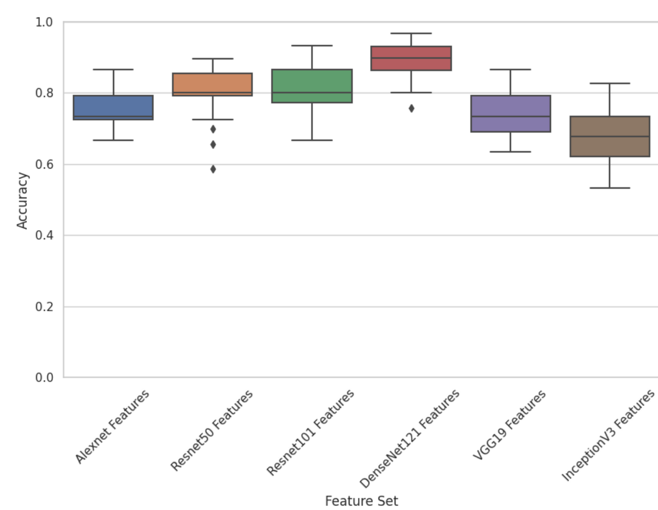

Fig. 11. Boxplot of the Deep Learning Feature Extractors. Visually, results of the Densenet121 are clearly better.

DenseNet121 consistently demonstrated superior performance, prompting the exclusive exploration of partial and full fine-tuning experiments with this model. In the realm of partial fine-tuning, a mean accuracy of 0.84 was achieved, while full fine-tuning yielded a slightly lower accuracy of 0.81, both falling short of the impressive 0.89 attained by DenseNet121. On the contrary, ViT demonstrated enhanced performance through partial fine-tuned training, achieving an accuracy of 0.85, while its feature extraction yielded an accuracy of 0.83. Given the notable improvement observed in ViT's performance through partial fine-tuning, it is reasonable to hypothesize that full fine-tuning training could further enhance its capabilities.

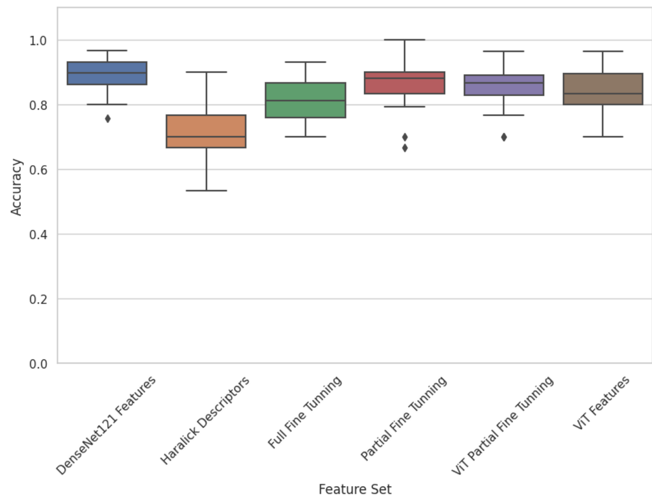

Fig. 12. Boxplot of Best Methods. Visually, results of the Densenet121 are clearly better.

Figure 12 shows a boxplot of all experiments together, expliciting the superiority of DenseNet121 besides the other methods.

VII. Conclusion

This paper explored machine learning methodologies for classifying public illumination lamp types and wattages based on visual descriptors. The experiments underscored that relying solely on mathematical visual descriptors extracted from images falls short in addressing the complexities of the classification task. Surprisingly, employing Haralick descriptors exclusively with the support vector machine classifier yielded the best result among the mathematical descriptor approaches, achieving a mean accuracy of 0.71. However, this performance still lags behind the state-of-the-art result in this domain, as reported by Silva et al. [29] with a mean accuracy of 0.86. It's noteworthy that the referenced study utilized a different type of data (obtained from radiometric sensors only) compared to the images employed in this work.

On the other hand, deep learning architectures performed significantly better, especially the DenseNet121-based approach that yielded a mean accuracy of 0.89, superior to the 0.86 obtained by Silva et al. [29], establishing a new state of the art in this problem domain. This noteworthy accomplishment affirms the initial expectations of this work, indicating that the task of classifying public lamps can indeed be efficiently executed using only images of the lamps. It also indicates the presence of transferable learning, a crucial aspect in real-world applications. The overall results obtained through these techniques are highly satisfactory, affirming the abundance of visual information within the images and facilitating public lamp type and wattage differentiation based only on image information.

The various feature selection approaches employed in this study did not manifest a discernible improvement in the obtained results. Despite rigorous experimentation with different methods, the impact on the overall performance of the classification task appeared to be negligible. This observation suggests that, in this context, the intrinsic characteristics of the features themselves or the complexity of the classification task may diminish the efficacy of feature selection in enhancing the results.

An important implication of this work is the simplification of data collection for the Public Lighting Images Dataset. Instead of necessitating a specially designed device capable of capturing both images and radiometric sensor data, the process now only requires a digital camera. As digital cameras embedded in smartphones continue to advance, there's potential to leverage smartphones for image capture, eliminating the need for specialized electronic devices to enhance the PL Dataset. Moreover, the economic viability of implementing the developed technique on a large scale is enhanced, as the requirement for building specific devices for data capture is no longer imperative.

Certainly, augmenting the number of instances in the PL Image Dataset is likely to independently enhance the performance of these techniques. The increase in data volume contributes to more robust machine learning training, fostering improved model generalization and accuracy. The principle that more data leads to better learning is a fundamental tenet in machine learning, and expanding the dataset is anticipated to positively impact the effectiveness of the employed techniques.

Alternatively, future work could explore increasing the dataset size by generating synthetic images through the application of generative adversarial networks (GANs) [11].

Furthermore, utilizing more powerful hardware could make full fine-tuning training of the ViT model feasible. There is strong evidence to suggest that this approach may enhance the model's performance, potentially establishing a new state-of-the-art in PL classification.

Public Lighting Image Dataset and Public Lighting Descriptors Dataset

To facilitate experimental comparisons of methods utilizing the Public Lighting Image Dataset and the Public Lighting Descriptors Dataset, both datasets are publicly available at Github.

References

[1] Raphael S Broetto and Flávio M Varejão. Heterogeneous feature models and feature selection applied to detection of street lighting lamps types and wattages. In *IECON 2016-42nd Annual Conference of the IEEE Industrial Electronics Society*, pages 933–938. IEEE, 2016.

[2] Raphael Santos Broetto, Flávio Miguel Varejão, José Gonçalves Pereira Filho, and André Bernadi Candeia. Automated image recognition of public lighting luminaries. In *2013 IEEE 20th International Conference on Electronics, Circuits, and Systems (ICECS)*, pages 657–660. IEEE, 2013.

[3] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794, 2016.

[4] Richard W Conners and Charles A Harlow. A theoretical comparison of texture algorithms. *IEEE transactions on pattern analysis and machine intelligence*, (3):204–222, 1980.

[5] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20:273–297, 1995.

[6] Thomas Cover and Peter Hart. Nearest neighbor pattern classification. *IEEE transactions on information theory*, 13(1):21–27, 1967.

[7] Thomas R Crimmins. A complete set of fourier descriptors for two-dimensional shapes. *IEEE Transactions on Systems, Man, and Cybernetics*, 12(6):848–855, 1982.

[8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[9] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *CoRR*, abs/2010.11929, 2020. URL https://arxiv.org/abs/2010.11929.

[10] Ellen R Girden. *ANOVA: Repeated measures*. Number 84. Sage, 1992.

[11] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014.

[12] Gösta H Granlund. Fourier preprocessing for hand print character recognition. *IEEE transactions on computers*, 100(2):195–201, 1972.

[13] Robert M Haralick, Karthikeyan Shanmugam, and Its' Hak Dinstein. Textural features for image classification. *IEEE Transactions on systems, man, and cybernetics*, (6):610–621, 1973.

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[15] Tin Kam Ho. Random decision forests. In *Proceedings of 3rd international conference on document analysis and recognition*, volume 1, pages 278–282. IEEE, 1995.

[16] Ming-Kuei Hu. Visual pattern recognition by moment invariants. *IRE transactions on information theory*, 8(2): 179–187, 1962.

[17] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

[18] M. Kanthi and Ravilla Dilli. Smart streetlight system using mobile applications: secured fault detection and diagnosis with optimal powers. *Wireless Networks*, 29 (5):2015–2028, July 2023. ISSN 1572-8196. doi: 10.1007/s11276-023-03278-9. URL https://doi.org/10.1007/s11276-023-03278-9.

[19] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems*, 30, 2017.

[20] J. Kittler. Feature set search algorithms. In C.H. Chen, editor, *Pattern Recognition and Signal Processing*, pages 41–60. Sijthoff and Noordhoff, Netherlands, 1978.

[21] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012.

[22] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[23] Ming Liu, Baogang Zhang, Tong Luo, Yue Liu, Boris A. Portnov, Tamar Trop, Weili Jiao, Huichan Liu, Yiwei Li, and Qingyuan Liu. Evaluating Street Lighting Quality in Residential Areas by Combining Remote Sensing Tools and a Survey on Pedestrians' Perceptions of Safety and Visual Comfort. *Remote Sensing*, 14(4):826, January 2022. ISSN 2072-4292. doi: 10.3390/rs14040826. URL https://www.mdpi.com/2072-4292/14/4/826. Number: 4 Publisher: Multidisciplinary Digital Publishing Institute.

[24] Ioannis Mavromatis, Aleksandar Stanoev, Pietro Carnelli, Yichao Jin, Mahesh Sooriyabandara, and Aftab Khan. A dataset of images of public streetlights with operational monitoring using computer vision techniques. *Data in Brief*, 45:108658, 2022. ISSN 2352-3409. doi: https://doi.org/10.1016/j.dib.2022.108658. URL https://www.sciencedirect.com/science/article/pii/S2352340922008630.

[25] P. Pudil, J. Novovičová, and J. Kittler. Floating search methods in feature selection. *Pattern Recognition Letters*, 15(11):1119–1125, 1994. ISSN 0167-8655. doi: https://doi.org/10.1016/0167-8655(94)90127-9. URL https://www.sciencedirect.com/science/article/pii/0167865594901279.

[26] J. Ross Quinlan. Induction of decision trees. *Machine learning*, 1:81–106, 1986.

[27] Mauricio GC Resende and Celso C Ribeiro. Greedy randomized adaptive search procedures: Advances, hybridizations, and applications. *Handbook of metaheuristics*, pages 283–319, 2010.

[28] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning internal representations by error propagation. Technical report, California Univ San Diego La Jolla Inst for Cognitive Science, 1985.

[29] I. C. Silva, R. M. Salgado, I. M. S. Varej ao, and

F. M. Varej ao. Analysis and improvement of machine learning models for detecting street lighting lamps. *Learning & Nonlinear Models*, 21(2):16–28, 2023. doi: 10.21528/lnlm-vol21-no2-art2.

[30] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[31] G. M. Soares, A. G. B. Almeida, R. M. Mendes, E. C. Teixeira, H. A. C. Braga, M. N. Machado, R. S. Broetto, M. V. H. B. Castro, H. O. Gomes Filho, F. M. Varejão, J. G. Pereira Filho, A. B. Candeia, and R. A. A. Sousa. Performance evaluation of a sensor-based system devised to minimize commercial losses in street lighting networks. In *2014 IEEE International Instrumentation and Measurement Technology Conference (I2MTC) Proceedings*, pages 336–341, 2014. doi: 10.1109/I2MTC.2014.6860763.

[32] GM Soares, AGB Almeida, RM Mendes, EC Teixeira, HAC Braga, MN Machado, RS Broetto, MVHB Castro, HO Gomes Filho, FM Varejão, et al. On the use of light sensors and pattern recognition techniques for automated detection of street lighting lamps. *Sensing Technology: Current Status and Future Trends III*, pages 67–92, 2015.

[33] Guilherme Marcio Soares, Alcindo GB Almeida, Raphael M Mendes, Estêvão Coelho Teixeira, Henrique AC Braga, Missael N Machado, Raphael Santos Broetto, Murillo VHB Castro, Helder O Gomes Filho, Flávio Miguel Varejão, et al. Detection of street lighting bulbs information to minimize commercial losses. In *2013 Seventh International Conference on Sensing Technology (ICST)*, pages 895–900. IEEE, 2013.

[34] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.

[35] Joan S Weszka, Charles R Dyer, and Azriel Rosenfeld. A comparative study of texture measures for terrain classification. *IEEE transactions on Systems, Man, and Cybernetics*, (4):269–285, 1976.

[36] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? *Advances in neural information processing systems*, 27, 2014.