ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 9, Issue 7, July 2022

# Evaluating Image Redundancies by Analyzing Pixel Based Similarity Ratios

[1] Janhavi Patil,* [2] Saumitra Jagdale

[1] [2] Open Cloudware.

Corresponding Author Email: [1] janhavi886@gmail.com, [2] saumitra@opencloudware.com

***Abstract*— While bifurcating a video at a specific rate and extracting frames from it, there might be an occurrence of repetitive similar frames which can cause redundancy of data. As a result, computation time increases, thus causing problems in determining the most accurate frame for further computation such as content extraction, feature extraction, etc. The intent of this paper is to introduce a solution to detect such lightly identical frames from the input video as well as to detect similar images having equal dimensions from a database. The proposed detection algorithm in this research paper mainly focuses on converting frames to NumPy array objects and then performing array-based numerical operations on them, which also involves comparing a pair of images pixel-by-pixel to determine the similarity ratio between them. It returns a new dataset of frames having minimal duplicate frames, and a compressed version of the original video can be re-constructed using this dataset of frames. The key concept is to eradicate anomalies caused by the redundancy of frames.***

***Index Terms*—Frames, Image Processing, Image-Pixels, Video Stream Computations, Image Evaluation**

## I. INTRODUCTION

A digital video stream is made up of individual frames, each one representing a time slice. While extracting these frames as images from the video, depending upon the frame rate, similar images can be extracted from the video stream. Saving these images in the device memory would take up unnecessary space and cause redundant anomalies. A 3D photo is a new emerging type of Image Optimization in the market. It adds stereoscopic vision; i.e., two images are shown simultaneously – one to each eye. It makes a photo more immersive. 3D image is just a 2D image with additional depth or perspective. Similarly, when it comes to a 2D video, frames are created to define sequences, whereas a 3D video has depth which makes it look more realistic. When it comes to processing such 3D[1] videos in real time, it would bring enormous technical problems involving storage issues as well. This algorithm aims to work on such technical issues. For working with such an enormous amount of data at real time, it would take a huge amount of time to process the video. Using this algorithm, working in real-time would become comparatively easier as, this algorithm can be used as a filtering mechanism to pre-process the video and delete the redundant frames and then process it further. Additionally, when one tries to read these images and extract data from them, the same data is extracted multiple times which causes inaccuracy. For instance, while extracting text from the content-based video, it may return the same text as the output multiple times when it undergoes optical character recognition.

## II. LITERATURE SURVEY

### *A. Adaptive Image-Based Plagiarism Detection*

This approach uses a combination of Convolutional Neural Network, Perceptual hashing, and Optical Character recognition. It also uses positional text matching, k-gram matching, and Ratio[2] hashing for specifically detecting semantically similar but visually different bar charts. All

these analytical methods are applied independently to the input and each of them returns a method-specific distance of input to make the conclusion and the relative delta in the distance is calculated.

Mostly suitable for academic documents, this approach makes use of PDF documents as an input, and converts them to JPEG files (above 7500 bytes) using an open-source library called poppler and then various image processing techniques are applied on it. To reduce computational load, this approach first makes use of CNN to classify images based on their similarity and suitability for being analysed using different analysis methods (whether the images are exactly similar, closely similar, etc). [3]The final evaluation depicts that the occurrence of a threshold value of 0.5 for an image shows that it is suspicious of plagiarism, while a value greater than 0.75 shows that it is highly suspicious.

### *B. Robust image hashing based on local features for image authentication*

A hash value of an image is a value that uniquely identifies the image based on its contents. This method makes the use of the image hashing method by using the features of selected effective blocks.[4] The input image is pre-processed firstly and resized to a standard size using bilinear interpolation. The image is divided into non-overlapping blocks. The

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 9, Issue 7, July 2022

number of sifting points is calculated for a block. If a block has more SIFT points, it relatively has more effective information in it. An intermediate hash value is calculated for each block using the colour, [5] texture, and shape features of these blocks. The final hash value is calculated using pseudo-randomly permuting the intermediate hash value. The similarity between the final hash values is calculated using Euclidean distance. This approach is also capable of detecting forged foreign objects in an image, it can also be used to find whether the input image is tampered or not.

### *C. Image similarity using Mutual information of regions*

This approach extends the existing mutual information algorithm into regional mutual information of a pair ofimages and is closely related to joint entropy. Region mutual information uses one pixel and its neighbor pixel to represent an image. Given two images, the individual entropies of those images and the [7] combined joint entropies are calculated. The entropy here is the measure of uncertainty in a random variable. The joint entropy decreases when there is a one-to-one mapping between the pixels of image A and their counterparts in image B and this joint entropy increases when the statistical relationship between image A and B weakens.

- pixels among the edges are ignored with the assumption that they won't have a very pronounced effect on the final entropy
- Mutual information fails to take geometry and considers only pixel values and not pixel positions

## III. METHOD

The main purpose of this algorithm is to identify the similarity between two images and then delete the redundant images. This algorithm can be broadly divided into two categories one for images that are extracted from a video and the latter being for images that are directly taken from the storage.

Images from the device when taken as input, may or may not have similar dimensions. Dimensions in this case are the length and width of a digital image. The shape of the NumPy array objects that are created for an image corresponds to the shape of the image as [9] well as its depth. Hence, for images with different dimensions, their arrays would also have unequal shapes and dimensions. In this case, it is not possible to calculate the difference between these arrays as it would be undefined. Hence, we can aim to normalize the dimensions of the two images by resizing them.

### *A. Image Processing*

When an image is converted into an array it has 3 dimensions, height width and depth where depth represents the colour.[11] When a computer retrieves an image, it will first break it down into three separate channels: red , green, and blue. For the development of this algorithm the colour dimension doesn't help us much, thus images are converted into grayscale images first before converting them into arrays. Converting the image into a grayscale image not only makes the code simpler but also makes the image analysis simpler. The grayscale intensity of the pixel is stored as an 8-bit integer giving 256 possible different shades of grey from black to white. [6] If the levels are evenly spaced then the difference between successive graylevels is significantly better than the grey level resolving power of the human eye. Gray-scale images are very common, in part because much of today's display and image capture hardware can only support 8-bit images. In addition, grayscale images are entirely sufficient for many tasks and so there is no need to use more complicated and harder-to-process colour images.

### *B. NumPy array objects*

Computers only understand data in the form of binary numbers. These numbers are often stored as scalars, vectors or in arrays and matrices. Each element of an array, corresponds to a pixel value in the original image. This means that the array's dimensions [8] are equal to the resolution of the image. Therefore, a color image with a resolution of 1920 x 1080 pixels will be broken down to 3 arrays with the same dimensions.

The values stored in each cell of the array represents the intensity of that channel for the corresponding pixel and a NumPy array is a grid of homogenous values. NumPy provides us with an array() function that allows us to create an array for performing numerous operations and computations over it.

### *C. FrameRate*

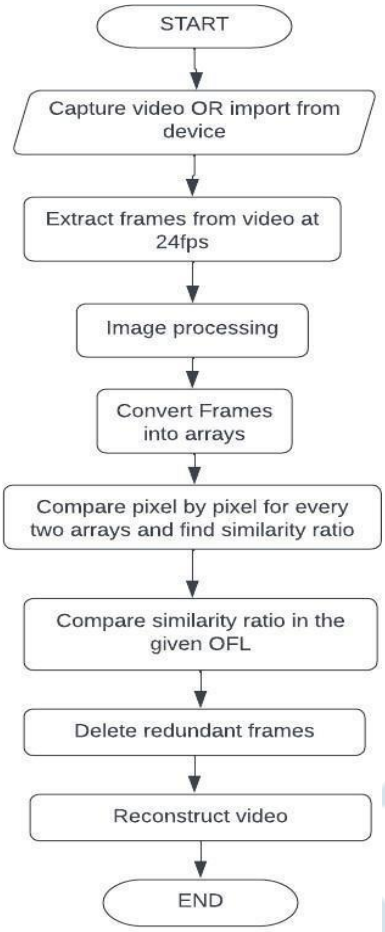
Images that are extracted from a video will by default have similar dimensions hence there's no need for normalization. The foremost step involves reading the video and bifurcating the video into frames at a specified frame rate.

Frames per second are the number of frames that get swapped across a line in a video. 24 fps is the minimum necessary for the human brain to recognize the series of pictures as a video hence the input video is bifurcated at the rate of 24 fps [10]. During the experimentation phase, the frame rate was chosen to be 1 frame per second which led to complications as 1 second was too long duration to miss any important frame of a video. Example, there might be a video case where within a second a PDF file is scrolled very quickly over the pages in a screen recording or video being used for an third umpire decision in the game of cricket, in such a scenario, the frame rate of 1 frame per second could lose an important section of the video and it won't be captured in the further analytical steps.

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 9, Issue 7, July 2022




*D. ReadingImages*

To read images and extract data from them, the efficiency of the OpenCV library is better compared to the Pillow library. OpenCV is 1.4 times faster than the pillow library.[3] For extracting frames from videos, we might deal with a huge number of frames depending upon the length of the video and the frame rate. For such large data, OpenCV is faster in case of processing speed and reliability.

*E. ImageProcessing*

Images are pre-processed so that we can efficiently extract insights from them. Firstly, they are converted into gray-scale so that there's no 3rd dimension and the image is hence converted to a 2D image. Upon converting the image into gray-scale or binary, image thresholding is performed over it for easier analysis.

*F. Similarity Ratio*

Converting these images into array objects returns a 2D array in which the dimensions represent the height and width of the image. Each element of the array represents the pixel value ranging from 0-to 255.[13] When 2 arrays of pixel values are subtracted, they return another array in which the zero elements represent that the specific pixel is common for both the images. By counting the number of zero elements in the returned array we can find out the similarity ratio which is the ratio of the total elements in the subtracted array and the number of zero elements in the subtracted array.

$$Similarity\ Ratio = \frac{Total\ Elements}{Zero\ Elements} \quad (1)$$

Similarity ratios for the same images come out to be 1 if the images are of the same dimensions. Constructing a dataset of the similarity ratios obtained from the input of 4 videos we can observe that the average of all the similarity ratios for similar images comes out to be 1.05002 and the standard deviation from the mean of the sample comes out to be 0.0688. Using this [12] statistical data when the normal distribution for all the datasets is calculated we find out that the similarity ratios of similar frames lie in the range of 1.00 to 1.09. If the similarity ratio for a pair of images comes out to be greater than 1.09, we can conclude that the frames are different. This dataset also contains some anomalies as some frames when checked were similar but their similarity ratio was greater than expected.

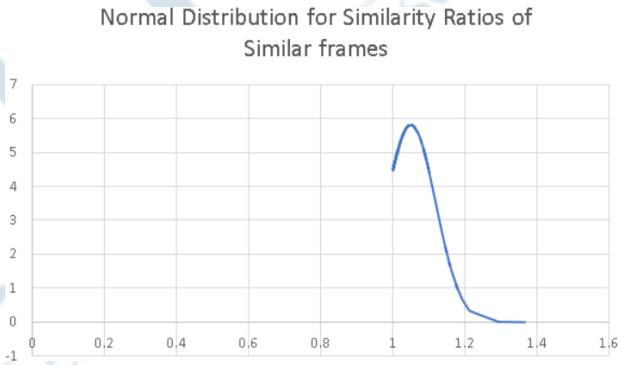



**Fig 1:** Normal Distribution for Similarity Ratios of Similar Images

For images taken out from a video, for exact similar frames, the similarity ratio ranges between 1.00 - 1.09

Anomaly 1: for text-based images, even if a pair of images contain the same text in them but the background, font, font size, etc is different, the similarity ratio comes out to be greater than 1.1, thus concluding that the images are not similar.

*G. OFL*

Using the above data, we can find whether two frames are similar or not. If the similarity ratio lies in the predefined range, we can delete one of the two frames and this could be done for all adjacent frames until we get a dataset of frames with almost no redundant frames. After this, we can reconstruct the video with the remaining frames which are not redundant amongst each other. To delete the redundant frames, the algorithm involves initially dividing the total number of frames into X lots where X is the OFL (Optimum Frame Lot). The similarity ratio frame of every frame is compared with the similarity ratio of every other frame and we jump into the next OFL only when we get a distinct frame

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 9, Issue 7, July 2022

and until then we keep deleting the redundant frames. Once we get a proper dataset of distinct frames, a new video using these frames is created. To get a proper OFL value for this algorithm, multiple OFL values were tried for multiple input videos. The newly constructed videos were manually analyzed and the data was accumulated using which we find the exact OFL value to be 13% of the number of frames present in the video.

| Number Of Frames | OFL value | Percentage Value |
|---|---|---|
| 71 | 9 | 12.676 |
| 69 | 10 | 14.49 |
| 1005 | 104 | 10.04 |
| 75 | 9 | 12 |

**TABLE 1.** OFL values

## IV. POTENTIAL IMPLEMENTATION

There are various imaging applications that take an image or a frame from a video as an input. Such applications need some kind of similarity measure to detect similar images and frames. The presented algorithm is helpful in determining the similarity ratio between two images which is based on comparing each pixel of one image with the corresponding pixel of another image. If a particular pixel at a particular position in both the images correspond to the same value, the difference between them would be zero. On performing this operation for all the pixel values, we can determine the extent of similarity between two images.


Image 1


Image 2

Image 3

Image 4

**TABLE 2.** Similarity Ratios for Images with different Dimensions

| Serial Number | Zero Elements | Similarity Ratio |
|---|---|---|
| Image 1, Image2 | 22176 | 6.127 |
| Image 3, Image 4 | 30594 | 47.008 |

This is also the fact why we cannot determine the similarity between two images having different dimensions as the shape of the matrix for each image would be different. This would return us an undefined value. Resizing both the images to a particular normalized dimension could be one of the possible solutions. But, in this case the range of similarity ratio will highly vary depending upon the image. But for exact similar images the similarity ratio will be exact unity. When the Matrices of two images are subtracted it results in a new matrix with similar dimensions. The number of zero elements in the new matrix represent the common region or the common pixel value between the two images. This algorithm will also be beneficial for feature extraction and also in detecting similar features between images. Also, it can be used for detecting empty space from images.

To verify this similarity detection algorithm, around 6 videos, each of it when bifurcated produced a dataset of 1500+ images. On finding the image similarity between each of these frames and then deleting the redundant frames, we are able to generate a new video which is reconstructed using the nonredundant frames. The length of the new video comes out to be very accurate with very little or almost no redundant frames. It also gets compressed and the length and size of the video is greatly reduced.

This algorithm will efficiently reduce computation power in many applications which would directly help in making the applications be faster. Predictive Maintenance is implemented using machine learning algorithms which helps in detecting potential equipment failures and also recommends the necessary actions that are to be taken. Hence, this algorithm also indirectly promotes hardware durability by some extent as it helps in acting as a filtering mechanism and conserving the device's power consumption and the necessary computations that are to be done.

## V. SCOPE

In near future, this algorithm can be beneficial in optical character recognition where we are supposed to recognize text in a text-based video and extract it. Also, this algorithm can be used to compress video files, thus, reducing the length as well as the size of the video which will, in turn, help manage the storage of a device. In the case of screen recording the device, even a minute of gap may cause the occurrence of more than 1440 redundant frames if the video is to be saved, and this might take up a huge amount of space in the device. The proposed algorithm can be performed on such large videos to compress them which will help in saving up space.

To minimize the storage as well as the time required to fetch information in the cloud, this algorithm would help in compressing the videos before saving them on cloud.

Moreover, it will also be beneficial in Metaverse and Virtual Reality applications. These applications basically work on 3D models which involves 3D videos. These videos

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 9, Issue 7, July 2022

store some common relevant and necessary information on the Cloud. Not only does storing on Cloud increase the computation time, but along with it might deteriorate the quality of the video.

For Edge AI, where the AI computation is done nearer to the network rather than storing them on Cloud. With Edge AI, data does not need to be sent to another machine to do the processing, instead the device itself handles the computations. While it proves its efficiency when it comes to dealing with cloud compute strain, there is also an increased chance of failure as most of the processing is done at real-time and a huge amount of data is to be processed by a device. In such a case too, eradicating redundant data would help in dealing with the problems related to failure of devices.

## VI. CONCLUSION

The paper proposes an Image Similarity Detection algorithm which is based on comparing two images pixel-by-pixel and then finding the similar pixels and hence determine the similarity ratio. The results of the experiments done shows that the proposed algorithm is capable of achieving high quality similarity detection. In near future, this algorithm can be expanded for images which by default don't have the similar dimensions. We could also aim to find for newer ways to normalize the dimension between two images. It could also be expanded to enhance the accuracy for text-based images having the same content but differing only by font and background. The implementation of this model when combined with several other existing applications would help in enhancing the speed, durability and functioning of these applications. To serve its purpose, this algorithm can be combined with character recognition and then be curated into a device that is capable of scanning the content from the camera present in the device and capturing it in the form of a video and return us the content present in the video. The similarity detection method would increase the efficiency of such models.

## REFERENCES

[1] Norman Meuschke, Christopher Gondek, Daniel Seebacher, Corinna Breitinger, Daniel Keim, Bela Gipp - An adaptive Image- based plagiarism detection approach- JCDL '18 - (2018)

[2] Qin Lv, Moses Charikar , kai Li - Image similarity search with compact data structures

[3] I. S. Jacobs and C. P. Bean, "Fine particles, thin films and exchange anisotropy," in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.

[4] Mrs. Asha K H, Manjunathswamy B E, Mrs. Chaithra A S – Python - Based Image processing - International journal of scientific research and management (2021)

[5] Daniel B. Russakoff, Carlo Tomasi, Torsten Rohlfing, and Calvin R. Maurer, Jr. - Image similarity using mutual information of regions.

[6] Yan ZHAO and Qian ZHAO - Robust Image Hashing Based on local Features for image Authentication - International Conference on Software, Multimedia and communication Engineering - (SMCE 2017)

[7] Bernice E Rogowitz, Thomas Frese, John R Smith, Charles A Bouman, Edward B Kalin - Perpetual Image similarity experiments – Human Vision and Electronic Imaging III 3299 - 1998

[8] Gal Chechik, Varun Sharma, Uri Shalit, Samy Bengio - Large Scale Online Learning of Image Similarity through Ranking - Journal of Machine learning Research 11 (3) - 2010

[9] Oliver Zendel. Christian Zinner - NAPHash: Efficient Hash to reduce Dataset Redundancy - 2021 International Conference on Electrical, Computer and Energy technologies(ICECET) - 2021

[10] Raman Maini, Himanshu Aggarwal - Study and Comparison of various image edge detection techniques - International Journal on Image Processing (IJIP) - 2009

[11] Haixia Chen, Xinyi Huang, Wei Wu, Yi Mu - Efficient and secure image authentication with robustness and versatility - Science China Information Sciences

[12] Yan Zhao, Xiaoran Yuan - Perceptual Image Hashing Based on Color Structure and Intensity Gradient - 2019

[13] Marcin Iwanowski, Arkadiusz Cacko, Grzegorz Sarwas – Comparing Images for Document Plagiarism Detection - Intyernational Conference on Computer Vision and Graphics - 2016