Proyecto IPOO

Image Transform

NervCode

Ingrid Echeverri	Jean Pierre Cardenas	Andres David Camargos	Jhan Alejandro Perez
			
1943542	1942703	1944140	1941003

Orientado a Objetos

Aplicar filtros en imagenes .ppm y .pgm

Image Vector

Atributos
Int fileName; vector<vector<int>> image Int rows Int columns Int width Int height Int maxRGB String header Int matrixStart
Metodos
ImageVector() ImageVector(string fileName) ~ImageVector() Int getRows() Int getColumns() Int getWidth() Int getHeight() Int getMaxRGB() String getHeader() Void start() Void convertToVector() vector<vector<int>> getVector()

Nos permite crear vectores de las imágenes. Este vector se copia dentro de **aImage[x][y]**, permitiéndonos tener 2 copias de la imagen.

```
for (int i = 0; i < vectorImagen.size(); i++)
{
  for (int j = 0; j < vectorImagen[i].size(); j++)
  {
    img->setPixel(i, j, vectorImagen[i][j]);
  }
}
```

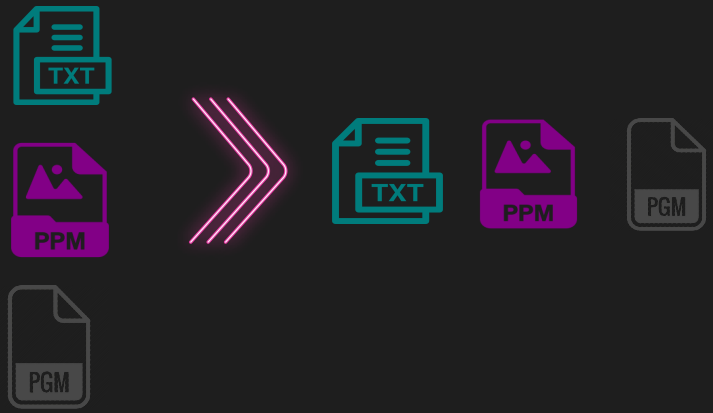
También nos permite extraer información de archivo imagen como, el header (P2,P3,P6), width,height y maxRGB.

```
ImageVector imageVector("images/" + fileToOpen);

int rows = imageVector.getRows();
int columns = imageVector.getColumns();
int height = imageVector.getHeight();
int width = imageVector.getWidth();
int maxRGB = imageVector.getMaxRGB();
string header = imageVector.getHeader();
```

En el VectorImage se cuentan las rows y las columns que forman a la matriz.

```
while (getline(file, line))
{
  this->rows++;
  int fileColumns = 0;

  for (int i = 0; i < line.length(); i++)
  {
    if (line[i] == ' ')
    {
      fileColumns++; //Counts the columns
    }
    if (fileColumns >= columns)
    {
      this->columns = fileColumns;
    }
  }
}
```

Lectura de formatos de imagen



Recibe formatos .TXT, .PPM o .PGM que convertirá en una imagen .PPM o .TXT

-> make run

Write the image file name to open with its extension (.ppm, .pgm, .txt), must be in images folder

Cat.ppm

Select a filter to apply:
1-Normal
2-Black and White
3-Negative
4-Sephia
5-Colorize
....


Filtros!

Filtros Negativos

Original

Formula:

inverso = 255 – colorActual

```
int negativeValue = 255 - picture->getPixel(i,j);
picture->setPixel(i,j,negativeValue);
```

Para el filtro negativo basta con restar del valor máximo en la escala RGB (255) el valor actual y eso nos dará el valor inverso ejemplo: blanco es 255, el inverso de blanco es negro ¿Cuánto vale negro?

Inverso = 255 - blanco = 255 - 255 = 0 = negro

Negativo

Esta fórmula se aplica a todos los píxeles de la imagen

Sepia

Para el filtro sepia se aplican una fórmula diferente por cada valor (R,G,B), al combinar esos 3 valores, se obtendrá un tono de marrón, por cada pixel y siguiendo este mismo procedimiento en toda la imagen, se lograra obtener la imagen en el tono deseado.

Original

Formulas:

R = (0.393 * R) + (0.769 * G) + (0.189 * B)
G = (0.349 * R) + (0.686 * G) + (0.168 *B)
B = (0.272 * R) + (0.534 * G) + (0.131 * B)

```
int tr = 0.393*r + 0.769*g + 0.189*b;//Tono de marrón para valor r
if(tr>= maxRGB) tr = maxRGB;
int tg = 0.349*r + 0.686*g + 0.168*b;//Tono de marrón para valor g
if(tg>= maxRGB) tg = maxRGB;
int tb = 0.272*r + 0.534*g + 0.131*b;//Tono de marrón para valor b
if(tb>= maxRGB) tb = maxRGB;
```

Sepia

Blanco y negro

Original

Formula:

Gris = (0.2989 * R) + (0.5870 * G) + (0.1140 * B)

```
//Tono de gris para todos los valores (r, g, b)
int gray = 0.2989 * red + 0.5870 * green + 0.1140 * blue;

//Se aplica el tono gris a todos los pixeles de la imagen
picture->setPixel(i,j,gray);//Tono de gris para pixeles con valor r
picture->setPixel(i,j+1,gray);//Tono de gris para pixeles con valor g
picture->setPixel(i,j+2,gray);//Tono de gris para pixeles con valor b
```

Blanco y negro

Para el filtro blanco y negro se aplica la misma formula en todos los valores (R,G,B), al combinar esos 3 valores, se obtendrá un tono de gris, por cada pixel y así dándole otro tono a la imagen.

Colorize

Para el filtro colorize se crea un vector con vectores los cuales contienen valores que al ser operados con los de la imagen original, da como resultado valores relacionados con el tono de color deseado. **Tener en cuenta que los valores de los que hablamos representan RGB.**

Ejemplo: Imagen Blanco y Negro a un tono verde

Original
(154, 154, 154)

Tono Verde
(82, 229, 77)

Usando la formula general, para llegar de original a verde:

colorizeR = 154 - 72 = 82
colorizeG = 154 + 75 = 229
colorizeB = 154 - 77 = 77

Se usa para todos los valores (R, G, B)

Colorize

Ejemplo: Imagen Blanco y Negro a un tono rosa

Original
(154, 154, 154)

Tono Rosa
(245, 55, 140)

Usando la formula general, para llegar de original a rosa:

colorizeR = 154 + 91 = 245
colorizeG = 154 - 99 = 55
colorizeB = 154 - 14 = 140

Se usa para todos los valores (R, G, B)

Gracias!

Proyecto aún en desarrollo,
Pre entrega 1 del proyecto