

2ej.



**UNIVERSIDAD NACIONAL AUTONOMA
DE MEXICO**

FACULTAD DE INGENIERIA

**SISTEMAS CONEXIONISTAS DE
CARACTERIZACION Y ASOCIACION DE
INFORMACION EN RECONOCIMIENTO DE
PATRONES**

T E S I S

QUE PARA OBTENER EL TITULO DE:

INGENIERO EN COMPUTACION

P R E S E N T A :

WALTERIO WOLFGANG MAYOL CUEVAS

DIRECTOR DE TESIS: ING. STALIN MUÑOZ GUTIERREZ



MEXICO, D. F.

1999

**TESIS CON
FALLA DE ORIGEN**

280361



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

ÍNDICE

1. Parte Primera.....	1
1.1. Se puede volar sin plumas.....	1
1.2. Cualidades de los modelos conexionistas.....	2
1.3. El esquema de Super-Patrones.....	8
2. Reconocimiento de Texturas por Medios Táctiles usando LVQ.....	10
2.1 Introducción.....	10
2.2 Trabajo relacionado.....	10
2.3 Sistema tegumentario humano y diferencias entre sensores.....	11
2.4 Plumilla sensora.....	12
2.5 Procedimiento.....	12
2.6 Resultados.....	13
2.7 Discusión.....	14
3. Reconocimiento de Rostros Empleando Coeficientes de Autocorrelación Local.....	18
3.1 Planteamiento del Problema.....	18
3.2 Metodología Escenario Estático.....	19
3.3 Coeficientes de Autocorrelación Local.....	19
3.4 Método de Clasificación.....	20
3.5 Resultados de Clasificación en Ambiente Estático.....	21
4. Red de Kanerva en 2D para Computación Óptica.....	23
4.1 Introducción.....	23
4.2 Red de Kanerva.....	23
4.3 Computación Óptica.....	24
4.4 Red de Kanerva Modificada en 2D.....	25
4.4.1 Prueba de llenado.....	26
4.4.2 Cantidad de Matrices del Espacio que se Cubren.....	26
4.5 Discusión y resultados.....	30
5. Discusión	
5.1 Redes de caracterización y asociación de información.....	31
5.2 Resumen.....	33

PRELIMINARES

Definición del problema:

Las técnicas de Inteligencia Computacional (CI) son muy variadas [23], entre éstas se encuentran las denominadas redes neuronales artificiales (RNA) o sistemas conexionistas, sistemas que permiten trabajar como mapeadores de funciones de muy diversos tipos y que trabajan con conocimiento al estilo numérico.

En esta tesis, se describirán y en ciertos casos se extenderán tres RNA.

Los sistemas que serán analizados en detalle son: la red de Kanerva, para la cual se presentará una extensión original a dos dimensiones, lo que permite que dicho modelo pueda implementarse con ventajas sobre otros paradigmas en computadoras paralelas especiales como las ópticas.

Los otros dos sistemas que se presentarán serán la Red de Regresión General o GRNN, aplicada a un problema de reconocimiento de rostros.

Así como se presentará el sistema de Cuantización Vectorial de Kohonen, aplicado a clasificación de texturas táctiles, para el cual se presentará un sistema original en hardware y software.

Con respecto a la relevancia del problema, es importante mencionar que las aplicaciones que se presentarán, son consideradas como difíciles y para los que no se tienen soluciones generales, como son el reconocimiento de rostros o el reconocimiento de texturas.

Por otro lado, en RNA, no está muy definido el cómo los diferentes paradigmas se relacionan, de tal manera que podamos saber qué ventajas de herramienta matemática pueden emplearse entre una "familia" de estos sistemas, o más aún, sería importante entender los mecanismos comunes que hacen que cada uno de estos sistemas funcionen.

Como objetivo a futuro, se tiene no solo el presentar aplicaciones en los problemas mencionados, sino el tratar de encontrar un mecanismo de funcionamiento común entre los sistemas involucrados.

Método:

Conviene subdividirlo como sigue:

Red de Kanerva: La red de Kanerva se presentó hace poco más de diez años, sin embargo, a pesar de sus prometedoras propiedades de funcionamiento sencillo y versátil, ésta no ha sido empleada de manera generalizada principalmente debido a que tiene requerimientos de memoria altos, comparativamente con los de otros sistemas. Se presentará una extensión original para el caso de dos dimensiones, en donde se mostrará cómo es que esta conversión la hace más eficiente, especialmente para computadoras de tipo paralelo. Se presentará una implementación en una computadora óptica que realiza parte de las funciones más costosas del modelo desde un punto de vista computacional.

El por qué se eligió una implementación óptica del sistema, es debido a que así se tiene una velocidad de procesamiento superior a las de una computadora convencional, cosa que ha probado ser útil para manejo de grandes volúmenes de información como imágenes.

Red de Kohonen LVQ: Aquí el sistema será evaluado en software, recibiendo datos de un equipo diseñado expresamente que muestra manualmente texturas por medios táctiles.

La investigación de métodos de reconocimiento de texturas de medios táctiles es escasa en la literatura consultada, puesto que en general el nivel de precisión buscado es alto. Sin embargo, aquí se presentará una aproximación barata y sencilla basada en técnicas de reconocimiento de voz y en una plumilla sensora diseñada, consistente en un piezoeléctrico cuya señal es transformada vía FFT.

El reconocimiento de las señales obtenidas, es realizado por un sistema de cuantización vectorial que ha probado su utilidad en el manejo de señales similares.

Red de Regresión General GRNN: Este sistema será probado en un problema de reconocimiento de rostros de frente sobre fondo uniforme. Es importante resaltar que éste es un problema difícil donde se tiene un escenario con objetos cambiantes en posición, iluminación, variedad de aditamentos (anteojos, aretes, barba, etc...) por lo que es importante la exploración de técnicas nuevas que permitan atacar el problema de manera eficiente.

En este caso, se propondrá una combinación de técnicas de pre-procesamiento y la red GRNN, para trabajar sobre un conjunto de por lo menos 50 personas.

Se elige la red de GRNN debido a que ésta pertenece a el grupo de las Radial Basis Function Networks, que han probado su eficiencia en el reconocimiento de señales fijas tipo plantilla, pero además ésta red permite ser entrenada sin iteraciones, siendo una ventaja indiscutible para cuando se requiere trabajar en ambientes en línea, donde las iteraciones cuestan tiempo de operación.

Referencias y bibliografía especializada¹:

- [1] R. Brunelli and T. Poggio, "Face Recognition: Features versus Templates", *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 15, no. 10, pp. 1042-1052, Oct. 1993.
- [2] F. Goudail, E. Lange, et al., "Face Recognition System Using Local Autocorrelations and Multiscale Integration", *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 18, no. 10, Oct. 1996.
- [3] L. Wiskott, J. Fellous, et al. "Face Recognition and Gender Determination", *Proc. Intern. Workshop on Automatic Face and Gesture Recognition*, Zurich, Suiza, pp. 92-97, 1995.
- [4] L. Wiskott, "Phantom Faces for Face Analysis", accepted for publication in *Pattern Recognition*, 1996. <http://www.neuroinformatik.ruhr-uni-bochum.de>
- [5] N. Otsu and T. Kurita, "A New Scheme for Practical, Flexible and Intelligent Vision Systems", *Proc. IAPR Workshop on Computer Vision-Special Hardware and Industrial Applications*, pp. 431-435, Tokyo, 1988.
- [6] D.F. Specht, "A General Regression Neural Network", *IEEE Trans. Neural Networks*, vol. 2, no. 6, pp. 568-567, 1991.
- [7] R. García and W.W. Mayol, "Reconocimiento de Imágenes Empleando Redes de Regresión General y la Técnica TVS", *Proc V Congreso Iberoamericano de Inteligencia Artificial*, Cholula, México, pp. 508-515, Nov. 1996.
- [8] Mayol W. Gómez E. 2D sparse distributed memory-optical neural network for pattern recognition. IEEE International Conference on Neural Networks ICNN94. Orlando, USA. June 28-July 2, 1994.
- [9] P.D. Wasserman, *Advanced Methods in Neural Computing*, VNR Publishers, New York, 1993 .
- [10] Lipman P. An introduction to computing with neural nets IEEE ASSP Magazine. 1987.
- [11] Yu, F. Taiwei L. & Xiang Y. Optical neural network with pocket sized liquid crystal televisions. Optical Society of America. 1990.
- [12] Xiang Y. Taiwei L. & Yu, F. Compact optical neural network using cascaded liquid crystal television. Applied Optics 1990.
- [13] Mayol W. & Gómez E. Técnicas ópticas paralelas para procesamiento en redes neuronales. Eight International Conference. UNAM CU. 1992
- [14] Mayol Cuevas W., J Juárez Guerrero & S Muñoz Gutiérrez. A First Approach To Tactile Texture Recognition. IEEE Systems, Man and Cybernetics Conf. San Diego Calif. USA. 1998.

¹ Es común a muchos, que descuidemos en ocasiones la bibliografía que encontramos en los textos y artículos que leemos pues caemos en la tentación de confiar que los autores son auténticos magos de la síntesis o certeros en sus apreciaciones sobre el trabajo de sus consultados, sin embargo, no hay mejor costumbre para un minero de la información que saber rastrear con prudencia las raíces de los textos que lee, apoyado de estas auténticas señales.

- [15] Rogers D. Statistical Prediction with Kanerva's Sparse Distributed Memory. RIACS Technical Report. 1989.
- [16] Rogers D. Weather Prediction Using a Genetic Memory. RIACS Technical Report. 90.6. 1990.
- [17] Gómez, E. & Mayol, W., Procesamiento de Imágenes utilizando Técnicas de Computación Óptica. XXXVI Congreso Nacional de Física. Acapulco, Gro., México, del 18 al 22 de octubre de 1993.
- [18] Aleksander, I. An Introduction to neural computing, Chapman & Hall. 1990.
- [19] Kohonen T. The "neural" Phonetic Type Writer, IEEE. 1988.
- [20] Yu, F. & Saganda, J. Optical Signal Processing Computing and Neural Networks. Wiley 1992.
- [21] V. Ramos-Viterbo, E. Gómez-Ramírez, W. Mayol-Cuevas (LINDA-UNAM), V. Martínez-Martínez, F. García de León Ramírez, O. Balderas-Pereyra. Computadora óptica: fundamentos, arquitectura y algoritmos. XXXVIII Congreso Nacional de Física. Zacatecas, Zacatecas, Mexico del 16 al 20 de octubre de 1995.
- [22] Muñoz-Gutiérrez, S; Mayol-Cuevas, W.; Sánchez-Guzman,R., A Toroidal Random Structure as Base for an Associative Machine. Technical Report #R09101995. Laboratorio de Investigación para el Desarrollo Académico (LINDA group). Div. Ing. Eléctrica. Facultad de Ingeniería UNAM. A.P: 70-500, C.P: 04510, México.
- [23] J. Bezdeck, What is Computational Intelligence? In Computational Intelligence imitating life, ed. J. Zurada, R. Marks, et.al. IEEE Press. NJ. 1994.

I Parte Primera

La palabra inteligencia, es una palabra muy fuerte, si pudiéramos responder a la pregunta ¿Qué es la inteligencia?, al momento "siguiente" podríamos construir una *Máquina Inteligente*, o tal vez podríamos (si tuviéramos a un bioquímico o bioquímica muy buena al lado) hacer una "pastillita" que nos de un poco más de eso. Honestamente, no creemos que el definir una palabra como esta nos ayude a hacer algunas de las cosas que queremos tratar en este ensayo, sin embargo, trataremos de investigar al respecto de la propiedad llamada Inteligencia para cierta etapa dentro del procesamiento de información dentro de un sistema artificial.

La cantidad de tareas atribuibles a la inteligencia es tan grande que sería poco práctico presentar aquí aunque sea el planteamiento de cómo intentar resolverlas.

Sin embargo, existe un conjunto de conductas asociados con la palabra inteligencia como el proponer y ejecutar acciones para responder a problemas que en general presentan más de una opción, por ejemplo, poder adivinar la clave de un buen amigo para jugarle una broma en su computadora sin probar con todas las opciones posibles. Para realizar tareas de este tipo decimos que requerimos de eso llamado inteligencia y lejos de definirla nos dedicaremos a intentar abordar algunas de las técnicas que permiten trabajar en problemas que requieren de una cierta experiencia para resolverse o problemas que realizamos cotidianamente los organismos biológicos de cierta complejidad, como identificar rostros o texturas; tareas que son solo casos particulares de tareas que combinadas con otras etapas, son atribuibles a la inteligencia y se agrupan difusamente en la sub-área denominada Reconocimiento de Patrones.

En este capítulo introductorio y en los subsecuentes no pretendemos dar un tratado completo de este muy vasto tema, en cambio nos hemos esforzado en intentar presentar de forma concreta y breve algunos de los métodos "avanzados" que caen en la categoría de red neuronal, empleadas aquí para el reconocimiento de patrones, incluyendo sólo las ecuaciones más básicas y prefiriendo una descripción más esquemática de la operación de estos sistemas de tal manera que sean accesibles para cualquier amante de la ciencia computacional. Es así que tenemos que sugerir la consulta de las referencias incluidas en cada sección para quien desee información mas extensa que sería impráctica de incluir en este trabajo.

1.1 Se Puede Volar sin Plumas

Con operaciones que nosotros llamaríamos suma y resta, un hombre primitivo podía calcular cuánto tiempo podía subsistir su familia con cierta comida almacenada, aún sin tener la comida físicamente o sin necesitarla realmente, es decir podía simular un evento: la distribución del alimento, incluso podía plantear diferentes condiciones de número de consumidores, diferentes raciones y demás parámetros; con el tiempo descubrió que concatenando operaciones resolvía problemas más complejos dando nacimiento a la idea de lo que hoy conocemos como algoritmo, una de las bases de la computación.

Gracias a la existencia de la sociedad, mientras unos individuos proponían formas de simular eventos, otros se dedicaban a tratar de entender y generar teorías de cómo es que los primeros proponían estos algoritmos, naciendo así la lógica del pensamiento, que ya vislumbra a lo que aquí nos atañe, pues ésta es raíz de la Inteligencia Artificial.

Se dio un gran avance cuando empleando esta lógica, que hoy conocemos como lógica clásica, se construyeron máquinas hechas con engranes y tarjetas perforadas que podían resolver problemas que hasta antes de ellas sólo resolvía el hombre con su cerebro, problemas como las premisas lógicas y cálculos como los necesarios en las ecuaciones diferenciales; al mismo tiempo y posiblemente sin querer se probó que la constitución física de una cosa que resolvía ese tipo de problemas, nada tenía que ver con la forma del cerebro humano, no tenían neuronas!, y esto es muy importante como veremos más adelante.

Claro que la lógica tuvo sus "detalles", en sus inicios buscó los pasos definitivos, las "leyes escritas sobre granito", leyes invariables que permitieran saber cómo se realizaban los procesos inteligentes; y por esta rigidez, que parece incompatible con la Naturaleza llena de imprevistos e imprecisiones, las máquinas lógicas no fueron suficientes para afrontar problemas mas "reales".

Más adelante, cuando existió un mejor entendimiento del funcionamiento de ciertas partes del cerebro, surgió la posibilidad de arriesgarse a construir máquinas que imitaran al cerebro, a través de elementos que imitaran sus neuronas, de ahí que a estos intentos, se les ha denominado Redes Neuronales.

Posteriormente, sabiendo que la imitación del cerebro no es perfecta con estos sistemas, sino más bien al nivel de una metáfora, se les ha denominado menos pretenciosamente como Modelos Conexionistas, término que emplearemos en lo sucesivo. Estos modelos lograron atacar problemas considerados difíciles para los sistemas basados en lógica clásica, sobre todo problemas que involucran señales del mundo real (voz, espectros, imágenes, etc.), y por ello han tenido un tremendo impacto en la disciplina.

Hasta aquí hemos comentado de manera muy rápida y general la evolución de los sistemas en la Inteligencia Artificial, y ya queremos pasar a ver modelos que resuelvan problemas, pero antes, es conveniente que iniciemos con ciertas aclaraciones: ¿por qué tratamos a los modelos que presentamos en este trabajo, sabiendo que existe una gran cantidad de modelos conexionistas?, ¿por qué estos en particular?, pues bien, dentro de los Modelos Conexionistas, existen dos grandes orientaciones, una es la de basar a los sistemas en estructuras biológicas y otra es la de basarlos en Matemáticas. De aquí retomamos la pregunta histórica: ¿para volar se necesitan plumas?

Situaciones parecidas al intento de imitar al cerebro mediante neuronas artificiales se han presentado muchas veces en la historia de la humanidad, cuando el hombre quería volar intentó imitar el vuelo de los pájaros, sus artefactos tenían incluso plumas. Después de un auténtico proceso evolutivo y de otro de resignación, tenemos aviones que vuelan no precisamente como lo hacen los pájaros (aunque seguimos añorando poder hacerlo con su gracia y eficiencia); este ejemplo nos sugiere que para poder realizar lo que llamamos máquinas inteligentes, quizás no necesitamos copiar hasta la última cualidad de las neuronas que poseemos, tal vez lo podemos hacer de otras maneras y en este trabajo preferimos tomar la inclinación de no relacionar a los modelos que presentaremos con modelos biológicos (sin embargo el copiar modelos biológicos para construir sistemas que resuelvan tareas complejas, también es válido a nivel científico y no deseamos implicar que se tenga que desechar por completo) porque preferimos tomar la inclinación de las Matemáticas para construir este tipo de modelos, esto es muy importante y hay que tenerlo muy en cuenta, hay que recordar a los aviones!

1.2 Cualidades de los Modelos Conexionistas

I. Alexander Oparin, quien planteó una de las primeras explicaciones modernas sobre el origen de la vida escribía:

".. no existe ninguna diferencia fundamental entre un organismo viviente y la materia que llamamos inanimada. La compleja combinación de manifestaciones y propiedades que es tan característica de la vida debe haber surgido en el proceso de la evolución de la materia..."

Con ésto, Oparin dejaba claro que toda la belleza de funcionamiento de un organismo podía ser explicada sin la necesidad de algo más que materia, este punto consideramos es muy importante para el investigador o el iniciado en el estudio de la Inteligencia Artificial, sencillamente le da la esperanza y permiso de trabajar al respecto, de lograr en un momento dado, construir o establecer condiciones que lleven a que "algo" tenga la propiedad tan interesante de la Inteligencia.

Nos parece evidente lo deseable que sería poder trasladar ciertas de nuestras habilidades a las máquinas, ya que estas pueden en principio realizar cosas muy difíciles para nosotros, como potencialmente llegar a lugares tales como el interior de una célula o explorar un cuerpo celeste distante, o tal vez algo más difícil: habitar y desenvolverse en una ciudad.

Además sería deseable tener máquinas generales es decir que podamos aplicarlas para diferentes problemas, y ya que estamos tan pretenciosos, quisiéramos también que trabajen en nuestro impreciso y cambiante mundo.

Éstas posiblemente son algunas de las metas más buscadas en Inteligencia Artificial.

Pero ¿a qué habilidades nos referimos?. Para acercarnos al objetivo de este capítulo de que sea una introducción general, tendremos que concretarnos en algunos problemas que son una pequeña parte de los problemas tratados por la Inteligencia Artificial, nos referimos a problemas como: discernir letras escritas o imágenes, decidir entre varias opciones de rutas para ser seguidas por un vehículo, predecir el clima de mañana, y problemas parecidos.

Algunos de estas tareas son cotidianamente atacadas por los humanos y otras inclusive por ciertos organismos muy simples que nos rodean.

Si nos detenemos un momento a analizar las tareas arriba mencionados, encontramos que tienen ciertas cosas generales en común, en ellas hay una entrada o estímulo que pueden ser las letras o la temperatura de los días anteriores, después hay un proceso que es el proceso mismo de la elección, o la predicción; y después existe una respuesta a ese proceso-estímulo, que denominamos salida y que puede ser: "lloverá mañana" o la mejor ruta es la "X".

Este fenómeno de relacionar estímulos con respuestas, en términos matemáticos se le denomina mapeo (se mapea el vector la entrada en el de salida), y se esquematiza en la figura 1.

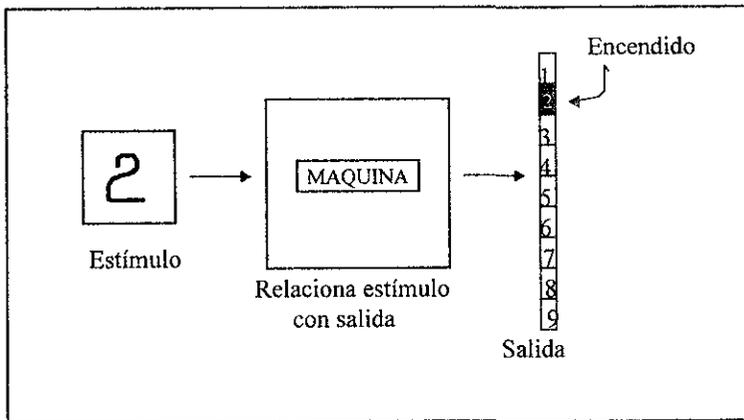


Figura1. Máquina clasificadora de números.

Inspeccionando levemente la máquina de la figura 1, por ahora de funcionamiento desconocido, podemos observar que parece que recibe estímulos y los clasifica como salida en alguno de los números de la derecha, indicado por un cuadro oscuro o encendido, descartando a las demás opciones (demás números) indicándolo con cuadros blancos o apagados; es decir parece que se trata de una máquina clasificadora de números del 1 al 9.

¿Qué propiedades necesita una máquina para poder realizar una tarea de clasificación de números como ésta?

Así es como llegamos a la primera de las cualidades de los modelos conexionistas: *el entrenamiento*.

A los seres biológicos de cierta complejidad parece que no les es suficiente la información genética que reciben de sus progenitores para poder sobrevivir en su entorno; es así que por ejemplo, los humanos hemos valorado la necesidad innata de la comunicación, desarrollando mecanismos complejos como los lenguajes, que permiten aumentar directamente nuestra supervivencia. Éste tipo de mecanismos generados se aprenden a manera de estructuras sintácticas o fonéticas, pero además de aprenderse, pueden olvidarse o modificarse conforme las exigencias del entorno, es así que los lenguajes han evolucionado, mezclándose con otros lenguajes viejos y nuevos. Si las máquinas que desarrollemos habrán de trabajar en nuestro caprichoso mundo, deberán de ser capaces de recibir y modificar la información que tengan para

poder sobrevivir, deben de ser capaces de ser entrenadas por nosotros y de preferencia en general por el mismo entorno.

El entrenamiento, se presenta en dos *sabores* principalmente: supervisado y no supervisado.

El entrenamiento supervisado consiste en la presentación de estímulos asociados con la respuesta que se debería dar al aparecer esos estímulos; como ejemplo está la presentación de una letra (estímulo) a un niño y la pronunciación correcta (respuesta) por la/el maestra(o)¹.

El entrenamiento no supervisado es aquel que se presenta cuando no hay maestro explícito que dé esa *respuesta correcta* para el estímulo dado, por lo que en general, el sistema propone una respuesta que se va mejorando hasta lograr un cierto resultado deseado o suficiente.

La mayoría de las situaciones de aprendizaje combinan los dos tipos de entrenamiento, el niño del ejemplo anterior es "entrenado" a asociar letra escrita con pronunciación, sin embargo este aprendizaje se puede descomponer en una serie de reacciones fisiológicas como abrir la boca, controlar las cuerdas vocales y demás acciones motoras, que de antemano cuando se esta aprendiendo por primera vez, y aunque parezca haber cierta instrucción natal sobre estos procesos, no se sabe exactamente como coordinarlas correctamente, y es así que poco a poco se va corrigiendo hasta tener una pronunciación que es de alguna forma "aceptada", pero los controles neuromotores que intervienen en esa coordinación parecen no "saber" de antemano cómo coordinarse para pronunciar una letra "a", proponen una respuesta y se mejora hasta lograr una aceptación o buena pronunciación.

Es así como una tarea específica involucra a los entrenamientos supervisado-no supervisado.

Volviendo a la figura 1, tenemos que esta máquina de alguna manera tendría que "conocer" cuando menos previamente las cosas con las que va a trabajar, tendrá que ser capaz de ser entrenada, con los números 1 al 9.

Para entrenar a nuestra máquina utilizaremos el concepto de *conjunto de entrenamiento*, en donde se dan diferentes variantes de un mismo tipo de estímulo, es decir le presentaremos varios números uno, varios números dos y así hasta completar los números que queramos que sean procesados.

Si además establecemos una respuesta deseada para cada tipo de numero de tal forma que al conjunto de los números uno le asociamos la cadena {100000000} que tiene un "encendido" en la primera posición y al conjunto de números dos le asignamos la cadena {010000000} que tiene un "encendido" en la segunda localidad, y hacemos lo propio para cada subconjunto de números (es decir damos una cadena identificadora para cada uno de los conjuntos), estaremos practicando un aprendizaje supervisado y nuestra máquina sabrá qué cosa debería de responder cuando se presente cada estímulo.

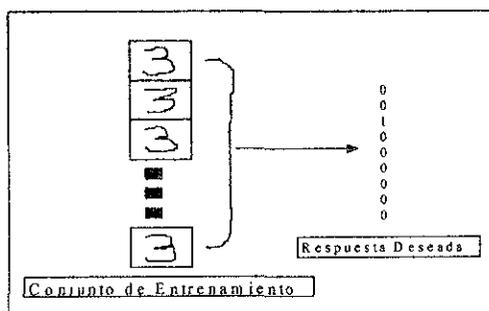


Fig.2 Conjunto de entrenamiento que contiene varios números tres y su respuesta propuesta, la posición del número uno (tercer lugar de arriba a abajo) indica que el conjunto es de números tres.

¹ N. del E. Al poner el femenino en primer plano no se establece una preferencia de este genero para uno u otro trabajo, sino que considerado esto en el contexto general de escritos, es solo para recalcar lo indistinto como equidad

Por cada número que vayamos a procesar deberemos de tener un conjunto como el de la figura 2 (con su respectiva respuesta asociada), uniendo todos estos conjuntos tenemos el conjunto de entrenamiento completo de nuestra máquina.

Teniendo el conjunto de entrenamiento podemos pasar a las siguientes etapas del aprendizaje que veremos con detalle en las secciones siguientes puesto que la manera en que las máquinas conexionistas son entrenadas no es única, depende del tipo particular de modelo utilizado; pero en términos generales el aprendizaje en estos sistemas consiste en cambios internos de la máquina (cambio de valores numéricos o conexiones) que permiten dar correctamente respuestas a los estímulos presentados.

Ahora bien, ¿cuántas maneras distintas hay de dibujar un número tres?,muchas!!!.

¿Tendremos que poner en nuestro conjunto de entrenamiento todos los posibles números tres?, sería algo *prácticamente imposible y muy absurdo, por fortuna esto no es necesario.*

Los humanos tenemos la capacidad de clasificar cosas que nunca habíamos visto, por ejemplo, si algún día nos encontramos con una silla desconocida, de todas maneras la clasificamos como silla por ciertas características claves que la hacen ser lo que es, características que podrían ser el tener cuatro patas, algo parecido a un respaldo y una asentadera. A esta habilidad se le conoce como *generalización* y esta es otra de las cualidades importantes que tienen los modelos conexionistas.

Hasta aquí hemos mencionado la habilidad del entrenamiento y la habilidad de la generalización.

Otra cualidad mas bien morfológica pero esencial de los modelos conexionistas es que tienen *muchos procesadores simples*, pero que conectados colectivamente resuelven más fácilmente problemas que a otros sistemas de un solo procesador les es mas difícil; si estos procesadores simples se organizan jerárquicamente, estratificadamente, de tal suerte que su acomodación sea en capas, son capaces de resolver problemas complejos más fácilmente, pero esto lo abundaremos en las secciones siguientes ya que podemos acomodar a estos modelos conexionistas de tantas maneras que la organización en capas (que gusta mucho a los mismos que les nombran como redes neuronales) es solo una forma de verlos.

Es conveniente anticipar que un sistema conexionista estará definido, diferenciado y clasificado con respecto a otros sistemas conexionistas por dos cualidades básicas: una **topología** que es el cómo se conectan o jerárquizan nuestros "procesadores simples" y una **regla de entrenamiento**, que se encarga de encontrar los valores, parámetros o pesos que requiera la topología mencionada para resolver la tarea que se le establezca. El encontrar la topología y la regla de entrenamiento adecuada no es un problema trivial así como tampoco lo es el encontrar una vez definidos estos parámetros los valores, o pesos que permiten resolver la tarea que especificamos v.g. cómo construir internamente una máquina conexionista que reconozca números manuscritos y después cómo encontrar los parámetros para que reconozca la clase "numero tres" específicamente.

¿A qué nos referimos con procesadores simples, aquellos que metafóricamente se han comparado con las neuronas biológicas?, estos procesadores simples son realmente simples en los modelos conexionistas que presentamos en este ensayo, y posiblemente esa sencillez es la que hace que trabajen. Por ejemplo uno de estos procesadores puede ser un simple *sumador* que realice la suma algebraica de los estímulos que le envían los otros procesadores y que esta suma sea enviada como salida a otros procesadores, si complicamos un poco el modelo e imitamos en lo general la respuesta de las neuronas reales, podemos agregar un *módulo de salida especial dentro de este procesador*, de tal manera que su salida no sea simplemente la suma algebraica sino esa suma procesada con una "función especial" que generalmente es una función llamada no-lineal, y aquí encontramos un concepto básico de la ciencia contemporánea, La No-linealidad.

Si tenemos varios leños y el objetivo es hacer una fogata, podemos ensayar varios métodos de ignición. Si los intentamos encender con un cerillo, seguramente la temperatura y duración de la flama no es suficiente y no los alcanzan a prender y por lo tanto no tenemos fogata; si en vez de un cerillo empleamos un soplete, no habrá problema, tendremos la fogata rápidamente, pero si en vez de soplete empleamos una antorcha de periódico seguramente lograremos también encender la fogata, es decir, el resultado será el mismo que el obtenido con el soplete a pesar de que el estímulo (temperatura de la flama) fue bastante menor, es así que para obtener la respuesta (lograr fogata) basta que el estímulo sobrepase cierto umbral, cierto nivel de temperatura, y si esto es así ya no importa que tanta más temperatura se

aplique, la respuesta se consigue, y por ello se dice que el comportamiento es no lineal, al no cambiar la respuesta del sistema proporcionalmente conforme cambia la intensidad del estímulo; veámoslo gráficamente:

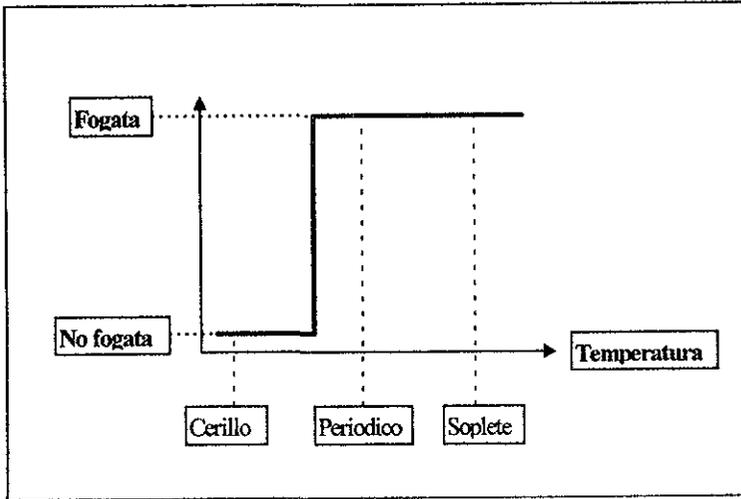


Figura 3. Comportamiento No-Lineal del encendido de una fogata con respecto a la temperatura alcanzada por diferentes materiales de encendido.

Más que pretender entender el mecanismo de encendido de las fogatas, la figura 3 ejemplifica cómo un fenómeno relativamente sencillo tiene un comportamiento *especial*, no-lineal; y el hecho de insistir en esta peculiaridad, es debido a que por lo menos las neuronas biológicas, nuestras células sensoriales y muchas otras cosas del mundo tienen este comportamiento, por lo que recordando la estructura de los procesadores simples de más arriba, a éstos les podemos incluir la propiedad de tener este tipo de respuesta, y de hecho se puede demostrar matemáticamente que el tener esta propiedad es una de las que hacen que los sistemas conexionistas funcionen y resuelvan problemas.

Aún falta algo importante por establecer, y esto es las conexiones que existen entre los mismos procesadores. Estas conexiones pueden entenderse como canales de comunicación que propagan el resultado de un procesador a otro, es decir si del procesador emisor sale un número "1" y al procesador receptor llega un "0.5", el canal multiplica por "1/2" la respuesta del emisor, este valor de multiplicación se conoce como peso y a pesar de que muchos de los modelos conexionistas no usan pesos (o usan en todas las conexiones un "1" como multiplicador, es decir no afectan lo que comunican) la idea de que la información se va modificando como se propaga dentro del sistema es muy interesante y muy útil como veremos más adelante.

Por lo que para completar la forma general de nuestros "procesadores neuronales" tenemos el esquema de la figura 4.

Es porpío mencionar que las representaciones gráficas si bien son útiles para darse una idea de cómo operan las cosas, no son necesariamente las mejores representaciones para implementar y trabajar con esas mismas cosas, por ejemplo, el modelo de Rutherford del átomo en forma de "sistema solar" sirvió bastante en su época para entender algunas cosas generales del micromundo, sin embargo para poder *hacer predicciones de nuevas partículas* o la cuantificación de las reacciones atómicas, esa representación no sirve de nada y por ello nuestro modelo de átomo contemporáneo es bastante lejano a la simplicidad de "sistemita solar", podemos decir que ni siquiera cabe una representación gráfica, más bien matemática al

estilo de las ecuaciones de Shrodinger o las matrices de Heisenberg (que nota aparte son equivalentes pero ambas matemáticas).

Con esto queremos advertir que las representaciones gráficas pueden ser un poco "tramposas" y nos pueden hacer entender las cosas equivocadamente, por ello empleamos en este ensayo una orientación un poco más con tendencia matemática al pretender que lo que aquí se diga ayude a la implementación de estos sistemas.

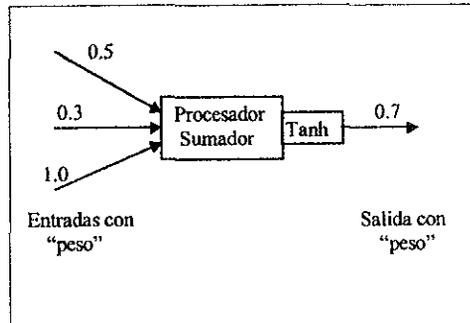


Figura 4. Esquema general de un procesador como los empleados en varios de los sistemas conexionistas. El "procesador sumador" realiza la suma algebraica de cada una de las entradas (que vienen multiplicadas por un peso) y a esa suma les aplica una función no lineal que en este caso es la Tanh vista más arriba.

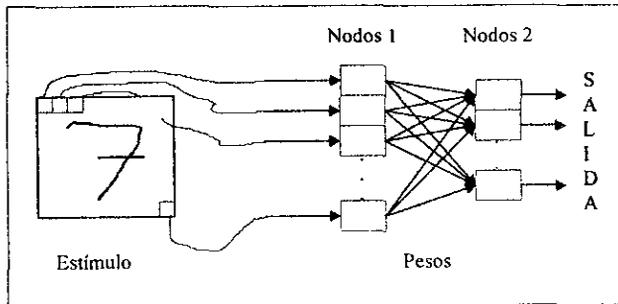


Figura 5. Esquema general de un sistema conexionista, cada nodo es como en la figura 4. Los Píxeles o elementos del estímulo se dan como entrada a los primeros nodos, la respuesta de estos se multiplica por los pesos y se propaga a los nodos de la capa siguiente que da finalmente la respuesta.

Relacionando el esquema de la figura 5 con el de la figura 1, vemos que tenemos más claro el esqueleto de nuestro sistema conexionista, podríamos ya establecer como le haríamos para reconocer números manuscritos, podemos adelantar que para poder obtener las respuestas deseadas ante los respectivos estímulos tenemos que encontrar los valores adecuados que resuelven el problema de clasificación de números, si encontramos los pesos adecuados, por ejemplo cada vez que presentemos un número tres obtendremos como salida el vector {001000000} y lo propio para cada número manuscrito como establecimos en la figura 2, pero como hemos mencionado por ahí, la manera de resolver los problemas es particular a cada sistema conexionista y preferimos tratar a cada uno por separado como sigue en las secciones siguientes.

1.3 El esquema de Super-Patrones.

Aquí comenzamos a tener contacto con nuestro primer modelo conexionista.

En particular analizaremos uno de los modelos más sencillos de reconocimiento, que incluso se encuentra como parte constitutiva de otros sistemas conexionistas más elaborados, se llama Modelo de Super-patrones o de Mapas de Ocurrencia.

Tomemos nuestro problema de clasificación de números y construyamos una máquina que puede identificarlos y decirnos "es un cuatro" o "es un cinco" indicándolo con un vector como el dispuesto en la figura 2.

Recordemos que necesitamos tener un conjunto de entrenamiento por cada símbolo o número a procesar.

Ahora, si tomamos un recuadro sobre el que escribimos el primer número tres de nuestro conjunto de entrenamiento, y encima el siguiente y el siguiente, y repetimos el proceso hasta haber sobre escrito el último de los estímulos de ese conjunto de entrenamiento, observaremos algo parecido a la lo que se muestra en la figura 6. Algo así como tener una sabana sobre la que con una brocha dibujamos los números, si dibujamos un mismo número varias veces sobre la sabana, y como casi siempre los haremos ligeramente diferente a los anteriores, obtendremos una zona en la que la brocha nunca tocó, otras que fueron tocadas solo pocas veces, y otras en donde la pintura es más gruesa pues la brocha tocó siempre. Si a cada zona le asignamos un "peso" o valor conforme a las veces que fue tocada la sabana, de tal forma que ese peso es mayor si fue tocada siempre y nulo si fue tocada nunca, habremos creado lo que se denomina super patrón o bien un mapa de ocurrencias para cada número, y este mapa es el que nos ayudará a resolver el problema de la clasificación.

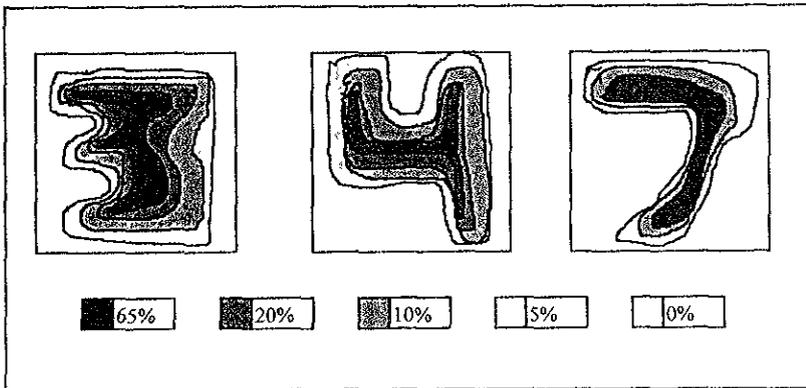


Figura 6. Mapas de ocurrencia de números manuscritos "3", "4" y "7".

El entrenamiento en este modelo conexionista en particular consiste en tener una de estas "sabanas" o mapas de ocurrencia por cada conjunto de entrenamiento que tengamos (por cada número) y como se aprecia esto es muy fácil de lograr en este sistema.

Pero después del entrenamiento viene la hora de la verdad, la clasificación de un estímulo que en general no se encontraba exactamente igual en los conjuntos de entrenamiento. Imaginemos que llega un número tres como entrada al sistema, ¿Qué operación tendremos que realizar para que la salida del sistema indique que es un número tres?, pues simplemente colocamos ese tres desconocido sobre cada una de los mapas de frecuencia, y vemos para cada uno de sus pixeles, en qué zona cae, si cae en una zona de 65% acumulamos un .65, si cae en una zona de 10% sumamos un .10, y así lo hacemos hasta el último pixel del estímulo; al final la suma resultante la dividimos entre el número total de puntos o pixeles que contiene el estímulo y así tenemos el resultado de ese mapa de ocurrencia con respecto a la entrada a

clasificar, si el resultado fuera por ejemplo .75, significa que se tiene 75% de certidumbre de que el número a clasificar pertenece a ese mapa de ocurrencias, en el caso que aquí tratamos, los resultados de comparar el estímulo con cada uno de los mapas de la figura 6 podrían ser algo así: 75% para el mapa de números "3", 15% para el de números "4" y 55% para el de números "7"; aplicando el criterio de asignar como ganador al mapa con la máxima certidumbre, tendríamos en nuestro ejemplo hipotético que ese número desconocido es asignado efectivamente como un tres ya que la mayor respuesta (75% de certidumbre) la tiene el mapa que elaboramos en nuestro entrenamiento para el conjunto de los números tres.

Es así que podemos construir una máquina que realice lo que comentamos anteriormente, y solamente requeriremos una matriz en donde escribir los números tanto para entrenamiento como para clasificación, una matriz de memoria para cada símbolo a procesar y después un comparador y un acumulador que nos dé la certidumbre de cada pixel de la matriz de entrada para obtener el resultado final, y así de fácil podemos construir un modelo conexionista simple pero poderoso!

Por supuesto que nuestro modelo tiene desventajas, en primer lugar la zona donde escribimos el estímulo es limitada, así como no puede (como está) clasificar números de diferentes escalas (números más pequeños o más grandes), tampoco a diferentes orientaciones o en diferentes posiciones dentro del recuadro, nuestro sistema está limitado, sin embargo no hay que desanimarse tempranamente, para ello existen métodos de preprocesamiento que permiten resolver los problemas de escalamiento, orientaciones y posición dentro del recuadro, estas técnicas se conocen como métodos invariantes y hay varios que en esta ocasión solamente los mencionaremos pues nos desviaría desglosarlos con el cuidado que merecen, entre estas técnicas se encuentran: La transformada de Fourier y Las técnicas de Momentos.

Con el planteamiento de este modelo de superpatrones, deseamos mostrar lo sencillo que puede ser uno de estos sistemas de nombre elegante; por otro lado, hemos descrito este sistema en particular no por azar ni por capricho, sino porque su estructura es base para sistemas como los que presentaremos en los capítulos siguientes, sistemas estos que pueden trabajar con las debidas condiciones, en uno o en todos los problemas ejemplificados en la figura 7.

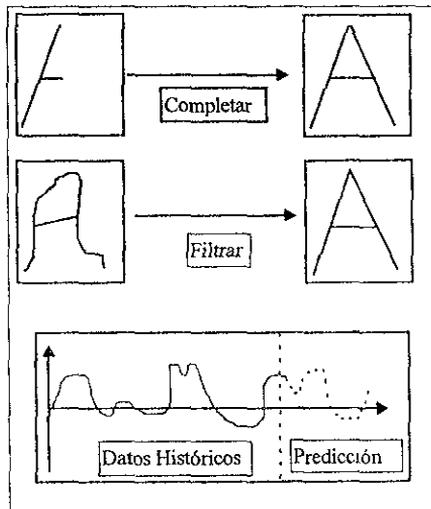


Figura 7. Esquemas de diferentes tareas realizables por los Sistemas Conexionistas.

II Reconocimiento de Texturas por Medios Táctiles usando LVQ¹

2.1 Introducción

De todo el conjunto de sentidos por medio de los cuales el hombre percibe su entorno, pocos son los que se han estudiado en el área de inteligencia artificial, los trabajos existentes son principalmente para visión y una pequeña sección de la audición humana: el reconocimiento de voz. Sorprende que los otros sentidos (tal es el caso del tacto) que podrían robustecer ampliamente las representaciones del mundo en sistemas artificiales hayan sido muy poco trabajados. Para el tacto, los trabajos existentes se limitan a sensores de presión, siendo que el tacto humano permite sentir múltiples variables físicas como temperatura, humedad y un sinnúmero de texturas.

Recientemente el interés de tener aparatos exo-sensores, ha motivado el desarrollo de sistemas que sirvan de interfaces entre el medio ambiente y los sensores humanos, este es el caso de los sistemas de tele-presencia, en donde se ha dado un desarrollo interesante de los sistemas táctiles. La mayoría de estos sistemas transmiten información al receptor humano solamente sobre la forma de los objetos y muy pocos sobre la textura y temperatura de los mismos. En este tipo de sistemas (y en general), se ha hecho muy poco en el área de reconocimiento automático de la información enviada.

Reconocer *in-situ* la información puede ser importante por varias razones, por ejemplo cuando los tiempos de transmisión son grandes, como en el caso de transmisiones actuales interplanetarias, en donde es prácticamente imposible responder rápidamente v.g. ante una eventualidad de un sistema de exploración geológica[19,20]; así mismo el reconocimiento automático de información sería importante en el caso de cuando los sistemas en los que está implementado son muchos y no están disponibles humanos para cada uno de estos sistemas, v.g. en un sistema distribuido de exploración médica.

2.2 Trabajo relacionado

El trabajo relacionado comprende los desarrollos en el entendimiento de los sistemas táctiles en humanos y animales, los sistemas táctiles en robótica y los exo-sensores y/o sistemas de tele-presencia, así como las técnicas de reconocimiento de señales dinámicas como la voz, sin embargo debido a la gran cantidad de trabajos existentes en estas áreas, nos limitaremos a mencionar algunos de los más recientes.

Algunos de los trabajos que colectan lo que se ha hecho en el sentido táctil son [2-4].

En Howe et. al. (1993) [9] se presenta posiblemente uno de los primeros esquemas para percepción dinámica táctil basado en sensores de deformación, más recientemente Okamura et. al.(1997) [15] presentan un sistema de exploración háptico basado en sensores táctiles, propio para la manipulación de objetos, sin embargo los sensores empleados son sensores básicamente dedicados a percibir presión sobre objetos deslizándose que poseen protuberancias aisladas mayores a 3mm En el trabajo de Okamura et. al. se comenta el reconocimiento de texturas finas, sin embargo ni un solo resultado, técnica o planteamiento en este sentido es presentado. Información dinámica para la detección temprana de un objeto que se resbala de un manipulador ha sido utilizada por autores varios v.g Tremblay & Cutkosky (1993) [13] y Cutkosky & Hyde (1993) [14], en donde el tipo de sensores utilizados impiden detectar texturas finas para su eventual reconocimiento.

¹ Este experimento se realizó en colaboración con Jesús Juárez Guerrero

Una parte importante del desarrollo de los sistemas de reconocimiento en general dependen fuertemente de los dispositivos (hardware) existentes, en el caso de reconocimiento táctil de texturas, es necesario contar con sensores mecánicos que puedan percibir las finas protuberancias que poseen los objetos. En el caso de los sensores táctiles, se ha trabajado con diferentes aproximaciones, que incluyen sobre todo piezoeléctricos, sensores ópticos, acelerómetros y sensores de deformación (strain gage) [4,5,9,22].

2.3 Sistema tegumentario humano y diferencias entre sensores

El tacto ha sido uno de los sentidos más ancestrales con el que los seres vivos han contado para percibir el mundo; en el caso de los humanos, éste ha sido tradicionalmente sustituto de sistemas perceptores más elaborados como la vista, además de ser usado cotidianamente para manipular y reconocer objetos muy diversos. Hacia el exterior, este sentido se manifiesta principalmente en la piel.

La piel es la cubierta de casi toda la superficie corporal, es una envoltura elástica deslizable bajo ligera tensión, con la capacidad de auto-regenerarse; en su superficie se encuentran surcos ramificados con patrones geométricos irregulares en las puntas de los dedos [1], mismos que sirven tanto para aumentar la fricción como también sirven para incrementar la sensibilidad táctil (Meano et.al. 1997) [12]. El volumen de la piel varía y posee un espesor medio de 2.2mm. En cuanto a la integración neuro-cutánea, se han detectado cinco principales receptores morfológicamente diferenciados, a los que se les ha semi-establecido un funcionamiento también diferenciado, es así que la sensación de dolor se asocia a las terminaciones nerviosas libres, la sensación de frío a los corpúsculos de Krause, la sensación de presión a los corpúsculos de Vater-Pacini, la sensación de tacto a los de Meissner y la sensación de calor a los de Ruffini; sin embargo se sabe que las concentraciones de dichos corpúsculos no es la misma en las diferentes zonas del cuerpo, así como se ha comprobado que las diferentes terminaciones nerviosas pueden detectar estímulos variados, y que a su vez producen sensaciones de frío, dolor, tacto, presión y calor, independientemente de la presencia o no de órganos histológicamente diferenciados [1]. Lo anterior sugiere que nuestra percepción cutánea y específicamente la de nuestro interés, la táctil, proviene del reconocimiento de una *configuración* de respuestas de diferentes sensores, más que de la simple respuesta de un solo tipo de sensor individual, lo cual parece importante de imitar en sistemas artificiales táctiles, además tener la evidencia de que una idea parecida ha sido probada recientemente con éxito para reconocimiento olfativo por Freund & Lewis (1995) [17]; a este punto de reconocimiento de configuraciones de respuesta volveremos más adelante.

Acerca de lo anterior es importante resaltar que en el caso de los sensores principalmente asociados con el tacto "fino", los corpúsculos de Meissner, poseen en las yemas de los dedos de la mano una profundidad de aproximadamente 0.35mm desde la superficie de la piel, y están localizados muy cerca de la epidermis (la parte más superficial de la piel), mientras que los corpúsculos que se dedican a la presión, los de Pacini, tienen un tamaño mayor y una concentración menor que los de Meissner, además de estar a una profundidad de cerca de 2mm, en el tejido graso subcutáneo, por debajo de la dermis.

De lo anterior se evidencia una profunda diferencia entre los dos sensores especializados para los diferentes estímulos tacto y presión; sin embargo la mayoría de las investigaciones en percepción táctil artificial parecen seguir la morfología y funcionamiento de los sensores de presión (en humanos los de Pacini) y no la de los asociados con el tacto (los de Meissner).

A pesar de que ambos sensores Meissner y Pacini son sensores que responden a excitación mecánica, parece existir una gran diferencia en los umbrales que los disparan, además de las variaciones en concentraciones y localización; y si bien como se mencionó más arriba los diferentes sensores tegumentarios pueden responder a los diferentes estímulos detectados por la piel, hay unos más adecuados

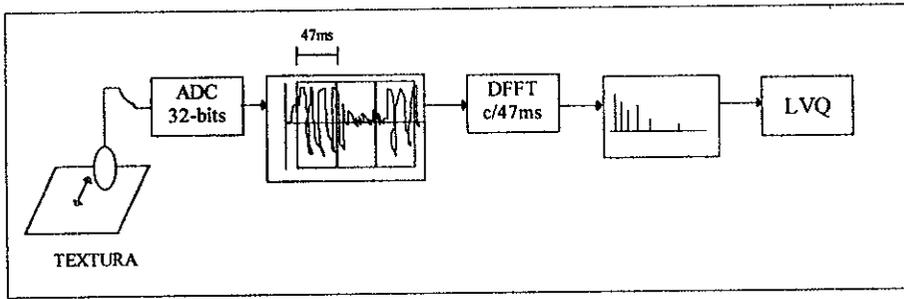


Figura 1. Esquema general del sistema.

que otros para ciertas percepciones, de ahí es que en este trabajo empleamos sensores mecánicos mucho más sensibles que los empleados convencionalmente para percepción táctil, que unidos con técnicas de reconocimiento de voz nos permiten trabajar con texturas sin precedente que incluyen desde materiales lisos como el vidrio, hasta texturas tan especiales como una barba humana.

2.4 Plumilla sensora

Desde que nuestro interés es el percibir texturas lo más finas posibles, y como se comentó en la sección anterior, para dicho tipo de percepciones se requiere de sensores que detecten cambios mecánicos sutiles, una posibilidad para tener un transductor de texturas es un micrófono, el cual se modifica fácilmente con excitación mecánica débil y además existe una gran cantidad de programas e infraestructura para procesamiento y visualización de las señales que dichos sensores entregan. En este trabajo se eligió un micrófono electret, no solamente por la disponibilidad comercial sino porque la construcción de los mismos en un futuro permitiría una alta escala de integración, con lo que podría tenerse una matriz con diferentes densidades y distribuciones de electrets; si bien hay una gran diferencia en tener muchos sensores y uno solo como en este trabajo, creemos que esta configuración experimental es suficiente para demostrar las posibilidades del sistema. En la figura 1 se muestra el esquema del sistema completo desarrollado, la figura 2 muestra la respuesta del sistema al ser deslizado sobre una protuberancia. La figura 2 muestra las señales obtenidas para dos texturas del experimento.

2.5 Procedimiento

Material	Material	Material
1. Poliuretano	7. Aluminio	13 Madera
2. Papel Bond	8. Unicel	14. Barba
3. Hule texturizado	9. Neopreno	15. PVC
4. Cartón	10. Papel texturizado	16. Mezclilla
5. Estireno.	11. Vidrio	17 Madera barnizada
6. Vinil texturizado	12. Palma de la mano	18 Lija agua #120

Tabla 1. Lista de los diferentes materiales usados en los experimentos de clasificación automática de texturas.

Se emplearon los 18 materiales listados en la tabla 1 para conducir el experimento.

Esta vez, los materiales fueron muestreados manualmente, frotando la plumilla (que tiene un recubrimiento de fieltro) en cinco diferentes orientaciones sobre el material: en círculo, horizontal, vertical, diagonal positiva y diagonal negativa. El lector puede darse una idea de lo que el transductor percibió, al escuchar de cerca el sonido que se produce al frotar un dedo contra dos texturas diferentes como la palma de la mano y la manga de la camisa. Se muestreó a 44.1 Khz con una definición de 16 bits; de dichas muestras se seleccionaron arbitrariamente dos segundos de cada orientación por cada material, lo que da 90 "palabras" o texturas en la base de datos (no se tomaron en cuenta los intervalos donde "aterriza" la plumilla sobre la textura, sino solamente las zonas cuando ya estaba frotándose).

Para realizar el procesamiento, se siguió una técnica común en reconocimiento de voz, que es el seccionamiento, aproximadamente cada 47ms (2048 datos) de señal se obtuvo la DFFT, el vector resultante se introdujo en un sistema clasificador supervisado basado en "supervectores" o templates LVQ de Kohonen (Kohonen 1988; Torkkola 1991) [6-8]. Las reglas de aprendizaje fueron las siguientes:

- si la entrada "x" fue clasificada en la clase correcta, aplicar la ec. 1 solamente al vector m_i más cercano.
- si la entrada "x" fue clasificada incorrectamente, aplicar la ec. 1 solamente al m_i más cercano de su clase y la ec. 2 únicamente al m_j ganador (el de la clase incorrecta).

donde:

$$m_i(t+1) = m_i(t) - \eta(t)[x(t) - m_i(t)] \quad (1)$$

$$m_j(t+1) = m_j(t) + \eta(t)[x(t) - m_j(t)] \quad (2)$$

η se eligió igual a 0.25. x es la FFT de un estímulo de 47ms de señal. Para cada clase se destinaron 30 vectores o superpatrones a ser adaptados, los cuales se inician con segmentos elegidos aleatoriamente de la señal de entrenamiento. La elección de los superpatrones ganadores para cada sección de 47ms fue bajo:

$$\min\{\|x - m_k\|\}$$

Posteriormente la elección de la clase definitiva por textura o "palabra" en el reconocimiento fue por mayoría, i.e. la clase que más estuvo presente considerando todos los segmentos de 47ms de la señal de entrada.

Se usaron conjuntos disjuntos de entrenamiento y de prueba. Se entrenó al sistema únicamente con una de las cinco orientaciones, la de círculo (ver esta sección al principio), y se probó al sistema con las demás orientaciones que el sistema no conoció durante el entrenamiento.

2.6 Resultados

En la tabla 2 observamos los resultados obtenidos en la clasificación de los diferentes materiales empleando las técnicas desarrolladas. Los resultados obtenidos son bastante prometedores, la mayoría de las texturas fueron clasificadas correctamente en su totalidad (93% de efectividad total). La clase #18 (lija) es la clase que más dificultades presenta, sin embargo es confundida con la clase #9 (Neopreno) y la #6 (Vinil texturizado), que poseen como la lija, comparable frecuencia espacial; las clases #15 (PVC) y la

#16 (mezclilla) poseen un solo error de clasificación. Cabe destacar que se entrenó una sola vez al sistema, y que se probó con orientaciones de muestreo que el sistema nunca conoció con antelación.

Por otro lado se puede decir que en general la forma de desplazamiento del sensor no afecta en el desempeño del sistema como se vio en la gran mayoría de las clases, inclusive en aquellas que tienen una textura orientada como la madera sin barnizar o el cartón corrugado; así mismo, en este trabajo quisimos presentar nuestros resultados de reconocimiento directo, es decir identificar al material simplemente por su textura y pretendiendo que ésta es "única" para cada material, lo anterior no es siempre cierto y bien se puede experimentar con agrupaciones de texturas más que con superpatrones individuales o bien ensayar otras estrategias como se detalla en las limitaciones y posibles soluciones descritas más adelante.

La base de datos de texturas, está disponible para fines de investigación en nuestra dirección de ftp [18]².

Clase de pertenencia	Clase asignada para la orientación de prueba:			
	Horizontal	Vertical	Diagonal positiva	Diagonal negativa
15	15	16	15	15
16	16	12	16	16
18	18	9	9	6
Las restantes	Clasificadas correctamente al 100%			

Tabla 2. Clasificación de los estímulos con diferente orientación para las diferentes clases (ver texto y tabla 1).

2.7 Discusión

Algunas de las principales características actuales del sistema y mejoras futuras son:

- Es un sistema de buen desempeño (ver tabla 2) para el reconocimiento táctil de texturas.
- Su implementación es sencilla y el equipo requerido para su implementación es de fácil acceso
- Al estar basada la clasificación en reconocimiento de voz, se pueden incorporar muchas técnicas, métodos y dispositivos existentes.
- No vemos impedimentos graves para trabajar en condiciones como debajo del agua (usando un hidrófono) o en condiciones de baja atmósfera (tomando las debidas precauciones de temperatura), debido a que el sistema es de contacto.
- El sistema puede ser complementado fácilmente con otras magnitudes físicas como la temperatura, mismas que ayudarían mucho a discernir de manera más fina sobre los objetos.
- Algunas de las principales limitaciones de la implementación que presentamos en este trabajo y algunas de sus posibles soluciones son las siguientes:

² Si es que no hay bloqueo.

- Si existe un ruido exterior intenso en el área de trabajo éste puede afectar el resultado de la clasificación; esto puede corregirse con técnicas sencillas de cancelación de ruido, básicamente se tendrían dos micrófonos: el sensor táctil y el que percibe el ruido externo.
- Para poder trabajar con una textura el sensor tiene que moverse. Esto parece ser una característica incluso en los equivalentes biológicos, en donde parece que es muy difícil discernir texturas con exploración estática; combinando lo que se planteó sobre cancelación de ruido y un sistema mecánico adecuado, creemos que se puede sustituir el sensado manual fácilmente y actualmente estamos trabajando en ello.
- Si el material a sensar vibra de manera no-monótona, podría afectarse la clasificación.ésto puede remediarse con cancelación de un sensor auxiliar estático sobre la textura; sin embargo en ambientes naturales, éste tipo de alteración es poco frecuente.
- El sensor tiene un tamaño considerable para ciertos escenarios (1cm³ aprox.). No vemos en esto un gran impedimento puesto que los piezoeléctricos pueden realizarse en alta escala de integración, además de que en su estado actual el sensor es una buena opción que ha demostrado buen desempeño (ver tabla 2) y es una alternativa a los sistemas de sensado matriciales.

Los trabajos en percepción táctil artificial en general y para el reconocimiento dinámico de texturas en particular son muy escasos en la literatura. En este trabajo presentamos una primera aproximación hacia el reconocimiento automático de texturas por métodos táctiles, empleando técnicas de procesamiento de señales sonoras. La percepción táctil es muy importante en seres biológicos y podría robustecer ampliamente las representaciones de sistemas artificiales del entorno donde éste es accesible táctilmente o bien, en ciertos casos, como sustituto de esquemas de percepción tradicional como la visión artificial, ya que los sistemas táctiles no son confundidos por sombras o decoloraciones en los objetos que pueden afectar mucho el desempeño de los sistemas visuales.

Referencias

- [1] F. Bustamante, Sistema tegumentario. Edit. Limusa. 1993.
- [2] H. Nicholls & M. Lee. Survey of robot tactile sensing technology. Int. Journal of Robotics Research, Vol. 8, No. 3, pp. 3-30. 1989.
- [3] R. Grupen et. al. Survey of general purpose manipulation. Int. Journal of Robotics Research. Vol. 8. No. 1. pp. 38-62. 1989.
- [4] R. Fearing. Tactile sensing mechanisms. Int. Journal of Robotics Research. Vol. 9. No. 3. June 1990.
- [5] B. Gray & R. Fearing. A surface micromachined microtactile sensor array. preprint. Dept. ECE, UC Davis 1996.
- [6] T. Kohonen, The "neural" phonetic typewriter. Computer, Vol 21, pp 11-22, March. 1988.
- [7] K. Torrkola, et. al., Status report of the finnish phonetic typewriter project. Procc. ICANN91, Espoo Finland, 24-28 June, 1991.
- [8] LVQ_Pak The Learning Vector Quantization Program Package, LVQ Programming Team. Helsinki U. of Technology. Lab. Comp. and Inf. Sci. 1992. Obtenido en: <http://www.hut.fi>
- [9] R. Howe & M.R. Cutkosky. Dynamic tactile sensing: perception of fine surface features with stress rate sensing. IEEE Trans. on Robotics and Automation. Vol 9, No. 2. April. 1993.
- [10] T. Maeno, K. Kobayashi & N. Yamazaki, Relationship between the structure of human tissue and the location of tactile receptors. Trans. Jpn. Soc. Mech. Eng., Vol. 63, No. 607, pp. 881-888.1997. (ver[23])

- [11] T. Maeno, K. Kobayashi & N. Yamazaki, Effect of geometry and property of finger tissue on stress/strain distribution near the tactile receptors. 10th Conference of the ESB. p. 312. 1996. (ver[23])
- [12] T. Maeno, K. Kobayashi & N. Yamazaki. Sensing Mechanisms of the partial incipient slip at the surface of cylindrical fingers during the precision grip. Bioengineering Conference ASME. BED-Vol. 35. pp. 117-118. 1997. (ver[23])
- [13] M. Tremblay & M. Cutkosky, Estimating friction using incipient slip sensing during a manipulation task. 1993 IEEE ICAR, May 2-6, Atlanta, GA. pp 363-368. 1993. (ver[19])
- [14] M. Cutkosky & J. Hyde, Manipulation control with dynamic tactile sensing. 6th ISRR, Oct. 2-5, USA. 1993. (ver[19])
- [15] Okamura, M. Turner & M. Cutkosky, Haptic exploration of objects with rolling and sliding. Proc. 1997 IEEE ICRA. Vol. 3, Apr. 20-25, Albuquerque USA. pp 2485-2490. 1997. (ver[19])
- [16] R. Fearing, G. Moy & E. Tan. Some basic issues in teletaction. IEEE ICRA97, Albuquerque NM. pp.3093-3100. 1997.
- [17] M. Freund & N. Lewis, A chemically diverse conducting polymer-based "electronic nose". Proc. Natl. Acad. Sci. USA. Vol 92, pp. 2652-2656. March 1995. Chemistry. 1995.
- [18] <http://132.248.59.55>
- [19] <http://cdr.stanford.edu/Touch/Touchpage.html>
- [20] <http://www.ai.mit.edu/projects/handar-haptics/haptics.html>
- [21] <http://touchlab.mit.edu/>
- [22] <http://hrl.harvard.edu/proyectos/squishy/finger.html>
- [23] <http://www.yamazaki.mech.keio.ac.jp/index2EN.html>
- [24] W.W. Mayol, J. Juárez & S. Muñoz. A first Approach to tactile texture recognition. IEEE Int. Conf. On Systems Man and Cybernetics SMC'98. La Jolla. USA. Octubre 11-14. 1998.

FALTA PAGINA

No. 17

III Reconocimiento de Rostros Empleando Coeficientes de Autocorrelación Local¹

3.1 Planteamiento del Problema

El reconocimiento de rostros desde el surgimiento de los primeros sistemas de visión por computadora, ha sido una tarea de interés para las diferentes áreas involucradas: procesamiento de señales, algoritmos de reconocimiento, coordinación visuo-motora y circuitería especializada.

Lo anterior obedece al deseo de que las máquinas puedan resolver varias tareas que los humanos realizamos basados en información facial; tareas que entre otras incluyen aspectos de seguridad y de descripción de los individuos (en base a su sexo, edad aparente, emociones, etc.).

En el caso particular del reconocimiento automático de rostros (RAR), el problema general podría ser enunciado de la siguiente manera: *Desarrollar un conjunto de sistemas que de la manera más eficiente posible sea capaz de identificar a un individuo conocido, en el mayor número de escenarios y condiciones posibles.*

Las dificultades presentes en el RAR incluyen: cambio y no uniformidad en la iluminación, cambios en estilo y color de cabello y/o barba, cambios en aditamentos del rostro (lentes, aretes, etc.), entre otros; además de que están presentes varios de los problemas convencionales de la visión por computadora: movimientos, oclusiones, rotaciones, escalas y demás.

Además de las dificultades inherentes del RAR, parece no existir en la literatura un consenso claro en cuanto al tipo de técnicas o métodos a emplear y son pocos los trabajos comparativos al respecto. Recientemente Brunelli y Poggio [1] comparan dos aproximaciones frecuentes en RAR: la aproximación basada en características geométricas que consiste en obtener información de distancias y ángulos entre características presumiblemente distintivas como: ojos, nariz y boca, y por otro lado, comparan la aproximación basada en plantillas, que consiste en considerar a la información extraída como un todo, en donde la plantilla más sencilla es la imagen completa del rostro; sus resultados indican un mejor desempeño de la aproximación de plantillas y actualmente existen varios RARs basados en este enfoque general [2,3,4], el enfoque del sistema aquí presentado cae dentro de la aproximación de plantillas.

En el presente capítulo estaremos interesados principalmente en las siguientes cuestiones:

- Desarrollar técnicas para un sistema general de reconocimiento de imágenes, que entre otras tareas pueda ser aplicado a rostros.
- Probar estas técnicas en ambientes estáticos (e.g. rostro fijo y fondo uniforme).
- Incorporar métodos rápidos de procesamiento y clasificación para facilitar situar el desempeño completo del sistema cerca del tiempo real.

Es así que la presentación restante del capítulo, la hemos dividido en dos secciones principales, la primera concerniente a la descripción de las técnicas de preprocesamiento y clasificación que empleamos para los escenarios estáticos y la segunda que se centra en la descripción y prueba de

¹Parte de esta investigación fue publicada en el II Taller Iberoamericano de Reconocimiento de Patrones La Habana Cuba Marzo 1997

algunos procedimientos previos al reconocimiento en escenarios no fijos. Finalmente presentamos una discusión de los resultados obtenidos.



Fig. 1. Nueve de las 500 imágenes empleadas en el presente trabajo para reconocimiento en ambientes estáticos.

3.2 Metodología Escenario Estático

Como situación de escenario estático consideraremos tomas fijas y frontales sobre un fondo de color uniforme de 50 sujetos¹: 29 hombres y 21 mujeres, que cuentan con cambios en posición y ligeros en escala, así como cuentan con diferentes gestos (ver figuras 1y2). Las imágenes se escalaron a un tamaño promedio de 100x90 pixeles y cuentan con una definición de 8 bits por pixel.

De cada uno de los sujetos se capturaron 10 tomas, lo que da una base de 500 imágenes empleadas.

Dentro de las diferentes pruebas siempre se usaron conjuntos distintos de entrenamiento y reconocimiento.

3.3 Coeficientes de Autocorrelación Local

Existen muchas técnicas para caracterizar imágenes como: momentos, FFT, wavelets, entre otras. Recientemente Otsu & Kurita [5] proponen una técnica de caracterización de imágenes conocida como coeficientes de autocorrelación local (CAL), que es de cálculo rápido e invariante a corrimientos, también reduce la cantidad de información a ser procesada. Esta técnica ha sido utilizada por Goudail et al. [2] en un sistema de RAR en ambiente estático, obteniendo un buen desempeño, sin embargo su sistema requiere de un número grande de entrenamientos y su escenario esta dispuesto de tal forma que hay un alto contraste entre la ropa, el fondo y el rostro. La técnica CAL es el método de preproceso empleado en el presente trabajo.

Se utilizaron las máscaras de la figura 3 para el cálculo de los veinticinco coeficientes de autocorrelación local. Cada una de estas mascaras es colocada sobre todos los puntos posibles de la imagen, en cada posición el producto de los elementos que se encuentran en oscuro se calcula, y todos los productos pertenecientes a una misma máscara se suman para obtener el coeficiente asociado a esa máscara. Formándose así el vector que caracteriza a la imagen. El tamaño de las

¹Se emplearon 40 individuos de la base de rostros de la Universidad de ESSEX (<http://cswww.essex.ac.uk>), los restantes fueron voluntarios de nuestra Universidad capturados en situaciones muy similares a la de los otros 40, esta base de datos puede ser obtenida para fines de investigación en nuestra dirección de <ftp://132.248.59.55>

máscaras puede ser cambiado aplicando una operación de dilatación como se muestra en la figura 4.



Fig. 2. Ejemplos de las distorsiones en posición y gestos que contienen las clases utilizadas.

Una vez obtenidos los vectores CAL se calcula el vector diferencias DCAL como $DCAL_i = CAL_{i+1} - CAL_i$. Con el uso de los vectores obtenidos con este sencillo procedimiento se logró un mejor reconocimiento que el obtenido con los vectores CAL. En la figura 5 se muestran ejemplos de dos sujetos y los diez vectores DCAL que conforman su clase.

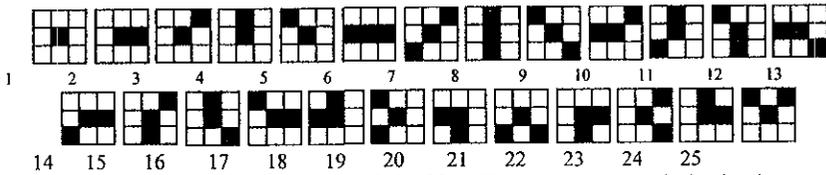


Fig. 3 Máscaras para el cálculo de los 25 coeficientes de autocorrelación local.

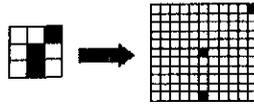


Fig. 4. Ejemplo de dilatación de una máscara de 3x3 a una de 11x11 como las empleadas en este trabajo.

3.4 Método de Clasificación

El método de clasificación seleccionado es la red neuronal de regresión general (GRNN) [6], esta red tiene varias ventajas que incluyen: aprendizaje no iterativo, prácticamente no tiene restricciones en cuanto al tipo de funciones que puede trabajar y tiene garantía de convergencia conforme el conjunto de entrenamiento crece; la GRNN ha sido probada por uno de los autores en otro sistema de reconocimiento de imágenes, comprobando sus características útiles para visión por computadora [7]. Un trabajo comparativo entre GRNN y la red backpropagation en una tarea de reconocimiento de imágenes sintéticas en 3D puede verse en [8], para una descripción detallada de la red, se sugieren [6 y 9].

Como primer paso para emplear la GRNN se establece un conjunto de entrenamiento formado por los estímulos y sus respuestas deseadas, en este caso formado por los vectores DCAL de un sujeto y su correspondiente etiqueta identificadora.

La ecuación de operación de la red GRNN es:

$$R_i = \frac{\sum_{j=1}^n h_j w_{ij}}{\sum_{j=1}^n h_j}$$

donde R_j es la respuesta en el nodo de salida j . W_{ij} es la respuesta deseada correspondiente al vector de entrenamiento U_i (vectores CAL) y la salida j .

$$h_i = e^{-\frac{(x - u_i)^T (x - u_i)}{\alpha^2 \sigma_i^2}}$$

h_i es la salida de un nodo "oculto". σ es el vector de coeficientes de desviación estándar de los patrones de entrenamiento U_i , $\alpha=0.1$, estos coeficientes están directamente relacionados con las habilidades de generalización de la red.

3.5 Resultados de Clasificación en Ambiente Estático

Utilizando la red GRNN y los vectores CAL con máscaras de 11×11 , se obtuvo un reconocimiento de 65.71%, empleando 3 entrenamientos y probando el sistema con las 7 imágenes restantes de cada uno de los 50 sujetos (50×7 patrones de prueba). Sin embargo, utilizando los vectores DCAL sugeridos en este artículo se obtuvieron mejores resultados, incluso para un solo entrenamiento, como se detalla en la tabla 1. Obsérvese que 3 entrenamientos por clase son suficientes para obtener un buen índice de desempeño.

# Entrenamientos por clase	# Patrones de Prueba por clase (Distintos a los de entrenamiento)	Porcentaje de Reconocimiento
1	9	94.33
2	8	95.51
3	7	98.38
4	6	98.42

Tabla 1. Desempeño del sistema de reconocimiento para diferente número de entrenamientos y con máscaras CAL de tamaño 11×11 , al operar con 50 personas o clases distintas. Nótese que se usaron conjuntos distintos de entrenamiento y reconocimiento.

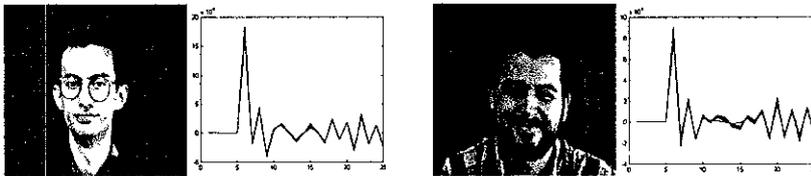


Fig. 5. Los 10 vectores DCAL de dos de los sujetos de la base de datos.

Referencias

- [1] R. Brunelli and T. Poggio, "Face Recognition: Features versus Templates", *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 15, no. 10, pp. 1042-1052, Oct. 1993.
- [2] F. Goudaill, E. Lange, et al., "Face Recognition System Using Local Autocorrelations and Multiscale Integration", *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 18, no. 10, Oct. 1996.
- [3] L. Wiskott, J. Fellous, et al. "Face Recognition and Gender Determination", *Proc. Intern. Workshop on Automatic Face and Gesture Recognition*, Zurich, Suiza, pp. 92-97, 1995.
- [4] L. Wiskott, "Phantom Faces for Face Analysis", accepted for publication in *Pattern Recognition*, 1996. <http://www.neuroinformatik.ruhr-uni-bochum.de>
- [5] N. Otsu and T. Kurita, "A New Scheme for Practical, Flexible and Intelligent Vision Systems", *Proc. IAPR Workshop on Computer Vision-Special Hardware and Industrial Applications*, pp. 431-435, Tokyo, 1988.
- [6] D.F. Specht, "A General Regression Neural Network", *IEEE Trans. Neural Networks*, vol. 2, no. 6, pp. 568-567, 1991.
- [7] R. García and W.W. Mayol, "Reconocimiento de Imágenes Empleando Redes de Regresión General y la Técnica TVS", *Proc. V Congreso Iberoamericano de Inteligencia Artificial*, Cholula, México, pp. 508-515, Nov. 1996.
- [8] P. Rayón, "Reconocimiento de Objetos Poliédricos con Redes Neuronales", *Tesis de Maestría*, CINVESTAV-IPN, 1995.
- [9] P.D. Wasserman, *Advanced Methods in Neural Computing*, VNR Publishers, New York, 1993.
- [10] S. Muñoz & W. W. Mayol. Una aproximación hacia el reconocimiento automático de rostros en varios escenarios. II Taller Iberoamericano de Reconocimiento de Patrones. La Habana Cuba Marzo 24-28. 1997.

IV Red de Kanerva en 2D para Computación Óptica¹

4.1 Introducción

Existen una gran cantidad de algoritmos de Redes Neuronales Artificiales (RNA) así como diferentes clasificaciones de éstos. Una de estas clasificaciones corresponde a las redes denominadas "sin pesos" o de "super patrones" [1,6,7,9,10,11] como: La red de Hamming, El mapa de características de Kohonen (LVQ, Learning Vector Quantizer), red de Aleksander y la Memoria Esparcida Distribuida de Kanerva (MED)². Esta última red permite de una manera más sencilla la construcción de memorias asociativas y la reconstrucción de señales.

En este trabajo se presenta la construcción de una red de 64 nodos (8 x 8) sobre una computadora óptica utilizando como componentes: un arreglo matricial de emisores de luz, moduladores espaciales y una cámara para retroalimentar la información resultante. El algoritmo es una modificación de MED para el reconocimiento de patrones bidimensionales.

4.2 Red de Kanerva

La estructura de la Red de Kanerva (RK) tiene ciertas similitudes con las memorias de acceso aleatorio³. En la red de Kanerva (Fig. 1) se tienen dos partes: la primera es un conjunto de direcciones binarias generadas aleatoriamente (DBA) que apuntan a la segunda parte que corresponde a la memoria de la red. El número de direcciones es igual o menor a 2^n donde n es el tamaño de los vectores de entrada.

El procedimiento de extracción de características utilizado es el siguiente: las localidades de memoria de la red se inicializan en cero, y con una *dirección de referencia o entrada*, se obtiene la distancia de hamming con todas las direcciones de la primera parte. Aquellas direcciones que sean menores a un umbral llamado *radio* son seleccionadas para colocar el patrón de entrada en la dirección correspondiente. La forma de escribir el patrón de entrada sobre la memoria depende de los valores de la entrada asociada, es decir, si es 1 el bit correspondiente se aumenta en 1, y si es cero se disminuye en 1. (fig. 1). Después se suman las localidades de memoria y se considera un umbral de 0, si es mayor a este valor se escribe un 1 y en el otro caso un 0.

En la etapa de reconocimiento el patrón a ser clasificado se utiliza como una dirección de referencia y se sigue el procedimiento mencionado anteriormente. Como puede observarse la entrada y la entrada asociada pueden ser diferentes de tal forma que la RK funciona como una memoria asociativa

Como se puede observar el radio activa zonas similares para un mismo patrón, por esta razón es que se dice que la información se *esparce* en la red, es decir, el patrón se distribuye en diferentes zonas de la matriz de direcciones. Resultados relacionados al comportamiento de la matriz aleatoria y algunos resultados sobre el número de matrices que pueden ser almacenados en un espacio se pueden encontrar en [13]

¹ Experimento hecho en colaboración con Eduardo Gómez-Ramírez & Victor Ramos-Viterbo (ULSA), Publicado en el II Taller Iberoamericano de Reconocimiento de Patrones. La Habana, Cuba, Marzo 24-28. 1997.

² SDM, sparse distributed memory

³ RAM, random access memory

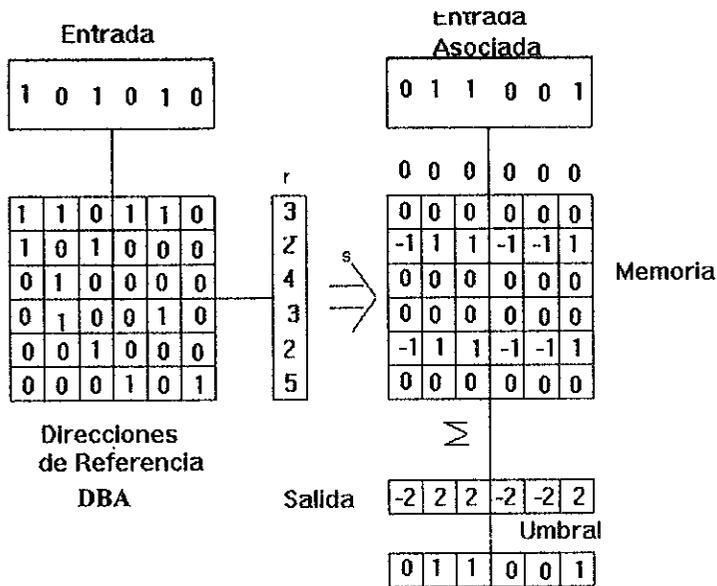


Figura 1. Memoria Distribuida Esparcida de Kanerva

4.3 Computación Óptica

A diferencia de la tecnología tradicional de computadoras donde el proceso se hace manipulando electrones, en una computadora óptica se utilizan fotones. El principal motivo es utilizar las características naturales de la luz: su velocidad y naturaleza paralela.

Una de las principales arquitecturas utilizadas está formada de tres partes [12]:

- 1) Fuente emisora de luz, en donde se despliega la imagen a ser transformada
- 2) El *modelo* se despliega en un modulador espacial de luz (SLM: Spacial Light Modulator). En esta parte se puede utilizar un acetato o un pantalla de cristal líquido.
- 3) El resultado se visualiza en una pantalla. Esta pantalla puede ser un vidrio traslúcido de tal forma que este resultado pueda ser capturado por una cámara de video (CCD), para su retroalimentación ya sea en la entrada o en el modulador. Esta arquitectura puede verse en la fig. 2.

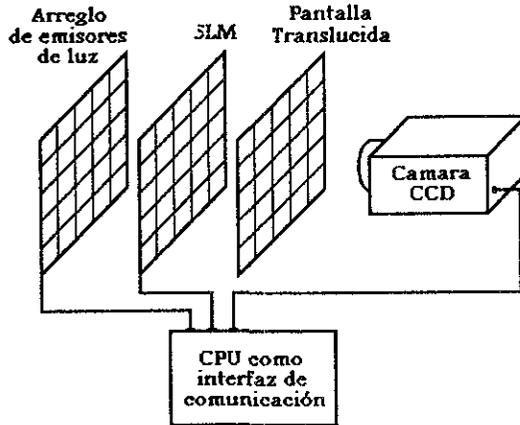


Fig. 2 Arquitectura Básica de una Computadora Óptica

La multiplicación de las matrices por la entrada se obtiene utilizando la técnica de "Shadow Casting" [12], y los resultados de la transformación son puntos en intensidades de luz. Este resultado es capturado con una cámara de video y retroalimentado utilizando una PC. En la siguiente tabla se puede observar un ejemplo para demostrar el funcionamiento de la técnica de Shadow Casting. En la primera columna se representan los emisores de luz que se encuentran prendidos, en la siguiente columna los traslapes de las sombras como resultado de la simulación en computadora (PC) y en la última columna el resultado en la computadora óptica.

4.4 Red de Kanerva Modificada en 2D

En este trabajo el procedimiento de extracción de características (DBA o primera parte de la RK) se hace en forma bidimensional, es decir, la forma en que se manipula la información es en forma matricial. Esto tiene varias ventajas, la principal es que se mantiene la naturaleza bidimensional de las entradas cuando se utiliza para imágenes o datos que tienen esta forma. Otra de las ventajas es que se reduce de forma significativa el espacio necesario para el almacenamiento ya que con solo desplazarse una localidad hacia los lados, la matriz considerada es completamente distinta.

El número de matrices aleatorias bidimensionales "C" de $m \times n$ que pueden ser colocadas en espacio de $i \times j$ es igual a:

$$C = [i - (m - 1)] * [j - (n - 1)] \quad (1)$$

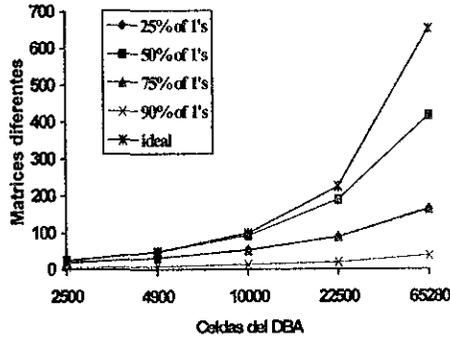
El método utilizado para multiplicar el patrón de entrada por la matriz de direcciones es el de distancia de Hamming [11]. Para las matrices aleatorias de la primera parte se generan aleatoriamente llenando con 1's y 0's. Es importante destacar que las matrices aleatorias comparten el espacio de la matriz. Por ejemplo, si el patrón de entrada está formado tiene dimensiones de 8×8 y la matriz de direcciones es de 50×50 , el número de matrices aleatorias contenidas sería igual a 1849 (utilizando la ec. 1), almacenadas en 2500 celdas o bits. Si el patrón de entrada fuera transformado en un vector se utilizarían 118 336 celdas para almacenar 1849 vectores aleatorios (64 entradas por 1849 matrices aleatorias). Como se puede observar existe una reducción significativa del espacio en el procedimiento de extracción de matrices y la ventaja de mantener la naturaleza bidimensional de las entradas.

Se hicieron dos pruebas experimentales para conocer el comportamiento de esta matriz aleatoria DBA.

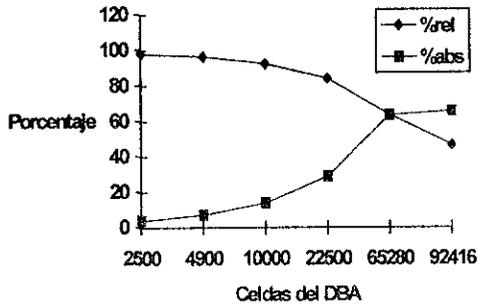
4.4.1 Prueba de llenado

La primera prueba de esa matriz bidimensional, creada con elementos binarios dispuestos aleatoriamente, consiste en un análisis del porcentaje óptimo de llenado tal que se obtenga el menor número de matrices redundantes. Los resultados se observan en la gráfica 1.

Las pruebas experimentales revelan que el llenado es óptimo cuando hay una probabilidad por celda de 50% de 1 o 0 como se esperaba.



Graf. 1. Matrices no redundantes (en cientos) contra número de celdas en matriz aleatoria DBA.



Graf. 2. Porcentaje relativo (número de matrices diferentes) vs Porcentaje absoluto (número de matrices del espacio) que poseen diferentes tamaños de matriz aleatoria DBA. Las pruebas se realizaron con entradas de 4X4.

4.4.2 Cantidad de Matrices del Espacio que se Cubren

En la segunda prueba (Gráfica 2) se indaga el número de matrices de 4 X 4 no redundantes que contienen diferentes tamaños de DBA. Así mismo se relaciona este número con el porcentaje del espacio vectorial que se cubre. Los resultados indican claramente (ver figura 4) que para un llenado aleatorio con 50% de

unos existe un número de celdas óptimo α , donde $\alpha=2^{m \times n}$ -siendo $m \times n$ el tamaño de la matriz a procesar.

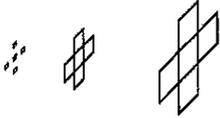
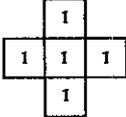
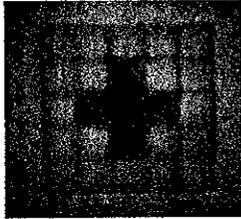
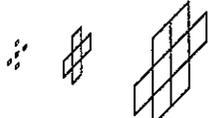
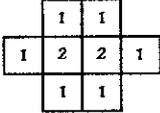
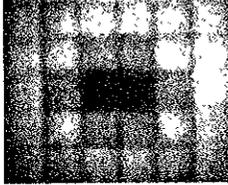
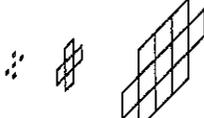
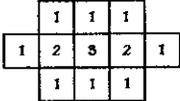
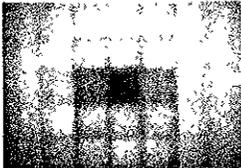
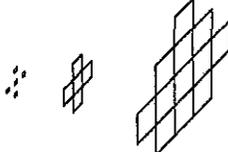
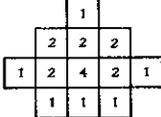
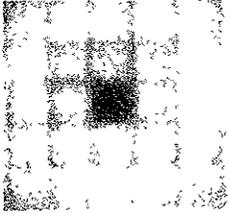
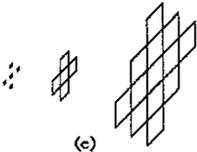
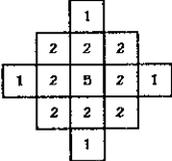
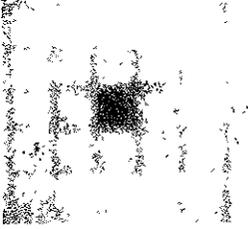
Nótese que estos resultados indican que para el punto analizado, cada elemento de la DBA puede "contener" o servir para representar 0.63 matrices diferentes de $m \times n$.

Dada la naturaleza de memoria esparcida del modelo y basados en los trabajos de Kanerva sobre las propiedades de espacios multidimensionales, no es necesario que DBA contenga todas las matrices del espacio para lograr un buen desempeño.

En la figura 3 se puede observar el esquema de la arquitectura utilizada para la implementación óptica. La cámara marcada como CCD1 recibe el resultado del primer estado entre la entrada de LEDs y la matriz aleatoria de direcciones, y después de aplicar el valor del radio se envía a la pantalla al segundo modulador marcado con el número 4.

Un punto en la matriz correspondiente a los valores activados en la etapa previa se utiliza para que por medio de la entrada asociada (L2) se coloque el patrón correspondiente. Como se puede observar en la RK la etapa de comparación con el conjunto de direcciones aleatorias si las dimensiones de los vectores son muy grandes, puede llegar a consumir una gran cantidad de tiempo de procesador, pero si se realiza mediante el procedimiento descrito anteriormente se desarrollaría a velocidad de la luz.

La fig. 4.1 muestra una matriz aleatoria de direcciones de 20×25 y una "T" como patrón de entrada. En las gráficas siguientes se muestra los resultados con diferentes umbrales y en la figura 4.2 se muestra el resultado considerando que la entrada asociada es igual que la entrada.

<i>Estado de la CO</i>	<i>Salida Deseada</i>	<i>Salida Experimental</i>
 <p>(a)</p>		
 <p>(b)</p>		
 <p>(c)</p>		
 <p>(d)</p>		
 <p>(e)</p>		

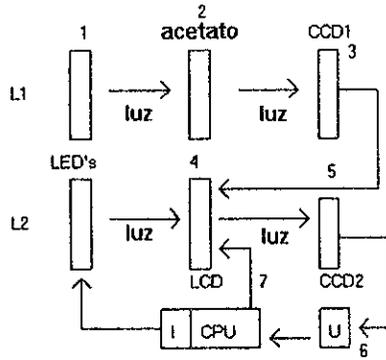


Figura 3. Diagrama de la implementación óptica

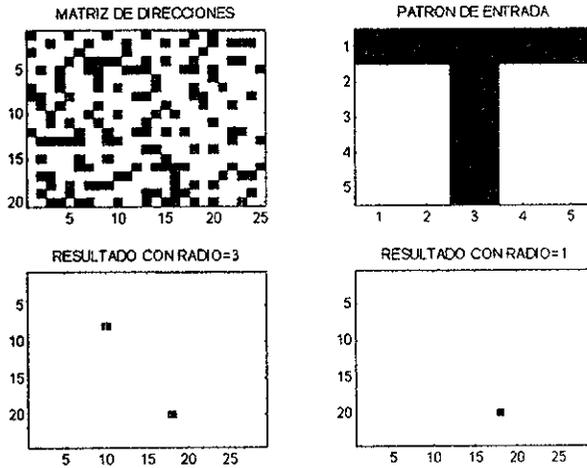


Fig. 4.1 Simulación de la Red de Kanerva en la computadora óptica



Fig. 4.2 Resultado de la Red de Kanerva

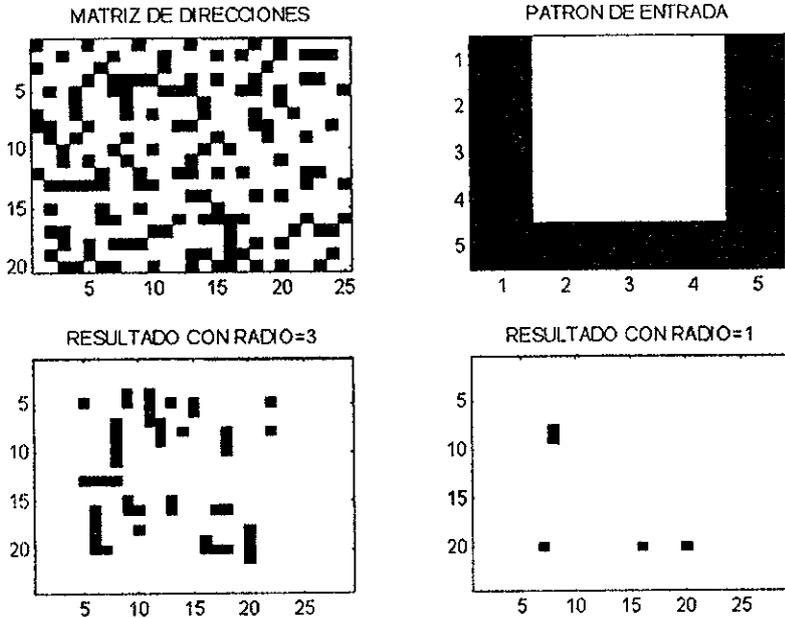


Fig. 5.1 Simulación de la Red de Kanerva en la computadora óptica

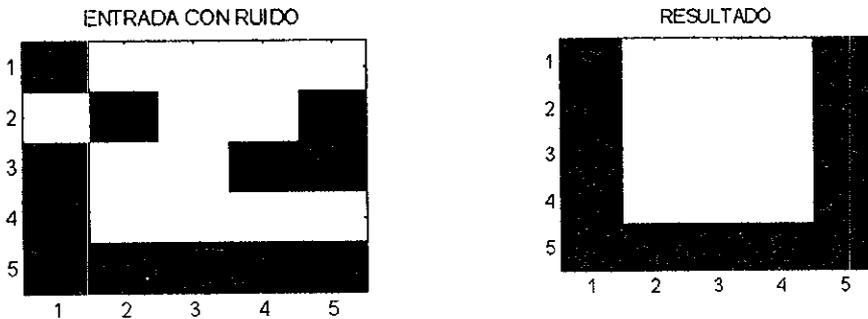


Fig. 4.2 Resultado de la Red de Kanerva para una entrada con ruido (simulación en PC)

4.5 Discusión y resultados

En este capítulo se presentó una modificación a la Red de Kanerva, transportando al sistema a un escenario de dos dimensiones y obteniendo así una reducción importante en cuanto al espacio de almacenamiento requerido para implementar la primera parte de la RK.

Por otro lado, se desglosan los puntos pertinentes para la implementación de este sistema en una computadora paralela óptica.

Otra ventaja en el planteamiento es que se mantiene la naturaleza bidimensional del patrón de entrada en el momento de procesarla.

Referencias

- [1] Lipman P. An introduction to computing with neural nets. IEEE ASSP Magazine. 1987.
- [2] Yu, F. Taiwei L. & Xiang Y. Optical neural network with pocket sized liquid crystal televisions. Optical Society of America. 1990.
- [3] Xiang Y. Taiwei L. & Yu, F. Compact optical neural network using cascaded liquid crystal television. Applied Optics. 1990.
- [4] Mayol W. & Gómez E. Técnicas ópticas paralelas para procesamiento en redes neuronales. Eight International Conference. UNAM CU. 1992.
- [5] Mayol W. Gómez E. 2D sparse distributed memory-optical neural network for pattern recognition. IEEE International Conference on Neural Networks ICNN94. Orlando, USA. June 28-July 2, 1994.
- [6] Rogers D. Statistical Prediction with Kanerva's Sparse Distributed Memory. RIACS Technical Report. 1989.
- [7] Rogers D. Weather Prediction Using a Genetic Memory. RIACS Technical Report. 90.6. 1990.
- [8] Gómez, E. & Mayol, W., Procesamiento de Imágenes utilizando Técnicas de Computación Óptica. XXXVI Congreso Nacional de Física. Acapulco, Gro., México, del 18 al 22 de octubre de 1993
- [9] Aleksander, I. An Introduction to neural computing, Chapman & Hall. 1990.
- [10] Kohonen T. The "neural" Phonetic Type Writer, IEEE. 1988.
- [11] Yu, F. & Saganda, J. Optical Signal Processing Computing and Neural Networks. Wiley 1992.
- [12] V. Ramos-Viterbo, E. Gómez-Ramírez, W. Mayol-Cuevas (LINDA-UNAM), V. Martínez-Martínez, F. García de León Ramírez, O. Balderas-Pereyra. Computadora óptica: fundamentos, arquitectura y algoritmos. XXXVIII Congreso Nacional de Física. Zacatecas, Zacatecas, Mexico del 16 al 20 de octubre de 1995.
- [13] Muñoz-Gutiérrez, S; Mayol-Cuevas, W.; Sánchez-Guzman, R., A Toroidal Random Structure as Base for an Associative Machine. Technical Report #R09101995. LINDA group. Div. Ing. Eléctrica. Facultad de Ingeniería UNAM. A.P: 70-500, C.P: 04510, México.

V Discusión

5.1 Redes de caracterización y asociación de información

Desde el primer capítulo, hemos mencionado que las redes que se presentan en las aplicaciones descritas en este trabajo, no fueron escogidas azarosamente para ser aplicadas a los problemas descritos, así como no es arbitrario presentarlas juntas en el mismo trabajo; de tal forma que en esta sección trataremos algunas de las propiedades que unen a estos paradigmas.

Existe desde hace algún tiempo [1-4], un conocimiento *superficial* de las propiedades generales de los diferentes modelos conexionistas, y existen algunas taxonomías de estos sistemas que se basan por ejemplo en cómo son sus tipos de entrada (binario/continuo) o como es su algoritmo de entrenamiento (supervisado/no-supervisado), o qué tipo de regla de aprendizaje emplean (iterativo/one-shot), entre otras características [3].

El hecho de establecer las diferencias importantes entre los sistemas conexionistas, lejos de ser trivial, puede aportar importante información del tipo de problemas que éstos pueden atacar con mejor o peor éxito, es así, que una taxonomía de RNA puede ayudar a saber qué requisitos o condiciones debe de cumplir un cierto problema para poder ser aplicado a una u otra red. La elección de grupos o categorías entre las RNA, resulta un trabajo no concluido y donde hay mucho que trabajar aún.

En este caso, se propone que una variable clasificadora importante de las RNA, es la de la manera en que estos sistemas de atacan el espacio solución. En la figura 1, podemos observar un espacio de entrada y uno de salida deseada; como lo hemos mencionado, la red, ejecuta v.g. la función $F: \mathcal{H}^n \rightarrow \mathcal{H}^m$ que mapea el espacio de entrada en el de salida.

Esta operación de mapeo, es realizada por los diversos tipos de sistemas conexionistas, sin embargo, no todos realizan esta operación de la misma manera

En el caso de las redes presentadas, es de particular interés para esta discusión, mencionar que éstas realizan su función mapeadora realizando dos operaciones principales que hemos denominado *caracterización* y *asociación*. Entendiendo por *caracterización*, una operación de discernimiento típicamente no-lineal, encargada de segmentar el espacio de entrada en categorías o *clusters* adecuadamente seleccionados, que después serán usados por la operación de *asociación* encargada de relacionar estos clusters con una cierta salida deseada por una función que resulta típicamente sencilla de definir (recordar v.g. el caso de GRNN donde la respuesta deseada simplemente se dispone a la salida de las funciones radiales)

El tipo de redes que pueden ser analizados bajo la perspectiva de caracterización y asociación son varias, entre ellas podemos encontrar además a las redes de Volterra [5], a las de regularización [6], a las de funciones sigmoide [7], a las redes de wavelets, entre otras.

Cabe mencionar que otros autores [1] agrupan a algunas estas redes dentro de las denominadas Generalized Single Layer Networks o GSLN, que cuentan con una sola capa de entrada de discriminación, sin embargo, la perspectiva de agrupar y asociar permite incluir y desarrollar sistemas de RNA con más de una capa de discriminación o con varias capas concatenadas de caracterización-asociación (CAAS).

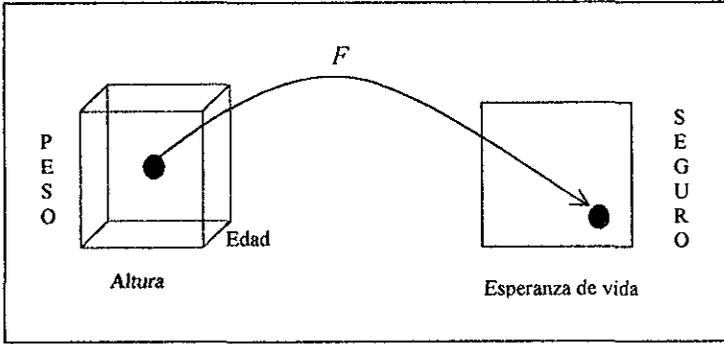


Figura 1. Ejemplo de función de mapeo realizado por una RNA.

Para el caso de redes CAAS con una etapa de discriminación, la red puede describirse más formalmente como sigue (figura 2).

Sea $x \in \mathcal{R}^n$ un vector de entrada, donde

$$x^T = [x_1 \ x_2 \ \dots \ x_n]$$

es mapeado dentro de un espacio mediante $F: \mathcal{R}^n \rightarrow \mathcal{R}^m$, $F: x \rightarrow x'$. Este mapeo se realiza mediante un conjunto de funciones ϕ típicamente no-lineales no necesariamente fijas (en GSLN si lo son) que son funciones base donde

$$\Phi = \{ \phi_i(x) \}_{i=1}^{m-1}$$

Con $\phi_i(x): \mathcal{R}^n \rightarrow \mathcal{R}^k$ (en GSLN la relación es aquí de $\mathcal{R}^n \rightarrow \mathcal{R}$) y $k \in \mathcal{R}$ con x' formada como:

$$x'^T = [1 \ \phi_1(x) \ \phi_2(x) \ \dots \ \phi_{m-1}(x)]$$

La salida y de la red se forma entonces como el producto interno de x' con un vector de pesos $w \in \mathcal{R}^m$

$$y = w^T x'$$

Para el caso de GRNN las $\phi(x)$ son iguales a $\phi(\|x - r_i\|)$ donde r_i son los centros de las radial basis functions y $\| \cdot \|$ es la norma Euclidiana, análisis similar es para el caso de LVQ donde las funciones no lineales tienen ahora como centro a los vectores de cuantización. En el caso de Kanerva, tenemos que los centros de las funciones son los vectores binarios y que sirven para ejecutar una función que va de $\alpha \rightarrow \mathcal{N}$ con $\alpha \in 2^n$ empleando la distancia de Hamming

Es nuestra intención en un trabajo futuro, realizar aproximaciones teóricas que nos permitan comparar la efectividad de estas redes de CAAS con respecto a v.g. las redes de tipo feedforward como backpropagation, sin embargo en este trabajo nos concretamos a presentar tres ejemplos de aplicación de estos sistemas.

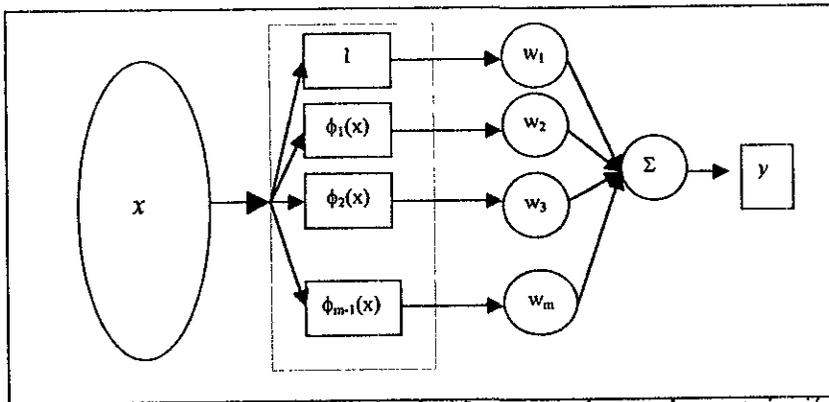


Figura 2. Esquema básico de una red del tipo CAAS para una sola etapa de caracterización.

Holden y Rayner [1], presentan una demostración de que redes GSLN como GRNN y la red de Kanerva, requieren menos entrenamientos que una red de tipo feedforward con unidades de umbral lineal para igual número de pesos. Este resultado apoya lo que en la práctica se ha visto para las redes CAAS y las GSLN, sin embargo se requieren más estudios desde el punto de vista teórico.

Por otro lado, las redes de tipo CAAS, tienen otras propiedades que parecen interesantes y que no son fácilmente presentes en redes de tipo feedforward, entre estas propiedades se encuentra el hecho de que en muchos casos existe independencia entre las etapas de caracterización y la de asociación, de tal manera que pudieran separarse y combinarse con otras etapas distintas, es decir una vez que el espacio fue segmentado, la respuesta deseada (contenida en la etapa de asociación) podría cambiarse por otra aparentemente sin afectar el desempeño del sistema. Esto es especialmente evidente en GRNN o en LVQ pero no vemos dificultad en poder diseñar un procedimiento para Kanerva u otras GSLN.

Desde nuestra perspectiva, la formalización de las redes CAAS puede ser un paso interesante para contar con una herramienta hacia la construcción de una taxonomía de RNA que sea más útil que las actuales, al enfocarse a la manera en cómo operan internamente los paradigmas ahí contenidos, en vez de resaltar si las entradas son binarias o continuas como hacen algunas de las taxonomías actuales.

Por último, es importante remarcar que a pesar de que se han descrito interesantes aplicaciones y propiedades de las redes CAAS, desde que mencionamos la importancia de la taxonomía, aceptamos que estas no son el tipo de redes que sean adecuadas para todos los problemas, son solamente un grupo que puede tener ventajas para atacar algunos de éstos.

5.2 Resumen

En este trabajo se ha presentado un conjunto de aplicaciones en el área de reconocimiento de patrones empleando redes del tipo de caracterización y asociación de información o CAAS. Estas redes tienen como denominador común el preparar el espacio de entrada y construir funciones que se obtienen por medio de procedimientos sencillos para construir la respuesta deseada.

En la práctica, las redes CAAS han mostrado una gran ventaja para trabajar con problemas considerados difíciles actualmente como los tratados en las aplicaciones presentadas, que en este caso incluyeron

reconocimiento automático de rostros, de texturas táctiles y una implementación y modificación para arquitecturas paralelas del modelo de la red de Kanerva.

Se introduce el concepto de redes CAAS que pueden considerarse un caso general de las redes GSLN y queda como trabajo futuro un desarrollo teórico que profundice en este tipo de redes de tal manera que se pueda iniciar una taxonomía de RNA con estas ideas, que permita tener una guía más clara para seleccionar entre los diferentes tipos de RNA.

En este trabajo no se encuentra una descripción detallada de todos los minuciosos aspectos de las redes y herramienta que los apoya pues este no es una recopilación del trabajo de otros autores, al lector interesado en profundizar en dichos temas se le sugiere consultar la bibliografía incluida.

Referencias

- [1] S. Holden & P. Rayner. Generalization and PAC Learning: Some new results for the class of generalized Single-Layer Networks. *IEEE Trans. On Neural Networks*. Vol 6. No. 2 1995.
- [2] D. Bailey. How to develop Neural Network. *AI Expert*. June 1990.
- [3] R.P. Lippmann. An introduction to computing with neural nets. *IEEE ASSP Magazine*. April 1997.
- [4] P.D. Wasserman, *Advanced Methods in Neural Computing*, VNR Publishers, New York, 1993.
- [5] P.W. Rayner & M. Lynch. A new connectionist model based on a non-linear adaptive filter In Proc. IEEE Int. Conf. Acoust., Speech, Signal Proc., May 1989.
- [6] T. Poggio and F. Girosi. Networks for approximation and learning. Proc. IEEE. Vol78 pp 1481-1497. Sept. 1990.
- [7] D.E. Rumelhart, J. McClelland and the PDP Group. Parallel distributed processing. Explorations in the microstructure of cognition, Vol 1. Cambridge. MIT Press. 1986.