

Filtros Artísticos y de Estilizado: Derretir, Solarizar, Viento, Hallar Bordes.

Arnoldt Leandro, M. Belén Crolla y Oscar Luis Risso

Trabajo práctico final de "Captura y Procesamiento Digital de Señales e Imágenes", II-FICH-UNL.

Resumen— En el presente trabajo se estudian, diseñan e implementan cuatro tipos de filtros ampliamente utilizados en la actualidad por software especializados en el tratamiento de imágenes.

Estos filtros son los llamados filtro derretir, filtro solarizar, filtro hallar bordes y filtro viento, los cuales modifican la imagen de entrada generando una nueva imagen con características particulares, tales como sus nombres lo indican.

En principio se realiza una breve introducción a la teoría de los filtros en imágenes para así poder entender mejor el objetivo. Seguidamente se explica el funcionamiento básico del filtro y luego se exponen los algoritmos implementados junto con las imágenes obtenidas luego de aplicar dicho algoritmo.

Finalmente se presenta una conclusión sobre todo el proceso realizado.

I. INTRODUCCIÓN

EL filtrado de una imagen es un herramienta ampliamente utilizada en el procesamiento digital de imágenes para varias aplicaciones.

Los filtros son acciones sobre las fotos que pueden producir una extensa gama de efectos especiales en la totalidad de la imagen o en zonas parciales de la misma.

De manera sencilla podemos entender el filtrado como un sistema por el cual una imagen de entrada es manipulada por las características del filtro con el objetivo de obtener a la salida una imagen con ciertas componentes modificadas, estas pueden ser: el brillo, contraste, ruido, saturación, tono, entre otras. Las modificaciones realizadas en las imágenes dependen del filtro aplicado en la misma.

Uno de los usos más importantes del proceso de filtrado es el de la restauración de imágenes o eliminación de ciertos errores introducidos en la imagen por distintos medios, como por ejemplo, errores en el sensor de CCD de una cámara digital.

Estos filtros de eliminación de errores son muy utilizados en el procesamiento digital de imágenes y por lo tanto existe una gran variedad, siempre dependiendo del tipo de error que contenga. Por ejemplo están los filtros de medias y filtros de orden, los cuales sirven para la reducción de ruidos por desenfoco, ruido gaussiano, ruido impulsivo, etc. Incluso se pueden aplicar filtros en frecuencias para otro cierto tipo de ruido, como es el caso del ruido periódico.

Sin embargo, los filtros estudiados en este trabajo cumplen la función de modificar la imagen para agregarle un efecto característico que resulta atrayente para el observador pero no necesario para un posterior procesamiento, es por ello que dichos filtros caen en la clasificación de filtros artísticos y filtro de estilizado.

Un problema de esto es que existe muy poca base

bibliográfica sobre este tipo de filtro ya que en su mayoría son utilizados por software especializados de diseño artístico sin ningún objetivo de análisis de imágenes y sus algoritmos o formas de implementación están muy pocos difundidos en el área.

Los filtros aquí presentados tuvieron su origen con las herramientas de software antes mencionadas y causaron una gran sensación de los usuarios. Fueron principalmente diseñados para el trabajo de personas en general que deseaban añadir efectos especiales a sus imágenes y rápidamente se extendieron al área de diseño, pudiéndose observar sus resultados en la actualidad en ciertas publicidades gráficas.

II. FILTRO DERRETIR

Este filtro trabaja sobre una cantidad de píxeles escogidas por el usuario y de manera simple tiene como objetivo llevar los píxeles más oscuros de la imagen hacia abajo y los más claros hacia arriba.

El proceso comienza tomando un píxel de la imagen de manera aleatoria y comparándolo con un píxel de misma columna pero de una cierta cantidad de filas más abajo (la cantidad de filas inferior es un parámetro fijado por el usuario). Si la intensidad del primer píxel tomado es menor que la intensidad del segundo píxel, estas se intercambian. Este proceso se repite tantas veces como el usuario halla elegido en principio.

Breve reseña del algoritmo:

```
s=-5;
//ban=cantidad seleccionada de pixeles por el usuario
for(k=0;k<ban;k++){
  //busqueda del pixel aleatoriamente
  rx=rand();
  rx1=RAND_MAX;
  i=int(((rx/rx1)*M));
  ry=rand();
  ry1=RAND_MAX;
  j=int(((ry/ry1)*N));

  //suma ponderada de las intensidades de los distintos
  //planos
  i0=0.3*R(i,j)+0.59*G(i,j)+0.11*B(i,j);
  i1=0.3*R(i,j-s)+0.59*G(i,j-s)+0.11*B(i,j-s);

  if((i<M)&&(j<N)&&((j-s)>0)&&((j+s)<N)){
    if(i0<i1) {
      //intercambio pixeles
      aux=R1(i,j);
      R1(i,j)=R1(i,j-s);
      R1(i,j-s)=aux;
    }
  }
}
```

```

aux=G1(i,j);
G1(i,j)=G1(i,j-s);
G1(i,j-s)=aux;
aux=B1(i,j);
B1(i,j)=B1(i,j-s);
B1(i,j-s)=aux;
}
}
}

```

A.Resultados Obtenidos:

Imagen Original



Imagen Resultante



A partir de estos resultados se puede observar que este proceso de intercambio de píxeles según sus intensidades genera un efecto o sensación de estar derritiéndose la imagen.

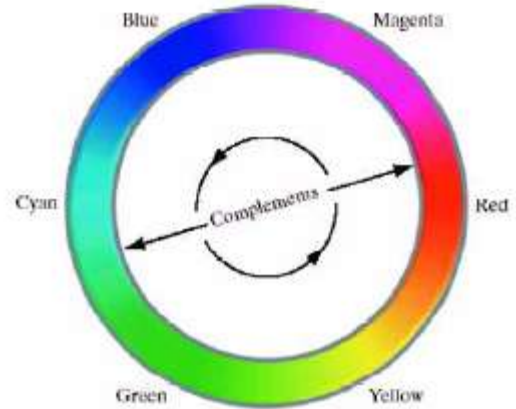
Se pudo observar también que variando la distancia entre píxeles que se intercambian (en el caso de las imágenes anteriores el valor elegido es 5) el efecto que produce en la imagen puede ser indeseado si no se elige adecuadamente. Por ejemplo, para un valor igual a 1, el efecto es casi inapreciable, mientras que para un valor muy grande, como 15, la imagen puede distorsionarse no logrando el resultado esperado.

Variando la cantidad de píxeles que selecciona el usuario, la imagen presenta mayor o menor efecto. Es recomendable tomar una cantidad de píxeles acorde al tamaño de la imagen.

III.FILTRO SOLARIZAR

Este filtro aplica una transformación en los colores de la imagen, aplicándole el complemento de color. Éste es análogo al negativo en imágenes en tonos de gris.

El efecto del filtro produce matices opuestos según el círculo de colores.



Para producir este efecto a cada píxel de la imagen con intensidad $I > 128$ se le asigna el valor $255 - I$. Si el valor de la intensidad es $I \leq 128$ se deja igual. Esto se realiza para cada capa RGB por separado.

Breve reseña del algoritmo:

```

// Solarizacion
for(i=0;i<M;i++){
for(j=0;j<N;j++){
if (R(i,j) > 128)
img2(i,j,0,0)=255-R(i,j,0,0);
if (G(i,j) > 128)
img2(i,j,0,1)=255-G(i,j,0,0);
if (B(i,j) > 128)
img2(i,j,0,2)=255-B(i,j,0,0);
}
}

```

A.Resultados Obtenidos:

Imagen Original



Imagen Resultante



A partir de las imágenes resultantes se puede observar y entender mejor aun el efecto de este filtro. Como antes de dijo, este efecto es análogo al negativo en las imágenes en tonos de grises, por lo tanto es fácil entender la variación brusca de colores en las imágenes resultantes.

IV.FILTRO VIENTO

La aplicación de este filtro se limita a los bordes de los distintos objetos de una imagen, por lo que primero se debió implementar un detector de bordes (en este caso Sobel). Seguidamente estos son clasificados para luego solo trabajar con aquellos que caen dentro de un rango de intensidad previamente fijado por el análisis de su histograma. De estos píxeles clasificados servirán aquellos que cumplan la condición de que la intensidad del píxel contiguo en la dirección en que se fijo el viento sea menor que la intensidad del píxel analizado.

Todos aquellos píxeles que cumplan las condiciones antes descritas serán los que se verán afectados por el filtro.

Este efecto de esfumado se realiza tomando un píxel "x" del conjunto seleccionado y replicar puntos continuos en la dirección del viento como la ponderación del píxel "x" y el actual. Una clave del buen efecto del viento es la variación aleatoria de la cantidad de replicas de un píxel del borde.

Breve reseña del algoritmo:

```
//aplicación del efecto de esfumado en el filtro viento
velocidad=2;
for(i=0;i<M;i++){
  for(j=0;j<N;j++){
    if((sobel(i,j)>35)and(sobel(i,j)<65)){//tomado viendo el
      histograma
```

```
      //numero aleatorio del largo del esfumado
      r=rand();
      r1=RAND_MAX;
      ra=int(((r/r1)*N));
      largo=int ((velocidad*ra)*pow(double (10),-1));
```

```
if((R(i,j)>=R(i+2,j))) { // canal R
  aux=R(i,j);
  s1=i;
  for(k=0;k<largo;k++){
    if (s1<M){
```

```
      p=(largo-k)*pow(double (largo),-1);
      img1(s1,j,0,0)=p*aux+(1-p)*R(s1,j); //esfumado
      s1=s1+1;
    }
  }
}
if((G(i,j)>=G(i+2,j))){ //canal G
  aux=G(i,j);
  s2=i;
  for(k=0;k<largo;k++){
    if (s2<M){
      p=(largo-k)*pow(double (largo),-1);
      img1(s2,j,0,1)=p*aux+(1-p)*G(s2,j);
      s2=s2+1;
    }
  }
}
if((B(i,j)>=B(i+2,j))){ //canal B
  aux=B(i,j);
  s3=i;
  for(k=0;k<largo;k++){
    if (s3<M){
      p=(largo-k)*pow(double (largo),-1);
      img1(s3,j,0,2)=p*aux+(1-p)*B(s3,j);
      s3=s3+1;
    }
  }
}
}}}
```

A.Resultados Obtenidos:

Imagen Original:



Imagen Resultante:



Imagen Original



Imagen Resultante:



```
p=(largo-k)*pow(double (largo),-1);
img2(s1,j,0,0)=p*aux+(1-p)*img(s1,j); //esfumado
s1=s1+1;
}}} }
```

B.Resultados Obtenidos:

Imagen Resultante



Como se puede apreciar en esta imagen el efecto de viento se realizo sobre un solo canal, dando una mejor sensacion de dicho efecto, ya que para el caso de las imagenes a color el esfumado posee diferentes colores producto de la ponderacion entre el pixel analizado (RGB) y el pixel contiguo en la misma dirección. Mientras que para el caso de la imagen en escala de grises, el esfumado se produce sobre sobre un pixel en tonalidades de gris, obteniendose un mejor efecto.

V.FILTRO HALLAR BORDES

Breve reseña del algoritmo para imágenes en escala de grises:

```
s1=0;
s2=0;
s3=0;
velocidad=2;
for(i=0;i<M;i++){
for(j=0;j<N;j++){
if((sobel(i,j)>80)&&(sobel(i,j)<120)){//tomado en
base al histograma
//aleatorio
r=rand();
r1=RAND_MAX;
ra=int(((r/r1)*N));
largo=int ((velocidad*ra)*pow(double (10),-1));

if((img(i,j)>=img(i+2,j))) {
aux=img(i,j);
s1=i;
for(k=0;k<largo;k++){
if (s1<M){
```

Los bordes de una imagen digital se pueden definir como transiciones entre dos regiones de niveles de gris significativamente distintos. Suministran una valiosa información sobre las fronteras de los objetos y puede ser utilizada para segmentar la imagen, reconocer objetos, etc.

La mayoría de las técnicas para detectar bordes emplean operadores locales basados en distintas aproximaciones discretas de la primera y segunda derivada de los niveles de grises de la imagen.

Para la realización de éste filtro se utilizó el detector de bordes Sobel, el cual esta basado en la primera derivada, cuya máscara utilizada es la siguiente:

Columna:			Fila:		
-1	-K	-1	1	0	-1
0	1	0	K	0	-K
1	K	1	1	0	-1

donde K = 2.

En primer lugar lo que se hace, al estar trabajando con

imágenes a color, es una separación de la imagen en sus tres canales, es decir R G B. Luego se procede a realizar la convolución entre cada canal con dichas máscaras y seguidamente hallar su modulo. Una vez hecho esto normalizamos el modulo e invertimos cada canal.

Finalmente a aquellos píxeles cuya intensidad por canal sea menor a 220 los dividimos por un escalar mayor a 1, logrando de esta forma acentuar aún más los bordes.

Breve reseña del algoritmo:

```
//Modulo
for(i=0;i<M;i++){
  for(j=0;j<N;j++){
    sobel(i,j,0)=(abs(auxr(i,j))+abs(aux2r(i,j)));
    sobel(i,j,0,1)=(abs(auxg(i,j))+abs(aux2g(i,j)));
    sobel(i,j,0,2)=(abs(auxb(i,j))+abs(aux2b(i,j)));
  }
}
```

```
//Normalizacion del modulo
sobel.normalize(0,255);
```

```
//invierto cada canal
for(i=0;i<M;i++){
  for(j=0;j<N;j++){
    sobel(i,j,0,0)=255-sobel(i,j,0,0);
    sobel(i,j,0,1)=255-sobel(i,j,0,1);
    sobel(i,j,0,2)=255-sobel(i,j,0,2);
  }
}
```

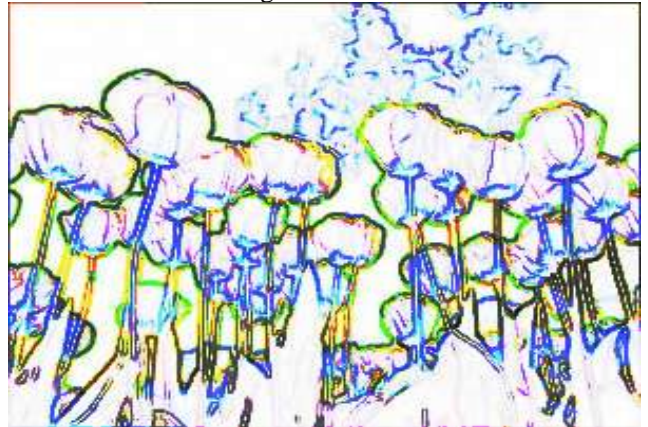
```
//Acentuado de Bordes
for(i=0;i<M;i++){
  for(j=0;j<N;j++){
    for(k=0;k<3;k++){
      if(sobel(i,j,0,k)<220)
        sobel(i,j,0,k)=sobel(i,j,0,k)/2.5;
    }
  }
}
```

A.Resultados Obtenidos:

Imagen Original



Imagen Resultante



VI.CONCLUSIÓN FINAL

Se puede concluir que la aplicación e implementación de los diferentes filtros fueron satisfactorios logrando en su mayoría los efectos esperados. Si bien las imágenes son simplemente para obtener un efecto agradable al usuario, esta implementación posee un gran contenido teórico ya que utilizan herramientas aplicables en el procesamiento digital de imágenes.

Nuevamente es importante aclarar la poca base bibliográfica sobre estos tipos de filtros, lo cual trajo aparejado la imposibilidad de medir los resultados y determinar así su buen funcionamiento. Por ello en varias ocasiones se utilizaron imágenes de prueba sacadas de publicaciones donde se exponían también los resultados del procesado con estos filtros. Gracias a ello se pudo determinar, de manera subjetiva, el correcto funcionamiento de los algoritmos aquí presentados.

VII.REFERENCIAS

- [1] <http://ie.fing.edu.uy/investigacion/grupos/gti/timag/trabajos/2004/efectos/frameset.htm>
- [2] <http://es.wikipedia.org/>