Name: - Bhaskar Gautam

Section: - IT (Purple)

Class Roll No. 26

University Roll No. 2004979

Proposed Algorithm

Input

Img={img1,img2,img3……}.

Output

Determine beauty of images.

Algorithm

Step1:- Select all images and get all images in a single directory (say in E: /CheckImage).

Step2:-Check that all images are in correct format if not then reject that image.

Step3:- Normalize them to same size (1800 * 1800).

Step4:- Change the all images in single orientation that is either in Portrait or Landscape Mode.

Step5:- Read all images by read() function.

Step6:- Determine best images by detImage(*ptr) function and call this function until last Image is not determined.

Step7:- Show the Beauty of image with the help of displayFun().

Step8:- End of Program.

Implementation

In read() function, we read and store all images in stack

{

For ex:

As we done In Matlab we read all the images by imread(Path) function which read particular Image.

for (i=1 to Img.length)

{

Read images and store it in stack.

}

ptr is the stack pointer which point the images in stack.

}

In detImage(*ptr) function we determine images which describe the all events

{

1st Approach-------

- Get one in memory with the help of *ptr then determine the number of pixels

  Present in that image (with the help of whosf- which show info which return the size

  Of which is the number of pixels in image).

- Get RGB value of each pixel with the help of function (rgb(:,:,1)->return red value, rgb(:,:,2)->return green value, rgb(:,:,3)->return blue value)
- Take sum of the of all rgb value of each pixel and calculate of average it.
- Number of colour determine by Formula $(2^{bit})^3$.
- If the average value is nearly equal to 16677216 then that image have more beauty in that image.
- If average value is in range 10000000 to 16677216, then that image contain average beauty.
- If average value is less than that range, then that image is rejected.

2nd Approach--------

Check that if image is too bright then return that image has less beauty.

(

If image contain brightness more than 50% and less than 10%, than that image contain less beauty

If range is in between 20 % to 40 % than that image contain more beauty

)

3rd Approach-------

Check that if image is too contrast then follow same procedure as in 2nd approach.

4th Approach-------

Check that if image is too saturated then follow same procedure as in 2nd approach.

5th Approach-------

Check that if an image contain too Shadow element then follow same procedure as in 2nd approach.

6th Approach-------

By determine the histogram of an image which is made with the help of pixels and colours, from this we get the threshold value

For ex:-

If $f(x, y) > T$ then $f(x, y) = 0$ else $f(x, y) = 255$

Where T is the threshold value

And if (histogram peaks are tall, narrow, symmetric, and separated by deep valleys)
Then that image contain more beauty else less beauty.

}

In displayFun(),we display the beauty of image which is calculated with the help of detImage() function.

For ex:-

```
For (i=1 to Img.length)
{
    Imshow("pointer of an Image")
}
```

NOTE:-

***We assume that all images are in color not in Black and white.**

***In Example section we use MATLAB function.**