



INSTITUTO POLITÉCNICO NACIONAL
UNIDAD PROFESIONAL INTERDISCIPLINARIA DE
INGENIERÍA Y TECNOLOGÍAS AVANZADAS



CARRERA: INGENIERÍA TELEMÁTICA

UNIDAD DE APRENDIZAJE: MULTIMEDIA

Practica 5: Paleta de colores con volcán



ALUMNO:

- Ayala Rodriguez Axel Alejandro
- 2023640150

DOCENTE: NOE SIERRA ROMERO

FECHA: 16/02/26

GRUPO: 3TM2

Código:

Filtro de 16 Tonos de Amarillo (Basado en estructura BMP)

1. Definimos la paleta de 16 tonos de amarillo (formato BGR para BMP)

Los valores van de oscuro (índice 0) a claro (índice 15)

```
paleta_amarilla = [  
    [0x00, 0x22, 0x33], [0x00, 0x33, 0x44], [0x00, 0x44, 0x55], [0x00, 0x55, 0x66],  
    [0x00, 0x66, 0x77], [0x00, 0x77, 0x88], [0x00, 0x88, 0x99], [0x00, 0x99, 0xAA],  
    [0x00, 0xAA, 0xBB], [0x00, 0xBB, 0xCC], [0x00, 0xCC, 0xDD], [0x00, 0xDD, 0xEE],  
    [0x00, 0xEE, 0xFF], [0x33, 0xFF, 0xFF], [0x66, 0xFF, 0xFF], [0x99, 0xFF, 0xFF]  
]
```

2. Abrir archivos

try:

```
file = open('./images/volcan.bmp', 'rb')
```

```
fileo = open('./images/volcan_yellow.bmp', 'wb')
```

Copiar metadata (54 bytes iniciales)

```
metadata = file.read(54)
```

```
fileo.write(metadata)
```

3. Configuración de rangos

El valor máximo de un color de 24 bits es 16,777,215

```
valor_maximo = pow(2, 24) - 1
```

Dividimos el total entre 16 para obtener el tamaño de cada "escalón"

```
tamano_rango = valor_maximo / 16
```

```
no_pix = 0
```

```
print("Procesando imagen...")
```



```
while True:
```

```
    pixel_data = file.read(3) # Leemos los 3 bytes (B, G, R)
```

```
    if len(pixel_data) == 3:
```

```
        # Convertimos los bytes del píxel a un valor entero para clasificarlo
```

```
        valor_int = int.from_bytes(pixel_data, byteorder='little')
```

```
        # Calculamos a qué índice de la paleta pertenece (0 a 15)
```

```
        # Usamos min() para asegurar que no se pase del índice 15
```

```
        indice = min(int(valor_int / tamano_rango), 15)
```

```
        # Escribimos los 3 bytes del color correspondiente de nuestra paleta
```

```
        color_seleccionado = paleta_amarilla[indice]
```

```
        fileo.write(bytes(color_seleccionado))
```

```
        no_pix += 1
```

```
    else:
```

```
        break
```

```
print(f'Proceso finalizado. Píxeles procesados: {no_pix}')
```

```
except FileNotFoundError:
```

```
    print("Error: Asegúrate de que la carpeta './images/' y el archivo existan.")
```

```
finally:
```

```
    file.close()
```

```
    fileo.close()
```


Imagen generada con la paleta:

