



**Instituto Politécnico Nacional
Escuela Superior de Cómputo**

INSTITUTO POLITÉCNICO NACIONAL



Práctica 7

Modificación del histograma 2

**Procesamiento Digital de Imágenes
Grupo 4BV1**

Alumno: Padilla García Andrea Miranda

Objetivo: Realizar las operaciones de ecualización sobre una imagen

Introducción

La ecualización transforma la imagen para intentar que su histograma quede lo más uniforme posible (distribución “plana”) sobre el rango de grises disponibles. Mejora el contraste general, especialmente en imágenes con poca variedad tonal, sin embargo, puede exagerar el ruido en imágenes si muchas regiones tienen niveles similares. se pueden aplicar diferentes **modelos de ecualización**, los cuales modifican el contraste de la imagen, entre ellos:

- Ecualización uniforme: La ecualización uniforme o clásica es el método más común. Para calcularla se emplea:

$$f = [f_{max} - f_{min}]D_P(P(i)) - f_{min}$$

En términos visuales, la ecualización uniforme expande el rango dinámico y aumenta la visibilidad de los detalles.

- Ecualización exponencial: Concentra los niveles de intensidad en el rango de valores más bajos (más oscuros). Puede ser útil para imágenes donde se quiere realzar detalles en las sombras. Se calcula con:

$$f = f_{min} - \frac{1}{\alpha} \ln[1 - D_P(P(i))]$$

- Rayleigh: Tiende a favorecer los valores más bajos de intensidad, pero con una forma de curva diferente a la exponencial. A menudo se utiliza para modelar intensidades en ciertas imágenes. Se calcula con:

$$f = f_{min} + \left[2\alpha^2 \ln \left(\frac{1}{1 - D_P(P(i))} \right) \right]^{1/2}$$

- Hiperbólica raíces: Se usan para transformaciones de contraste específicas, buscando acentuar rangos de intensidad específicos, a menudo los valores medios o bajos. Se calcula con:

$$f = \left[(f_{max}^{1/pot} - f_{min}^{1/pot})D_P(P(i)) + f_{min}^{1/pot} \right]^{pot}$$

- Hiperbólica logarítmica: Se usa para concentrar el brillo donde importa y expandiendo contraste donde está bajo. Se calcula con:

$$f = f_{min} \cdot \left[\frac{f_{max}}{f_{min}} \right]^{D_P(P(i))}$$

Como vemos, se requieren los valores de la probabilidad y la distribución de probabilidad de la imagen, que calculamos en prácticas pasadas.

Desarrollo

Para esta práctica se implementan aspectos realizados previamente en la práctica 6, entonces algo del código implementado aquí es *copy paste* de la práctica 2, solo adaptando para realizar únicamente las operaciones sobre imágenes en escala de grises.

Hablando de las funciones importantes para implementar las diferentes ecualizaciones, tenemos lo siguiente:

```
public static BufferedImage ecualizar(BufferedImage img, String tipo,
    double alpha, double pot) {
    int[] hist = calcularHistograma(img);
    double[] p = calcularProbabilidad(hist, img.getWidth() *
    img.getHeight());
    double[] D = calcularDistribucionAcumulada(p);
    BufferedImage salida = new BufferedImage(img.getWidth(),
    img.getHeight(), BufferedImage.TYPE_INT_RGB);
    double fmin = 0.0;
    double fmax = 255.0;
    double epsilon = 1e-8; // evitar log(0)

    for (int y = 0; y < img.getHeight(); y++) {
        for (int x = 0; x < img.getWidth(); x++) {
            int rgb = img.getRGB(x, y);
            int gris = (int) ((0.299 * ((rgb >> 16) & 0xFF))
                + (0.587 * ((rgb >> 8) & 0xFF))
                + (0.114 * (rgb & 0xFF)));
            double dp = D[gris];
            double nuevo = 0;

            switch (tipo) { //fórmulas de la tabla
                case "Uniforme":
                    nuevo = fmin + (fmax - fmin) * dp - fmin;
                    break;

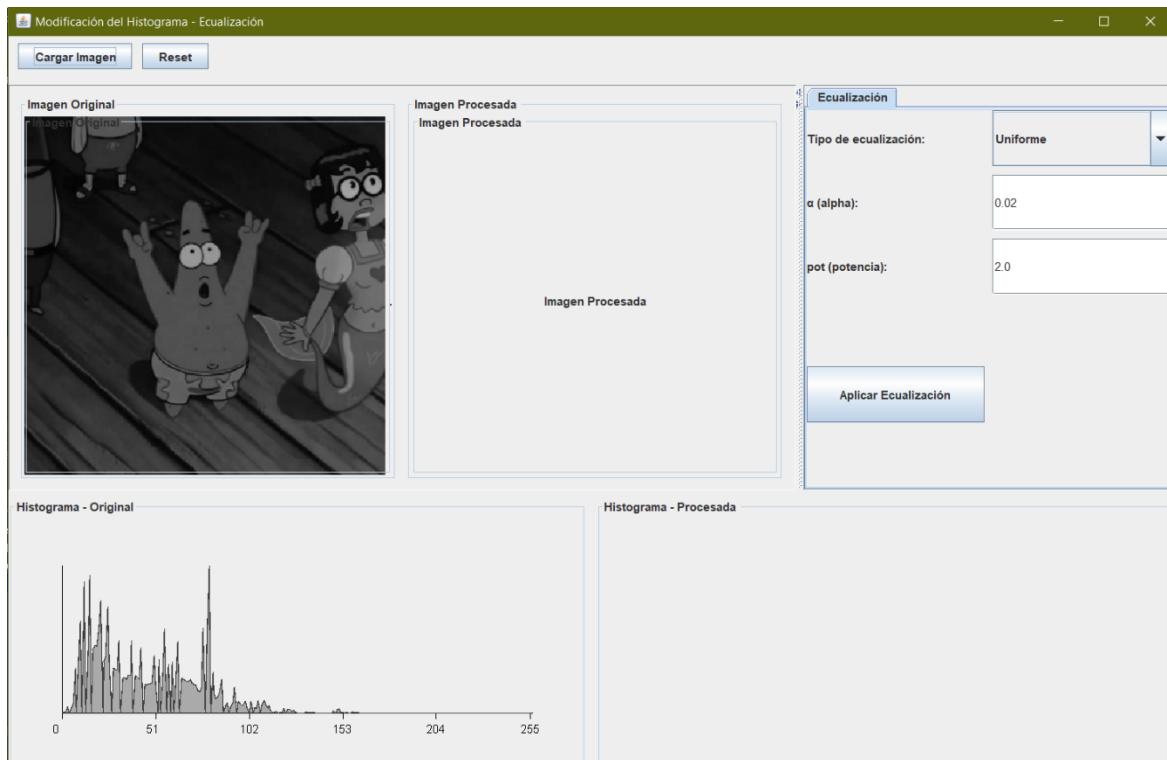
                case "Exponencial":
                    nuevo = fmin - (1.0 / alpha) * Math.log(1.0 - dp
                    + epsilon);
                    break;

                case "Rayleigh":
                    nuevo = fmin + Math.sqrt(2 * alpha * alpha *
                    Math.log(1.0 / (1.0 - dp + epsilon)));
                    break;

                case "Hiperbólica Raíces":
                    nuevo = Math.pow((Math.pow(fmax, 1.0 / pot) -
                    Math.pow(fmin, 1.0 / pot)) * dp
                    + Math.pow(fmin, 1.0 / pot)), pot);
                    break;
            }

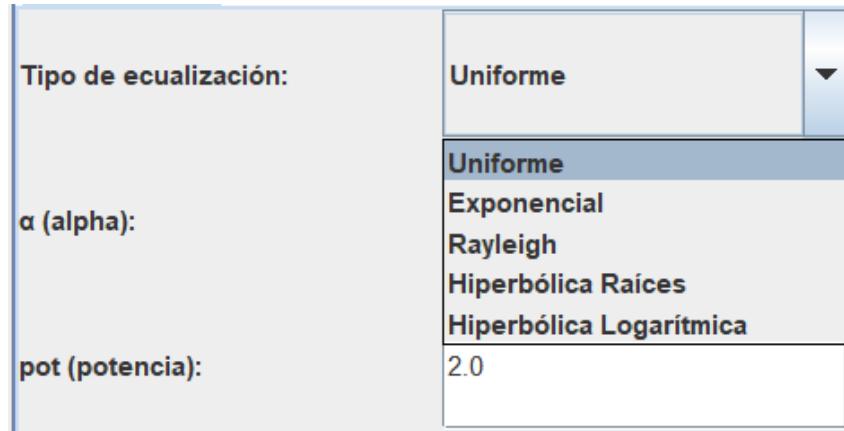
            int nivel = (int) Math.max(0, Math.min(255, nuevo));
            salida.setRGB(x, y, new Color(nivel, nivel,
            nivel).getRGB());
        }
    }
}
```

El diseño de la GUI se realizó de manera tal que resulta en una ventana de este estilo:



donde el panel lateral derecho permite aplicar las modificaciones mencionadas antes y le permite al usuario modificar valores permitidos para corregir la imagen cargada.

Al desplegar la opción de tipo de ecualización, se desprenden las ecualizaciones vistas en teoría:



Conclusión

Esta práctica conllevó conceptos e implementaciones bastante interesantes, me resulta interesante como el modificar valor de una variable (hablando de la ecualización) afecta demasiado la imagen a grado tal de hacer que se vea totalmente oscura. Las modificaciones realizadas a los histogramas son de gran ayuda al momento de querer ajustar una imagen que no tiene un rango adecuado para apreciar la información y detalles en ella.

Referencias

Kenneth, L., & Jian, H. (2015). Histogram equalization. Sciencedirect.
<https://sciencedirect.com/topics/computer-science/histogram-equalization>

Neel. (2024, 24 octubre). Quick Guide to Histogram Equalization for Clearer Images. Analytics Vidhya. <https://analyticsvidhya.com/blog/2022/01/histogram-equalization/>