# PART 1

## What is quantization,sampling and quantization error?

It is a problem that the world, which is an analog image, is expressed in a limited number of pixels with a digital image. It's like mapping a large dataset to a smaller dataset. Due to the hardware we have, we do not have an infinite number of pixels to represent an infinite number of points. For this reason, we apply quantization and sampling during the analog image to digital image conversion. Determining pixel coordinates is related to sampling, and assigning pixel amplitude values to threshold values is related to quantization. The sampling rate determines the spatial resolution of the digitized image, while the quantization level determines the number of gray levels in the digitized image. During quantization and sampling, we experience some losses as we map a larger data to a smaller one. One of these losses is the quantization error. In this section, we will try to get rid of or reduce this quantization error by using the Floyd Steinberg dithering algorithm.

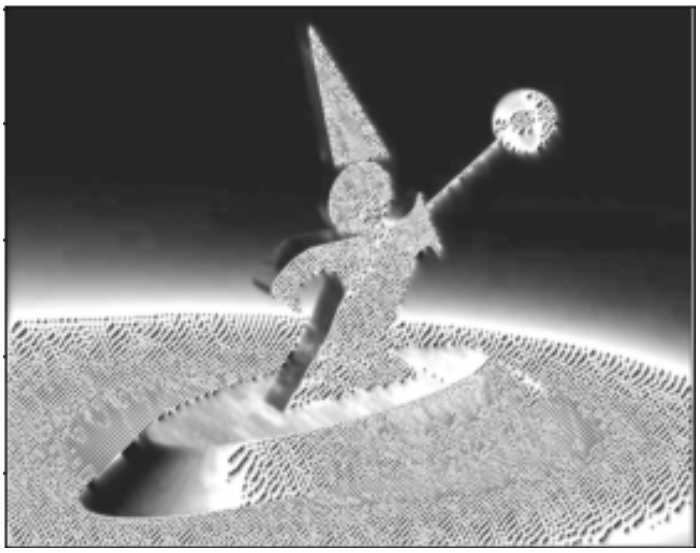
## What is Dithering?

Dithering in image processing is a technique used to simulate colours or shading. The basic concept behind dithering is adding noise, or additional pixels, to a digital file. In graphics, dithering adds random patterns of pixels to improve the image quality while avoiding banding. Thus, it allows us to fill in pixels and pixel values that are missing due to quantization or that we cannot represent. The Floyd Steinberg dithering algorithm also helps us in the face of quantization error problems.

## How is the given method achieves to prevent quantization error? Explain with examples.

The algorithm performs dithering using error diffusion. In this way, the operations performed by the algorithm for one pixel also affect other neighboring pixels. We solve the situations that we cannot do with a simple thresholding, by including neighboring pixels and spreading the error. By algorithm, quantization error is specified as the discrepancy between values for the pixel represented with a full colour palette, and the pixel values when using the smaller palette. This error is mitigated by spreading

to neighboring pixels, as it will cause serious problems when faced by a single pixel.

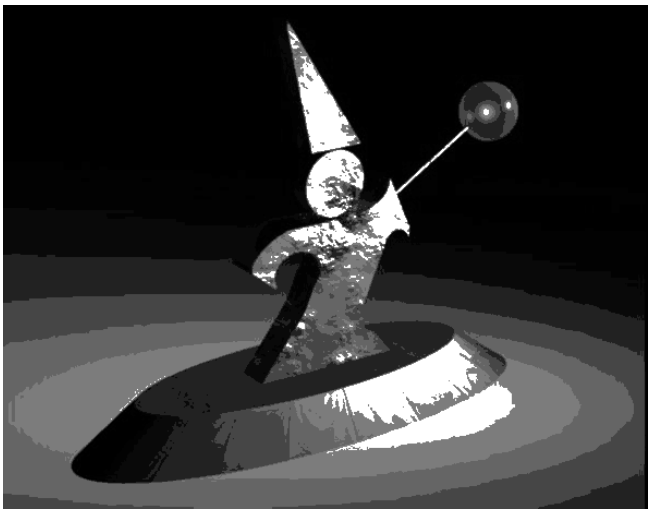


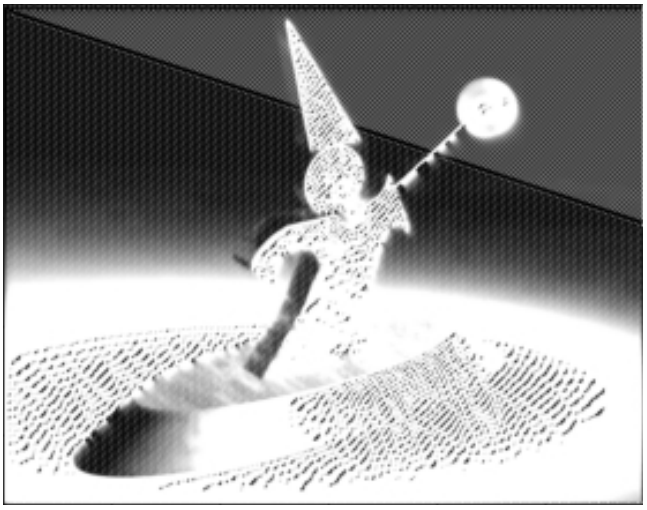
The above pictures were obtained using the q =2 value. Since our threshold number will be 2 in the quantization on the left, we have almost created a binary image. And the quantization error is too much, a very different image from the original image. Although the same quantitation value is used in the right-hand side, the error is spread over the pixels with the Floyd Steinberg algorithm and a much closer image to the original image is obtained.

**What is the behavior of the algorithm for different q parameters? Compare the quantized image and dithered image for different q parameters.**

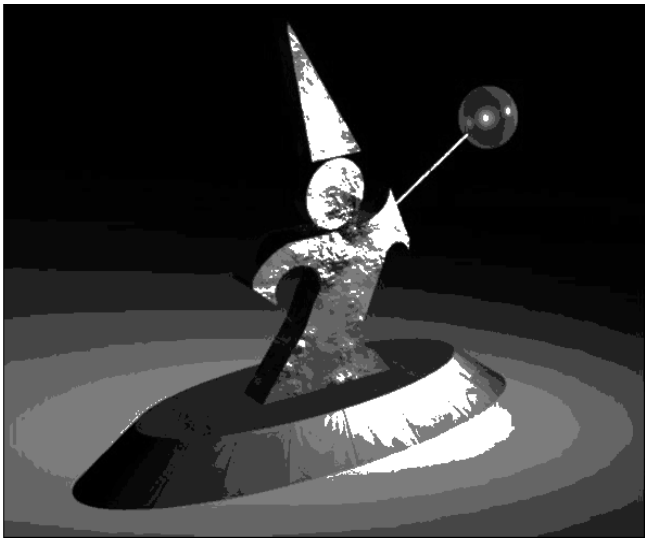
Increasing the quantization value of q allows us to have more threshold values and to express the value of a pixel with a closer value. Therefore, if we talk about quantization, increasing the q value gives us more realistic images. However, the Floyd Steinberg Dithering algorithm works better when the q value is low and the quantization error value is high. With the increase of q, which is the quantization value, we already obtain more realistic pixel values, but with this algorithm, we can unnecessarily increase both one pixel and neighboring pixel values, move them away from realistic values, and obtain images with higher intensity values. Below you can see some examples of this. As the Q value increases, the quantization image becomes more realistic, while after a certain level, the increase in the q value causes us to obtain images with higher intensity values than normal for the Floyd Steinberg algorithm.

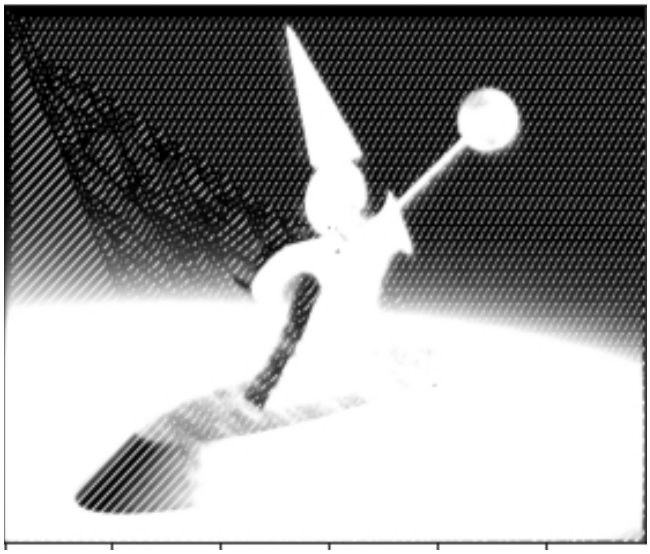
q=4



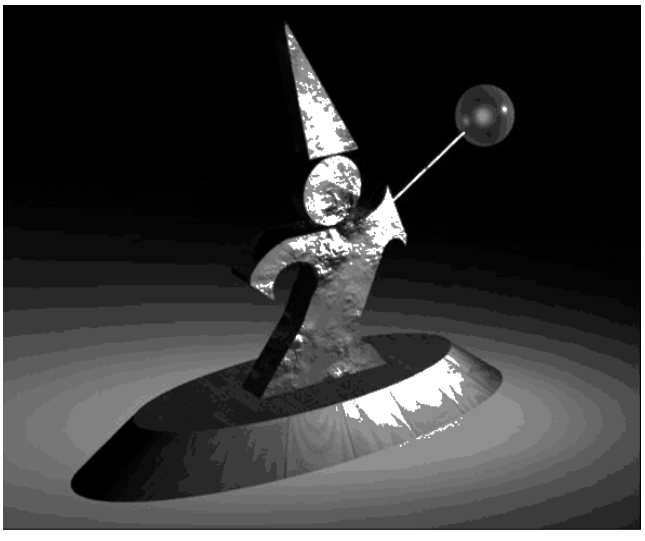


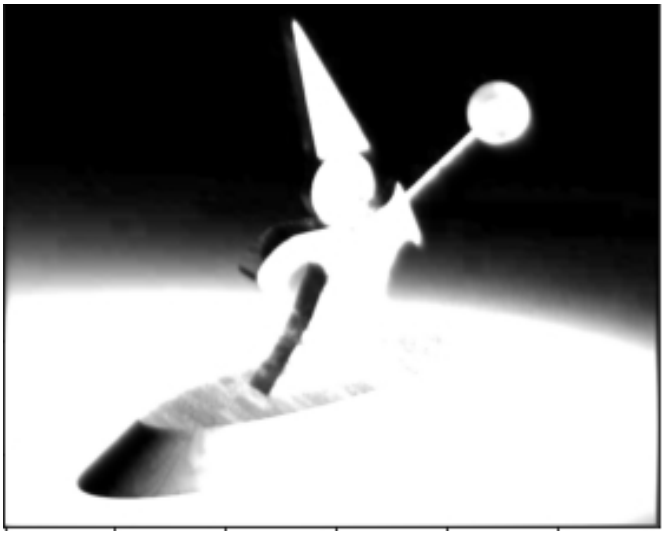
q=8



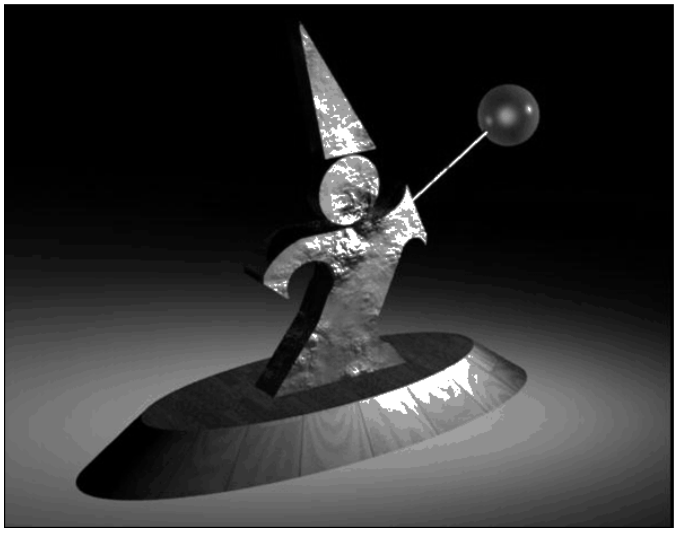


q=16





q=32



**What are the disadvantages of Floyd-Steinberg dithering algorithm? Explain with examples**

Although the Floyd Steinberg algorithm gives us much better results for low q values, it also has some disadvantages.

- For high q values, it changes the pixels and neighboring pixels of the picture that has already been quantized to be lifelike, and increases the intensity due to the algorithm, making the picture appear brighter than it is.

- Requires extra memory and work, as neighboring pixel operations are required for each pixel.

- One of its most important disadvantages is that it does not allow parallel processing as neighboring pixels are processed for each pixel. Pixel operations need to be done sequentially.

- It tends to improve the edges of an image more, giving less successful results for other parts of the image.