

Práctica 1 - MTF del ojo humano

Daniel Torres Robledo
shadow.cat6333@gmail.com

Andrés Fuentes Hernández
andres7233@hotmail.com

Abstract—En este documento se describen las aplicaciones de la *MTF*, el cálculo de ecuaciones para generarla, su implementación en *python*, así como experimentos utilizando este estímulo con diferentes personas a distintas distancias, que para obtener medidas invariantes a la distancia de observación, el cálculo de esta frecuencia será expresado en ciclos/grado.

Index Terms—*MTF*, *SVH*, *python*, sinusoidal modulada, experimentos, invariantes

I. Objetivo

- Encontrar la *MTF* del ojo experimentalmente.
- Encontrar la frecuencia de máxima sensibilidad del ojo humano.
- Generar los cálculos necesarios para obtener la función sinusoidal en frecuencia y amplitud.
- Desplegar el estímulo visual sinusoidal modulada.

II. Introducción

La *MTF* (*Modulation Transfer Function*) o función de transferencia de modulación se ha convertido en una herramienta muy utilizada para especificar el rendimiento y la resolución de toda clase de sistemas ópticos y de visión, que van desde lentes, cintas magnéticas y películas hasta telescopios, la atmósfera y el ojo humano.

La *MTF* es usada para caracterizar sistemas lineales e invariantes en el espacio, sin embargo, a pesar de que el Sistema Visual Humano (*SVH*) no cumple con estas dos propiedades, la *MTF* se ha utilizado para caracterizarlo bajo condiciones de iluminación controladas.

La *MTF* también puede interpretarse como la capacidad de un sistema óptico para percibir contraste, es decir, la capacidad de resolver o diferenciar líneas a una determinada frecuencia espacial.

Para entender de manera simplificada el concepto de frecuencia espacial, hay que notar que las frecuencias espaciales bajas equivalen a repeticiones de patrones muy separados espacialmente (por ejemplo, líneas muy separadas) y que las altas equivalen a repeticiones más compactas.

Hay que considerar que todo sistema de visión tiene un límite superior a partir del cual ya no le es posible distinguir más detalles. Este límite está directamente relacionado con la frecuencia de *Nyquist* [4]. En los estudios de sensibilidad del *SVH* a base de experimentación psico-física, se ha encontrado que su *MTF* consiste en una función del tipo paso-banda con un pico o frecuencia espacial máxima en el rango de 2 a 6 ciclos por grado [1] [2].

En esta practica se examinará la sensibilidad del *SVH* a distintas frecuencias espaciales; la idea general de crear un

patrón de sinusoidales, cuya frecuencia aumenta en el eje horizontal y el contraste en el eje vertical. La envolvente del patrón visibles generalmente muestra un comportamiento similar a la curva de la *MTF*.

Comprobaremos que dicho pico, que representa la frecuencia máxima de sensibilidad del ojo humano se moverá conforme el observador se aleja o acerca del estímulo. Para obtener una medición invariante a la distancia de observación, el cálculo de esta frecuencia de observación debe ser expresado en ciclos/grado.

III. Desarrollo

En esta sección se muestra el cálculo para obtener las constantes de la función sinusoidal modulada en frecuencia y en amplitud.

Sea la función I , que dará los valores de cada pixel de la imagen:

$$I(n, m) = g(n) \cdot \sin(2\pi \cdot f(m))$$

$$h = \text{altura de la imagen}$$

$$w = \text{ancho de la imagen}$$

Sea g la función de contraste, donde es necesario encontrar las constantes c, d :

$$g(n) = d \cdot e^{-c \cdot n}$$

Para el eje horizontal, utilizaremos f :

$$f(m) = a \cdot e^{b \cdot x}$$

Por lo que la frecuencia instantánea esta definida como la derivada de f :

$$f'(m) = a \cdot b \cdot e^{b \cdot x}$$

Es necesario encontrar las constantes a, b tal que $f'(w) = 1/2$ para obtener la frecuencia máxima, y $f'(0) = 1/w$ para encontrar la frecuencia mínima, en *ciclos/segundo*

$$1) f'(0) = a \cdot b \cdot e^{b \cdot 0} = a \cdot b \cdot 1 = a \cdot b = 1/w$$

$$2) f'(w) = a \cdot b \cdot e^{b \cdot w} = 1/2$$

Para encontrar b , sustituir 1 en 2:

$$(1/w) \cdot e^{b \cdot w} = 1/2$$

$$e^{b \cdot w} = w/2$$

$$\ln(e^{b \cdot w}) = \ln(w/2)$$

$$b \cdot w \cdot \ln(e) = \ln(w/2)$$

$$b = \frac{\ln(w/2)}{w \cdot \ln(e)}$$

$$b = \frac{\ln(w/2)}{w \cdot \ln(e)}$$

$$b = \frac{\ln(w/2)}{w}$$

Por lo que para encontrar a sustituimos b en 1)
 $a \cdot b = 1/w$

$$a = \frac{1}{b \cdot w}$$

Por lo tanto ya tenemos los valores de a, b :

$$a = \frac{1}{b \cdot w}$$

$$b = \frac{\ln(w/2)}{w}$$

Para encontrar las constantes c, d de $g(n)$:

$$g(n) = d \cdot e^{-c \cdot n}$$

Donde $\epsilon > 0$.

$$1)g(0) = d \cdot e^{-c \cdot 0} = d = \epsilon$$

$$2)g(h) = d \cdot e^{-c \cdot h} = 1$$

Para encontrar c , sustituimos 1 en 2:

$$\epsilon \cdot e^{-c \cdot h} = 1$$

$$\ln(\epsilon \cdot e^{-c \cdot h}) = \ln(1)$$

$$\ln(\epsilon) + \ln(e^{-c \cdot h}) = 0$$

$$-c \cdot h \cdot \ln(e) = 0$$

$$c = \frac{\ln(\epsilon)}{h}$$

Por lo tanto, para $\epsilon > 0$, en este caso utilizaremos $\epsilon = 0.02$:
 $d = \epsilon = 0.02$

$$c = \frac{\ln(\epsilon)}{h}$$

A continuación se muestra el estímulo MTF utilizando dichas constantes:

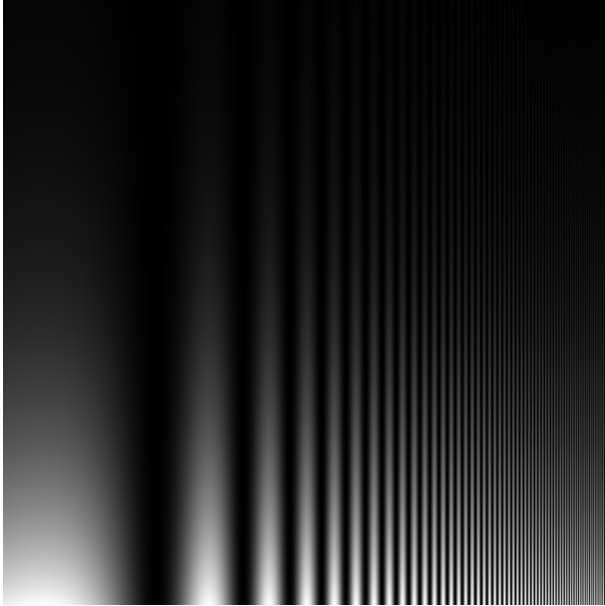


Fig. 1. Estímulo visual (sinusoidal modulada).

Para calcular los valores que se reportarán en los resultados son:

X : coordenada x donde se encontró experimentalmente el punto más alto visible por el usuario.

$\phi = f'(x) = a \cdot b \cdot e^{b \cdot x}$: Qué es la frecuencia instantánea evaluada en x .

$dd = w/screensize$: Es la distancia entre dos pixeles, que es la cantidad de pixeles entre la medida de la imagen en centímetros.

$\alpha = \tan^{-1}(dd/distancia) \cdot 2 \cdot \pi$: Resultado en grados/pixels, el valor de distancia es la distancia a la que se encuentra el observador en centímetros.

$$f = \frac{1}{\phi \cdot \alpha}$$

Frecuencia de observación en ciclos/grado

IV. Resultados

Para este experimento, se utilizó el estímulo visual MTF descrito en la sección de Desarrollo, el cual fue mostrado a tres diferentes personas a las distancias de $30cm$, $60cm$ y $100cm$, en donde fueron calculados para cada distancia los valores de x , ϕ , α y frecuencia de observación f . La descripción de estas ecuaciones también se encuentra en el apartado de Desarrollo.

En este experimento se utilizaron las siguientes constantes:

- $height = 728$: pixeles de altura de la imagen
- $width = 728$: pixeles de longitud de la imagen
- $screensize = 13cm$ medida de la imagen en centímetros
- $distancia \in [30cm, 60cm, 100cm]$: distancias en centímetros a las que se encuentra el usuario
- $\epsilon = 0.02$: constante para calcular el contraste.

	$D_1 = 30cm$	$D_2 = 60cm$	$D_3 = 100cm$
X	315	250	325
ϕ'	0.018	0.010	0.019
α	6.779	4.718	3.207
f	8.371	20.364	16.316
X	317	367	327
ϕ'	0.018	0.027	0.019
α	6.779	4.718	3.207
f	8.236	7.893	16.054

Fig. 2. Tabla de resultados, experimento sin lentes y en morado resultados con lentes 1.1.

	$D_1 = 30cm$	$D_2 = 60cm$	$D_3 = 100cm$
X	416	409	357
ϕ'	0.040	0.038	0.025
α	6.779	4.718	3.207
f	3.694	5.617	12.590
X	375	398	403
ϕ'	0.029	0.035	0.036
α	6.779	4.718	3.207
f	5.149	6.140	8.674

Fig. 3. Tabla de resultados, experimento sin lentes y en morado resultados con lentes 1.2.

	$D_1 = 30cm$	$D_2 = 60cm$	$D_3 = 100cm$
X	245	250	245
ϕ'	0.010	0.010	0.010
α	6.779	4.718	3.207
f	14.758	20.364	31.193

Fig. 4. Tabla de resultados, experimento 1.3.

V. Conclusiones

Algunas de las medidas salieron del rango definido en la introducción, el cual es de 2 a 6 ciclos por grado. Esto puede ser debido a falta de control de la iluminación, así como ruido al momento de dar *click* sobre la imagen utilizando un ratón inalámbrico a distancia de la imagen, o realizar movimientos adicionales, acercarse o alejarse del punto de visión, para tratar de ver el punto más alto de la curva en la imagen.

En el experimento en donde se tuvo mayor control, se puede observar como el valor de x se mueve a la izquierda conforme se aleja de la imagen. Sin embargo, a distancias lejanas, el valor de la frecuencia de observación sale de los parámetros, con y sin lentes, donde coincide que la persona tiene problemas para ver a distancias lejanas.

Entre los necesario para tener en control la iluminación, es la referente al dispositivo en donde se muestra el estímulo, en este caso, si el brillo de la pantalla estaba al máximo, los picos de la curva se muestran más altos en la imagen.

VI. Código fuente

A continuación se muestra la implementación de las ecuaciones mostradas en el Desarrollo para calcular el estímulo visual y los valores reportados en los resultados de x , ϕ , α y frecuencia de observación f .

Es posible modificar las constantes referentes al tamaño de la imagen, la medida en centímetros de ésta, y la distancia a la que se encuentra el usuario en centímetros, esto en las variables definidas en el *main* del programa.

Para la implementación, la función sinusoidal de multiplicó por 0.5 y se sumó 0.5 para ajustar los valores al rango $[0, 1]$, ya que estaban en $[-1, 1]$.

REFERENCES

- [1] Pratt, W. k., Digital Image Processing, John Wiley & Sons Inc, 2001.
- [2] Levine, M.D., Vision in man and machine, McGraw-Hill, 1985.
- [3] Norman Koren, *Resolution and MTF curves in scanners and sharpening*. Available at <http://www.normankoren.com/Tutorials/MTF2.html>. [Accessed Agosto 31, 2019]
- [4] Norman Koren, *Nyquist frequency*. Available at https://en.wikipedia.org/wiki/Nyquist_frequency. [Accessed Agosto 31, 2019]

```

1 import cv
2 import math
3 from random import randint
4
5 def user_x(event,x,y,flags,param):
6     if event == cv.EVENT_LBUTTONDOWN:
7         param = map(float, param)
8         n_pixels = param[0]
9         a = param[2]
10        b = param[3]
11        screen_size = param[4]
12        user_distance = param[5]
13
14        snap = a*b*math.e**(b*x)
15        d = n_pixels/screen_size
16        alpha = math.atan(d/user_distance)*2.0*math.pi
17        f = 1.0/(snap*alpha)
18        print "Distance:%.3f, X:%.3f, Snap:%.3f, Alpha:%.3f, F:%.3f"%(
19            user_distance, x, snap, alpha, f)
20
21 def create_mtf(name, width, height, screen, distance_user):
22     out = cv.CreateImage((width, height), 8, 1)
23
24     b = math.log(width/2.0)/width
25     a = 1.0/(b*width)
26
27     d = 0.82
28     c = math.log(d)/height
29
30     x_vector = [0.5+0.5*math.sin(2*math.pi*a*math.e**(b*val)) for val in
31                 xrange(width)]
32     y_vector = [d*math.e**(-c*val) for val in xrange(height)]
33
34     for i in xrange(width):
35         for j in xrange(height):
36             out[j,i] = 255.0*(x_vector[i]*y_vector[j])
37
38     cv.ShowImage("mtf", out)
39     cv.MoveWindow("mtf", 0, 0)
40
41     cv.SetMouseCallback("mtf", user_x, [width,height,a,b,screen,distance_user])
42
43     cv.SaveImage("mtf.png", out)
44     cv.WaitKey(0)
45
46 if __name__ == "__main__":
47     size = 728
48     screen_cm = 13.0
49     distance_user_cm = 70.0
50     create_mtf("out.png", size, size, screen_cm, distance_user_cm)

```

Fig. 5. Código utilizado para la implementación en *python*.