

Aplicación de los Sistemas Evolutivos al Tratamiento de Imágenes

Fernando Galindo Soria

Escuela Superior de Cómputo (ESCOM)

Instituto Politécnico Nacional

Av. Miguel Othón de Mendizábal y Av. Juan de Dios Bátiz s/n

Zacatenco, Ciudad de México

07738 MÉXICO

fgalindo@ipn.mx

4 de Mayo de 1998

INTRODUCCIÓN HISTÓRICA.

Los Sistemas Evolutivos[1] surgieron a principio de los 80's como una respuesta a la necesidad de contar con sistemas capaces de transformarse en tiempo real a partir de los cambios ocurridos en su ambiente.

Bajo esta idea, en 1986, Sandra Camacho Villanueva, Patricia Gómez Rendón y Jesús Olivares Ceja, tres estudiantes-investigadores de la Licenciatura en Ciencias de la Informática de la Unidad Profesional Interdisciplinaria de Ingeniería y Ciencias Sociales y Administrativas (UPIICSA) del Instituto Politécnico Nacional (IPN) desarrollaron el primer Sistema Evolutivo que funcionó en forma real, conocido como Si-VE (Sistema de Visión Experto) [5], el cual tiene la capacidad de reconocer imágenes en 2D a partir de sus propiedades topológicas.

El mismo grupo de investigadores construyó a finales de 1986 un nuevo sistema evolutivo el cual tiene la capacidad de representar imágenes simples como cuadrados o triángulos en término de gramáticas con atributos y es capaz de reestructurar en tiempo real estas gramáticas, con lo que permanentemente está evolucionando. A partir de esta idea, se desarrolló el concepto de lenguajes de trayectoria[6] y se construyeron otros sistemas de este tipo, tanto dentro de los cursos como en trabajos de titulación[7].

Ahora bien, *los Sistemas Evolutivos orientados al tratamiento de imágenes como tales surgieron a partir de los trabajos que inicio Cuitláhuac Cantú Rohlík[8] a principios de los 80's cuanto era también estudiante-investigador en la UPIICSA* y que concretó en 1988 con una serie de productos orientados al tratamiento de imágenes, mismos que desarrolla y comercializa actualmente en su empresa VICON.

Los trabajos de Cuitláhuac han impactado fuertemente en el desarrollo de los Sistemas Evolutivos, pero la característica de productos de mercado de estos trabajos ha dificultado la difusión de sus técnicas y métodos, por lo que, solo algunos de sus ideas básicas se han difundido en foros y cursos de Inteligencia Artificial (IA) [17].

Por otro lado, Ángel Cesar Morales Rubio[2][3] desarrollo en forma independiente una serie de ideas y conceptos sobre tratamiento de imágenes que por un lado tienen mucho en común con los trabajos de Cuitláhuac Cantú y que por otro aclaran y amplían el área.

Parte del objetivo de este trabajo es difundir algunos de los métodos desarrollados por estos y otros investigadores, con el fin de que una persona sin conocimientos previos sobre el área sea capaz de empezar a aplicarlos y resolver problemas cada vez de mayor complejidad.

I. MANEJO DE IMÁGENES COMO DATOS ENORMES.

En primer lugar partiremos de que el sistema más simple para reconocer una imagen se reduce a una comparación, para lo cual veremos a una imagen como un conjunto de puntos, donde cada punto toma cierto valor que indica su color, por lo que al final de cuentas una imagen se puede ver como un registro de $n*m$ números (donde n es el número de renglones y m el de columnas). De donde, saber si una imagen es igual a otra, se reduce a comparar dos registros, y si éstos son iguales, entonces, las imágenes son iguales.

En particular las imágenes en una fotografía, televisión o monitor de computadora están formadas por una gran cantidad de puntos conocidos como pixeles, que al estar juntos dan la ilusión de estar unidos. Estos puntos se arreglan en una matriz, de tal manera que forman la imagen en un plano. En una imagen típica se pueden tener del orden de 500 x 600 puntos o sea alrededor de 300,000 pixeles, donde cada pixel toma un valor con el cual se indica su color.

En los sistemas más simples el pixel toma dos valores (0 ó 1) (blanco ó negro) por lo que sólo se requiere un bit para representarlo, pero si quiero manejar 16 posibles colores en cada punto ya se requieren 4 bits por pixel, en su momento existen sistemas que permiten manejar 64 (6 bits), 256 (8 bits) por

pixel) y hasta $2^{72} \approx 4 \times 10^{21} = 4$ mil millones de billones de “colores” para cada punto.

En este trabajo supondremos que cada punto ocupa un byte, o sea que una imagen de 300,000 pixeles ocupa en principio 300,000 caracteres y requiere ser almacenada de alguna forma dentro de la máquina. Existen muchas formas de almacenarla, pero la más simple consiste en guardar todos los puntos uno detrás de otro o sea que primero se almacenan todos los puntos del primer renglón, inmediatamente todos los del segundo y así sucesivamente hasta que se termina la imagen. O sea que *toda la imagen se puede ver como un dato enorme*.

Existen muchas otras técnicas para guardar una imagen, pero en términos prácticos y para facilitar la explicación asumiremos que una imagen se puede ver como un dato formado por muchos bits, con lo que, todas las técnicas para manejo de datos (comparación, búsqueda, ordenamiento, suma, resta, operaciones lógicas, concatenación, etc.) le son aplicables.

II. SISTEMAS EVOLUTIVOS DE REESCRITURA PARA TRATAMIENTO DE IMÁGENES.

Por ejemplo, se puede crear un “archivo” con dos columnas (imagen, nombre_imagen) y hacer un programa que busque una imagen en el archivo, si la encuentra diga como se llama, sino que pregunte su nombre y la inserte como un nuevo renglón. Lo anterior es la base de un sistema evolutivo de reescritura[4] orientado al tratamiento de imágenes y fácil de construir ya que en esencia su código consta de unas cuantas líneas como se puede ver a continuación

```

main()
{
  mientras no_fin
  {
    lee imagen_nueva
    mientras archivo_de_imagenes ≠ fin_de_archivo
      e imagen_nueva ≠ imagen
      lee la siguiente imagen del archivo
    si archivo_de_imagenes ≠ fin_de_archivo
      escribe nombre_imagen
    sino
      {
        pide nombre de la imagen_nueva
        almacena en el archivo imagen_nueva, nombre_imagen
      }
  }
}

```

Como se puede ver, en su forma más simple, el problema de reconocimiento se reduce a un problema de búsqueda, ahora bien si se tienen muchas imágenes o se tienen imágenes parecidas pero no idénticas la situación se complica.

El primer problema es de estructura de datos y en principio se puede resolver de múltiples formas, por ejemplo, ordenando las imágenes, creando un directorio con valores iniciales de la imagen, obteniendo un número representativo (hashing) de la imagen, almacenándola en forma compacta o fractal, etc.

III. NORMALIZACIÓN DE IMÁGENES MEDIANTE REJILLAS.

El problema de reconocer imágenes parecidas es más interesante y requiere de más tratamiento. La primera solución al problema de imágenes similares consiste básicamente en introducir un contador, de tal manera que, cuando se van comparando dos imágenes, se lleva la cuenta de cuantos puntos son iguales y al final se ve en que porcentaje fueron similares, si el porcentaje es mayor que algún número prefijado (por ejemplo 90%) se asume que las dos imágenes representan lo mismo, sino, se entra al proceso evolutivo para ver como se llama la nueva imagen.

El anterior, sigue siendo un método sencillo, sin embargo, no resuelve problemas como el de imágenes “movidas”, agrandadas o achicadas, este tipo de problemas tiene que ver con lo que en el área se conoce como búsqueda de

invariantes, donde una invariante es una característica de un objeto que no depende de su posición o tamaño, por ejemplo un triángulo es una figura cerrada de tres lados y el número de lados no cambia independientemente del tamaño o posición del triángulo. Por lo que a continuación y sin meteremos en detalles presentaremos una técnica, que aunque no demostraremos que produce invariantes, da buenos resultados prácticos.

Durante los 80's Cuitláhuac Cantú[17] introdujo dentro de sus sistemas evolutivos lo que llamamos una rejilla variable. La explicación de esta idea se dividirá en dos partes, en la primera explicaremos lo que es una rejilla y mas adelante se verá lo que es una rejilla variable.

Una rejilla se construye básicamente, tomando una imagen y sin importar su tamaño sustituyéndola por otra imagen de tamaño fijo (por ejemplo 100 x 100) conocida como imagen normalizada. O sea que, si se tiene una imagen de 500 x 500 se sustituirá por una de 100 x 100 y si se tiene otra de 300 x 320 también la deja en 100 x 100, con lo cual no importa cual sea su tamaño original al final tengo imágenes del mismo tamaño y por lo tanto el tamaño de las imágenes deja de ser un factor a tomar en cuenta. (Nuevamente, se puede argumentar que esto no funciona si se comparan perros con casas, pero es conveniente recordar que todo es poco a poco y paso a paso).

Existen muchas técnicas para construir la imagen normalizada, incluyendo:

- 1) Tomar un dato y varios no, por ejemplo si la imagen es de 500 x 500 y se pasa a 100 x 100, se toman los datos de 5 en 5.
- 2) Tomar datos en forma probabilística con una distribución uniforme, por ejemplo en el problema de 500 x 500 a 100 x 100, se están pasando de 250,000 puntos a 10,000 o sea uno de cada 25, entonces la probabilidad de tomar un punto de la imagen original es de 1/25avo.
- 3) Tomar el valor promedio de un grupo de datos, en nuestro ejemplo, la imagen original de 500 x 500 se divide en pequeños bloques de 5 x 5, se obtiene el valor promedio de cada bloque y este valor forma parte de la imagen normalizada.

Con estas y otras técnicas se puede obtener una imagen normalizada en la cual pequeñas variaciones de posición o de tamaño son absorbidas, por lo que, en lo que sigue del trabajo se supone que estamos trabajando con una imagen normalizada (aunque mucho de lo que veremos se puede aplicar a las

imágenes originales, ya que, existen problemas que manejan datos de tamaño y posición similares).

IV. MANEJO DE IMÁGENES MEDIANTE FIRMAS.

Si se observa con cuidado desde que se introdujo la búsqueda de objetos similares y mas adelante el manejo de las rejillas, realmente ya no se están comparando objetos iguales sino solo similares, ahora bien, si por mala suerte la primera imagen de un objeto no es la mas representativa, cuando llega una imagen nueva del objeto, es probable que el sistema evolutivo no la encuentre, por ejemplo, si la primera imagen es una foto borrosa de mi perro, la siguiente vez que llegue una foto de mi perro es muy posible que los puntos iguales sean muy pocos y por tal motivo el sistema no la reconozca.

Con el fin de evitar lo anterior, desde que surgieron los primeros métodos de reconocimiento de formas y en particular de imágenes se ha usado que, en lugar de almacenar una imagen de un objeto lo que se almacena es el promedio de múltiples imágenes o formas del mismo objeto conocido como *firma del objeto*. O sea tomo muchas fotos de mi perro desde la misma posición y almaceno la “foto promedio” del perro.

Entonces si se compara una imagen nueva del perro es más probable que el sistema la asocie con el perro promedio y no con un gato promedio.

Nuevamente existen múltiples técnicas para encontrar y almacenar la firma. La más simple consiste en que dadas dos imágenes se suman punto a punto y se dividen entre dos y el resultado se almacena en la máquina. El problema en este caso es que fácilmente se llega a errores de truncamiento. Otra opción consiste en sumar muchas imágenes y sacar el promedio. Por ejemplo si se tienen n imágenes del perro $I_1, I_2, I_3, \dots, I_n$ la imagen promedio I_p se obtiene mediante la expresión:

$$I_p = \frac{\sum_{j=1}^n I_j}{n}$$

donde la suma y división se realiza pixel a pixel.

Una alternativa que da mejor resultado en algunos casos consiste en almacenar directamente el resultado de la suma y mantener aparte un contador que indica el numero de imágenes acumuladas.

Por ejemplo si se tienen las siguientes imágenes:

I_1	3	0	5	0	8	9	3
I_2	3	0	4	0	7	9	3
I_3	4	0	5	0	7	9	3

La imagen acumulada (I_A) $I_A = \sum_{j=1}^n I_j$ quedaría como sigue :

I_A	10	0	14	0	22	27	9	3
-------	----	---	----	---	----	----	---	---

donde el 3 final indica el número de imágenes que se acumularon.

Esta representación tiene la ventaja de que los errores de truncamiento se reducen.

Ahora bien, si se quiere reconocer una imagen nueva (I_N) :

I_N	3	0	5	0	7	9	3
-------	---	---	---	---	---	---	---

Una técnica puede ser la siguiente :

Se multiplica la imagen nueva por el número de imágenes (3 en el ejemplo anterior):

I_N	9	0	15	0	21	27	9
-------	---	---	----	---	----	----	---

Se resta en valor absoluto, punto a punto de la imagen acumulada :

	I_A	10	0	14	0	22	27	9
-	I_N	9	0	15	0	21	27	9
=	I_D	1	0	1	0	1	0	0

el resultado se divide punto a punto entre el número de imágenes

$$\boxed{I_D \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0}$$

I_D es el resultado del proceso y en este ejemplo indica que las imágenes son prácticamente iguales.

Ahora bien el proceso anterior se puede representar con la siguiente ecuación

$$I_D = \frac{I_A - I_N * n}{n}$$

donde si sustituimos

$$I_A = \sum_{j=1}^n I_j$$

queda

$$I_D = \frac{(\sum_{j=1}^n I_j) - I_N * n}{n} = \frac{\sum_{j=1}^n I_j}{n} - I_N = I_P - I_N$$

Es decir, que I_D es la diferencia entre la imagen promedio (I_P) y la imagen nueva (I_N)

$$I_D = I_P - I_N$$

A pesar de que las ecuaciones indican que los dos métodos son equivalentes, en términos prácticos y computacionales son diferentes. (En lo que sigue asumiremos que lo que se almacenan son las imágenes acumuladas, ya que esto facilita la explicación).

V. MANEJO DE MATRICES EVOLUTIVAS.

Existen otros métodos para ver si dos imágenes son similares y la mayoría se basan en el cálculo de la distancia entre las imágenes. Específicamente las técnicas de comparación con una imagen o patrón promedio se han utilizado dentro del área de reconocimiento de formas cotidianamente.

Sin embargo desde el principio se ha adolecido de varios problemas, por ejemplo, para usar estas técnica primero se tiene que “enseñar” al sistema, o sea darle un conjunto de imágenes seleccionadas que le permitan encontrar el

promedio, lo cual ocasiona que antes de poder usar el sistema se realice un trabajo previo que a veces es monótono y en algunos casos impráctico.

Por otro lado, si se quiere reconocer un nuevo objeto o alguna otra vista de un objeto ya reconocido se necesita detener el sistema y realizar un nuevo proceso de aprendizaje. Para atacar los problemas anteriores, Cuitláhuac Cantú desarrolló dentro de sus sistemas evolutivos un método que en esencia consta de lo siguiente:

Se parte de que originalmente el sistema se encuentra vacío y cuando llega la primera imagen se busca, como no se encuentra, se le pregunta al usuario como se llama, almacenándose la imagen, un contador n que indica que se tiene una imagen y su nombre.

La siguiente vez que llega una imagen el sistema la busca y si la encuentra similar con la imagen almacenada entonces las acumula (suma la nueva imagen a la que ya existía) e incrementa el número de imágenes acumuladas (n), si llega una imagen que no se reconoce entonces el sistema pregunta su nombre, si el nombre de la imagen corresponde a alguna existente entonces acumula sobre esta e incrementa el contador, si es una imagen nueva la almacena en el siguiente registro libre junto con su contador y nombre.

Como se puede observar este método reúne la característica de que esta evolucionando en forma natural y encontrando la imagen acumulada (y por tal, la imagen promedio $I_p = I_A / n$), *con lo que no existe un proceso previo de aprendizaje y otro de aplicación, sino que por el proceso natural de reconocer las imágenes va evolucionando.*

Al conjunto de vectores que representan las imágenes que reconoce el sistema se le conoce como matriz evolutiva[9]. (Las matrices evolutivas constituyen por si solas una herramienta con múltiples aplicaciones [12][13][14][15][16][19]).

Los trabajos de Cuitláhuac tiene también la característica de que el sistema está permanentemente evolucionando, ya que, cuando entra una imagen el sistema la acumula sobre algún vector de la matriz y mientras no entre una nueva imagen o se le indique lo contrario la vuelve a acumular y la sigue acumulando indefinidamente. Por lo que, una imagen dada puede tener una gran repercusión sobre su representación interna.

VI. REJILLAS VARIABLES Y PIRÁMIDES.

Ahora bien, si el sistema sumara la misma imagen una vez y otra y otra y k veces esto sería equivalente a sumar una copia de la imagen multiplicada por k y tal vez no resultaría tan práctico, por lo que, en este punto conviene recordar que, en su técnica Cantú introduce una rejilla variable, que consiste básicamente en un mecanismo de muestreo cuyos límites se están moviendo aleatoriamente.

Esto es, cada que la rejilla toma una “fotografía” de la imagen, esta es ligeramente diferente y como en un proceso dado se toman k fotografías y todas se acumulan sobre el mismo vector, el efecto es equivalente a tomar k imágenes ligeramente diferentes del mismo objeto con lo que la velocidad de convergencia crece rápidamente.

El contar con una rejilla variable permite subsanar algunos de los problemas de las técnicas tradicionales, ya que, *se esta partiendo de un sistema que trata con imágenes similares, movidas y parciales.*

Son similares porque, aunque sea la misma imagen, la rejilla toma puntos diferentes y los acumula, de entrada la rejilla se esta moviendo lo que es equivalente a que la imagen se esta moviendo, o sea que, si posteriormente se da una imagen similar o movida para el sistema es simplemente otro caso.

El movimiento de la rejilla se puede complicar tanto como se quiera, por ejemplo puede ser hacia los lados y arriba/abajo, también puede ser pendular con lo que se puede hacer que la rejilla gire cierto numero de grados a la izquierda y derecha lo que equivalen a “ver” la imagen inclinada, como todo se acumula en el mismo saco, al sistema le es indiferente que la imagen llegue derecha o inclinada.

También este enfoque de rejilla variable facilita el reconocimiento de imágenes bajo información parcial, ya que la misma rejilla en ciertos momentos sólo “ve” fragmentos de la imagen.

Para lograr un mayor nivel de reconocimiento Cuitláhuac ha desarrollado otras técnicas y métodos aún no difundidos y de los cuales solo algunos se

han presentados parcialmente en cursos y exposiciones[17] como por ejemplo la técnica de fractalización que se pueden resumir como rejillas de rejillas.

Para entender la fuerza de las rejillas de rejillas variables que acumulan información sobre una matriz evolutiva nos tenemos que referir a los trabajos de sistemas evolutivos desarrollados en forma independiente por Ángel Cesar Morales Rubio[2][3][18].

En estos trabajos Morales Rubio llegó a la conclusión de que *una imagen se puede ver como una pirámide en múltiples dimensiones y que se mueve*.

Para lo cual, primero construye una rejilla variable sobre la imagen I_0 con lo que obtiene una imagen nueva I_1 (recuérdese que la rejilla obtiene una muestra de la imagen original, pero esta muestra se puede seguir viendo como una imagen, de la misma forma que un sonido digitalizado se sigue manejando como sonido) ahora construye una rejilla sobre la imagen I_1 obteniéndose una imagen I_2 y así sucesivamente hasta llegar a un valor I_n , que ya no se puede reducir.

Con esta técnica Morales Rubio obtiene una pirámide de rejillas cuya base es la imagen original y la punta es una rejilla de tamaño 1. Esta pirámide tiene prácticamente almacenada la información de la imagen en diferentes niveles de agregación.

Pero no queda ahí la cosa, un día Morales Rubio me dijo “y sin embargo se mueve”, al principio no le entendí, por lo que me explicó que estamos cometiendo un grave error ya que las técnicas de reconocimiento partían de que las imágenes eran objetos fijos y si por algún motivo la imagen se movía lo primero que hacíamos era detenerla, pero por otro lado los sistemas de visión de la naturaleza (como el ojo humano, de la rana, etc.) aprovechan que los objetos se mueven para reconocerlos y aun más el ojo en forma natural está moviéndose permanentemente.

Al aplicar esta idea Angel Morales planteó que una imagen se puede ver como un objeto que se mueve, con lo cual llegamos nuevamente al concepto de una pirámide de rejillas variables (En este punto es importante comentar que a pesar de que Cuitláhuac y Ángel no se conocían, generaron ideas similares y llegaron independientemente a resultados parecidos).

Ahora bien, la rejilla variable es el mecanismo de percepción de la matriz evolutiva, con lo que, tal vez la pareja pirámide de rejillas variables/matrices evolutivas sea un buen modelo de los mecanismos de manejo de información percepción/almacenamiento/procesamiento de los seres vivos y en particular de la visión natural.

VII. MANEJO DEL PUNTO DE ATENCIÓN.

El símil entre mecanismos informáticos y biológicos no se queda ahí, sino que se puede continuar por ejemplo con el concepto de punto de atención (o punto de visión o fijar la atención) que consiste básicamente en un mecanismo que permite fijar la atención o la vista en un punto o zona específica de una escena.

Una forma de introducir este concepto a nuestro sistema de visión automatizado para buscar una imagen dentro de una escena, puede consistir en un mecanismo que primero realiza movimientos grandes sobre la escena y donde se posiciona compara mediante una rejilla rala (con pocos puntos) y si resulta válida la comparación aplica rejillas cada vez de mayor densidad.

Por otro lado si la comparación original no resulta debe volver a saltar y a comparar y así continuar hasta que encuentre algo o se le indique que realice otra función.

Un sistema sencillo de visión aplicando las técnicas anteriores puede atacar básicamente dos tipos de problemas:

- 1) Encontrar un objeto en un escena dada.
- 2) Reconocer (o conocer si no se conocía) un objeto que se encuentra dentro de una escena.

Encontrar un objeto en una escena se puede realizar básicamente mediante la técnica planteada anteriormente y consiste básicamente en movimientos aleatorios del punto de atención (que es una rejilla variable poco densa) sobre la escena hasta que se obtiene una colisión satisfactoria y aplica rejillas de mayor densidad (esto se parece a un método de ordenamiento conocido como ordenamiento de peine, que es un método de burbuja modificado para que las burbujas suban más rápido en las primeras etapas). En particular, reconocer un objeto que se encuentra dentro de una escena se reduce a fijar la atención (poner la rejilla) sobre el objeto a reconocer y aplicarle las técnicas anteriores.

CONCLUSIÓN.

En este trabajo se presentaron una serie de técnicas y métodos que se han desarrollado desde principios de los 80s en el área de los sistemas evolutivos aplicados al tratamiento de imágenes.

La idea fue mostrar por un lado una serie de resultados que han costado en algunos casos años de esfuerzo obtener y que se pueden llegar a perder. Estos trabajos se han ido obteniendo mediante un trabajo continuo y consistente y han permitido que poco a poco se desarrolle una comunidad de investigadores sobre esta área, con resultados propios y que buscan que nos consolidemos cada vez más, para en su momento competir dentro del contexto mundial y no ser simples receptores y seguidores de otros desarrollos.

Es por lo anterior que, en este trabajo se presentaron los resultados en una secuencia que permita a una persona interesada empezar desde cero y poco a poco ir insertando la siguiente capa, buscando que en su momento llegue a construir su propio camino.

FUENTES DE INFORMACIÓN.

- [1] Galindo Soria, Fernando, "*Sistemas Evolutivos*", en Boletín de Política Informática, INEGI-SPP, México. Septiembre de 1986
- [2] Morales Rubio, Ángel Cesar, "*Sistema Evolutivo para Tratamiento de Imágenes*", trabajo de titulación, IPN-UPIICSA México, 1992.
- [3] Morales Rubio, Ángel Cesar, "*Algoritmo Evolutivo Para Tratamiento de Imágenes*", IPN-UPIICSA México
- [4] Galindo Soria, Fernando, "*Sistemas Evolutivos de Reescritura*", en Memoria del curso tutorial de Sistemas Evolutivos, Ier Congreso Internacional de Investigación en Ciencias Computacionales, Metepec México, 1994
- [5] Camacho Villanueva, Sandra, Gómez Rendón, Guadalupe Patricia, Olivares Ceja, Jesús Manuel, "*SI-VE: Sistema de Visión Experto*", IPN-UPIICSA, México, 1986
- [6] Galindo Soria, Fernando, "*Sistemas Evolutivos de Lenguajes de Trayectoria*", En VI Reunión de Inteligencia Artificial, Memorias, Ed. Limusa, Junio 1989, Querétaro, Qro.
- [7] Olicón Nava, Carlos, "*Sistema Evolutivo Manejador de Imágenes en 2D con movimiento*", trabajo de titulación, IPN-UPIICSA México 1992
- [8] Cantú Rohlík, Adalberto Cuitláhuac, "*Sistemas Evolutivos para Reconocimiento de Imágenes*", IPN-UPIICSA
- [9] Galindo Soria, Fernando, "*Una Representación Matricial para Sistemas Evolutivos*", Conferencia magistral Simposium Internacional de Computación, IPN-CENAC, 1993
- [10] Tisher, "*Pc Interno*", ed. Marcombo, 1992

- [11] Jiménez Aviña, J. Antonio, Galindo Soria, Fernando, "*Diseño y Construcción de Sistemas Interactivos, Aplicando Experiencias Basadas en el Tratamiento de Imágenes*" en Memorias de la 2da Conferencia de Ingeniería Eléctrica CIE/96, CINVESTAV-IPN, 1996
- [12] Torres Hernández, Luis E, Longoria, Luis C., Rojas Salinas, Antonio, "*Aplicación de los Sistemas Evolutivos en el Análisis de Espectros de Rayos Gamma*", en Memorias del Cie/95, Primera Conferencia de Ingeniería Eléctrica CIE/95, CINVESTAV-IPN, Septiembre 11-13 de 1995 México, D.F.
- [13] De La Cruz Sánchez, Eduardo, Longoria Gándara, Luis C. Carrillo Mendoza, Rodolfo A., "*Sistema Evolutivo para el Diagnóstico de Fallas en Maquinas Rotatorias*", en Memorias del Cie/95, Primera Conferencia de Ingeniería Eléctrica CIE/95, CINVESTAV-IPN, Septiembre 11-13 de 1995 México, D.F.
- [14] Arzola Carvajal, Irene, Cruz Reyes, José Rafael, "*Sistema Evolutivo para el Reconocimiento de Texto Taquigrafico*", en Memorias del Cie/95, Primera Conferencia de Ingeniería Eléctrica CIE/95, CINVESTAV-IPN, Septiembre 11-13 de 1995 México, D.F.
- [15] García García, Diana Karla, Salcido Bustamante, Sergio, Ventura Silva, Alfonso, "*Sistema Evolutivo de Reconocimiento de Formas en dos Dimensiones*", en Concurso Nacional de Ciencia y Tecnología, CONADE, México, 1996
- [16] Olivares Ceja, Jesús Manuel, "*Sistema Evolutivo Reconocedor de Textos*", IPN-CIC, México, febrero, 1997
- [17] Cantú Rohlík, Adalberto Cuitláhuac, comunicaciones personales y cursos de Representación del Conocimiento, Inteligencia Artificial y Redes Neuronales
- [18] Morales Rubio, Ángel Cesar, comunicación personal
- [19] Galindo Soria, Fernando, "*Expert Systems Based on Evolutive Matrix*" , IPN-ESCOM, México, abril 1997