

**RECONOCIMIENTO DE FORMAS MEDIANTE COORDENADAS LOGPOLARES
CON INVARIANCIA A TRASLACIONES, ROTACIONES Y ESCALAJES**

**Bruno Cernuschi Frías*, Alejandro Blumenkrans*
Enrique Carlos Segura****

* Facultad de Ingeniería, Universidad de Buenos Aires.

** Consejo Nacional de Investigaciones Científicas y Técnicas, Argentina.

RESUMEN. Uno de los objetivos principales en visión por computadora es el reconocimiento de formas independientemente de su ubicación, orientación o tamaño. Los métodos no estadísticos sólo son útiles cuando el objeto posee la orientación y tamaño del patrón de referencia. Se propone una transformada que resulta invariante a rotaciones, escalaje y traslaciones.

1. LA TRANSFORMADA DE FOURIER.

Dada una función continua $g(x, y)$, su transformada de Fourier (TF) se define como:

$$F[g(x, y)] = G(\mu, \omega) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} e^{-i(\mu x + \omega y)} g(x, y) dx dy \quad (1)$$

Entre las propiedades de la TF, es bien conocida la invariancia de su módulo ante desplazamientos, es decir:

$$|F[g(x, y)]| = |F[g(x - x_0, y - y_0)]| \quad (2)$$

así como su no invariancia ante rotaciones y escalajes, i.e., dado el cambio de coordenadas

$$\begin{aligned} x_r &= x \cos \alpha - y \operatorname{sen} \alpha \\ y_r &= x \operatorname{sen} \alpha + y \cos \alpha \end{aligned}$$

resulta

$$F[g(x_r, y_r)] = G(\mu_r, \omega_r) \quad \text{con} \quad \begin{cases} \mu_r = \mu \cos \alpha - \omega \operatorname{sen} \alpha \\ \omega_r = \mu \operatorname{sen} \alpha + \omega \cos \alpha \end{cases}$$

y dado el cambio

$$\begin{aligned} x_e &= ax \\ y_e &= ay \end{aligned}$$

resulta

$$F[g(x_e, y_e)] = G(\mu_e, \omega_e) \quad \text{con} \quad \begin{cases} \mu_e = \frac{\mu}{a} \\ \omega_e = \frac{\omega}{a} \end{cases}$$

2. COORDENADAS LOGPOLARES Y LA TF.

Proponemos un primer cambio de coordenadas:

$$\mu = \rho \cos \varphi$$

$$\omega = \rho \operatorname{sen} \varphi$$

Definiendo ahora:

$$G_p(\rho, \varphi) = G(\rho \cos \varphi, \rho \operatorname{sen} \varphi) = F[g(x, y)]$$

es fácil comprobar que:

$$F[g(x_r, y_r)] = G_p(\rho, \varphi - \alpha)$$

(se reduce la rotación a un desplazamiento sobre φ), pero para el escalaje aún no hay tal reducción, ya que:

$$F[g(x_s, y_s)] = G_p\left(\frac{\rho}{a}, \varphi\right)$$

Introduciendo un segundo cambio:

$$\rho = e^h$$

resulta

$$\frac{\rho}{a} = \frac{e^h}{a} = e^{(h - \log a)} = e^{h'} \quad \text{con } h' = h - \log a$$

El cambio de coordenadas a adoptar sería entonces:

$$\mu = e^h \cos \varphi$$

$$\omega = e^h \operatorname{sen} \varphi$$

(3)

Ahora:

$$G_{\log p}(h, \varphi) = G_p(e^h, \varphi) = G(e^h \cos \varphi, e^h \operatorname{sen} \varphi) = F[g(x, y)]$$

Rotando y escalando $g(x, y)$:

$$x_{rs} = ax \cos \alpha - ay \operatorname{sen} \alpha$$

$$y_{rs} = ax \operatorname{sen} \alpha + ay \cos \alpha$$

Se obtiene que:

$$F[g(x_{rs}, y_{rs})] = G_{\log p}(h - \log a, \varphi - \alpha)$$

con lo cual (por (2)):

$$|F[F[g(x_{rs}, y_{rs})]]| = |F[F[g(x, y)]]| \quad (4)$$

3. TRANSFORMADA LOGPOLAR.

a) Caso continuo

De acuerdo con el resultado obtenido en (4), ya estamos en condiciones de diseñar nuestra transformada, que se calcula según la siguiente secuencia de subtransformadas

- | | | |
|--|---|---|
| i) Transformada de Fourier | } | Obtenemos invariancia al desplazamiento |
| ii) Módulo | | |
| iii) Conversión a coordenadas logpolares | } | Obtenemos invariancia a rotación y escalaje |
| iv) Transformada de Fourier | | |
| v) Módulo | | |

b) Desarrollo para el caso discreto

Trabajaremos con un lattice rectangular finito $M \times N$ y la función:

$$X(i, j) = \begin{cases} i = 0, \dots, M-1 \\ j = 0, \dots, N-1 \end{cases}$$

con la transformada discreta de Fourier (DFT)

$$\text{DFT}[X(i, j)] = X(q, k) = \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} X(i, j) e^{-\sqrt{-1}(2\pi/Mik + 2\pi/Njq)} \quad (5)$$

Siempre y cuando todos los objetos rotados, escalados y trasladados permanezcan dentro del lattice, los resultados obtenidos para el caso continuo siguen siendo válidos.

c) Conversión a coordenadas logpolares para el caso discreto

De la secuencia de subtransformadas a seguir en 3.a) la única que no nos queda definida para el caso discreto es la conversión a coordenadas logpolares. En este caso adoptaremos el cambio de coordenadas (3) de la siguiente manera:

Para todo el lattice $M \times N$ en el espacio (h, φ) barreemos sus coordenadas y vemos con qué punto se corresponde en el espacio (i, j) :

$$(h, \varphi) \longrightarrow (e^h \cos \varphi, e^h \sin \varphi) \quad (6)$$

Observando la exp. (6) vemos que surgen los siguientes inconvenientes:

- 1) No podemos barrer íntegramente el lattice (i, j)
- 2) $(e^h \cos \varphi, e^h \sin \varphi)$ no es un par ordenado de enteros
- 3) Para h chicos hay una gran pérdida de resolución

Proponemos hacer lo siguiente:

- a) Centrar la $\text{DFT}[X(i, j)]$ para $(k, q) = (0, 0)$ en el centro del cuadrado (tomemos el caso $M = N$)

b) Barremos e^h entre 1 y $\frac{M}{2}$. Haciendo:

$$e^{h \log \frac{M}{2} \cdot \pi^{h-1}} \begin{cases} h = 0 : e^h = \rho = 1 \\ h = M - 1 : e^h = \rho = \frac{M}{2} \end{cases}$$

c) Barremos $\hat{\varphi}$ entre 0 y 2π rad

$$\hat{\varphi} = \varphi \frac{2\pi}{M}$$

d) Determinamos $X(h, \varphi)$ por interpolación lineal: Si $\begin{cases} i_0 = e^h \cos \varphi \\ j_0 = e^h \sin \varphi \end{cases}$ con $i < i_0 < i + 1$, $j < j_0 < j + 1$, $\ell = i - i_0$, $m = j - j_0$:

$$X(h, \varphi) = X(i, j) \cdot (1 - \ell) \cdot (1 - m) + X(i + 1, j) \cdot \ell \cdot (1 - m) + X(i, j + 1) \cdot (1 - \ell) \cdot m + X(i + 1, j + 1) \cdot \ell \cdot m.$$

e) Descartamos los valores de $X(i, j)$ no barridos por (h, φ) en los ángulos. Esto implicará una cierta pérdida de altas frecuencias, que puede llegar a perjudicar nuestro clasificador.

f) Consideramos que la pérdida de resolución cerca de $(k, q) = (0, 0)$ no afecta demasiado el reconocimiento de formas.

4. EXPERIMENTOS.

La simulación realizada consistió en investigar la capacidad de la transformada logpolar para clasificar los 10 dígitos. Estos fueron generados sobre un raster binario de 7×5 pixels.

Los caracteres fueron modificados, a saber: trasladados, rotados y escalados sobre una imagen de 16×16 pixels. Si consideramos que las imágenes usuales son de 256×256 pixels, vemos que hemos trabajado bajo condiciones de baja resolución extremas, elemento que acentúa nuestros resultados.

Luego, cada uno de los 10 dígitos modificados fueron clasificados según uno de los 10 caracteres originales. Ya que nuestro mayor acento en este trabajo es en la transformada, y no en el método clasificador, hemos efectuado la comparación entre las respectivas transformadas de las muestras a clasificar y los patrones originales por cuadrados mínimos.

O sea que, dada la transformada logpolar de los caracteres originales:

$$X_\omega = \{X_\omega(i, j), \quad i, j = 0, \dots, M - 1\} \quad \omega = 0, \dots, 9$$

Y la transformada logpolar de los caracteres modificados:

$$Y_\mu = \{Y_\mu(i, j), \quad i, j = 0, \dots, M - 1\} \quad \mu = 0, \dots, 9.$$

Decidiremos que la transformada Y_{μ} corresponde al carácter 'K' si el error cuadrático medio:

$$e^2_{\mu\omega} = \sum_{i=0}^{M-1} \sum_{j=0}^{M-1} [Y_{\mu}(i,j) - X_{\omega}(i,j)]^2 \quad \begin{cases} \mu = 0, \dots, 9 & (\text{filas}) \\ \omega = 0, \dots, 9 & (\text{columnas}) \end{cases}$$

es mínimo para $\omega = k$. Obtenemos así la matriz de los errores cuadráticos de 10×10 :

$$E = \{e^2_{\mu\omega}; \mu, \omega = 0, \dots, 9\}$$

que nos da el error cuadrático medio de las transformadas de los 10 caracteres (dígitos) modificados con respecto a las transformadas de los 10 caracteres originales. Cuando la clasificación es correcta el elemento del mínimo (que será subrayado en todos los casos), estará en la diagonal principal.

a) Traslación

Al realizar como primeros dos pasos en la transformada logpolar la DFT y obtenido su módulo (3.a); y al no haber ninguna fuente de error aquí, la invariancia a traslaciones es perfecta. La diagonal principal de E es nula para traslación únicamente, o bien no sufre ninguna alteración la matriz E toda, cuando se combina traslación con rotación.

Es por ello que no incluimos ninguna simulación al respecto.

b) Rotación

Las figs. 1.0 a 1.9 nos muestran los caracteres originales y los respectivos rotados $0,5$ radianes ($28,6^\circ$). Obsérvese la deformación que acarrea la rotación con tan baja resolución.

La Tabla I nos da la correspondiente matriz E . Vemos que los elementos de la diagonal principal son siempre los menores de cada fila, por lo que los 10 dígitos son clasificados correctamente.

Para 90° ($\pi/2$ radianes), al no haber deformación del dígito, los elementos de la diagonal principal deberían tender a 0. Esto se evidencia en las figs. 2.0 a 2.9; y en la Tabla II. El orden de los elementos de la diagonal principal es 10^{-3} .

Procediendo de esta manera hemos construido la Tabla III, que para distintos ángulos de rotación nos da la cantidad de caracteres elegidos correctamente y la suma de los elementos de la diagonal principal, como indicativo de la desviación de las transformadas, idealmente invariantes.

Tal como lo hemos adelantado, al realizar estos experimentos en 32×32 pixels, la Tabla VII muestra cómo los 10 dígitos son clasificados correctamente para cualquier ángulo (patrones de 14×10 pixels).

c) Escalaje

Hemos observado una gran sensibilidad de la transformada al escalaje. Los resultados son satisfactorios para escalajes imperceptibles con tan baja resolución.

d) Ruido

Si bien la teoría no prevé inmunidad de esta transformada al ruido, al ser la TF una función integradora, es esperada cierta capacidad en tal sentido.

Se procedió a contaminar las imágenes con ruido binario independiente; es decir que hay una probabilidad p de que cada pixel, independientemente, invierta su valor de 0 a 1 o viceversa.

Para un nivel de ruido del 1.1% los resultados son las figs. 3.0 a 3.9 y la matriz F de la Tabla IV. Sólo clasificó erróneamente el dígito '3'. Para un nivel de ruido del 2.5% obtenemos las figs. 4.0 a 4.9 y la Tabla V. Aquí, con una considerable deformación de los caracteres, clasificó correctamente 6 de los 10 caracteres.

5. CONCLUSIONES.

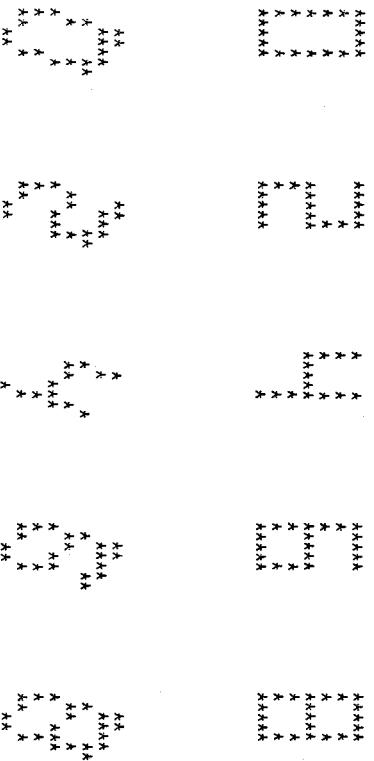
Hemos presentado un método para clasificar objetos en dos dimensiones basado en las propiedades de las coordenadas logpolares y su relación con la transformada de Fourier en dos variables.

Los resultados de combinar rotación y traslación han sido excelentes, aun con imágenes de tan baja resolución (16×16 pixels). En el caso del escalaje los resultados no han sido tan satisfactorios pues al realizar la primera transformada la información respecto del tamaño se difunde muy rápidamente y sólo es recuperable para corrimientos de tamaño muy pequeños.

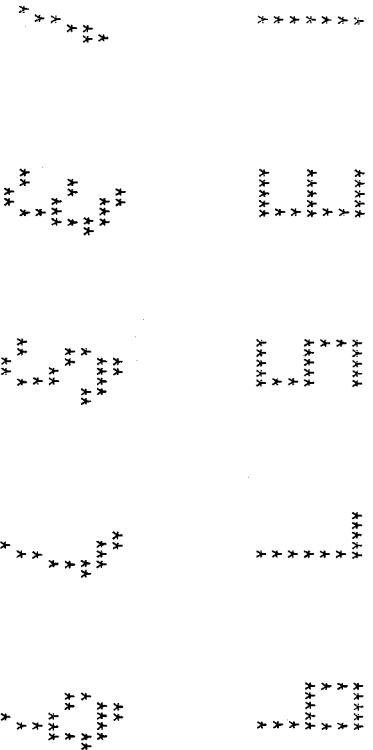
El método, al estar compuesto por subtransformadas usuales y rápidas (FFT o cambio de coordenadas) es de fácil implementación. Si tenemos presente que las imágenes usuales son de 256×256 pixels y que los objetos pueden ocupar una buena parte de la imagen, los resultados de este método son excelentes.

BIBLIOGRAFIA

- Segura, E.C. y Cernuschi Frías, B.: *Pattern classification using neural networks, with invariance under translations, rotations and scaling*; 1989 (preprint).
- Rajalevu, A., Musavi, M.T. y Shirvaikar, M.V.: *A neural network approach to character recognition*; Neural Networks, vol. 2, pp. 387-393, 1989.
- Kahan, S.T., Pavlidis, T. y Baird, W.: *On recognition of printed characters of any font and size*; IEEE Trans. Pattern Recognition and Machine Intelligence, PAMI-9, 274-285, 1987.
- Lifferman, J.: *Les méthodes rapides de transformation du signal: Fourier, Walsh, Hadamard, Haar*; Paris, Masson, 1980.



Figs. 1.0-1.9



<u>248.4</u>	868.3	349.0	379.4	345.4	346.3	370.3	544.9	455.1	317.2
849.6	<u>144.4</u>	856.6	869.1	598.0	851.7	992.6	366.4	1104.3	840.1
452.2	886.7	<u>215.8</u>	286.1	347.2	314.7	368.7	572.1	457.4	329.3
452.1	905.5	231.4	<u>206.9</u>	362.1	250.5	278.8	588.2	371.4	267.1
481.7	558.9	422.6	499.9	<u>172.2</u>	411.8	548.8	284.6	669.1	401.0
451.0	917.3	296.4	255.9	380.5	<u>221.1</u>	288.9	592.6	420.2	276.4
427.0	990.3	309.7	256.7	423.6	238.0	<u>207.6</u>	667.1	314.0	266.1
494.3	495.6	477.5	483.4	250.6	481.8	598.3	<u>176.4</u>	716.3	454.5
464.3	1135.9	390.4	383.2	567.8	406.6	267.4	818.6	<u>235.4</u>	349.8
435.5	1000.2	357.1	318.9	432.9	307.1	241.3	679.7	331.7	<u>122.0</u>

Tabla I



Figs. 2.0-2.9



<u>0.0</u>	873.3	409.7	424.7	396.0	409.7	384.3	558.5	444.9	340.6
873.3	<u>0.0</u>	892.4	898.2	622.3	892.4	1025.5	391.3	1125.3	873.9
409.7	892.4	<u>0.0</u>	189.5	365.5	357.9	313.6	574.2	378.7	332.7
424.7	898.2	189.5	<u>0.0</u>	362.1	189.5	222.5	573.1	328.4	289.0
396.0	622.3	365.5	362.1	<u>0.0</u>	341.0	450.7	315.3	570.9	299.1
409.7	892.4	257.9	189.5	341.0	<u>0.0</u>	223.6	571.5	378.7	254.3
384.3	1025.5	313.6	222.5	450.7	223.6	<u>0.0</u>	697.6	199.9	238.9
558.5	391.3	574.2	573.1	315.3	571.5	697.6	<u>0.0</u>	818.2	551.6
444.9	1125.3	378.7	328.4	570.9	378.7	199.9	818.2	<u>0.0</u>	370.0
340.6	873.9	332.7	289.0	299.1	254.3	238.9	551.6	370.0	<u>0.0</u>

Tabla II

0	10	0.00
10	10	190.37
20	6	245.46
30	9	209.15
40	8	236.02
50	6	275.96
60	4	281.55
70	10	220.91
80	10	169.33
90	10	0.00
100	10	190.37
110	6	245.46
120	9	209.15
130	8	236.02
140	6	275.96
150	4	281.55
160	10	220.91
170	10	169.33
180	10	0.00
190	10	190.37
200	6	245.46
210	9	209.15
220	8	236.02
230	6	275.96
240	7	237.83
250	10	220.91
260	10	169.33
270	10	0.00
280	10	190.37
290	6	245.46
300	9	209.15
310	8	236.02
320	6	275.96
330	7	237.83
340	10	220.91
350	10	169.33
360	10	0.00

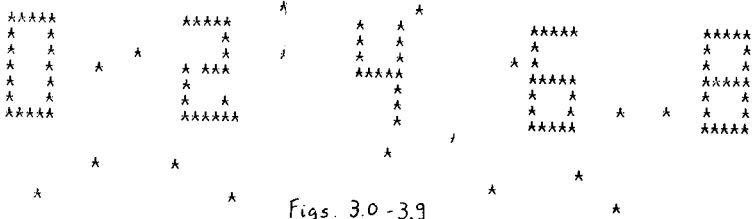
Tabla III

0.005	10	106.03
0.006	10	111.49
0.007	10	137.53
0.008	9	134.81
0.009	10	129.45
0.010	9	148.05
0.011	9	181.26
0.012	10	176.52
0.013	10	200.76
0.014	10	133.11
0.015	9	150.33
0.016	7	212.55
0.017	7	244.09
0.018	8	187.75
0.019	8	221.46
0.020	8	193.88
0.021	7	218.19
0.022	7	293.17
0.023	8	212.75
0.024	6	251.08
0.025	9	218.06
0.026	8	263.41
0.027	8	290.31
0.028	7	245.35
0.029	8	260.70
0.030	4	335.60
0.031	8	271.55
0.032	8	269.56
0.033	6	306.36
0.034	5	331.67
0.035	5	314.29
0.036	7	305.81
0.037	5	320.49
0.038	7	342.50
0.039	6	306.95
0.040	6	320.25
0.041	5	341.13
0.042	5	337.71
0.043	7	343.66
0.044	5	333.26
0.045	6	342.67
0.046	6	324.36
0.047	6	361.11
0.048	7	323.85
0.049	5	367.41
0.050	5	370.51

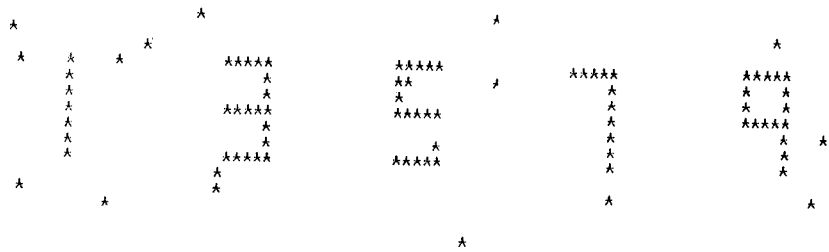
Tabla VI

0	10	0.00
10	10	1359.07
20	10	1648.11
30	10	1667.20
40	10	1657.71
50	10	1629.02
60	10	1681.49
70	10	1663.77
80	10	1311.16
90	10	13.94
100	10	1359.19
110	10	1648.11
120	10	1667.20
130	10	1657.71
140	10	1628.98
150	10	1681.46
160	10	1663.78
170	10	1311.33
180	10	0.00
190	10	1359.07
200	10	1648.11
210	10	1667.20
220	10	1657.71
230	10	1629.02
240	10	1647.07
250	10	1663.77
260	10	1311.16
270	10	13.94
280	10	1359.19
290	10	1648.11
300	10	1667.20
310	10	1657.71
320	10	1628.98
330	10	1647.03
340	10	1663.78
350	10	1311.33
360	10	0.00

Tabla VII

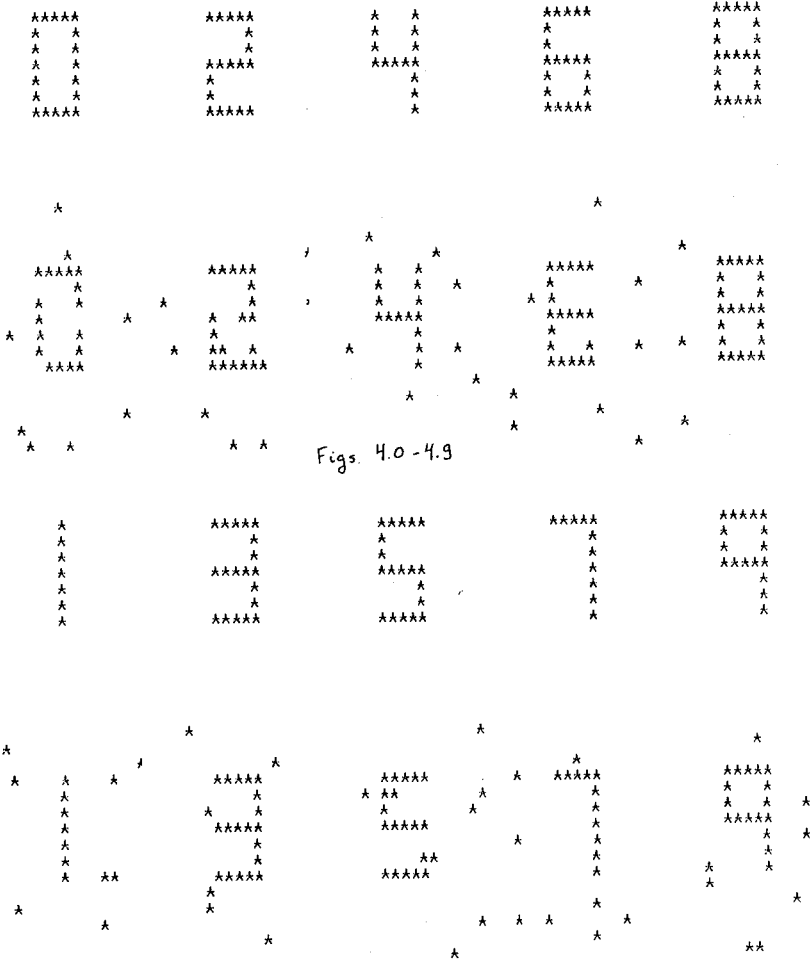


Figs. 3.0-3.9



<u>201.5</u>	816.9	451.1	453.9	366.6	442.9	451.7	498.2	530.4	391.3
834.4	<u>323.8</u>	825.9	850.2	576.4	827.7	982.0	342.8	1108.9	830.5
420.2	844.8	282.1	310.2	346.7	337.3	410.7	492.0	520.1	379.5
101.7	927.6	<u>202.5</u>	204.0	360.1	247.9	259.0	588.8	357.7	253.2
412.8	657.8	407.3	429.5	<u>176.8</u>	404.6	497.9	341.0	623.2	332.1
465.7	935.5	276.3	248.6	381.1	<u>191.2</u>	298.5	604.8	441.9	319.9
409.1	1016.7	327.4	277.0	436.1	234.4	<u>170.4</u>	678.9	310.3	220.5
581.4	397.7	600.8	587.2	345.8	594.7	712.2	<u>113.1</u>	829.1	573.1
428.7	1121.6	363.7	332.7	562.4	375.4	199.9	807.4	<u>98.8</u>	351.4
345.9	878.2	324.0	295.4	289.9	249.1	245.7	561.8	369.5	<u>110.3</u>

Tabla IV



474.6	689.3	535.8	565.5	376.8	538.7	643.5	<u>407.0</u>	756.3	555.6
837.6	396.9	820.0	847.0	584.3	817.7	980.7	<u>375.1</u>	1111.8	840.2
432.5	840.5	<u>404.1</u>	453.9	416.5	471.6	535.6	511.3	622.5	498.3
399.5	923.1	<u>266.8</u>	276.7	355.6	256.2	296.1	592.2	404.4	271.1
505.3	644.3	446.5	489.9	<u>251.3</u>	440.7	583.2	376.2	700.9	452.8
470.2	968.5	327.7	304.3	414.5	<u>267.6</u>	317.1	631.6	447.3	320.9
444.1	915.2	<u>275.5</u>	291.2	369.9	275.3	324.3	580.5	446.6	293.0
694.0	567.3	693.3	699.9	513.5	707.8	816.5	<u>356.4</u>	931.5	696.4
402.3	1091.7	338.5	318.1	524.0	350.1	198.1	772.6	<u>148.2</u>	329.4
324.1	848.7	357.9	389.8	289.7	354.1	378.8	517.9	488.8	<u>248.5</u>

Table V