Int. j. adv. multidisc. res. stud. 2024; 4(3):943-946

Received: 14-04-2024
Accepted: 24-05-2024

International Journal of Advanced Multidisciplinary Research and Studies

ISSN: 2583-049X

The Effect of Grayscale, CLAHE Image and Filter Images in Convolution Process

Panca Mudjirahardjo
Deptartment of Electrical Engineering, Faculty of Engineering, Universitas Brawijaya, Indonesia

Corresponding Author: **Panca Mudjirahardjo**

Abstract

Convolution process for Convolutional Neural Network (CNN) need a source image and a kernel. Kernel is a filter image with a certain size, such as 3×3, 5×5, 7×7, and so on. The source image or input image is an image in RGB or processed image. The processed image are in color model or grayscale format. The color models are such as RGB, HSV, CMYK and YIQ. Whereas the grayscale format such as Contrast Limited Adaptive Histogram Equalization (CLAHE). In this paper, we demonstrate the effect of grayscale and CLAHE formatted image and several filter images in convolution process, both for static image and real time frame captured by a camera. The experiment is conducted using programming language C++ and openCV library. The experiment results show the distinction between the various of input images and various of the filter images.

Keywords: Grayscale, CLAHE, Convolution, Kernel, Filter Images

Introduction

In recent days, computer vision takes the interest task in many fields. In computer vision, the source image or video before fed to classifier, it needs to be processed to enhance the image and extract the feature. Feature extraction stage is required to show the different points of object classes and to reduce the computation time. One of the interest platform for object detection and classification is CNN based. In CNN consists of several layer of convolution process to extract feature.

The distinctive object will increase the classification accuracy. This property is often difficult to obtain because the light intensity on an object is not uniform. Sometimes the objects get enough light, sometimes not enough light. One way to overcome this lack of light is to process histogram images, namely histogram equalization. Histogram equalization stretches the histogram distribution so that it does not accumulate at certain light intensities. One of these histogram equalization methods is CLAHE.

Object's feature can be obtained from the convolution process of the input image with a filter. Usually, the output of this convolution process is the detection of edges of an object in certain directions according to the filter used. So that a filter corresponds to certain directions. These particular directions can be obtained by increasing the amplitude of the desired direction and suppressing the amplitude of the undesired direction.

The proposed method

In this section, we briefly explain the proposed method to study the effect of image format and filter images in convolution process. The method is shown in Fig 1.

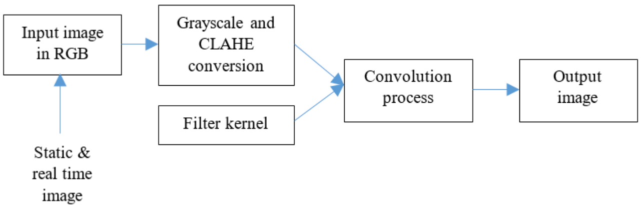


Fig 1: The proposed method of this study

In Fig 1, the input image from static image from a file and real time frame captured by a camera. Then the RGB image converts into grayscale image. This grayscale image, then convert into CLAHE image for contrast enhancement with histogram equalization. Convolution process requires two images, this is done by an input image in CLAHE format and a filter kernel. The result of convolution process can be shown in output image.

Contrast Limited Adaptive Histogram Equalization (CLAHE)

Adaptive histogram equalization (AHE) is an image pre-processing technique used to improve contrast in images. It computes several histograms, each corresponding to a distinct section of the image, and uses them to redistribute the luminance values of the image. It is therefore suitable for improving the local contrast and enhancing the definitions of edges in each region of an image. However, AHE has a tendency to over amplify noise in relatively homogeneous regions of an image. A variant of adaptive histogram equalization called contrast-limited adaptive histogram equalization (CLAHE) prevents this effect by limiting the amplification.

Ordinary AHE tends to overamplify the contrast in near-constant regions of the image, since the histogram in such regions is highly concentrated. As a result, AHE may cause noise to be amplified in near-constant regions. Contrast Limited AHE (CLAHE) is a variant of adaptive histogram equalization in which the contrast amplification is limited, so as to reduce this problem of noise amplification [1-5].

In CLAHE, the contrast amplification in the vicinity of a given pixel value is given by the slope of the transformation function. This is proportional to the slope of the neighborhood cumulative distribution function (CDF) and therefore to the value of the histogram at that pixel value. CLAHE limits the amplification by clipping the histogram at a predefined value before computing the CDF. This limits the slope of the CDF and therefore of the transformation function. The value at which the histogram is clipped, the so-called clip limit, depends on the normalization of the histogram and thereby on the size of the neighborhood region. Common values limit the resulting amplification to between 3 and 4.

It is advantageous not to discard the part of the histogram that exceeds the clip limit but to redistribute it equally among all histogram bins [1].

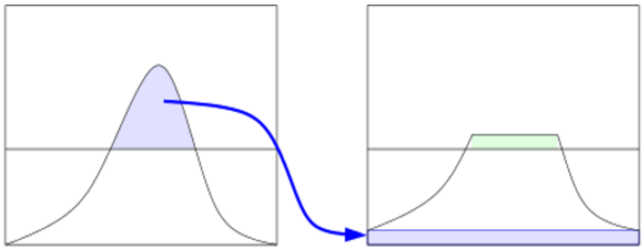


Fig 2: The histogram distribution in CLAHE [1]

The redistribution will push some bins over the clip limit again (region shaded green in the Fig 2), resulting in an effective clip limit that is larger than the prescribed limit and the exact value of which depends on the image. If this is undesirable, the redistribution procedure can be repeated recursively until the excess is negligible.

The CLAHE algorithm has three major parts: Tile generation, histogram equalization, and bilinear interpolation. The input image is first divided into sections. Each section is called a tile. Histogram equalization is then performed on each tile using a pre-defined clip limit. Histogram equalization consists of five steps: Histogram computation, excess calculation, excess distribution, excess redistribution, and scaling and mapping using a cumulative distribution function (CDF). The histogram is computed as a set of bins for each tile. Histogram bin values higher than the clip limit are accumulated and distributed into other bins. CDF is then calculated for the histogram values. CDF values of each tile are scaled and mapped using the input image pixel values. The resulting tiles are stitched together using bilinear interpolation, to generate an output image with improved contrast.

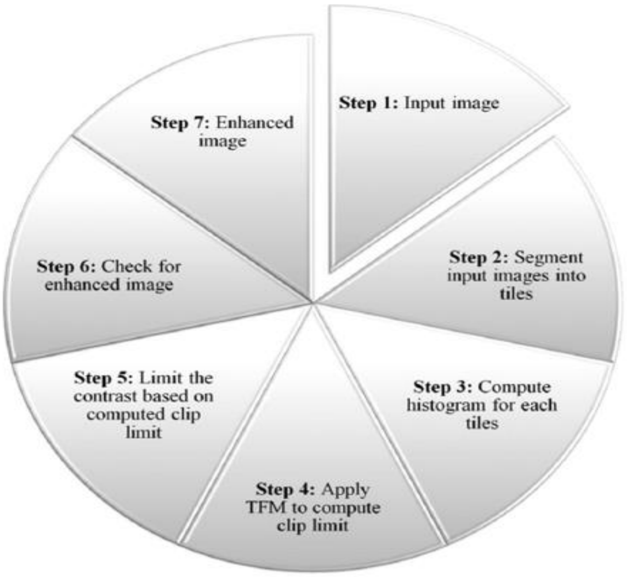


Fig 3: Steps followed in CLAHE algorithm [5].

To increase image contrast, use the CLAHE algorithm as below. Grayscale and color photos can both be processed using this approach.

CLAHE algorithm steps are as follows [5]:

Step 1: Input image
Step 2: Segment input images into tiles
Step 3: Compute histogram for each tiles
Step 4: Apply TFM to compute clip limit
Step 5: Limit the contrast based on computed clip limit
Step 6: Check for enhanced image
Step 7: Enhanced image.

Fig 3 illustrates the steps to be followed in the CLAHE algorithm. Prior to creating a histogram for each context region, a given input image is first separated into context regions. So that various portions of the image may be easily linked, a mapping function is used to produce an image mapping. The image noise is subsequently reduced using an interpolation approach. This enables us to lessen the noise in particular regions of the image. Although the method denoises the image, it does not do so fully.

Filter kernel

A filter, or kernel, in a CNN is a small matrix of weights that slides over the input data (such as an image), performs element-wise multiplication with the part of the input it is currently on, and then sums up all the results into a single output pixel. This process is known as convolution.

Filters are at the heart of what makes CNNs work. They are the primary component that helps the model extract useful features from the input data.

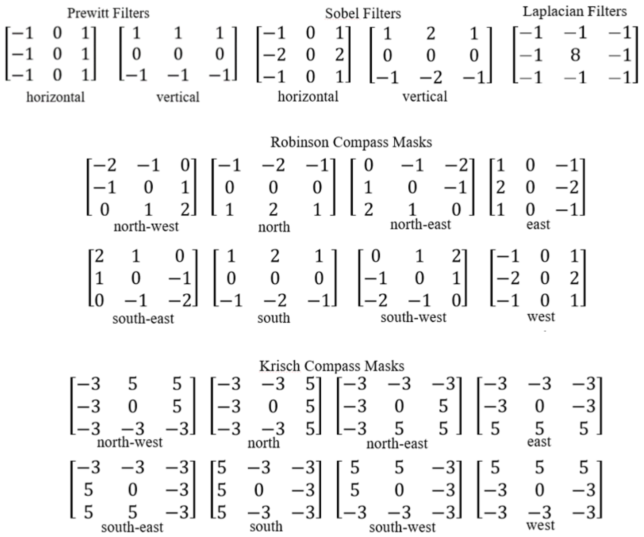


Fig 4: Types of filter 3×3 used in this experiment [7]

2D Convolution Operation

Convolution is a technique widely used in image processing. Convolution is a mathematical operation on two functions that produces a third function expressing how the shape of one is modified by the other.

Many image processing results come from a modification of one pixel with respect to its neighbors. When this modification is similar in the entire image g, it can be mathematically defined using a second image h which defines the neighbor relationships. This results in a third image f. This is the so-called convolution and it is denoted with $*$:

$$f(x,y) = (g*h)(x,y) = \sum_{m}\sum_{n} g(x-m,y-n)\,h(m,n) \quad (1)$$

Intuitively, the convolution "spreads" each pixel $(m,)$ in g following h and proportionally to the intensity (m,n). Fig 5 gives an example of the computing of a particular pixel.

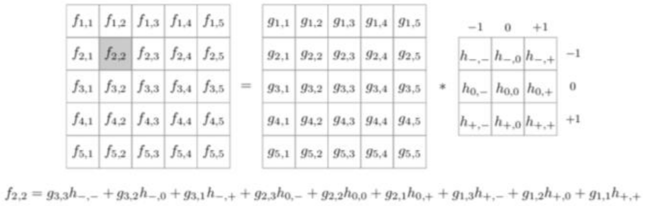


Fig 5: Example for computing the pixel (2,2) of f [6]

The Experimental Result

In this section, the experimental procedure and result are briefly explain. This experiment is performed using programming language C++ and openCV library.

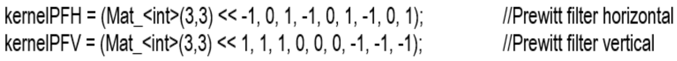
The programming code to convert the RGB input image into grayscale is:

```
cvtColor(imgOriginal,imgGrey,COLOR_BGR2GRAY);
```

The programming code to convert the grayscale image into CLAHE image with clip limit 4, are:

```
Ptr<CLAHE> clahe = createCLAHE();
clahe->setClipLimit(4);
clahe->apply(imgGrey,imgClahe);
```

To create the filter kernel 3×3 is as follows:



```
kernelPFH = (Mat_<int>(3,3) << -1, 0, 1, -1, 0, 1, -1, 0, 1);          //Prewitt filter horizontal
kernelPFV = (Mat_<int>(3,3) << 1, 1, 1, 0, 0, 0, -1, -1, -1);          //Prewitt filter vertical
```

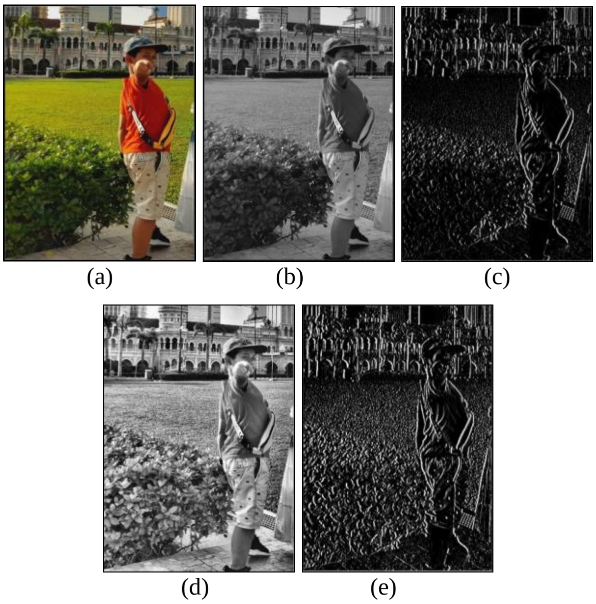
Then the convolution process is performed in filter2D(src,dst,ddepth,kernel,anchor,delta,BORDER_DEFAULT) as:

```
filter2D(imgClahe,output, -1, kernel, Point(-1, -1), 0, 4);
```

Where the arguments denote:

- *src*: Source image.
- *dst*: Destination image.
- *ddepth*: The depth of *dst*. A negative value (such as -1) indicates that the depth is same as the source.
- *kernel*: The kernel to be scanned through the image.
- *anchor*: The position of the anchor relative to its kernel. The location *Point(-1,-1)* indicates the center by default.
- *delta*: A value to be added to each pixel during the correlation. By default it is 0.
- *BORDER_DEFAULT*: We let this value by default.

The result images are depicted in Figure 6, 7 and 8. Figure 6 is processing of static image, to compare the convolved image with grayscale input and CLAHE input. It is shown in Figure 6(e) more texture than (b). Fig 7 shows some of convolved image. Fig 8 is processing of the real time frame that captured by a camera. In Fig 8, the experiment is performed at light and no light condition. It is shown that the CLAHE image can improve the contrast image and affecting to the convolution result.



(a) (b) (c)

(d) (e)

Fig 6: Original and convolved images (a) the original image (b) grayscale image (c) the convolved output of (b) (d) CLAHE image (e) the convolved output of (d)

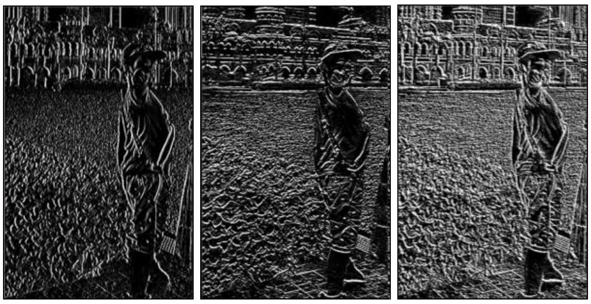


Fig 7: The convolved images of (a) Prewitt filter of horizontal (b) Sobel filter of vertical (c) Krisch's mask of north-west

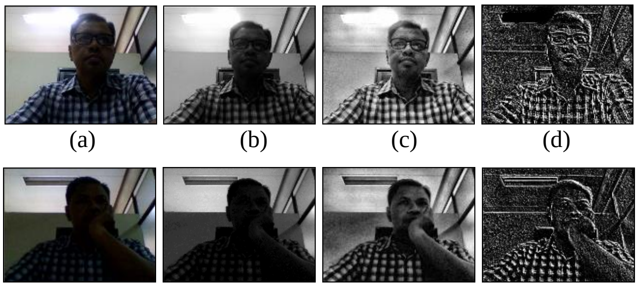


Fig 8: Processing real time video, (top) at light condition, (bottom) no light condition, (a) original frame (b) grayscale frame (c) CLAHE frame (d) convolved frame.

Conclusion

From the above study, the implementation of CLAHE image improves the image contrast then increase the texture pattern of convolution output. This texture pattern is almost same at light and no light condition. This output becomes a feature for the next classifier stage. The distinctive pattern will become the feature of the object.

References

1. Pizer SM, Amburn EP, Austin JD, *et al*. Adaptive Histogram Equalization and Its Variations. Computer Vision, Graphics, and Image Processing. 1987; 39:355-368.
2. Zuiderveld K. Contrast Limited Adaptive Histogram Equalization. In: P. Heckbert: Graphics Gems IV, Academic Press, 1994. ISBN 0-12-336155-9.
3. Sund T, Moystad A. Sliding window adaptive histogram equalization of intra-oral radiographs: Effect on diagnostic quality. Dentomaxillofac Radiol. 2006; 35(3):133-138.
4. Vidhya GR, Ramesh H. Effectiveness of contrast limited adaptive histogram equalization technique on multispectral satellite imagery, Proc. Int. Conf. Video Image Process, 2017, 234-239.
5. Venkatesh S, John De Britto C, Subhashini P, Somasundaram K. Image Enhancement and Implementation of CLAHE Algorithm and Bilinear Interpolation. Cybernetics and systems: An International Journal, 2022.
6. Vincent Mazet. Convolution. Basics of Image Processing — (Université de Strasbourg), 2020-2024. https://vincmazet.github.io/bip/filtering/convolution.html
7. Filters in convolutional neural networks, 2022. https://blog.paperspace.com/filters-in-convolutional-neural-networks/.