Image Processing for Model Training and Deep Learning

A project report
submitted in partial fulfilment of the requirements for the award of the degree of

Bachelor of Technology
in **Computer Science &**
Engineering

By

ANKIT KUMAR
B.Tech. 5TH Semester
Enrolment No.: U2051006

Under the Supervisions of

Dr. Shiv Prakash

Department of Electronics and Communication
J K Institute of Applied Physics & Technology
UNIVERSITY OF ALLAHABAD

November, 2022

CANDIDATE'S DECLARATION

I, **Ankit Kumar,** hereby certify that the work, which is being presented in the report, entitled **Image processing for Model Training and Deep Learning**, in partial fulfillment of the requirement for the award of the Degree of **Bachelor of Technology in Computer Science and Technology** and submitted to the institution is an authentic record of my own work carried out during the period *October 2022* to *November-2022* under the supervision of **Dr. Shiv Prakash** at the Department of Electronics and Communication, University of Allahabad. The matter presented in this report has not been submitted elsewhere for the award of any other degree or diploma from any Institutions.

I declare that I have cited the reference about the text(s) /figure(s) /table(s) /equation(s) from where they have been taken. I further declare that I have not willfully lifted up some other's work, para, text, data, results, etc. reported in the journals, books, magazines, reports, dissertations, theses, etc., or available at web-sites and included them in this report and cited as my own work.

Date: 29/11/2022

Signature of the Candidate
Ankit Kumar

CERTIFICATE FROM THE SUPERVISOR

This is to certify that the **Mr. Ankit Kumar** has carried out this project/dissertation entitled **Image processing for Model Training and Deep Learning** under my supervision.

Date: 29/11/2022

Signature of the Supervisor
Dr. Shiv Prakash
(Supervisor)
Seal/Designation

ACKNOWLEDGEMENT

I would like to express my sincere thanks and gratitude my supervisor in this project, **Dr. Shiv Prakash** for their consistent support and motivation during the development of this project. His supervision and support in this project have a great role in completion of this project on time.

ABSTRACT

This report aims to break the problems intended to processing image in data feeding and model training. First the picture must undergo image processing before being firsthand for model training and conclusion. The sole purpose of pre-processing is to increase the image quality so that later we can do an further analysis. It allows us to eliminate distortions and improve specific qualities that are essential for the application we are working on. Preprocessing includes adjustments to the size, orientation, and color but is not limited to it. These characteristics change depending on the application. An image must be preprocessed in order for the software to work correctly and produce the desire results.

Content

CHAPTER 1: INTRODUCTION

BACKGROUND AND MOTIVATION

Preprocessing is necessary to prepare picture data for model input. For example, the convolutional neural networks fully connected layers it is favored that all the images must be in the arrays which are of the same size.

PROBLEM STATEMENT AND OBJECTIVES

The sole purpose of pre-processing is raise the image quality so that we can analyze it more effectively. It allows us to eliminate distortions and improve specific qualities that are crucial for the task. Preprocessing includes adjustments to the size, orientation, and color but is not limited to it. These characteristics change depending on the application

PROJECT CATEGORY

This project comes under the category of Image processing.

CHAPTER 2: DESIGN DETAILS, PROPOSED APPROACHES, SYSTEM AND ALGORITHMS

INTRODUCTION

In a machine learning or a computer vision project, what we need is data. In our case, image data. Unfortunately, a few problems associated with image data include inaccuracy, complexity and inadequacy. This is why before building a computer vision model, it is important that the data is preprocessed to achieve the desired results.

SYSTEM DESIGN/REQUIREMENT

The work demonstrated over here can be implement in system with specification higher than or equal to 1 core CPU & minimum 2 GiB of RAM in any of Windows/ MacOS/Linux Distribution.

CHAPTER 3: TOOL/TECHNOLOGY DETAILS, AND IMPLEMENTATION

TOOL/TECHNOLOGY USED

Git

Google Colab

Kaggle

Linux

Python Programming Language

IMPLEMENTATION

- Importing the Dataset

First we load the dataset to google collab. To import the datasets, we need to import some of the libraries

Tensorflow
Keras
Os
Glob
Skimage io
Random
Numpy
Matplotlib

We have loaded our dataset and the next step is visualization of data

- Data preprocessing

Before using the images we need to standardize them since images are in different formats such as natural, fake, grayscale, etc., we need to standardize them before using them only after that we can use it for neural network application
Here are some image preprocessing techniques:

i. Grayscale conversion
ii. Normalization
iii. Data Augmentation
iv. Image standardization

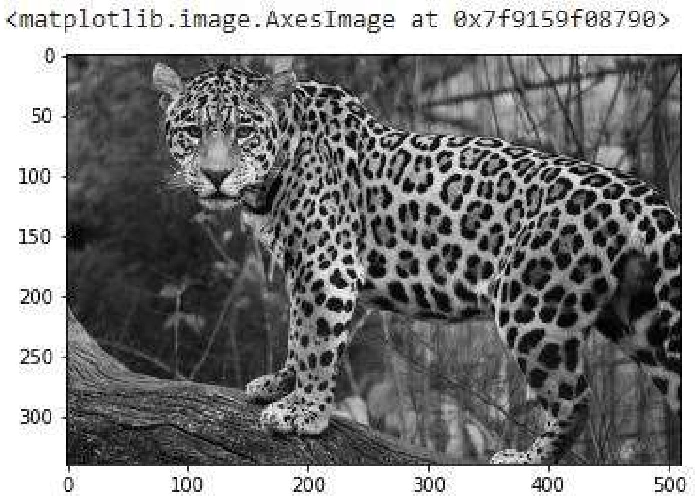
CHAPTER 4: TESTING, VERIFICATION, VALIDATION, AND RESULTS AND DISCUSSIONS

- **Grayscale conversion**

Grayscale is nothing but converting images from color to monochrome. It is normally used to reduce computational complexity in machine learning algorithms. Code below will convert the original image to a grayscale image:

```
gray_image = skimage.color.rgb2gray(image)
plt.imshow(gray_image, cmap = 'gray')
```

Output:



```
<matplotlib.image.AxesImage at 0x7f9159f08790>
```

- **Normalization**

It is used to scale the data of an attribute so that it falls in the smaller range such as minus -1 to 1 or 0.0 to 1.0 when multiple attributes are having values on different scales this may lead to poor data models for data mining operations so they are normalized to bring all the attributes on same scale.

Decimal scaling:

$$v' = \frac{v}{10^{j}}$$

Min-max normalization formula:

$$v_i' = \frac{v_i - min}{max - min}(new\_max - new\_min) + new\_min$$

Z-score normalization:

$$z = \frac{x - \mu}{\sigma}$$

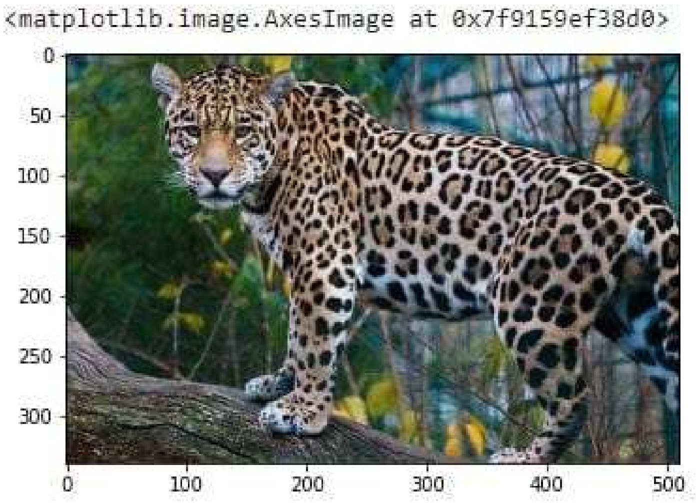
$$\mu = \text{Mean}$$
$$\sigma = \text{Standard Deviation}$$

the code below normalize our data.

```
norm_image = (gray_image - np.min(gray_image)) / (np.max(gray_image) -
np.min(gray_image))
```

```
plt.imshow(norm_image)
```

Output:



```
<matplotlib.image.AxesImage at 0x7f9159ef38d0>
```

- **Data Augmentation**

There are two types of augmentation:

Offline augmentation – Offline augmentation is used for small datasets. Which is applied in the data preprocessing step.

Online augmentation- Used only for large datasets. It is usually applied in real-time.

  - Shifting

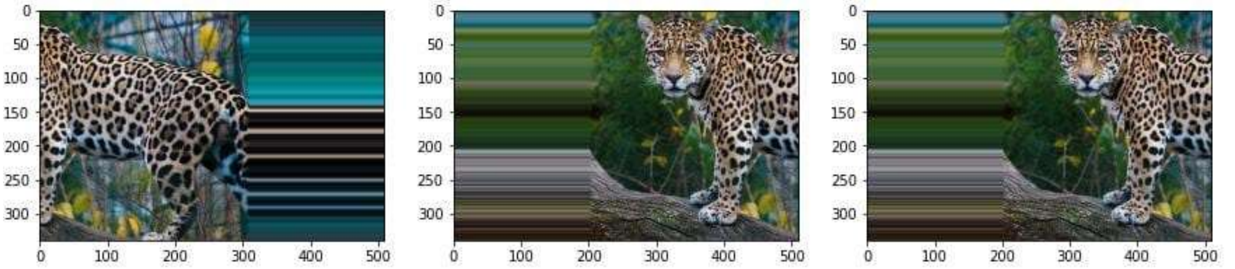
A process in which image pixels are shifted horizontally or vertically is called Shifting.

example below shifts horizontally:

```
# import libraries
from numpy import expand_dims
from keras.preprocessing.image import load_img
from keras.preprocessing.image import img_to_array
from keras.preprocessing.image import ImageDataGenerator

# convert to numpy array
data = img_to_array(image)
# expand dimension to one sample
samples = expand_dims(image, 0)
# create image data augmentation generator
datagen = ImageDataGenerator(width_shift_range=[-200,200])
# create an iterator
it = datagen.flow(samples, batch_size=1)
fig, im = plt.subplots(nrows=1, ncols=3, figsize=(15,15))
# generate batch of images
for i in range(3):
# convert to unsigned integers
image = next(it)[0].astype('uint8')
# plot image
im[i].imshow(image)
```

Output:



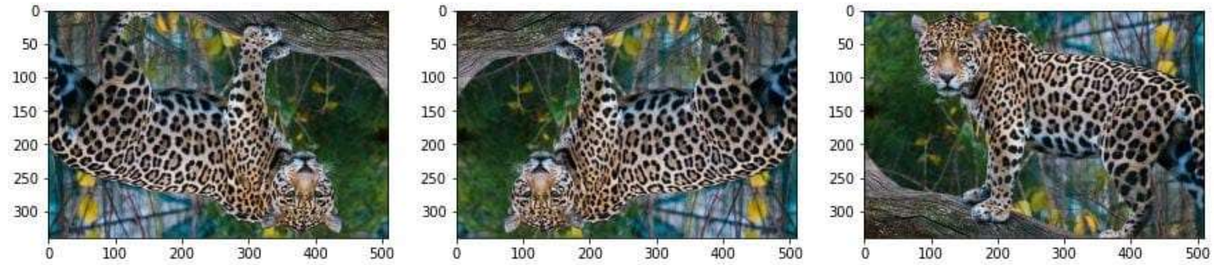
- Flipping

When the image pixel rows and columns are interchanged either vertically or horizontal.

The code below, change the Image-DataGenerator parameters as shown below:

```
# ImageDataGenerator for flipping
datagen = ImageDataGenerator(horizontal_flip=True, vertical_flip=True)
```

Output:



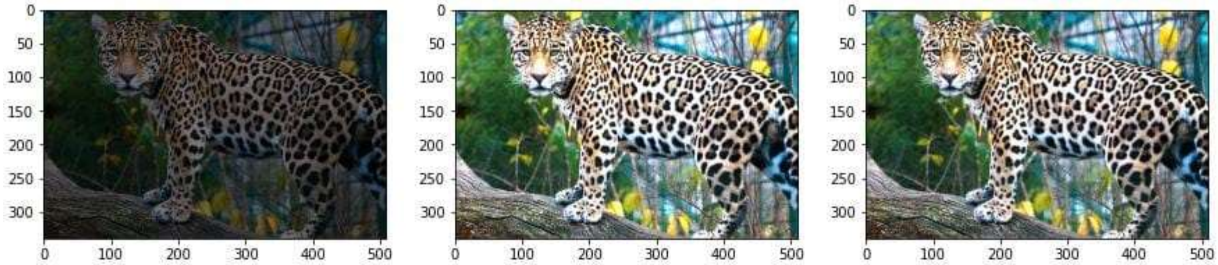
- Changing brightness

This is the process of increasing or decreasing image brightness or contrast.

From the code above, change the Image-DataGenerator parameters, as shown below:

```
datagen = ImageDataGenerator(brightness_range=[0.5,2.0])
```

Output:


0
50
100
150
200
250
300
0
100
200
300
400
500

CHAPTER 5: CONCLUSION AND FUTURE WORK/EXTENSION

CONCLUSIONS

The contribution of this report is to remove the ambiguity & conflicts of heterogeneity in feeding multiple image data in a model training.

SCOPE FOR FUTURE WORK

The project can be extended to solve many problems in image processing and further application of homogeneous image dimension and properties.

REFERENCES

- International Journal of Engineering Research & Technology (IJERT) ISSN: 2278-0181 IJERTV4IS030552 www.ijert.org (This work is licensed under a Creative Commons Attribution 4.0 International License.) Vol. 4 Issue 03, March-2015
- K. Sumithra, S. Buvana, R. Somasundaram Mtech Students Department of Computer Science and Engineering, Manakula Vinayagar Institute of Technology, Pondicherry.
- VIVA-Tech International Journal for Research and Innovation Volume 1, Issue 1 (2018) ISSN(Online): 2581-7280 Article No. 10 PP 1-10
- International Journal of Computer Applications (0975 – 8887) Volume 52– No.2, August 2012 34 Applications of Image Processing in Agriculture: A Survey Anup Vibhute Assistant Professor, BMIT, Solapur(India) S K Bodhe Phd, Professor, College of Engineering , Pandharpur