A Thesis/Project/Dissertation Report

on

Visualizing image differences using Python and OpenCV

Submitted in partial fulfillment of the
requirement for the award ofthe degree of

B.Tech CSE

(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)

Under The Supervision of Name of Supervisor : Mr. S. Prakash
School of Computing Science and Engineering, Galgotias University

Assistant Professor
Greater Noida,
Uttar Pradesh, India.

S. No	Enrolment Number	Admission Number	Student Name	Degree / Branch	Sem
1	19021011641	19SCSE1010466	KUSHAGRA GUPTA	B.tech/CSE	5
2	19021011364	19SCSE1010169	HARSHIT GUPTA	B.tech/CSE	5

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
GALGOTIAS UNIVERSITY, GREATER NOIDA
INDIA
DECEMBER, 2021

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING
GALGOTIAS UNIVERSITY, GREATER NOIDA

CANDIDATE'S DECLARATION

I/We hereby certify that the work which is being presented in the thesis/project/dissertation, entitled **"Visualizing image differences using python and open CV"** in partial fulfillment of the requirements for the award of the Galgotias

University submitted in the School of Computing Science and Engineering of Galgotias University, Greater Noida, is an original work carried out during the period of month, Year to Month and Year, under the supervision of Name… Designation, Department of Computer Science and Engineering/Computer Application and Information and Science, of School of Computing Science and Engineering , Galgotias University, Greater Noida

The matter presented in the thesis/project/dissertation has not been submitted by me/us for the award of any other degree of this or any other places.

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Supervisor Name

Designation

<u>CERTIFICATE</u>

The Final Thesis/Project/ Dissertation Viva-Voce examination of Name:_____

Admission No.:_____ has been held on _____ and his/her work is

recommended for the award of Name of Degree _____

Signature of Examiner(s)   **Signature of Supervisor(s)**

Signature of Project Coordinator   **Signature of Dean**

Date: December, 2021

Place: Greater Noida

ACKNOWLEDGEMENT

We have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. We would like to extend our sincere thanks to all of them. We are highly indebted to Mr. V. Gokul Rajan for their guidance and constant supervision as well as for providing necessary information regarding the project & also for their support in completing the project.

We would like to express our gratitude towards the member of Galgotias University for their kind co-operation and encouragement which help us in completion of this project.

We would like to express our special gratitude and thanks to industry persons for giving me such attention and time. Our thanks and appreciations also go to our colleague in developing the project and people who have willingly helped us out with their abilities.

ABSTRACT

In the recent past years, there has been as surge in the usage of internet around the world, billions of people are over the internet visiting different source of website but normal users can't differentiate between an authentic websites and a forged fake one. Sometimes its leads to potential scams or frauds, its becomes now quite normal nowadays. By these fake images attackers can spread misinformation among the general mass.

Our project helps you to visualize image differences and check whether the image is same as the true image or not , this project can be used in various fields like in the medical field to examine and compare various patient reports, to identify fake counterfeit bills, in the game industry to make games etc.

KEYWORDS- Python, OpenCV,Scikit-image, imutils,SSIM, CFS,HVS

TABLE OF CONTENTS

CHAPTER 1
INTRODUCTION

The situation of twenty first century is consider as time of tech. Technology, today, assumes a great process in our life. It is considered as a premise of improvement of a digital system. But you may have noticed that nowadays attackers use many different ways to spread misinformation to users by images or can leads to many attacks by forged websites. One example is phishing. Attackers can manipulate images ever-so-slightly to trick unsuspecting users who don't validate the URL into thinking they are logging into their banking website — only to later find out that it was a scam. Comparing logos and known User Interface (UI) elements on a web page to an existing data set could help reduce phishing attacks. Developing a phishing detection system is obviously much more complicated than simple image differences, but we can still apply these techniques to determine if a given image has been manipulated.

Digital images are subject to a wide variety of distortions during acquisition, processing, compression, storage, transmission and reproduction, any of which may result in a degradation of visual quality. For applications in which images are ultimately to be viewed by human beings, the only "right" method of quantifying visual image quality is through subjective evaluation. In practice, however, subjective evaluation is usually too inconvenient, time-consuming and expensive. The goal of research in objective image quality assessment is to develop quantitative measures that can automatically predict perceived image quality. A true picture quality measurement can play an assortment of jobs in picture handling applications. To start with, it very well may be used to progressively screen and change picture quality. For model, an organization computerized video server can inspect the nature of video being communicated to control and dispense streaming assets. Second, it can be used to optimize algorithms and parameter settings of image processing systems. For instance, in a visual communication system a quality measurement can aid the ideal plan of pre-separating and spot task calculations at the encoder what's more of ideal recreation, mistake covering and postfiltering calculations at the decoder. Third, it can be used to benchmark image processing systems and algorithms. Objective picture quality measurements can be arranged by the accessibility of a unique (distortion-free) image, with which the distorted image is to be analysed. Most existing methodologies are known as full-reference, which means that a total reference picture is thought to be known. In numerous down to earth applications, in any case, the reference picture is not available, and a no-reference or "blind" quality assessment approach is desirable. In a third sort of strategy, the reference picture is just to some degree accessible in the form of a set of extracted

features made available as side information to help evaluate the quality of the distorted image. This is referred to as reduced-reference quality assessment.

Objective image quality metrics can be classified according to the availability of an original (distortion-free) image, with which the distorted image is to be compared. Most existing approaches are known as full-reference, meaning that a complete reference image is assumed to be known. In many practical applications, however, the reference image is not available, and a no-reference or "blind" quality assessment approach is desirable. In a third type of method, the reference image is only partially available, in the form of a set of extracted features made available as side information to help evaluate the quality of the distorted image. This is referred to as reduced-reference quality assessment. This paper focuses on full-reference image quality assessment. The simplest and most widely used full-reference quality

metric is the mean squared error (MSE), computed by averaging the squared intensity differences of distorted and reference image pixels, along with the related quantity of peak signal-to-noise ratio (PSNR). These are appealing because they are simple to calculate, have clear physical meanings, and are mathematically convenient in the context of optimization. But they are not very well matched to perceived visual quality. In the last three decades, a great deal of effort has gone into the development of quality assessment methods that take advantage of known characteristics of the human visual system (HVS). The majority of the proposed perceptual quality assessment models have followed a strategy of modifying the MSE measure so that errors are penalized in accordance with their visibility and discusses its difficulties and limitations on the hypothesis that the HVS is highly adapted for extracting structural information. As a specific example, we develop a measure of structural similarity that compares local patterns of pixel intensities that have been normalized for luminance and contrast. So ,we can let's compute the difference between two images , and view the differences side by side of images using OpenCV, scikit-image, and Python.

TOOLS AND TECHNOLOGY USED

Before getting into any specific features, we should have basic ones in our system that we are going to use –

1. Python - Python is an interpreted, high-level, general-purpose programming language, Python can be easy to pick up whether you're a first time programmer or you're experienced with other languages. The following pages are a useful first step to get on your way writing programs with Python. It is friendly and easy to learn The community hosts conferences and meetups, collaborates on code, and much more. Python's documentation helps along the way, and the mailing lists keeps in touch. Python supports a lot of applications, the Python Package Index (PyPI) hosts thousands of third-party modules for Python both Python's standard library and the community-contributed modules allow for endless possibilities. Python is developed under an OSI-approved open source license, making it freely usable and distributable, even for commercial use so we will use many python modules to get determine the difference between the images.

2. Imutils - It is a computer vision package that includes a series of OpenCV + convenience functions to make basic image processing functions such as translation, rotation, resizing, skeletonisation, displaying Matplotlib images, sorting contours, detecting edges, among others quite easy. A series of convenience functions to make basic image processing functions such as translation, rotation, resizing, skeletonization, and displaying Matplotlib images easier with OpenCV and both Python 2.7 and Python 3.

a) Translation
Translation is the shifting of an image in either the x or y direction. To translate an image in OpenCV you would need to supply the (x, y)-shift, denoted as (tx, ty) to construct the translation matrix M: Translation equation
And from there, you would need to apply the cv2.warpAffine function.
Instead of manually constructing the translation matrix M and calling cv2.warpAffine, you can simply make a call to the translate function of imutils.

```
# translate the image x=25 pixels to the right and y=75 pixels up

translated = imutils.translate(workspace, 25, -75)
```

b) Rotation

Rotating an image in OpenCV is accomplished by making a call to cv2.getRotationMatrix2D and cv2.warpAffine. Further care has to be taken to supply the (x, y)-coordinate of the point the image is to be rotated about. These calculation calls can quickly add up and make your code bulky and less readable. The rotate function in imutils helps resolve this problem.

```
# rotate the image and display it

        rotated = imutils.rotate(bridge, angle=angle)

        cv2.imshow("Angle=%d" % (angle), rotated)
```

c) Resizing

Resizing an image in OpenCV is accomplished by calling the cv2.resize function. However, special care needs to be taken to ensure that the aspect ratio is maintained. This resize function of imutils maintains the aspect ratio and provides the keyword arguments width and height so the image can be resized to the intended width/height while (1) maintaining aspect ratio and (2) ensuring the dimensions of the image do not have to be explicitly computed by the developer.

```
        # resize the image and display it

        resized = imutils.resize(workspace, width=width)

        cv2.imshow("Width=%dpx" % (width), resized)
```

d) Skeletonization

Skeletonization is the process of constructing the "topological skeleton" of an object in an image, where the object is presumed to be white on a black background. OpenCV does not provide a function to explicitly construct the skeleton, but does provide the morphological and binary functions to do so.
For convenience, the skeletonize function of imutils can be used to construct the topological skeleton of the image.

```
# skeletonize the image

gray = cv2.cvtColor(logo, cv2.COLOR_BGR2GRAY)

skeleton = imutils.skeletonize(gray, size=(3, 3))

cv2.imshow("Skeleton", skeleton)
```

e) Listing Paths to Images

The paths sub-module of imutils includes a function to recursively find images based on a root directory.

```
from imutils import paths

for imagePath in paths.list_images("../demo_images"):

        print imagePath
```

f) Sorting Contours

The contours returned from cv2.findContours are unsorted. By using the contours module the the sort_contours function we can sort a list of contours from left-to-right, right-to-left, top-to-bottom, and bottom-to-top, respectively.

3. OpenCV - OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in the commercial products. Being a BSD-licensed product, OpenCV makes it easy for businesses to utilize and modify the code. The library has more than 2500 optimized algorithms, which includes a comprehensive set of both classic and state-of-the-art computer vision and machine learning algorithms. These algorithms can be used to detect and recognize faces, identify objects, classify human actions in videos, track camera movements, track moving objects, extract 3D models of objects, produce 3D point clouds from stereo cameras, stitch images together to produce a high resolution image of an entire scene, find similar images from an image database, remove red eyes from images taken using flash, follow eye movements, recognize scenery and establish markers to overlay it with augmented reality, etc. OpenCV has more than 47 thousand people of user community and estimated number of downloads exceeding 18 million. The library is used extensively in companies, research groups and by governmental bodies. Along with well-established companies like Google, Yahoo, Microsoft, Intel, IBM, Sony, Honda, Toyota that employ the library, there are many start-ups such as Applied Minds, VideoSurf, and Zeitera, that make extensive use of OpenCV. OpenCV's deployed uses span the range from stitching street view images together, detecting intrusions in surveillance video in Israel, monitoring mine equipment in China, helping robots navigate and pick up objects at Willow Garage, detection of swimming pool drowning accidents in Europe, running interactive art in Spain and New York, checking runways for debris in Turkey, inspecting labels on products in factories around the world on to rapid face detection in Japan. It has C++, Python, Java and MATLAB interfaces and supports Windows, Linux, Android and Mac OS. OpenCV leans mostly towards real-time vision applications and takes advantage of MMX and SSE instructions when available. A full-featured CUDAand OpenCL interfaces are

being actively developed right now. There are over 500 algorithms and about 10 times as many functions that compose or support those algorithms. OpenCV is written natively in C++ and has a templated interface that works seamlessly with STL containers.

4. skimage - Scikit-image is a Python package dedicated to image processing, and using natively NumPy arrays as image objects. This chapter describes how to use scikit-image on various image processing tasks, and insists on the link with other scientific Python modules such as NumPy and SciPy. Scikit-image is a collection of algorithms for image processing and computer vision. The main package of Skimage only provides a few utilities for converting between image data types; for most features, you need to import one of the following sub-packages: Colour , Data , Draw , Exposure , Filters .. and so on . Skimage gives us a lot of features like-

a) Input, Output, Datatype and Color Spaces
Works with all data formats supported by the Python Imaging Library (or any other I/O plugin provided to imread with the plugin keyword argument) and it also works with URL image paths.

b) Datatypes - Image ndarrays can be represented either by integers (signed or unsigned) or floats. Different integer sizes are possible: 8-, 16- or 32-bytes, signed or unsigned.

c) Color Spaces- Color images are of shape (N, M, 3) or (N, M, 4) (when an alpha channel encodes transparency)
shape Routines converting between different color spaces (RGB, HSV, LAB etc.) are available in skimage. color : color.rgb2hsv, color.lab2rgb, etc. Check the docstring for the expected data type (and data range) of input images.

d) Image Pre-processing and enhancement- Local filters replace the value of pixels by a function of the values of neighboring pixels. The function can be linear or non-linear while Non-local filters use a large region of the image (or all the image) to transform the value of one pixel

e) Mathematical Morphology- Probe an image with a simple shape (a structuring element), and modify this image according to how the shape locally fits or misses the image.
Erosion = minimum filter. Replace the value of a pixel by the minimal value covered by the structuring element, Dilation: maximum filter, and Opening: erosion + dilation.

CHAPTER 2
LITERATURE SURVEY

As we have noticed use of internet has been increasing globally , which means it's will play an increasingly important role in the distribution of digital information in the future . But the spread of fake images on the internet is a cause of great concern .It is specifically designed to plant a seed of mistrust and exacerbate the existing social and cultural dynamics by misusing political ,regional and religious .In the past two years as well , 37% people have been victims of fraud by the fake websites.

Consumers reported losing more than 3.3 billion to fraud in 2020, up from 1.8 billion in 2019 and over 3.2 billion images are shared online daily as well , many of them are leads to mis-information for the group of users . This images sometimes becomes the reasons of riot or create disturbance in the society.

As we have noticed use of internet has been increasing globally , which means it's will play an increasingly important role in the distribution of digital information in the future . But the spread of fake images on the internet is a cause of great concern .It is specifically designed to plant a seed of mistrust and exacerbate the existing social and cultural dynamics by misusing political ,regional and religious .In the past two years as well , 37% people have been victims of fraud by the fake websites .

Consumers reported losing more than 3.3 billion to fraud in 2020, up from 1.8 billion in 2019 and over 3.2 billion images are shared online daily as well, many of them are leads to mis-information for the group of users. This images sometimes becomes the reasons of riot or create disturbance in the society.
The ubiquitous availability of easy-to-use software for editing digital images brought about by rapid technological advances of the 21st century has dramatically decreased the time, cost, effort, and skill required to fabricate convincing visual forgeries. Often distributed through trusted sources such as mass media outlets, perhaps unknowingly, these manipulated images propagate across social media with growing frequency and sophistication. Moreover, the technology that allows for manipulating or generating realistic appearing images has far outpaced the technological development of methods for detecting fake imagery and even experts often cannot rely on visual inspection to distinguish authentic digital images from forgeries. Bad actors can thus easily publish manipulated visual content to deceive their viewers, inflicting cognitive stress, exploiting prior beliefs, or influencing individuals' decisions and actions.

EXISTING PROBLEMS

Although it is difficult to say how prevalent undetected occurrences of fake imagery are, numerous examples have been exposed in which manipulated images have caused substantial harms at individual, organizational, and societal levels. For instance, an image of Senator John Kerry and Jane Fonda sharing the stage at a Vietnam era anti-war rally emerged during the 2004 presidential primaries as Senator Kerry was campaigning for the Democratic nomination. The forged photograph, however, was created by compositing together two separate photos that separately depicted Kerry and Fonda.

The edited image showing them together gave the false impression that Kerry shared the controversial anti-war views of activist Jane Fonda (Light, 2004). In a more recent example, in January 2014, the Associated Press news agency fired its Pulitzer prize-winning photographer Narciso Contreras for digitally removing an object from one of his widely distributed photographs of the Syrian civil war (The Guardian, 2014).This case has stirred an ongoing and contested discussion about the authenticity of digital photographs, the potential repercussions of image manipulation, and the ethics code in photojournalism. Numerous other examples exist where fake imagery has been used to distort the truth and manipulate viewers (For more examples, see http://pth.izitru.com/).

It is unclear how prevalent are instances of undetected photo manipulation. The damage done by manipulated imagery is real, substantial, and persistent. Studies suggest that manipulated images can distort viewer's memory (Wade et al., 2002)—therefore further enhancing the credibility of these images—and even influence decision-making behaviors such as voting (Bailenson et al., 2008; Nash et al., 2009). Moreover, even when individuals do become aware of the true nature of a forgery, the harmful impact of misinformation on their perception, memory, emotions, viewpoints, and attitude toward a topic can linger (Sacchi et al., 2007). Quite often the distribution of fake images will far surpass the distribution of any correction or attempt to expose the forgery (Friggeri et al., 2014). The factors combine to make image manipulation an extremely effective and difficult to combat manipulation method.

CHAPTER 3
FUNCTIONALITY

An Image Signal whose quality is being assessed can be considered as an amount of an undistorted reference signal and an error signal. A widely adopted assumption is that the loss of perceptual quality is directly related to the visibility of the error signal. The simplest implementation of this concept is the MSE, which objectively quantifies the strength of the error signal. But two distorted images with the same MSE may have very different types of errors, some of which are much more visible than others. Most perceptual image quality assessment approaches proposed in the literature attempt to weight different aspects of the error signal according to their visibility, as determined by psychophysical measurements in humans or physiological measurements in animals. This approach was pioneered by Mannos and Sakrison, and has been extended by many other researchers over the years.

Framework
The figure below illustrates a generic image quality assessment framework based on error sensitivity.   Most perceptual quality assessment models can be described with a similar diagram, although they differ in detail.
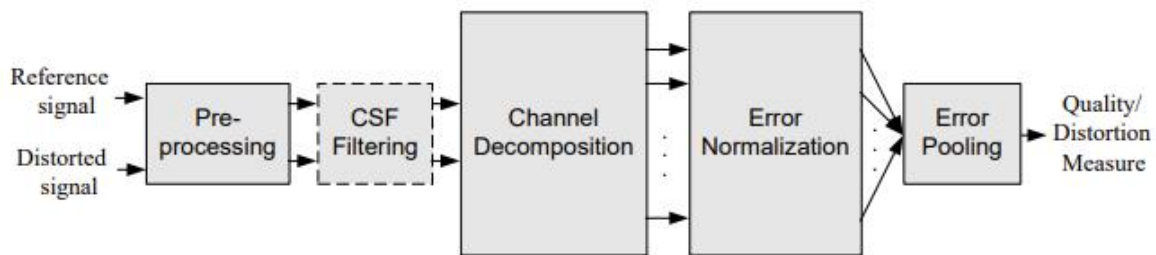
a) Pre-processing
This stage normally plays out an assortment of essential tasks to dispose of known mutilations from the pictures being analyzed. In the first place, the mutilated and reference signals are appropriately scaled and adjusted. Second, the sign
may be changed into a color space that is more fitting for the HVS.
Third, quality evaluation measurements might have to change over the digital pixel values stored in the computer memory into luminance values of pixels on
the display device through pointwise nonlinear transformations. Fourth, a low-pass filter simulating the point spread function of the eye optics may be applied. Finally, the reference and the distorted images may be modified using a nonlinear point operation to simulate light adaptation.

b) CSF Filtering

The contrast sensitivity function (CSF) portrays the sensitivity of the HVS to various spatial and temporal frequencies that are present in the visual stimulus. Some picture quality measurements incorporate a phase that weights the signal according to this function (typically implemented using a linear filter that approximates the quency response of the CSF). Nonetheless, many recent metrics choose to execute CSF as a base-affectability standardization factor after channel decomposition.

c) Channel Decomposition

The images are typically separated into sub bands that are selective for spatial and temporal frequency as well as orientation. While some quality assessment methods implement sophisticated channel decompositions that are believed to be closely related to the neural responses in the primary visual cortex many metrics use simpler transforms such as the discrete cosine transform (DCT) or separable wavelet transforms Channel decompositions tuned to various temporal frequencies have also been reported for video quality assessment.

d) Error Normalization

The error (difference) between the decomposed reference and distorted signals in each channel is calculated and normalized according to a certain masking model, which takes into account the fact that the presence of one image component will decrease the visibility of another image component that is proximate in spatial or temporal location, spatial frequency, or orientation. The normalization mechanism weights the error signal in a channel by a space-varying visibility threshold. The visibility threshold at each point is calculated based on the energy of the reference and/or distorted coefficients in a neighborhood (which may include coefficients from within a spatial neighborhood of the same channel as well as other channels) and the base-sensitivity for that channel. The normalization process is intended to convert the error into units of just noticeable difference (JND)

e) Error Pooling

The last phase of all quality measurements must consolidate the the normalized error signals over the spatial degree of the picture, and across the various channels, into a single value. For most quality assessment methods, pooling takes the form of a Minkowski norm: $E(\{e_{l,k}\}) = (\Sigma_i \Sigma_k |e_{l,k}|^\beta)^{1/\beta}$ where $e_{l,k}$ is the normalized error of the k-th coefficient in the l-th channel, and β is a constant exponent typically chosen to lie between 1 and 4.

Limitations

The underlying principle of the error-sensitivity approach is that perceptual quality is best estimated by quantifying the visibility of errors. This is essentially accomplished by simulating the functional properties of early stages of the HVS, as characterized by both psychophysical and physiological experiments.

In specific, the HVS is a complex and exceptionally non-straight system, yet most models of early vision depend on direct or semi direct administrators that have been portrayed utilizing confined and oversimplified boosts. Hence, blunder affectability approaches should depend on various solid suppositions and speculations

The Quality Definition Problem

The most fundamental problem with the traditional approach is the definition of image quality. In particular, it is not clear that error visibility should be equated with loss of quality, as some distortions may be clearly visible but not so objectionable. An obvious example would be multiplication of the image intensities by a global scale factor.

The Suprathreshold Problem

The psychophysical experiments that underlie many error sensitivity models are specifically designed to estimate the threshold at which a stimulus is just barely visible. These measured threshold values are then used to define visual error sensitivity measures, such as the CSF and various masking effects. However, very few psychophysical studies indicate whether such near-threshold models can be generalized to characterize perceptual distortions significantly larger than threshold levels, as is the case in a majority of image processing situations.

The Natural Image Complexity Problem

Most psychophysical experiments are conducted using relatively simple patterns, such as spots, bars, or sinusoidal gratings. For example, the CSF is typically obtained from threshold experiments using global sinusoidal images. The masking phenomena are usually characterized using a superposition of two (or perhaps a few) different patterns. But all such patterns are much simpler than real world images, which can be thought of as a superposition of a much larger number of simple patterns. Can the models for the interactions between a few simple patterns generalize to evaluate interactions between tens or hundreds of patterns? Is this limited number of simple-stimulus experiments sufficient to build a model that can predict the visual quality of complex-structured natural images? Although the answers to these questions are currently not known.

The Cognitive Interaction Problem

It is widely known that cognitive understanding and interactive visual processing (e.g., eye movements) influence the perceived quality of images. For example, a human observer will give different quality scores to the same image if s/he is provided with different instructions. Prior information regarding the image content, or attention and fixation, may also affect the evaluation of the image quality. But most image quality

metrics do not consider these effects, as they are difficult to quantify and not well understood.

STRUCTURED SIMILARITY BASED ASSESSMENT

Normal picture signals are profoundly organized Their pixels show solid conditions, particularly when they are spatially general, and these conditions convey significant data about the construction of the items in the visual scene. The Minkowski error metric depends on pointwise signal differences which are free of the fundamental signal construction Although most quality measures based on error sensitivity decompose image signals using linear transformations, these do not remove the strong dependencies, as discussed in the previous section. The inspiration of our new methodology is to track down a more straightforward method for contrasting the constructions of the reference and the mutilated signs.

a) New Method
a new framework for the design of image quality measures was proposed, based on the assumption that the human visual system is highly adapted to extract structural information from the viewing field It follows that a proportion of primary data change can give a decent guess to apparent picture contortion.
This new way of thinking can be best perceived through correlation with the error sensitivity philosophy. First, the error sensitivity approach estimates apparent mistakes to measure picture debasements, while the new way of thinking thinks about picture debasements as seen changes in underlying data. An example is discussed below-where the original "Boat" image is altered with different distortions, each adjusted to yield nearly identical MSE relative to the original image. Even after this, the images can be seen to have drastically different perceptual quality. With the error sensitivity philosophy, it is difficult to explain why the contrast-stretched image has very high quality in consideration of the fact that its visual difference from the reference image is easily discerned. But it is easily understood with the new philosophy since nearly all the structural information of the reference image is preserved, in the sense that the original information can be nearly fully recovered via a simple pointwise inverse linear luminance transform (except perhaps for the very bright and dark regions where saturation occurs). On the other hand, some structural information from the original image is permanently lossed in JPEG compressed. And the blurred images and thus they are given lower quality scores when compared to contrast stretched and mean shifted images.

Second, the error-sensitivity paradigm is a bottom-up approach, simulating the function of relevant early-stage components in the HVS. The new paradigm is a top-down approach, mimicking the hypothesized functionality of the overall HVS. This, on the one hand, avoids the suprathreshold problem mentioned in the previous section because it does not rely on threshold psychophysics to quantify the perceived distortions. On the other hand, the cognitive interaction problem is also reduced to a certain extent because probing the structures of the objects being observed is thought of as the purpose of the entire process of visual observation, including high level and interactive processes. Third, the problems of natural image complexity and decorrelation are also avoided to some extent because the new philosophy does not attempt to predict image quality by accumulating the errors associated with psychophysically understood simple patterns. Instead, the new philosophy proposes to evaluate the structural changes between two complex-structured signals directly.

b) Structural SIMilarity(SSIM) Index

We construct a specific example of a structural similarity quality measure from the perspective of image formation. A previous instantiation of this approach was made in and promising results on simple tests were achieved. In this paper, we generalize this algorithm, and provide a more extensive set of validation results. The luminance of the surface of an object being observed is the product of the illumination and the reflectance, but the structures of the objects in the scene are independent of the illumination. Consequently, to explore the structural information in an image, we wish to separate the influence of the illumination. We define the structural information in an image as those attributes that represent the structure of objects in the scene, independent of the average luminance and contrast. Since luminance and contrast can vary across a scene, we use the local luminance and contrast for our definition.

First, the luminance of each signal is compared. Assuming discrete signals, this is estimated as the mean intensity then Second, we remove the mean intensity from the signal. In discrete form, the resulting signal $x - \mu_x$ corresponds to the projection of vector x onto the hyperplane We use the standard deviation (the square root of variance) as an estimate of the signal contrast. An unbiased estimate in discrete form The contrast comparison $c(x, y)$ is then the comparison of σ_x and σ_y.

Third, the signal is normalized (divided) by its own standard deviation, so that the two signals being compared have unit standard deviation. Finally, the three components are combined to yield an overall similarity measure
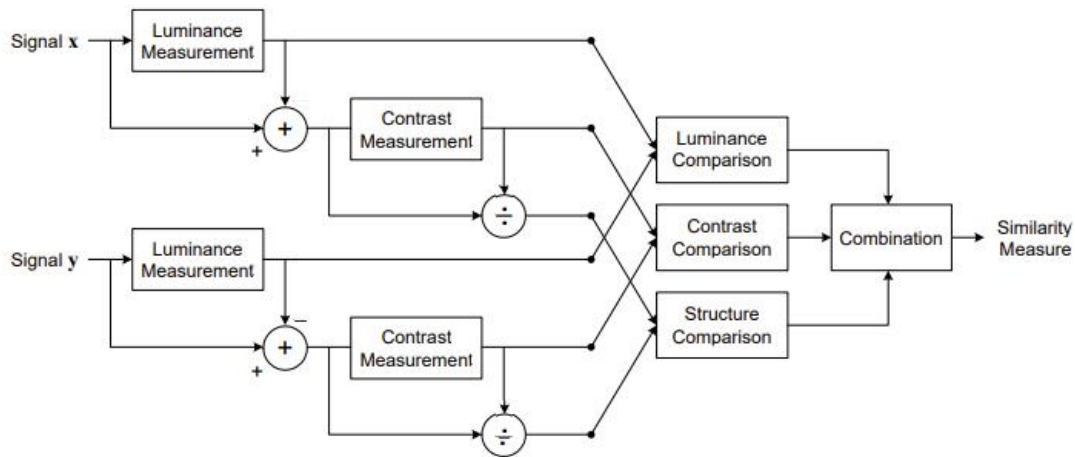
Fig. 3. Diagram of the structural similarity (SSIM) measurement system.

c) Image Quality Assessment using SSIM index

For image quality assessment, it is useful to apply the SSIM index locally rather than globally. First, image statistical features are usually highly spatially non-stationary. Second, image distortions, which may or may not depend on the local image statistics, may also be space-variant. Third, at typical viewing distances, only a local area in the image can be perceived with high resolution by the human observer at one time instance And finally, localized quality measurement can provide a spatially varying quality map of the image, which delivers more information about the quality degradation of the image and may be useful in some applications.

WORKING OF PROJECT

[1]
```python
from skimage.metrics import structural_similarity
import imutils
import cv2
from google.colab.patches import cv2_imshow
```

[2]
```python
image_one = cv2.imread("/content/image_part_001.jpg")
image_two = cv2.imread("/content/image_part_002.jpg")

gray1 = cv2.cvtColor(image_one, cv2.COLOR_BGR2GRAY)
gray2 = cv2.cvtColor(image_two, cv2.COLOR_BGR2GRAY)
```

[3]
```python
(score, diff) = structural_similarity(gray1, gray2, full=True)
diff = (diff*255).astype("uint8")
print("SSIM: {}".format(score))
```

Output-

```
SSIM: 0.9153203351442917
```

[4]
```python
thresh = cv2.threshold(diff, 0, 128, cv2.THRESH_BINARY_INV|cv2.THRESH_OTSU) [1]
cnts =cv2.findContours(thresh.copy(), cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE)
cnts = imutils.grab_contours(cnts)
```

[5]
```python
no_of_differences = 0
for c in cnts:
  (x,y,w,h) = cv2.boundingRect(c)
  rect_area = w*h
  if rect_area > 10:
    no_of_differences += 1
    cv2.rectangle(image_one,(x,y),(x+w, y+h),(0,0,255), 2)
    cv2.rectangle(image_two,(x,y),(x+w, y+h),(0,0,255), 2)
print("No of differences= ", no_of_differences)
cv2_imshow(image_one)
cv2_imshow(image_two)
cv2_imshow(diff)
cv2.waitKey(0)
```

Output-
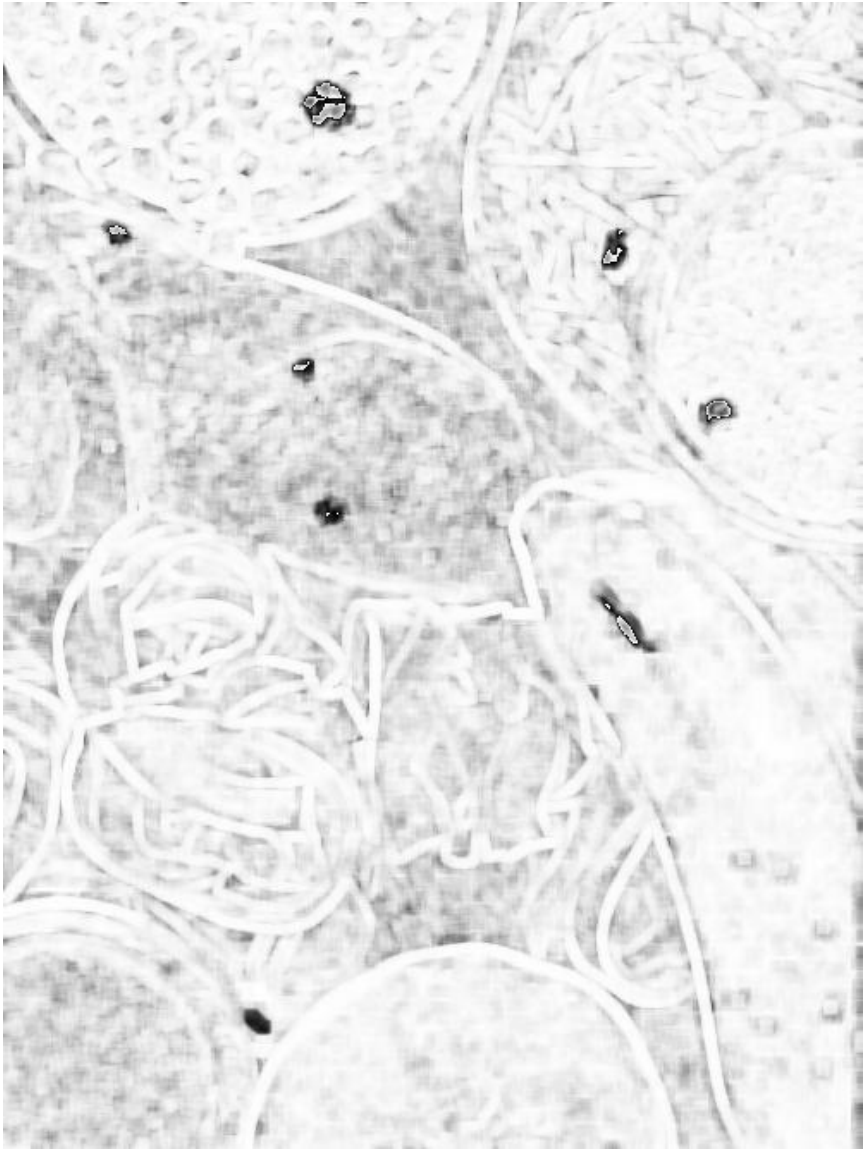
```
No of differences=  545
```

OUTPUT

a) Test Case 1

Output of Test Case 1[1]

Output of Test Case[2]

b) Test Case 2

Output of Test Case 2 [1]

Output of Test Case 2[2]

CHAPTER 4
CONCLUSION AND FUTURE SCOPE

In conclusion, this research proposes a Model used to visualise image difference or Structural Similarity based on various Image Computation Algorithms. Following the data collecting, greyscale conversion , building and implementing the model, this model can differentiate between two images and show there differences.

In this paper, we briefly discussed the work's motivation. The model's learning and performance task was then demonstrated. The method has attained a reasonable level of accuracy using basic Image Computational tools and simplified methodologies.
It may be used for a wide range of purposes like analyzing images has a varied number of jobs in present time, it can be used in the forensic field for analysis of set of differentiating between a set of fingerprints or in hospitals to compare a set of a patient's reports or even to make different visual challenging games, there are more possibilities related to statistical study of data and analysis. It can be used in the field of astronomy as well to locate objects that fluctuate in brightness or move against the star field. The given project can be used to verify the authenticity of an image to avoid fraud and prevent further scams as well,
Many hands will will reach out to use this service. The implemented approach will make a contribution to various types of applications that can be worked on with this project.

CHAPTER 5
REFERENCES

[1] E. Baudrier and A. Riffaud. A method for image local-difference visualization. In *Proceedings of the 9th International Conference on Document Analysis and Recognition*, volume 2 of *ICDAR '07*, pages 949–953, Curitiba, Paran, Brazil, Sept 23-26 2007. IEEE Computer Society.

[2] S. Bruckner and T. Moller. Result-driven exploration of simulation parameter spaces for visual effects design. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1467–1475, Oct 2010.

[3] H. Chen, L. Wei, and C. Chang. Nonlinear revision control for images.*ACM Trans. Graph.*, 30(4):105:1–105:10, July 2011.

[4] B. Clarkson and A. Pentland. Unsupervised clustering of ambulatory audio and video. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, ICASSP '99, pages 3037– 3040, Phoenix, Arizona, USA, Mar 15-19 1999. IEEE Computer Society.

[5] A. Cockburn, A. Karlson, and B. B. Bederson. A review of overview+detail, zooming, and focus+context interfaces. ACM Comput. Surv., 41(1):2:1–2:31, Jan 2009.
[6] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. IEEE Trans. Pattern Anal. Mach. Intell., 24(5):603–619,May 2002.
[7] M. DaSilva, S. Zhang, C. Demiralp, and D. H. Laidlaw. Visualizing the differences between diffusion tensor volume images. In Proceedings of the ISMRM Workshop on Diffusion MRI: What Can We Measure?, ISMRM '02, pages 237–238, St. Malo, France, Mar 10-12 2002.

[8] S. M. Drucker, G. Petschnigg, and M. Agrawala. Comparing and managing multiple versions of slide presentations. In Proceedings of the 19th.Annual ACM symposium on User Interface Software and Technology, UIST '06, pages 47–56, Montreux, Switzerland, Oct 15-18 2006. ACM.

[9] R. O. Duda and P. E. Hart. Pattern Classification and Scene Analysis.

[10] M. B. Eisen, P. T. Spellman, P. O. Brown, and D. Botstein. Cluster analysis and display of genome-wide expression patterns. Proceedings of theNational Academy of Sciences, 95(25):14863–14868, Dec 1998.

[11] D. M. Eler, M. Y. Nakazaki, F. V. Paulovich, D. P. Santos, M. C. F Oliveira, J. E. S. B. Neto, and R. Minghim. Multidimensional visualization to support analysis of image collections. In Proceedings of the XXI Brazilian Symposium on Computer Graphics and Image Processing, SIBGRAPI '08, pages 289–296, Washington, DC, USA, 2008. IEEE Computer Society.

[12] C. Fowlkes, C. L. Hendriks, S. Ker¨anen, G. Weber, O. R¨ubel, M. Huang, S. Chatoor, L. Simirenko, A. DePace, C. Henriquez, A. Beaton, R. Weiszmann, S. Celniker, B. Hamann, D. Knowles, M. Biggin, M. Eisen, and J. Malik. A quantitative spatiotemporal atlas of gene expression in the drosohpila blastoderm. Cell, 133:364–374, 2008.

[13] D. Gareth and C. Min. Visualising video sequences using direct volume rendering. In Proceedings of Vision, Video and Graphics, VVG '03, pages 103–110, University of Bath, UK, Jul 10-11 2003.

[14] M. Gleicher, D. Albers, R. Walker, I. Jusufi, C. D. Hansen, and J. C. Roberts. Visual comparison for information visualization. Information Visualization, 10(4):289–309, Oct 2011.

[15] H. Hagh-Shenas, V. Interrante, C. Healey, and S. Kim. Weaving versus blending: a quantitative assessment of the information carrying capacities. of two alternative methods for conveying multivariate data with color. In Proceedings of the 3rd Symposium on Applied Perception in Graphics and Visualization, APGV '06, pages 164–164, Boston, Massachusetts,USA, July 28-29 2006. ACM.

[16] B. Hollingsworth, S. Reichenbach, Q. Tao, and A. Visvanathan. Comparative visualization for comprehensive two-dimensional gas chromatography. J Chromatogr A, 1105(1-2):51–8, 2006.

[17] J. Lubin, "The use of psychophysical data and models in the analysis of display system performance," in Digital images and human vision (A. B. Watson, ed.), pp. 163–178, Cambridge, Massachusetts: The MIT Press, 1993.

[18] D. J. Heeger and P. C. Teo, "A model of perceptual image fidelity," in Proc. IEEE Int. Conf. Image Proc., pp. 343–345,1995.

[19] E. P. Simoncelli, W. T. Freeman, E. H. Adelson, and D. J. Heeger, "Shiftable multi-scale transforms," IEEE Trans. Information Theory, vol. 38, pp. 587–607, 1992.

[20] A. B. Watson, "DCT quantization matrices visually optimized for individual images," in Proc. SPIE, vol. 1913, pp. 202–216, 1993.

[21] A. B. Watson, J. Hu, and J. F. III. McGowan, "DVQ: A digital video quality metric based on human vision," Journal of Electronic Imaging, vol. 10, no. 1, pp. 20–29, 2001.

[22] A. B. Watson, G. Y. Yang, J. A. Solomon, and J. Villasenor, "Visibility of wavelet quantization noise," IEEE Trans. Image Processing, vol. 6, pp. 1164–1175, Aug. 1997.

[23] A. P. Bradley, "A wavelet visible difference predictor," IEEE Trans. Image Processing, vol. 5, pp. 717–730, May 1999.

[24] Y. K. Lai and C.-C. J. Kuo, "A Haar wavelet approach to compressed image quality measurement," Journal of Visual Communication and Image Representation, vol. 11, pp. 17–40, Mar. 2000.

[26] C. J. van den Branden Lambrecht and O. Verscheure, "Perceptual quality measure using a spatio-temporal model of the human visual system," in Proc. SPIE, vol. 2668, pp. 450–461, 1996.