Morphological Image Processing to reduce Bad Textures

Raghav Hari Krishna V S*
CECN @ Department of Computer Science and Engineering
Texas A&M University

Abstract

In this Project report, we study the effects of morphological image processing in reducing noisy textures which negatively affects neural networks when used for training. We apply a combination of Opening and Closing operations combined with two types of extraction methods to study how the important features in an image are made prominent while the stone noise is reduced.

1 Problem Statement: Bad Textures

Textures in images are often used as features in understanding the spatial arrangement of color or intensities in an image. It is a prominent feature to extract and use for training machine learning and computer vision systems.

In some cases, these textures are not necessarily useful. A major example would be noise such as the salt and pepper noise. There have been research on how these salt and pepper noise can be removed completely such as the work by Hasegawa et.al[1]. But there is one domain of images where none of these techniques work and that is reducing noise in image of stone sculptures.

This niche problem is what we define as 'Stone Noise'. Stone noise is essentially a bad texture, that does not contribute as useful features in training neural networks and more often than not, negatively affects the learning. This is due to the similarity in texture between the background of the sculpture and the sculpture itself.

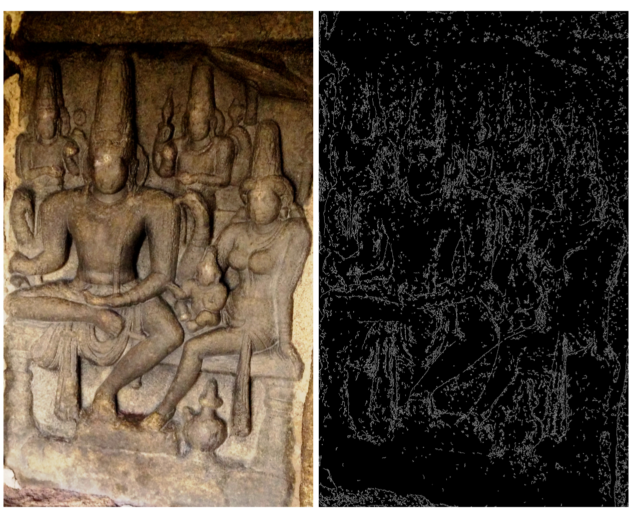

Figure 1: *A stone sculpture and its edge detected counterpart*

As we can see in Figure 1, performing a simple edge detection shows us the amount of noise which is present in the image. Getting rid of this noise is a much harder problem than one would expect. This report studies the extent to which morphological techniques such as opening, closing, and boundary extractions can help not just in reducing the noise, but also making the important features such as the outlines and depth features to be made more prominent.

*e-mail:vsraghavhk@tamu.edu

2 Image Pre-Processing

Image pre-processing is an improvement of image data that suppressing unwilling distortions or enhances some features important for further processing [2]. Usage of pre-processing is very common in neural networks in and machine vision where it all depends on data. From modifying data to fit your algorithm to data augmentation for increasing data size and variance in images. Two such common methods are segmentation and morphology.

2.1 Current Techniques

Segmentation

Segmentation is the process of partitioning an image into multiple segments. The goal of segmentation is to simplify the object so that it is easier to analyze [3]. More precisely, the process assigns each pixel in the image with a label such that all the pixels with the same label share the same characteristics. Such as a building or the sky. This method works with color images but works well only if there is stark difference in intensities, contrast, or have well defined gradient lines to separate the objects.

Morphology

Morphological image processing is a collection of non-linear operations related to a shape or the morphology of the features in the image [4]. To put it in simpler terms, a structuring element is used to morph the image where the the element either 'hits' or 'fits' the object in the image. Common operations include erosion and dilation of the image which are explained in detailed further in the report. Morphology is generally applied to and works best for binary images.

3 The Basics: Morphology

Morphology, as explained in the previous section, is generally used in binary images. But they can be made to work in gray-scale and color images as well. Instead of using 0's and 1's for the pixel values, we take a scale between 0 and 1 and go about doing morphology and re-scale them back to 0-255 scale. For color images, we do the same for each channel separately. Following are the general morphological operations explained.

3.1 Erosion

The idea of erosion is to erode away the boundary of the foreground object in the image.So in a binary image where the object is in white and the background in black, the white pixels are reduced and the thickness of the object essentially reduces. This process is useful in removing small white spots, detaching two connected objects and so on. In this project with color images, it helps by reducing the thickness of all the outlines it can detect (gradients) thus softening small noise artifacts, and reducing or removing any bright reflections and white spots.

3.2 Dilation

Dilation is the exact opposite of erosion. In binary images, we dilate the white pixels and fill any black spots within the object, making the object thicker. In color images, dilation reduces those small shadow features, flushes out the small gradients in the stone, and thins out the large gradients. We use a square kernel of size 7x7 pixels which is large enough to remove the stone perforations but small enough to not affect the outline of the stone sculpture.

Figure 2: *Eroded and Dilated images of stone sculpture in Fig 1*

In the above figure we see that erosion and dilation work in opposite ways. Erosion has caused black spots to appear in places where there are very bright lighting artifacts and made the image darker whereas dilation has caused white spots to appear in places where there are very dark crevices and perforations and made the image brighter than the original.

3.3 Opening

Opening process is simply the combination of erosion followed by dilation. This should essentially remove the noise around the object and fill the removed pixels with data from nearby pixels. When applied our test image, the result looks as if the image has been slightly blurred and roughened up so that the bright spots are not visible anymore and the gradients are more prominent.

3.4 Closing

The closing process is the combination of dilation followed by erosion process. This closes small holes and gradients and replaces them with the neighboring pixels, removing the noise altogether. When applied to our test image, it looks like the noise has been reduced but the image is brighter than usual and the gradients have softened.

Figure 3: *Opened and Closed images of stone sculpture in Fig 1*

Neither result from Figure 3 seems to have completely solved our problem of reducing bad textures and making outlines better, since each has its own drawbacks. Opened images, have a certain roughness to it that when passed through an edge detector, gives us some unwanted edges. The closed image has some loss in outlines when passed through edge detection and still has plenty of noise.

3.5 Boundary extraction

Boundary extraction has the same goal as edge detection but through morphology. Instead of finding gradients for edges, we simply obtain the outline of the object by finding the difference between the source image and an eroded form of the source image. This way, we get rid of all the pixels which do not change between source and eroded image, and store only the boundary of the image.

3.6 Morphological Extraction

Similar to Boundary Extraction, Morphological Extraction is another technique where we obtain the boundary of the object by finding the difference between the the eroded source image and dilated source image. This process gives more features as the difference gives all the features in the image which go through significant change between the two operations. Thus we end up with all the boundaries of every part of the image.

Figure 4: *Boundary and Morphological extractions of source image after applying Opening followed by Closing*

The Boundary extraction of the image gives sharp outlines and less noise but is dull and does seem to not capture certain features clearly. The Morphological extraction on the other hand captures all the features well while allowing very only the large noise to show. Overall the morphological extraction seems to be what we want.

4 Methods Applied

Usage of Color Images instead of Binary

Morphological operations are defined for binary images and does not color images. But in this project we use color images as inputs and we do not convert it to binary. This is done by processing each channel separately and simply scaling the pixel values from 0-255 to 0-1, processing, and scaling back to 0-255. This is done for two major reasons.

- The images we process, when processed as binary images, does not give the results we expect as the noise in the images are too high and overwhelms the number of good features we need to retain.

- When converted to grayscale, the results are equal to color images, and the output being color or grayscale does not affect the results we need.

Thus we simply use color images for ease and to save computation time of converting to grayscale, as speed is one of the goals of this project.

4.1 Masking

The first attempt was to process these images using masks to mask the object, and flush out the background and then process the object separately. This was in hopes that the noise in the background is completely removed while reducing noise within the object, keeping the features we need. But masking negatively affected the whole process in the following ways.

- The code to stick masks onto the image before processing was terribly slow since we individually check whether to apply the mask in the pixel or not.

- Once the background was flushed out, adding back the original image onto the processed image with mask took over 3 seconds per image, even if the mask area was small.

- The morphological operations after flushing out background did not create results which were beneficial. Instead it caused noise in the background features.

- Precise outline masks could work in increasing the value of the outlines, but the interior features are lost and the whole masking process was not worth the processing time. Image without masking works better.

4.2 Edge Detection

Edge detection works on the understanding that discontinuities in the brightness of pixels correspond to a discontinuity in the object and the surroundings within the image. Although this may not be true in all cases, it works well in drawing outlines of objects when there is good difference in brightness in the image.

But the objects in our images do not have separate lighting, different textures, or even different colors. Thus edge detection simply gives a mess of important outlines and a overwhelming amount of noise. An example of edge detected output from the input image is given in Figure 1. But edge detection of morphological extraction output after closing and opening the image gave very good binary image results similar to the the morphological extraction. This is explained in results later in the report.

4.3 Morphological processes

Although one might call this process as filtering rather than morphology at this point, because we use color images, the processes used in this image processing are morphology concepts. That being said, we tried 10 different combinations of closing (C) and opening (O) which are listed below:

- C, CO, COO

- O, OC, OCC, OCO, OCOC, OOC, OOCC

After each of these operations, we extracted the boundary through boundary extraction (hence referred as boundex) as well as morphological extraction (hence referred as (morphex). We then use a canny edge detector on the morphex images and compare the output with that of morphex images.

5 Results

Best morphology combo: Closing→Opening (CO)

It was interesting to find that, the number of operations (greater than 3) on the image matters significantly since, the more number of operations, the more blurry the image gets and the outlines are lost. But the lower the number of operations (less than 2), the outlines are sharper but the noise is not removed enough. In sequences which started with opening, the the noise actually increased when compared to operations that started with closing the image. Another aspect which was considered was that the extraction method does erode the image once, thus acting as a third layer of operation after Closing and Opening.

Best boundary extraction method: Morphex

Morphex or Morphological Extraction gave the best results in terms of strong outline features while extracting the least noise. Boundary extraction on the other hand extracted more inner features, but also extracted noise from all over the image, and the outlines were less prominent than inner features when compared to morphex. Both these methods gave some color artifacts in their outputs, but this is not a problem to us since the overall outline is still good and the color can work as an additional feature in training.

Edge detection after morphex

We used a canny edge detector on the morphex images with a hysteresis threshold of 75(min) and 85(max). Although there is small loss of important outlines when compared to morphex images, the canny edge detected images were decent. But between the two, morphex images without edge detection is the winner since the edge detection required different threshold for our different test images, which is neither convenient for speed not required as we plan to use this process for batch processing 1000s of images.

Figure 5: *morphex and canny edge morphex images of stone sculpture in Fig 1*

6 Conclusion

The definition of a successful result is subjective in this problem. In fact it also depends on the neural network used for training after this image processing. But it was a wonderful learning experience to work without knowing the results you would get, and the results turned out to be better than expected. All the resultant images from the code is available at the end of this report as well as in the results folder.

References

1 Hasegawa, Masaya et al. “Removal of Salt-and-Pepper Noise Using a High-Precision Frequency Analysis Approach.” IEICE Transactions 100-D (2017): 1097-1105.

2 Sonka M., Hlavac V., Boyle R. (1993) Image pre-processing. In: Image Processing, Analysis and Machine Vision. Springer, Boston, MA

3 https://en.wikipedia.org/wiki/Image_segmentation

4 https://en.wikipedia.org/wiki/Mathematical_morphology

5 https://www.cs.auckland.ac.nz/courses/compsci773s1c/lectures/ImageProcessing-html/topic4.htm