**Ex No:4**

**Date:**

**Augmentation Using Image Data Generator from Keras**

**Aim:**

To perform image augmentation using Keras ImageDataGenerator and visualize the transformed

images.

**Procedure:**

1.Import required libraries (numpy, matplotlib, ImageDataGenerator).

2.Load and resize the image to (224, 224).

3.Display the original image using matplotlib.

4.Create an ImageDataGenerator with augmentation parameters like rotation, zoom, brightness,

shear, and horizontal flip.

5.Generate multiple augmented images using datagen.flow().

Display the augmented images in a subplot grid.

Code:

import numpy as np

import matplotlib.pyplot as plt

from tensorflow.keras.preprocessing.image import ImageDataGenerator, img\_to\_array, load\_img

# Set image path

img\_path = '/content/unnamed.png' # Change to your actual path

# Load and resize image

image = load\_img(img\_path, target\_size=(224, 224))

image\_array = img\_to\_array(image)

image\_array = np.expand\_dims(image\_array, axis=0)

# Show original image

plt.figure(figsize=(5, 5))

plt.imshow(np.array(image).astype("uint8"))

plt.title("Original Image")

plt.axis("off")

plt.show()

# Define ImageDataGenerator

datagen = ImageDataGenerator(

rotation\_range=40,

zoom\_range=0.2,

brightness\_range=[0.5, 1.5],

shear\_range=0.2,

horizontal\_flip=True,

fill\_mode='nearest'

)

# Generate augmented images

aug\_iter = datagen.flow(image\_array, batch\_size=1)

augmented\_images = [next(aug\_iter)[0].astype('uint8') for \_ in range(4)]

# Display augmented images

plt.figure(figsize=(15, 4))

for i, aug\_img in enumerate(augmented\_images):

plt.subplot(1, 4, i + 1)

plt.imshow(aug\_img)

plt.title(f"Keras Aug {i + 1}")

plt.axis("off")

plt.tight\_layout()

plt.show()

**Output:**



|  |  |
| --- | --- |
| COE (20) |  |
| Record (20) |  |
| VIVA (10) |  |
| Total (50) |  |

**Result:**

Successfully generated and visualized multiple augmented variations of the original image using

Keras.