Faculty of Mathematics
and Computer Science

Development methods for intelligent systems (TRSI 2021)

Light source presence in images

Filip Comșa

Department of Computer Science, Babeş-Bolyai University
1, M. Kogalniceanu Street, 400084, Cluj-Napoca, Romania
E-mail: filipcomsa5@gmail.com

Abstract

Together with the evolution of technology and the continuously increasing speed of computation, a huge wave of innovation has started to happen in many different fields, led by artificial intelligence and machine learning algorithms, thus being able to automate many of the processes that would require human interaction. This paper aims to provide information on how one such 'basic' problem has been solved, the problem of deciding whether an image contains a light source or not. Through a light source we can mean both artificial sources, such as a bulb or a LED, and also natural lighting, which during the day could be a window, or the sun itself, and during the night any visible celestial bodies. What's more, we can also consider reflections, like a window or some metallic objects, as a light source. This paper will show the process of deciding this, through trial and error. Multiple models were tried, some just for their architecture, other models were taken as pretrained and tuned for own own dataset. AutoML was also taken into consideration, but the final results indicated that the best model was, in fact, one made. Data augmentation and preprocessing played a crucial part in our algorithm, bringing improvements in some cases of up to 20%.

Keywords: AI; Artificial Intelligence; ML; Machine Learning; Image Classification; CNN; Light; Source; Image; Classification; Algorithm; Architecture; Model; Keras; Tensorflow; Light Source; Reflections; Artificial Lighting; Detection; Presence; Photograph; Augmentation; Data; Preprocessing; Processing

1. Introduction

The idea behind this project is to create a machine learning model based on deep learning which should be capable of classifying certain images by the follow criteria: whether an image contains, or not, a visible light source.

In the real world, such a project could have a humanitarian purpose, as a light polluting system, which has a negative effect on wild life, it contributes to the raising levels of carbon dioxide from the atmosphere, and, even more than that, it has a negative effect on human health, having sleep depravation effects in some cases.

This problem, of light pollution, has become an international problem specially because the majority of the world population lives under a sky affected by this fenomenon. Any person may easily identify this fenomonem, being quite visible for the clear eye when looking at the sky during night time. According to statistics coming from ”World Atlas of Artificial Night Sky Brightness”, 80% of the world's population is affected by this, percentages being even higher in the United States of America and in Europe, at 99%.

2. Implementation Details

2.1. Dataset

Even though the problem of detecting light sources has a great importance for the environment, there exists no public available datasets. Therefore, in order to find a solution for solving the issue of light pollution, a dataset containing 700 high quality images was created. The photos are splitted evenly as follows:

- 350 photos which contain turned on lights
- 350 photos which contain turned off lights

2.2. Methodology

In order to classify an image as whether it contains or not a light source, first there is the need for a clear definition for what a light source is.

In this case, by a light source is described either as an artificial source of light, such as a bulb, but also as a source of natural light, such as the sun going through a window. Moreover, the reflections can also be considered light sources. The goal of the proposed approach is to be able to recognize all of these sources.

The solution proposed is to implement a trainable end-to-end classification model for lighting sources. In order to obtain a high accuracy, several approaches were analyzed:

- Existing Architectures - EfficientNet
- AutoML - AutoKeras
- Matrix Multiplication Power

2.3. Data Preprocessing

[5]Before using the photos from the dataset as input for the model, the photos are subjected to pre-processing. For this stage, several actions were applied, as follows:

- All pictures were resized to a standard resolution: 250x250 pixels
- The the RGB interval was transformed from [0, 255] to [0, 1]

- The luminance formula using dot product is applied: P = R * 0.2126 + G * 0.7512 + B * 0.0722. The final image will look like a black and white image, but the lighting will be enhanced.

- Matrix multiplication is enforced multiple times, which cause dark spots to become darker and lighter spots to become lighter. The result is a feature enhancement, whereas the features seeking for this approach are the lightest of spots.

2.4. Model Architecture

2.4.1. EfficientNet

EfficientNet[6] is a Convolutional Neural Network architecture and scaling method that is optimized to scaling uniformly to all dimensions of depth, width and resolution using a compound coefficient.

The approach of compound scaling is based on the intuition that if the input image is bigger, then the network needs more layers to increase the receptive field and more channels to capture more fine-grained patterns on the bigger image.

2.4.2. AutoKeras

AutoKeras[1] is an open-source library for performing AutoML for deep learning models. The search is performed using so-called Keras models via the TensorFlow tf.keras API.

Automated Machine Learning provides methods and processes to make Machine Learning available for non-Machine Learning experts, to improve efficiency of Machine Learning and to accelerate research on Machine Learning.

3. Experiments

Through multiple tries, it was found a model that is up to 85% in terms of accuracy. The properties for the best model and settings are described below.

3.1. Setup

The models proposed were trained and validated on the following:

- **EfficientNet** was trained using an M1 Pro processor having 10 core CPU, 16 core GPU and 16 core Neural Engine.

- **AutoKeras** was run on Intel Core i7-6700HQ processor, 4 core CPU, due to the fact that is not available for M1 architecture yet. Ergo, a lower end computer had to be used, dramatically increasing training times (from minutes to hours), while reaching inconclusive results.

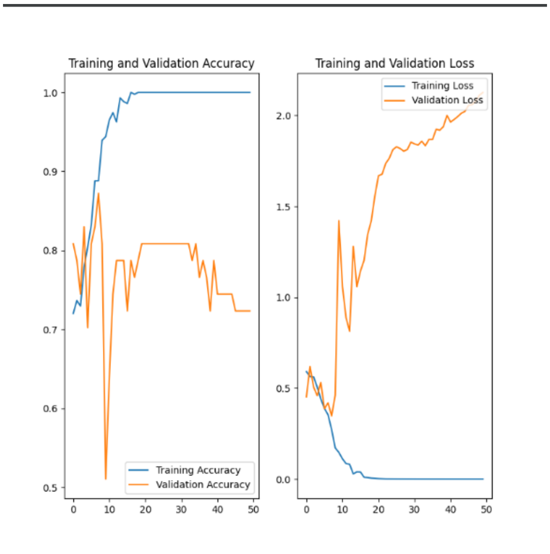
The dataset was split in the following way:

- 90% of the data was used for training
- and 10% for validation.

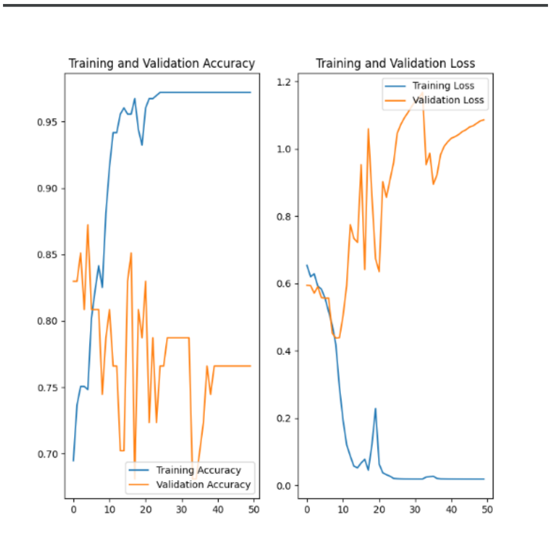
3.2. Results and Analysis

[2]Several experiments were conducted for EfficientNet model, from BX from B0 to B5. The results were not very good, having reached a maximum of 65% validation accuracy.

Epochs	Final Validation Accuracy	Epoch with Max Val Acc	Graph	Max Val Acc	Data Augmentation Matrix Multiplication Power
50	0.7234	8	0	0.8723	4
50	0.766	17	1	0.8511	4
50	0.6809	4	2	0.8298	4
50	0.7872	27	-	0.8723	None
50	0.8298	45	3	0.8723	2
50	0.7447	7	4	0.8723	2
50	0.7021	12	5	0.8936	2
50	0.7234	3	6	0.8723	2
50	0.7447	2	7	0.8723	3
50	0.7447	6	8	0.8723	3
50	0.7872	10	9	0.8723	3



Graph 0



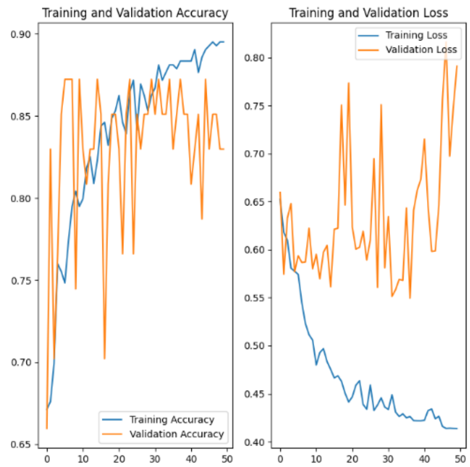
Graph 1

After multiple failed tries, it was found that with the available hardware AutoKeras will not be able to find a satisfactory model for the presented use case. The maximum accuracy obtained was under 50%, after 6 hours of training.

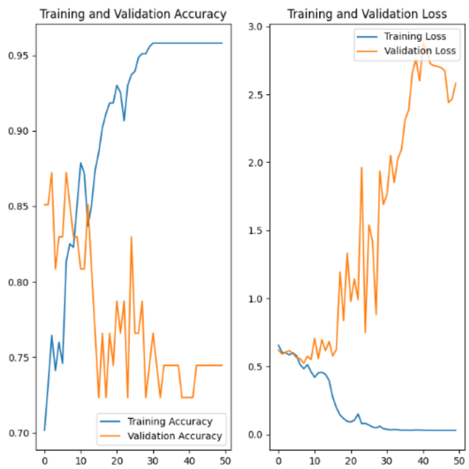
The average accuracy was 73%, the better models and parameters getting an up to 85%, and the worst as low as 54% (though that would be quite rare).

Finding the best augmentation settings: The course of multiple training attempts is presented through table 2.1 As we can notice from the table and the graphs, it looks like the simple matrix multiplication seems to have the best results, having a power of only 2.

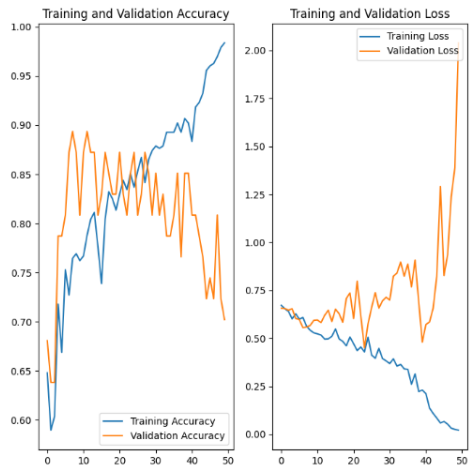
Graph 2



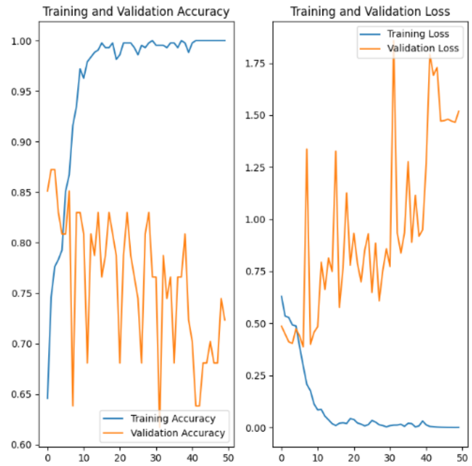
Graph 3



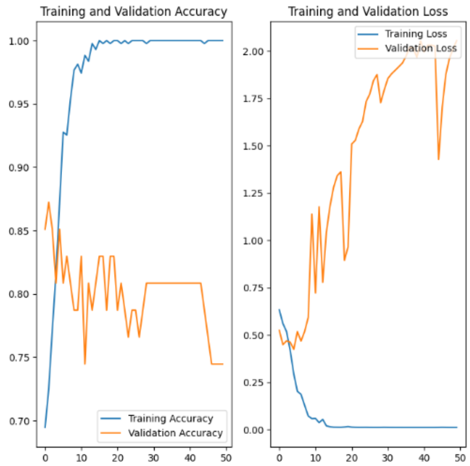
Graph 4



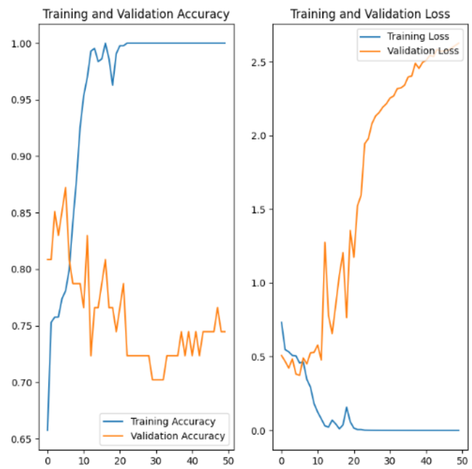
Graph 5



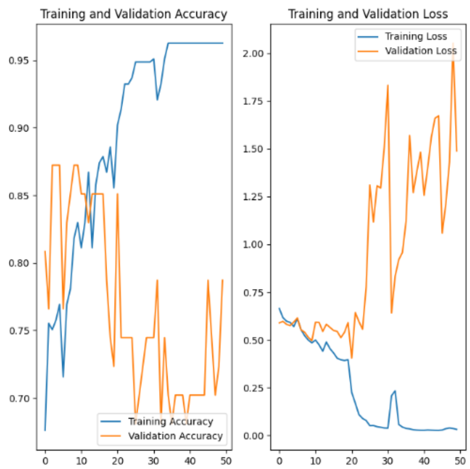
Graph 6



Graph 7



Graph 8



Graph 9

4. Other sections to be added

5. Related Work

When it comes to related work, it seems that not much work has been done in this direction. By the looks of it, machine learning models for light source detection have not been a subject of great interest to researchers for some reason.

While machine learning is a very innovative field for many and it has been used to experiment with any possible automation, light source detection seems to have been avoided in this way.

So far, most light detection algorithms take different approaches, very far from intelligent technique of detection, mostly using usual algorithms. One such example of work where light detection is involved is [4]. This paper discusses about some very complex mathematical equations through which they want to detect not necessarily the source of light from an image, but rather to identify how the lighting is emitted and how it appears on certain objects, and it also attempts to collect intensity information for these light sources. The algorithm involved performs different transformarmations on the image, both preprocessing kind but also some 3D transformations to better understand the environment. The final algorithm that it proposes involved the calculation of azimuth and zenith angles which is used to detect the intensity and to detect where light is coming from. While in the previously mentioned paper the researchers seem to focus on a single light source, the same idea is later applied in this paper [3] by the same authors to detect multiple light sources.

While the work mentioned above seems to be closely related to light detection, it seems to be quite for from what our intention is with our project.

6. Real World Applications

Being able to tell that there is - or there isn't - a light source in an image becomes very useful when our intention is to create a sensor or to do light detection in real time. As we have previously discussed, light pollution is one very important issue that most of the world - specially the richer countries - are fighting, and light detection mechanisms can become a great asset in this combat. An algorithm that is capable of telling whether in a certain spot there is or isnt a light source becomes quite important because it could allow tracking of light sources across large areas - for example through drones being equipped with camera tools which also use this project could very well be able to create a map of light sources - and that is just one very vague example. More serious tools could be created for the experts that are trying to automate the process of identifying light pollution sources.

What's more, light detection algorithms can become very useful in the context of smart homes. This project could be very well used as a sensor to notify a smart home hub to perform certain actions depending on the light - for example depending on whether the sun is visible, like automatically close the curtain of a window if direct light is coming through strongly such that it may disturb us, or our sleep.

7. Conclusions and future work

After many experiments, it is becoming clear that data augmentation and pre-processing can play a great role when it comes to this kind of classifying problem. Finding the right combination of data augmentation and preprocessing filters is not always an easy task, which is why it takes a lot of experimentation to reach a definite conclusion, and having a small dataset is definitely not helpful.

Due to the small size of the used dataset (700 photos), gaining clear results from any experiment is next to impossible, thus every time an "experiment" was conducted (through an experiment we mean a different configuration of a data augmentation + preprocessing, followed by a training and a final algorithm evaluation) it was necessary to rerun the same experiment multiple times. To be able to test as many things in the limited time available, we have chosen to always run the same experiment 3 times, and perform an analysis on the results of each experiment trial, while keeping in mind the fact that this is actually a single experiment, so we sort of "averaged" the information.

Unexpectedly so, the combination of luminance equation and matrix multiplication seems to be very helpful in our use case (please keep in mind that not any black and white equation along with matrix multiplication will have the same effect on our data - the effect of enhancing light features, it is totally possible that for different black and white formulas to have quite the opposite effect when adding in matrix multiplication - that is to reduce light source features) - so helpful that in fact, is became the most important part of this project. The importance of preprocessing should never be overlooked, specially when you are able to find a way to increase the wanted features and decrease the unwanted ones.

When it comes to future work, a starting point should be the extension of the dataset, and once that is done, I believe it would be a good idea to revisit both EfficientNet BX and AutoKeras - because they will most certainly perform better when more data is available to them, such that they are able to build the weights more efficiently - that is to learn better.

Future work can definitely go way beyond all of this. Different algorithms can be tried out and some other ways of preprocessing and augmentation can be tried out. Image segmentation, if used correctly, while being a bit more advanced than a CNN, could most probably also improve the results of this classification problem - that is to check whether it has found or not an object in the image.

To conclude this paper here, while we were able to get quite a strong accuracy percentage with what we have implemented here, there are a lot more things to try, and a much higher accuracy can, and should, be reached. Biggest drawback that we had to fight was the lack of data - if we can solve this issue first, we can get much further with this research.

References

[1] Haifeng Jin, Q.S., Hu., X., 2019. Auto-keras: An efficient neural architecture search system. Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery Data Mining , 1946–1956.

[2] Jiajia Zhou, Bolong Huang, Z.Y., Bünzli, J.C.G., 2019. Emerging role of machine learning in light-matter interaction. Light: Science Applications 8, No. 84, 1–7.

[3] Lopez-Moreno, J., Garces, E., Hadap, S., Reinhard, E., Gutiérrez, D., 2013. Multiple light source estimation in a single image. Computer Graphics Forum 32, 1–12.

[4] Lopez-Moreno, J., Hadap, S., Reinhard, E., Gutierrez, D., 2009. Light source detection in photographs, in: Andujar, C., Lluch, J. (Eds.), CEIG 09 - Congreso Espanol de Informatica Grafica, The Eurographics Association. pp. 161–167.

[5] Shorten, C., Khoshgoftaar, T.M., 2019. A survey on image data augmentation for deep learning. Journal of Big Data 6, No. 60.

[6] Tan, M., Le., Q.V., 2019. Efficientnet: Rethinking model scaling for convolutional neural networks. International Conference on Machine Learning .