



INSTITUTO POLITÉCNICO NACIONAL



CENTRO DE INVESTIGACIÓN EN COMPUTACIÓN

**“RECONOCIMIENTO DE CARACTERES
MEDIANTE MÁQUINAS ASOCIATIVAS
POR MUESTREO ALEATORIO”**

Tesis que para obtener el grado de
Maestro en Ciencias de la Computación
presenta

ING. JIMÉNEZ HERNÁNDEZ HUGO

Director de Tesis
DR. JESÚS FIGUEROA NAZUNO

**México, D.F.
Noviembre, 2004**



INSTITUTO POLITECNICO NACIONAL
COORDINACION GENERAL DE POSGRADO E INVESTIGACION

ACTA DE REVISIÓN DE TESIS

En la Ciudad de México, D.F. siendo las 16:00 horas del día 21 del mes de Abril del 2004 se reunieron los miembros de la Comisión Revisora de Tesis designada por el Colegio de Profesores de Estudios de Posgrado e Investigación del:

Centro de Investigación en Computación

para examinar la tesis de grado titulada:

**"RECONOCIMIENTO DE CARACTERES CON MÁQUINAS
ASOCIATIVAS POR MUESTREO ALEATORIO"**

Presentada por el alumno:

JIMÉNEZ
Apellido paterno

HERNÁNDEZ
materno

HUGO
nombre(s)

Con registro: B

0	1	1	3	8	5
---	---	---	---	---	---

aspirante al grado de: **MAESTRO EN CIENCIAS DE LA COMPUTACIÓN**

Después de intercambiar opiniones los miembros de la Comisión manifestaron **SU APROBACION DE LA TESIS**, en virtud de que satisface los requisitos señalados por las disposiciones reglamentarias vigentes.

LA COMISIÓN REVISORA

Director de la tesis

Dr. Jesús Guillermo Figueras Nazari
Secretario

Presidente

Dr. Segundo Levachine Pavlovich
Segundo Vocal

Dr. Luis Pastor Sánchez Fernández
Tercer Vocal

Dr. Edgardo Manuel Felipe Riverón
Suplente

Dr. Cornelio Yáñez Márquez

M. en C. María Elena Aguilar Jauregui

EL PRESIDENTE DEL COLEGIO

Dr. Juan Luis Díaz de León Santiago



INSTITUTO POLITECNICO NACIONAL
CENTRO DE INVESTIGACION
EN COMPUTACION
DIRECCION

Dedicatoria

El trabajo y esfuerzo consumado en este documento se los dedico a las siguientes personas:

Mis padres (Manuel Francisco Jiménez Morales y Edith Hernández Lázaro), por el apoyo incondicional brindado en todo este tiempo, puesto que sin ellos no estaría hoy aquí.

Mi novia (Ana Marcela Herrera Navarro), por toda tu comprensión, todo tú apoyo, porque me has hecho seguir siempre adelante, siempre motivado.

Mis hermanos (Francisco Jiménez Hernández y Edith Jiménez Hernández), con ustedes siempre he contado, y siempre me han apoyado sin importar las situaciones.

Mi tía (Lizbeth Hernández Lázaro), por todo el apoyo incondicional que me has dado todos estos años.

Agradecimientos

Este documento que se presenta, no es el resultado individual, si no por el contrario, colectivo de varias personas a las que quiero agradecer el apoyo material, moral o afectivo, que me han brindado a lo largo de estos años.

A Ernesto Bautista Thompson, gracias a tus enseñanzas, comentarios y lo principal tu amistad.

Al Dr. Jesús Figueroa, que durante este tiempo, que más que una relación de alumno profesor, es una relación de amistad, que me ha acompañado y enseñado a lo largo de estos años.

A todos los miembros del Laboratorio de Sistemas Distribuidos y Paralelo (hoy cómputo de alto rendimiento) por todos aquellos detalles, pequeños o grandes que hay hecho que hoy presente este documento.

A aquellos compañeros del centro, que en su momento me han dado la mano, facilitándome el desarrollo de la tesis.

Al Centro de Investigación en Computación (CIC) por el apoyo de los profesores y académicos.

Al Instituto Politécnico Nacional por el fomento a la investigación.

A CONACYT, por el apoyo económico del cual he disfrutado, que ha permitido dedicarme al desarrollo de este trabajo.

Índice

ÍNDICE	I
RESUMEN	1
ABSTRACT	2
INTRODUCCIÓN	3
PLANTEAMIENTO DEL PROBLEMA	3
OBJETIVOS DE LA TESIS.....	4
DELIMITACIONES	4
HIPÓTESIS DE TRABAJO.....	5
ORGANIZACIÓN	5
1 ANTECEDENTES	7
1.1 RECONOCIMIENTO DE PATRONES	7
1.1.1 Reconocimiento de caracteres.....	8
1.2 MÉTODOS DE CLASIFICACIÓN.....	10
1.2.1 Modelos de clasificación basados en coincidencia.....	13
1.2.2 Clasificadores óptimos estadísticos	15
1.2.3 Redes neuronales artificiales	16
1.2.4 Memorias asociativas.....	18
1.2.4.1 Modelo Hopfield.....	20
1.2.4.2 Modelo ADAM.....	21
1.2.4.3 Modelo BAM	21
1.2.4.4 Modelo SDM	21
1.2.4.5 Modelo de cómputo asociativo ASC.....	23
1.3 DISCUSIÓN	25
2 EL MODELO DE MEMORIA DISTRIBUIDA DISPERSA DE KANERVA	27
2.1 PATRÓN Y SIMILITUD	27
2.1.1 La representación.....	27
2.1.2 Definición de patrón y similitud	29
2.2 EL ESPACIO $\{0,1\}^N$	30
2.2.1 Conceptos del espacio $\{0,1\}^n$	30
2.2.2 Propiedades del espacio $\{0,1\}^n$	34
2.3 CONCEPTOS DE MEMORIA DISPERSA.....	36
2.3.1 Elementos de memoria como puntos en el espacio $\{0,1\}^n$	36
2.3.2 Conceptos de memoria en $\{0,1\}^n$ y almacenamiento distribuido.....	36
2.4 DISCUSIÓN	38
3 MODELO DE MUESTREO ALEATORIO DE IMÁGENES	40
3.1 MUESTREO DE IMÁGENES.....	40
3.2 MUESTREO ALEATORIO.....	42
3.2.1 Representación	43
3.2.2 Codificación	44
3.2.3 Cálculo de patrones.....	47
3.2.4 Optimización del muestreo	50
3.3. DISCUSIÓN	59
4 MÉTODO PROPUESTO DE RECONOCIMIENTO DE CARACTERES	62
4.1 DESCRIPCIÓN DEL MÉTODO PROPUESTO.....	62
4.1.1 Cálculo de los patrones representativos	63

4.1.2 Fase de reconocimiento.....	67
4.1.3 Análisis de complejidad del algoritmo.....	69
4.2 PRUEBA Y ANÁLISIS DE RESULTADOS.....	70
4.2.1 Descripción del método experimental.....	70
4.2.1.1 Descripción de pruebas.....	70
4.2.1.2 Parametrización.....	71
4.2.2 Análisis de resultados.....	72
4.3 DISCUSIÓN.....	79
CONTRIBUCIONES.....	81
CONCLUSIONES.....	82
APÉNDICE. VARIANTES DE CARACTERES DE LETRAS.....	84
GLOSARIO DE TÉRMINOS.....	88
SIMBOLOGÍA.....	91
ÍNDICE DE FIGURAS.....	92
ÍNDICE DE TABLAS.....	94
ÍNDICE DE CÓDIGOS.....	95
BIBLIOGRAFÍA.....	96

Resumen

La tarea de reconocimiento de patrones no tiene una solución general. Un caso particular de la misma es el reconocimiento de caracteres. El conocimiento anticipado de los *descriptores* necesarios para *clasificar* e *identificar* a un grupo de variantes de caracteres, es una tarea abierta, donde el uso de los modelos de representación de información simbólicos permite dar soluciones diferentes a las de los modelos aritméticos clásicos.

En este trabajo se presenta un método para el reconocimiento de caracteres empleando los conceptos de *asociatividad* y *similitud en modelos simbólicos*. La información se representa como cadenas *simbólicas* de longitud definida. La codificación y extracción de información se realiza con un Método de Muestreo en imágenes, sobre un conjunto de diferentes tipos de caracteres, donde cada carácter se representa como una imagen de tamaño definido. El muestreo de imágenes selecciona puntos aleatorios de la imagen, donde la eficiencia del mismo se garantiza utilizando un método de optimización. A partir de la codificación de los diferentes tipos de caracteres, se encuentran *patrones simbólicos* característicos por cada carácter del alfabeto y se definen los criterios para *decidir* si un carácter es *similar* a un patrón característico previamente codificado.

El método de reconocimiento de caracteres se evalúa mediante método experimental, que mide su eficiencia con un conjunto de diferentes tipos de caracteres. El modelo experimental muestra cómo es que los diferentes parámetros del método afectan la eficiencia de reconocimiento.

Abstract

The task of pattern recognition has not general solution. A particular case is the character recognition. The expect knowledge of the needed *descriptors* for *classify* and *identify* to set of character variants is open task, where the use of symbolic information models give us different approaches to classic arithmetic models.

In this work present one method for the character recognition, using concepts like *associativity* and *similarity* in symbolics models. The information is represented as symbolic strings, where the length is defined. The encoding and the information extraction is made by a sampling image method, using a set of different characters, each character is represented as an image of defined size. The image sampling chooses randomly a set of dots from the image, after the significance of those sampling dots are warranted by an optimization method. From the encoding of set of different characters, looks for characteristic symbolic patterns by each letter of the alphabet and define the judgments for decide when a give us character is *similar* to one pattern previously encoded.

The character recognition method is evaluated by an experimental method, which quantify the efficiency to recognize a set of different characters. Experimentally the model shows how different models parameters affect the recognition efficiency.

Introducción

El *reconocimiento de patrones* es un problema que no tiene una solución general con los métodos existentes en computación [1]; las soluciones propuestas resultan ser particulares y específicas para cada tipo de problema y están limitadas por la forma de codificar y representar la información del problema que analizan [2, 3].

En este trabajo se presenta un método particular de reconocimiento de caracteres que emplea modelos simbólicos de representación y conceptos de memorias asociativas. La codificación de la información se realiza utilizando un muestreo aleatorio sobre imágenes, donde por medio de un mecanismo de optimización, se identifica la *mejor información* que ayuda a obtener mejores resultados de reconocimiento de caracteres. El mecanismo de optimización del muestreo consiste en la modificación de los puntos de muestreo, mediante el análisis de la información que ha sido codificada en forma de *patrones*.

El método se evalúa mediante un modelo experimental, que consiste en medir la eficiencia del reconocimiento empleando un conjunto de diferentes tipos de caracteres. En el modelo experimental se describe un conjunto de parámetros y cómo influyen estos en el reconocimiento.

Planteamiento del problema

El *reconocimiento de patrones* es el proceso de distinguir algo de acuerdo a ciertas características [1]; el *reconocimiento* por computadora no tiene una solución general, porque resulta difícil encontrar las *características adecuadas (descriptores)* y una forma de codificarlas, para poder discriminar de forma *eficiente* al problema; las herramientas y modelos empleados para el reconocimiento dependen del contexto de la aplicación. La capacidad de clasificación de un modelo depende de cómo se codifica la información y los operadores existentes sobre la información codificada. Los diferentes modelos de codificación y de representación pueden facilitar la tarea de reconocimiento [2, 3].

El reconocimiento de caracteres por computadora es un caso particular del reconocimiento de patrones (por computadora), y por tanto, tampoco existen métodos generales para el reconocimiento de los mismos. En este trabajo se plantea una metodología para el reconocimiento de caracteres que no define *a priori* los *descriptores*. El método se basa en la elección de una codificación simbólica y en un muestreo probabilístico.

La codificación simbólica junto con un Método de Muestreo, a diferencia de las técnicas existentes, no presupone los descriptores necesarios para representar cada carácter, sino que

emplea una representación simbólica a través de un mapeo en un espacio métrico en $\{0, 1\}^n$.

La representación simbólica se auxilia a través de un muestreo probabilístico, que debe ser lo suficientemente significativo como para recolectar la información asociada a cada carácter, de tal manera que sea posible construir cadenas que representen a cada una de las variantes de los caracteres. La elección del mejor conjunto de puntos de muestreo, que maximiza la eficiencia del reconocimiento, no es una tarea sencilla y es una de las tareas primordiales de este trabajo.

Objetivos de la tesis

El objetivo general del trabajo es:

- Presentar un método para el reconocimiento de caracteres.
- Evaluar la eficiencia del método utilizando un modelo experimental.

Los objetivos particulares que se persiguen son:

- Emplear un modelo de representación de información simbólico no aritmético para la construcción de cadenas de símbolos de longitud n , mapeados a vectores n dimensionales [3].
- Definir un método para extraer y codificar la información de diferentes tipos de caracteres expresadas como imágenes, empleando un Método de Muestreo.
- Proponer un método de búsqueda de patrones simbólicos que permita clasificar los diferentes caracteres.
- Obtener la distribución de los puntos de muestreo sobre las imágenes de los diferentes tipos de caracteres.
- Comprobar experimentalmente el desempeño del método de reconocimiento propuesto ante distintos tipos de caracteres.

Delimitaciones

El problema del reconocimiento es dependiente del contexto, no se puede hablar de una heurística o método general para desarrollar herramientas de reconocimiento de propósito general.

El método de reconocimiento de caracteres que se propone consiste en extraer la información por medio de un muestreo que caracterice a cada imagen, donde cada imagen representa a los símbolos del alfabeto $\{a\dots z\}$, de acuerdo a sus características.

Se considera un conjunto de imágenes donde cada imagen representa a un carácter. No se consideran invariantes de rotación, escala o color.

Cada carácter está normalizado a un tamaño de $m \times n$ píxeles, en monocromático. Se consideran 26 símbolos distintos por cada tipo de carácter que se tiene (considerando solo letras mayúsculas o sólo letras minúsculas).

El método de reconocimiento de caracteres consta de dos etapas:

- Etapa de aprendizaje o cálculo de cadenas representativas: consiste en extraer la información de un conjunto de caracteres, y encontrar *patrones representativos* de estos mismos.
- Etapa de prueba: donde el método clasifica y reconoce un carácter de acuerdo a la información extraída en la etapa anterior.

Hipótesis de trabajo

- El uso de modelos de representación de información simbólicos (no aritméticos) permite resolver problemas particulares de reconocimiento de patrones [4, 5]. La codificación, representación y operación de la información sobre estos modelos puede ser diferente a un modelo aritmético.
- El uso de modelos de representación simbólica para representar objetos, en un espacio métrico con dimensionalidad grande facilita el reconocimiento de patrones [6, 7].
- Los modelos aleatorios permiten dar aproximaciones a problemas donde *no* se tiene una solución particular [8, 9, 10, 11].
- El concepto de *inexactitud* [8] permite dar perspectivas distintas para aproximarse a conceptos como *patrón* y *similitud* para cierto tipo de problemas, en particular el reconocimiento de caracteres [12].

Organización

La estructura de este documento es la siguiente:

En el capítulo de Antecedentes se da un panorama general de lo que es el reconocimiento de caracteres, mostrando un análisis general de los clasificadores empleados en esta tarea como son las redes neuronales y las memorias asociativas.

En el siguiente capítulo se muestra el modelo de Espacio de Memoria Distribuido de Kanerva [13] mostrando sus principales características; se definen conceptos particulares de *similitud* y *patrón* para este trabajo.

En el capítulo de muestreo aleatorio de imágenes, se expone un método de codificación y representación de la información (en este caso de caracteres) en el espacio métrico $\{0,1\}^n$; se detalla el proceso que se sigue para realizar el reconocimiento, se define un modelo de representación de la información y una manera de extraer la información sobre las imágenes, que contienen a las variantes de caracteres.

En el capítulo cuatro, se presenta un método de reconocimiento de caracteres. Este método se evalúa mediante un proceso experimental, midiendo la eficiencia del reconocimiento sobre un conjunto de variantes de caracteres y se determina la complejidad del método sobre un modelo computacional de la máquina de Turing.

Por último se dan las conclusiones obtenidas y trabajos e inquietudes de líneas de investigación derivados de este trabajo.

Los capítulos están relacionados como se representa en la **Figura 1**.

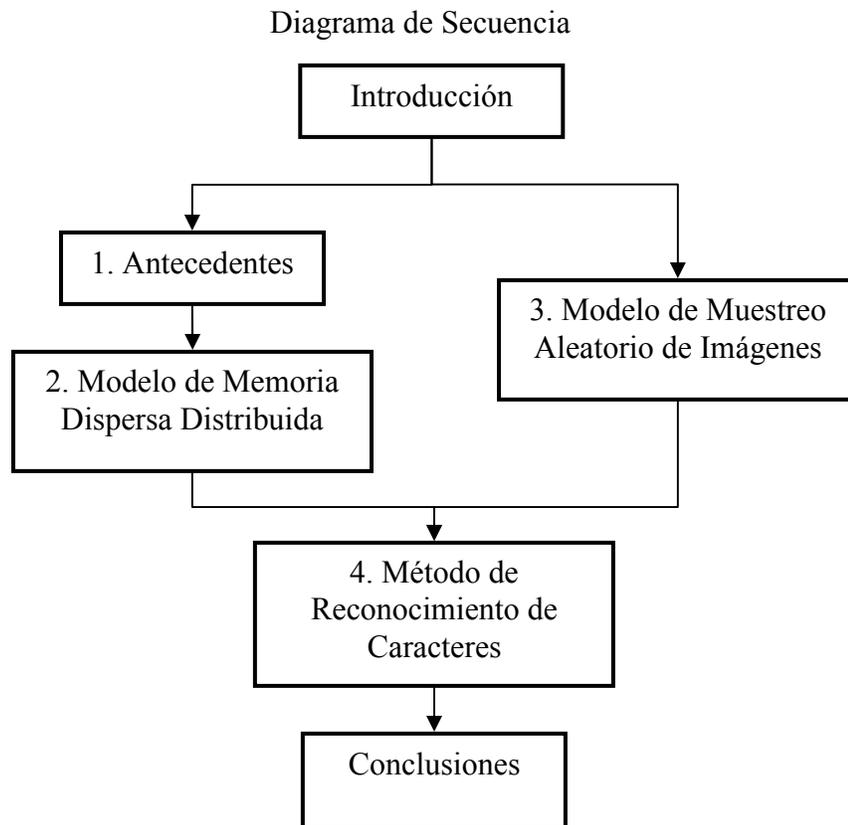


Figura 1. Diagrama de relación de capítulos de la tesis.

1 Antecedentes

En este capítulo se muestra el marco conceptual del reconocimiento de patrones y las herramientas utilizadas en el de reconocimiento de caracteres. Se detallan los métodos teóricos de clasificación más utilizados para realizar esta tarea y se analizan a nivel general las memorias asociativas, que permiten dar una solución distinta a los métodos tradicionales en reconocimiento de caracteres. Al final se muestra una discusión sobre los modelos de clasificación mostrados.

Los temas abordados en este capítulo sirven de fundamento e introducción a los modelos de computación asociativa, que llevan al desarrollo de nuevos modelos de reconocimiento de caracteres.

1.1 Reconocimiento de patrones

Establecer un modelo de análisis para clasificar y reconocer las *estructuras* de un fenómeno por sus *características*, es una tarea dependiente del contexto; las soluciones proporcionadas mediante un modelo son específicas a casos acotados, resultando que los modelos conceptuales no son generalizables [1, 2, 3], además muchas de las situaciones están limitadas por *eventos* externos al fenómeno, que no pueden ser caracterizados. Algunas de estas situaciones que influyen en el reconocimiento y clasificación son las siguientes:

1. Se asume que el fenómeno tiene estructura.
2. Ruido experimental (resultado de la medición del fenómeno).
3. Capacidad del modelo de representación.
4. Variables externas.
5. Estados o situaciones del fenómeno no contemplados.

Estas situaciones en conjunto, permiten que el fenómeno a caracterizar tenga un número grande de situaciones distintas (grados de libertad) en donde los atributos necesarios para su análisis resulten de difícil obtención, sean insuficientes o no sean los adecuados para los fines que se presenta [14]. Contar con modelos y técnicas con un costo computacional bajo, ayuda a dar soluciones a problemas de reconocimiento y clasificación.

Definición 1

“Reconocimiento de patrones es el acto de tomar datos crudos y acciones para poder clasificar estos datos en categorías de acuerdo a características presentes en los datos”.
[1]

El proceso de *reconocer* es un proceso que depende del contexto del fenómeno que se analiza. Reconocer es identificar de acuerdo a ciertos criterios, los cuales son tomados de la naturaleza del fenómeno; si los criterios se toman de acuerdo a la evolución en tiempo del comportamiento del fenómeno se consideran dinámicos, por el contrario, si se consideran en un instante determinado, se les denomina estáticos.

Los criterios son representaciones simbólicas o numéricas, para hacer abstracción del fenómeno que se analiza. Cada criterio, representa un *atributo* o *descriptor* de algún *evento* o *característica* particular del fenómeno [3]. Dependiendo del detalle del análisis del fenómeno se emplean uno o varios descriptores donde cada descriptor se codifica bajo un modelo de representación. El modelo de representación para que sea el adecuado debe considerar al menos dos operaciones: comparación y similitud, la primera para determinar cuándo dos elementos son iguales, y la segunda indica cuán parecidos son dos elementos.

1.1.1 Reconocimiento de caracteres

Un contexto particular del reconocimiento de patrones es el reconocimiento de caracteres en textos. El objetivo fundamental es extraer la información necesaria en forma de atributos que permita identificar y clasificar cada carácter que pueda estar presente en un documento escrito. Al *símbolo* o *dibujo* que denota a un carácter en particular se le llama grafía. El problema que se presenta al hacer un análisis sobre textos es que la forma, tamaño y estilos de las grafías no son uniformes y depende del tipo de documento que se desea analizar.

Dentro del reconocimiento de caracteres se pueden hacer dos distinciones, de acuerdo al origen del escrito:

1. Caracteres generados de moldes artificiales (impresos).
2. Caracteres manuscritos (generadas por humanos).

Para este trabajo sólo se analiza el caso de caracteres generados por moldes artificiales. La variedad de las grafías depende de factores tales como: escala, rotación y texturizado dentro de un texto. Contar con un número definido de *descriptores* sobre las grafías no es posible debido a la gran variedad de situaciones en que una sola grafía puede presentarse. Por ejemplo, en el caso de la carácter “h” se pueden dar varios ejemplos de diferentes estilos de caracteres, algunos de estos se muestran en la **Figura 2**, pero no sólo el problema es referente a la cantidad de variantes de caracteres si no también del contexto, porque siguiendo con el ejemplo del carácter “h” éste puede aparecer en contextos donde su identificación no resulta sencilla por medios computacionales (ver la **Figura 3**), entonces establecer modelos adecuados para identificar a los distintos caracteres depende de métodos especializados.

h, *h*, *h*, h, h, h, **h**, *h*, h, h, h,...

Figura 2. Diferentes estilos para representar un mismo carácter o letra.

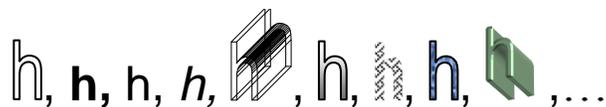


Figura 3. Variantes de estilo en un mismo carácter.

Buscar métodos que ayuden a encontrar información que sea significativa para *describir* una grafía, combinado con métodos teóricos de clasificación, permite establecer las bases suficientes para un reconocedor de caracteres.

En la literatura especializada existen varios métodos que dan soluciones particulares a problemas que están acotados en un contexto definido [2], en la **Figura 4** se muestran algunos métodos tradicionales para reconocer caracteres. Cada método ofrece buenos resultados en contextos particulares, pero ninguno ofrece soluciones generales. La capacidad de cómputo necesario varía en función de cada método.

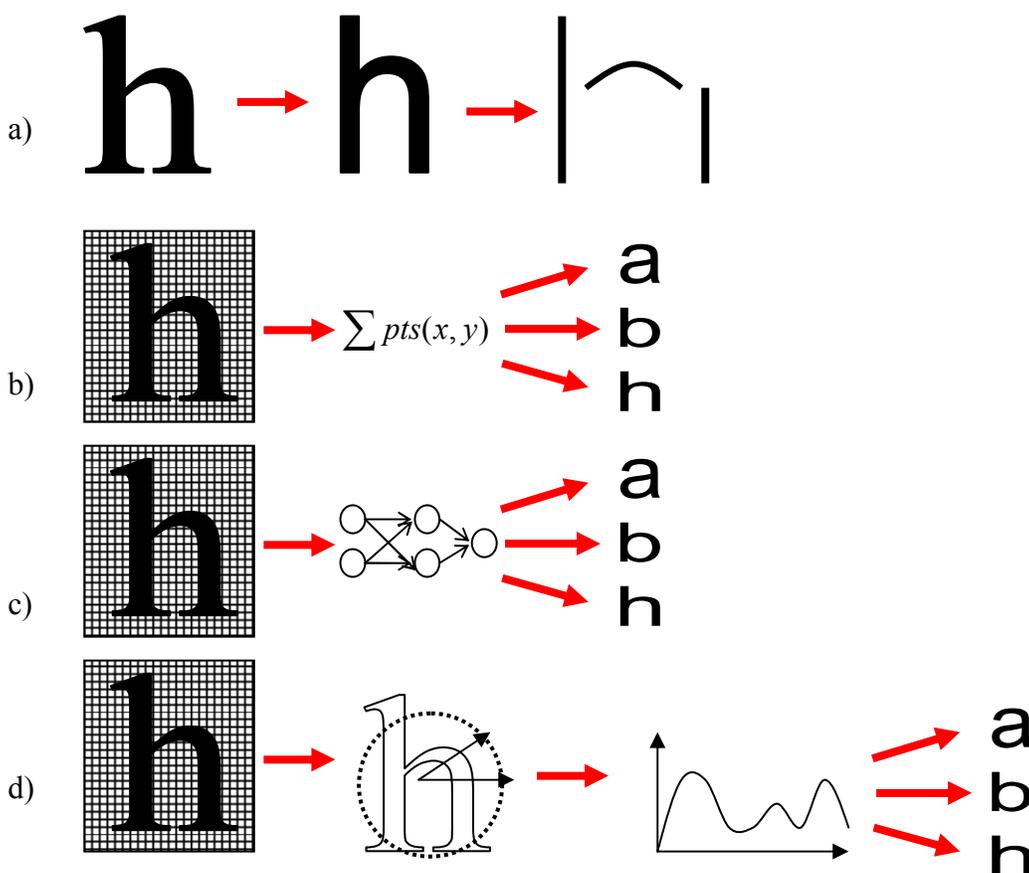


Figura 4. Diferentes maneras de realizar el reconocimiento de caracteres: a) por descomposición de formas primitivas, líneas, y curvas; b) por áreas del carácter ocupado; c) utilizando redes neuronales; d) construyendo gráficas de distancia a partir del centro de la imagen.

De los métodos descritos en la literatura, la mayoría de ellos caracterizan a un carácter por medio de la imagen de este, representada como un conjunto de datos y caracterizarlos por algún método de codificación en un *conjunto de descriptores numéricos*, que sirve de

entrada a algún método de clasificación. Este método usualmente es una transformación numérica que bajo ciertos umbrales indica la *clase* (letra) a la que es coincidente la entrada bajo un esquema de tolerancia (ver **Figura 5**).

Proponer maneras diferentes de resolver el problema, permite tener alternativas con un costo computacional distinto que ayuden a dar soluciones adecuadas a ciertos contextos. En este trabajo se muestra cómo, al emplear diferentes estructuras para codificar la información, y las diferentes maneras de representarla, se pueden construir maneras distintas con un coste computacional bajo (entiéndase por costo computacional como el número de operaciones necesarias para realizar una tarea [15]).

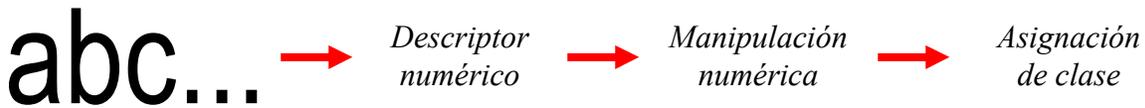


Figura 5. Proceso del reconocimiento de patrones mediante técnicas clásicas.

Los *descriptores*, al ser empleados sobre *características* particulares *cuantifican*, bajo un *esquema* de codificación, los estados que presentan las grafías de los caracteres, sin embargo, la diversidad de situaciones de una grafía no está limitada a un número definido, y no se puede saber *a priori*, qué *descriptores* son los adecuados para abstraer todas aquellas características que permitan identificar a cada carácter [1].

El empleo de modelos simbólicos basados en dos estados permite codificar la información como cadenas simbólicas [6], donde la estructura de las mismas, bajo algún modelo de codificación representa características particulares de los caracteres. Si la información codificada en cadenas es la *necesaria* por algún medio de codificación para representar a algún carácter, entonces la cadena es la abstracción de este carácter. Emplear modelos aleatorios [8, 9, 10] ayuda a encontrar soluciones a problemas que son difíciles de resolver, dando soluciones tentativas subóptimas que en contextos definidos son muy útiles. Unir dos modelos de estas características proporciona las bases suficientes para proponer procesos distintos para el reconocimiento de caracteres.

En el resto del capítulo se muestran algunos modelos teóricos de decisión, analizando en particular las memorias asociativas, de donde se toman algunos conceptos que ayudan a desarrollar un proceso automático para reconocer caracteres empleando cadenas simbólicas.

1.2 Métodos de clasificación

La descripción del fenómeno depende de la forma de codificación. Generalmente los descriptores son valores *numéricos reales*, aunque también pueden ser simbólicos, o cualquier otro valor que pueda ser cuantificable. Cuando se emplean varios descriptores para representar alguna situación, por notación son vistos como vectores de *n tuplas* donde cada componente representa a cada atributo; por ejemplo, si se tienen tres *descriptores* para identificar cierto tipo de aves (tamaño, peso, color), por lo general se denota como un

vector x , de la forma $x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$, donde cada $x_i \in \mathbb{R}$ o algún subconjunto de \mathbb{R} , denota el

atributo i -ésimo [1, 3]. Una función de *decisión* se define para operar sobre elementos en \mathbb{R}^n , donde cada elemento x se considera como un punto en un espacio de dimensión n ; un ejemplo de un espacio métrico se muestra en la **Figura 6**, donde se representa un espacio métrico en el que cada eje se corresponde con cada uno de los descriptores. En la **Figura 6**, se muestra el espacio métrico correspondiente tres descriptores para caracterizar a aves, en este espacio se observa que un ave particular se representa por un punto en este espacio, además que aves de una misma especie están dispersas en áreas definidas, también que especies distintas están esparcidas en conglomerados distintos, pero hay situaciones que estos conglomerados se encuentran solapados o muy juntos (ver **Figura 7**).

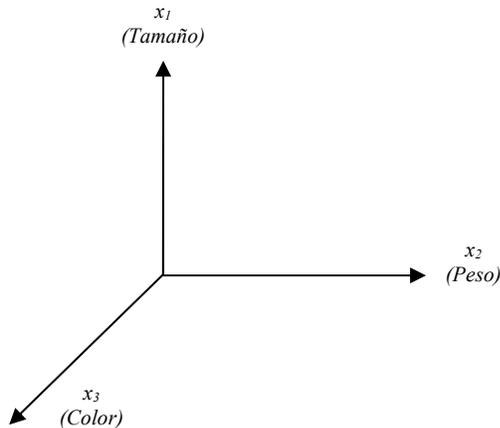


Figura 6. Representación del espacio de atributos del problema ejemplo.

La construcción de una *función de clasificación* tiene por objetivo asignar una *clase* a un elemento x del espacio, esta es una función que generaliza ciertas características de los elementos dentro del espacio. La búsqueda de funciones de clasificación adecuadas al problema es el objetivo de los *métodos teóricos de decisión*. La función de *clasificación* se denota por $d : D_1 \times D_2 \times \dots \times D_n \rightarrow V$, donde D_i representa el dominio del descriptor i -ésimo y V representa cualquier valor *numérico*, por lo general el conjunto de los reales, en notación también se expresa como $d(x)$, donde x es un vector de atributos. Esta función presenta las siguientes características [3]:

1. $d(x) - d(y) \geq 0$
2. $d(x) - d(x) = 0$

La construcción de d , consiste en definir áreas o secciones dentro del espacio de atributos que permitan clasificar a los elementos expresados dentro de éste. Supóngase que se consideran dos tipos de aves: el *azulejo* y el *periquito australiano*, con los atributos que se muestran en **Figura 7**, se desea clasificar a un conjunto aves, entonces se denota por \otimes a las aves que corresponden a un *periquito*, y \diamond a las aves que corresponden a un *azulejo*, esquemáticamente la **Figura 7** muestra la codificación de distintos ejemplares en el

espacio. Construir una función de *decisión* d , para clasificar a cada ave consiste en encontrar la curva que *corte* o que *separe* a ambos grupos, pero dependiendo de la dispersión los elementos caracterizados en este espacio, la construcción de la función d puede resultar difícil, en donde aproximaciones, con cierto grado de error, a esta función d , resultan útiles. Obsérvese como tres funciones distintas pueden generar resultados diferentes en la clasificación de las aves en la **Figura 7** (por cuestiones esquemáticas se presentan sólo dos atributos del ejemplo del ave). En el caso de la construcción de una función lineal, resulta más sencillo, pero hay elementos que son clasificados erróneamente (línea de la forma raya punto). En el caso de construcción de funciones no lineales, se tienen por ejemplo a las cuadráticas (representada por la línea constituida por puntos pequeños) que tienen un error menor en la decisión de elementos. Por último, la línea representada por puntos grandes separa mejor el área para definir cuándo un elemento pertenece o no a una clase determinada, pero a costa de una dificultad mayor.

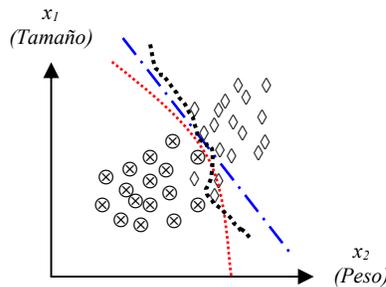


Figura 7. Codificación de elementos por atributos.

Es evidente la dificultad para generar la tercera función, y a veces nos basta con dar aproximaciones en donde la complejidad computacional y el tiempo de respuesta determinan la precisión para discriminar elementos en el espacio. En el caso de que se conozca el número de clases, en la literatura consultada se definen métodos que ofrecen aproximaciones para encontrar la función de discriminación d (en este trabajo los métodos de agrupación con un número de clases no definido se excluyen, debido a la naturaleza del problema, donde las clases se definen y son los diferentes caracteres o grafías del alfabeto). Algunos de estos métodos se esquematizan en **Figura 8**, tomados de [7].

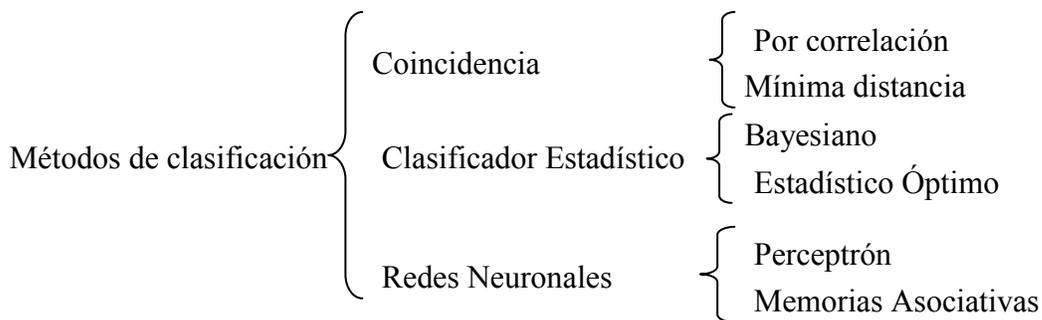


Figura 8. Tipos de clasificación tradicionales.

1.2.1 Modelos de clasificación basados en coincidencia

Los modelos de clasificación basados en coincidencia consisten en definir una función d y un umbral u que en conjunto permitan decidir si un elemento bajo la función d se encuentra dentro de un umbral u , de manera con la combinación de ambos, se pueda definir la clase a la que corresponde el elemento a analizar. La manera de definir el umbral u , y la función d , determinan el tipo de método por coincidencia que se emplea. Básicamente los métodos de coincidencia se pueden agrupar en los siguientes grupos [3]:

- Clasificador por mínima distancia.
- Coincidencia por correlación.

Clasificador por mínima distancia.

Dado un conjunto de descriptores de la forma $x = [x_1 \dots x_n]$, cada elemento codificado bajo estos descriptores se observa como un punto en el espacio $D(x_1) \times \dots \times D(x_n)$, donde $D(x_i)$ denota el dominio del atributo x_i , los elementos que resultan *parecidos* por los atributos mencionados, se encuentran dispersos en cúmulos. Los elementos que se consideran pertenecientes a la clase c_i , se agrupan de manera que se calcule el elemento distintivo a ellos (el elemento medio), el proceso se repite para todas las clases c_i que se tienen en el problema. La decisión de si un elemento dado pertenece a una clase, consiste en encontrar la distancia del elemento a clasificar (denotado por y), que tenga la *distancia mínima euclidiana* (se pueden considerar otras distancias). Una aproximación lineal para la función de decisión d entre dos elementos distintivos c_i y c_j se construye a partir de la recta normal a la distancia media entre los dos elementos.

En la **Figura 9** se observa que, en el caso en que las clases se encuentren muy dispersas y juntas, las aproximaciones lineales a la función d no pueden ser suficientes para resolver el problema de decisión, dando aproximaciones que no necesariamente son lineales.

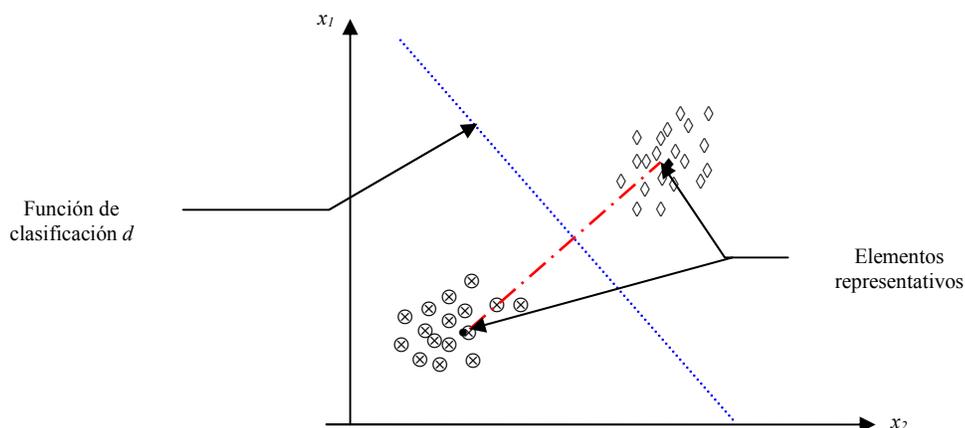


Figura 9. Esquema del criterio de mínima distancia.

El criterio de mínima distancia es aplicable cuando el número de clases involucradas es relativamente bajo y los elementos a clasificar se encuentran esparcidos dentro de conglomerados bien definidos. Aproximaciones distintas a las lineales son una forma de ganar potencia de reconocimiento a costa de una complejidad mayor, algunas de ellas son funciones cuadráticas, exponenciales o logarítmicas básicamente.

Emplear el criterio de mínima distancia, con otros modelos en casos particulares ofrece, desde el punto de vista computacional maneras económicas de definir un criterio de decisión para clasificar elementos. Un caso particular es el de este trabajo donde los elementos se ven como puntos en un espacio n dimensional con gran cantidad de clases y algunas veces con cúmulos solapados.

Coincidencia por Correlación

Este método se basa en tener un conjunto de descriptores x , de la forma

$$x = \begin{bmatrix} x_{11} & \dots & x_{m1} \\ \vdots & \ddots & \vdots \\ x_{1n} & \dots & x_{mn} \end{bmatrix}, \text{ que consiste en verificar en un espacio } n \text{ dimensional, dónde existe la}$$

combinación más cercana a los descriptores dentro de x [3].

Este método se emplea cuando se desea buscar un patrón (una imagen) de dimensiones $m \times n$, dentro de otro (i) de dimensiones $p \times q$, donde $m < p$ y $n < q$. La correlación permite definir una función que al *empalmar* (realizar la diferencia entre la imagen y el patrón que se busca en una posición determinada) sobre una región en i , el patrón m , identifique en donde la coincidencia de los descriptores de x sea mejor. El proceso descrito anteriormente se itera sobre el área en que se desea buscar dichas coincidencias de manera que se obtiene un patrón i' que representa el área buscada dentro de i , de manera y denote aquellas regiones de mayor coincidencia sobre x . Esquemáticamente se observa en **Figura 10**, y en **Figura 11**, se observa un ejemplo del método.

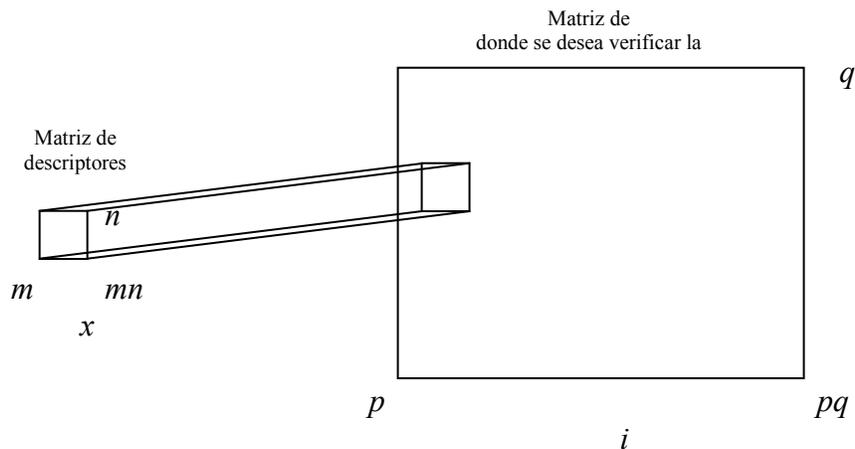


Figura 10. Esquema del funcionamiento de la coincidencia por correlación.

Este método tiene la desventaja de que no es tolerante a invariantes de rotación y escala, entonces es necesario un *preprocesamiento previo*, considerando a manera de ejemplo x , como un patrón de una imagen, sobre x , o sobre i , para encontrar las mejores coincidencias. Una manera de atacar los problemas de invarianza de rotación y escala es pasar a dominio de la frecuencia, vía la Transformada de Fourier, aplicando correspondencia por correlación.

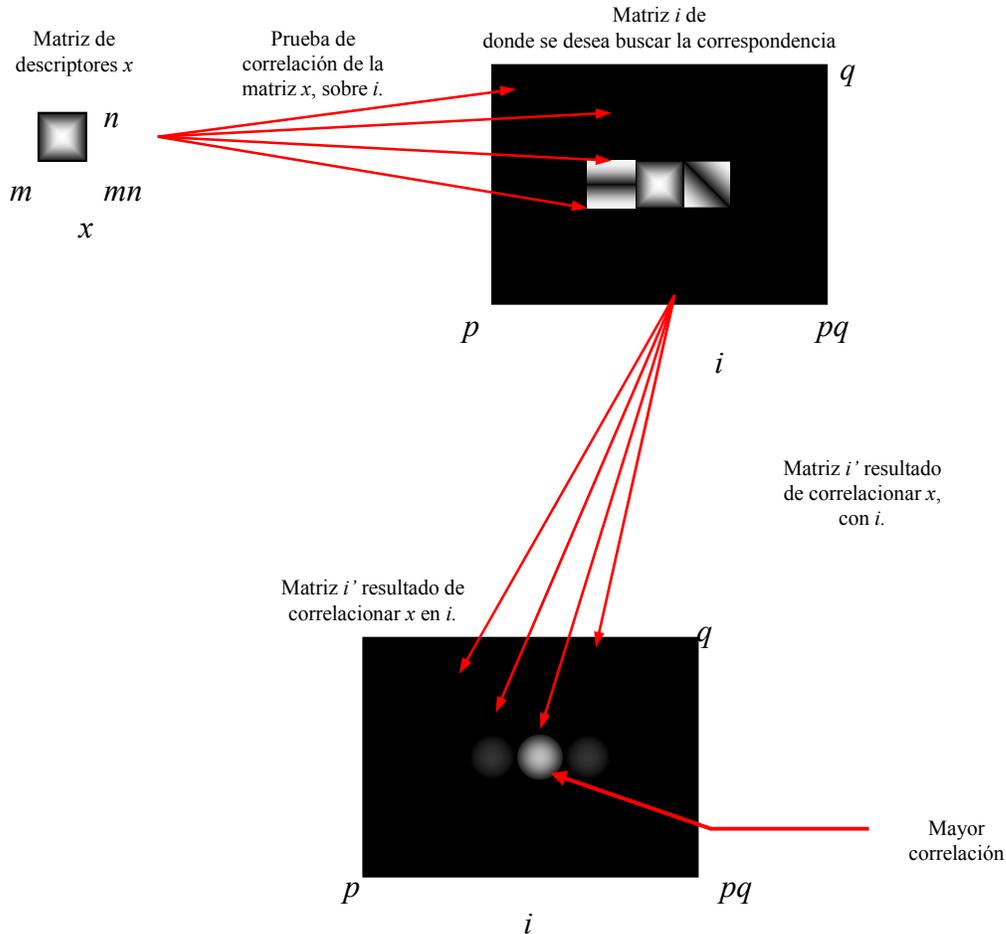


Figura 11. Esquema del proceso de correspondencia por correlación.

1.2.2 Clasificadores óptimos estadísticos

Este tipo de clasificadores consiste en construir la función de decisión d como la probabilidad de que se pueda seleccionar o no un elemento x en una clase w_i . Estos métodos son útiles debido a que en reconocimiento de patrones, añaden el concepto de aleatoriedad sobre las clases que normalmente se generan.

La probabilidad particular de que un patrón x_i sea de la clase w se denota por $p(w_i/x_i)$; entonces también se puede expresar que la probabilidad de que se incurra en un error de clasificación al asignarle la clase w_j , que se denota por L_{ij} . Entonces del conjunto posible de clase existentes denotado por M , el error promedio al clasificar x_i a la clase w_j , se expresa como la sumatoria de todas las probabilidades de error, denotado por la ecuación (1).

$$r_j(x_i) = \sum_{k=1}^M L_{kj} p(w_k / x_i) \quad (1)$$

esta sumatoria se conoce también como *promedio condicional de perdida*.

Un clasificador estadístico tiene M posibles clases para asignar, dado algún patrón desconocido. En el caso de que se computen todos los $r_i(x)$, ($r_1(x), \dots, r_M(x)$) para cada patrón x , y se asigne a cada x la clase con menor error de perdida, se tiene que el promedio con respecto a todas las decisiones será mínimo. Si se desea minimizar la probabilidad de perdida, entonces el clasificador se denomina clasificador *Bayesiano*, es decir se asigna al patrón x la clase w_i que cumple que $r_i(x) < r_j(x)$ para toda $j=1, 2, \dots, M$, donde $i \neq j$. Entonces un clasificador Bayesiano asigna al patrón x la clase w_i como se expresa en (2).

$$p(x / w_i)P(w_i) > p(x / w_j)P(w_j) \quad (2)$$

para todo $j=1, \dots, M, i \neq j$

donde $p(x/w_k)$ es la función de densidad de la probabilidad del patrón x para la clase w_k , y $P(w_k)$ es la probabilidad de ocurrencia de la clase w_k . La función de decisión para un clasificador Bayesiano es expresada de la forma como en (3).

$$d_j(x) = p(x / w_j)P(w_j) \quad (3)$$

1.2.3 Redes neuronales artificiales

Las Redes Neuronales Artificiales, son un modelo computacional que toma de metáfora el funcionamiento del cerebro, que consiste en la interconexión de entidades de proceso independientes (neuronas) que intercambian información (pesos) [16]. El modelo de Red Neuronal Artificial (RNA) se considera un modelo de caja negra debido a que se logra un resultado, pero no queda explícita la forma en que se logra [17].

Una RNA en computación se caracteriza como un conjunto de nodos interconectados con una dirección definida (una digráfica), donde cada conexión (ver **Figura 12**), tiene asociada un peso (w), este peso, es tomado de la metáfora, como el grado de enlace que tienen dos neuronas. Una RNA tiene una dirección de flujo, que indica la dirección en que fluye la información y se opera con esta, las neuronas que no tienen una neurona anterior se les conoce como neuronas de la capa de entrada; las neuronas que no tienen una neurona posterior, se les conoce como neuronas de la capa de salida; el resto de las neuronas, recibe el nombre de capas intermedias.

Cada neurona representa una entidad de proceso, este proceso esta definido por una función $y=f(x)$, también conocida como *función de transferencia*, donde x es la representación numérica (entrada) de la información de la neurona anterior. La estructura de una neurona se observa en **Figura 13**.

Una arquitectura de RNA se define por la función de transferencia y por la disposición de las neuronas que la conforman. La construcción de una RNA consiste en dos etapas:

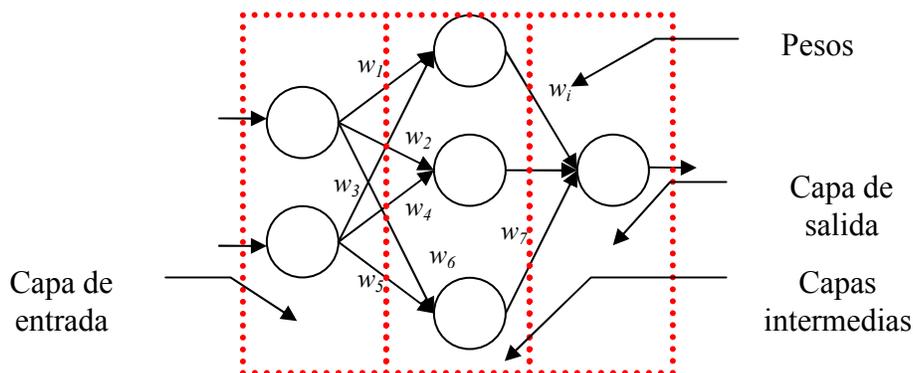


Figura 12. Estructura de una RNA.

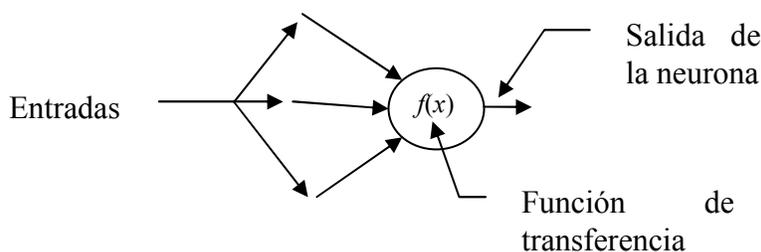


Figura 13. Estructura de una neurona.

Entrenamiento: Es el proceso donde se ajustan los pesos w de acuerdo a una función de transferencia y un conjunto de ejemplos de entrada, para los que se desea una salida definida. El proceso de entrenamiento con fines prácticos es visto como un problema de optimización, que consiste en encontrar el error mínimo del conjunto de entrada asociada a la salida de cada ejemplar de este conjunto.

Prueba: Sobre un conjunto de datos no necesariamente empleados en la fase de entrenamiento, se obtiene una salida.

Una RNA es un modelo computacional, donde se ha codificado un problema en un espacio de soluciones, pretendiendo segmentarlo en secciones por aproximaciones lineales [17, 18, 19]. Las capacidades de reconocimiento dependen de la arquitectura de red empleada (ver **Figura 14**).

Las RNA's en reconocimiento de patrones son importantes, porque pueden ser usadas como clasificadores universales. Las ventajas que tienen las RNA's consisten en que permiten entradas con ruido o imperfecciones sin que esto influya en el resultado. La capacidad de segmentar el espacio de soluciones es lo que permite usar a una RNA como un clasificador. El ajuste de los pesos entre las interconexiones de cada neurona, se determina a partir de un conjunto de ejemplos, que previamente se conoce su clasificación.

Se considera a una RNA como un modelo supervisado, debido a que es necesario contar con una métrica de error, que sirve de guía para el ajuste de los pesos; a una RNA una vez

entrenada, en la mayoría de las situaciones no se alteran los pesos asignados entre cada neurona.

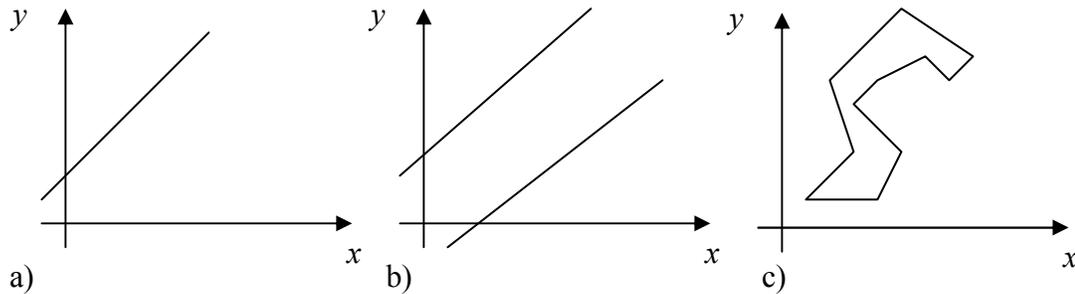


Figura 14. Diferentes arquitecturas de RNA sobre el espacio de soluciones: a) de una capa; b) dos capas; c) multicapa.

Una arquitectura muy usada en el reconocimiento de patrones es la de Back Propagation [18], por su facilidad de construcción. Al emplear RNA's tipo back propagation, los resultados que se obtienen están alrededor de 80 a 90 por ciento, pero modificaciones se elevan hasta 95 o más. Algunos trabajos de reconocimiento de caracteres empleando una variante de Back Propagation se ven en [19, 20, 21, 22, 23], donde por fines prácticos se crea un conjunto de RNA's para identificar a cada carácter y en la que para cada variante se entrena una RNA (Una variante es alguna situación en que pueda encontrarse un carácter, por ejemplo, rotada con un ángulo definido, escalada a un tamaño definido, etc.).

1.2.4 Memorias asociativas

Las Memorias Asociativas (MA) son un modelo abstracto computacional, que toma de metáfora el concepto de asociación de conceptos en *categorías* [24]. El funcionamiento de las MA's consiste en que dados dos conjuntos A y B, se *asocian* los elementos del conjunto A, con los elementos del conjunto B. En general se habla de dos conjuntos A y B, pero las *asociaciones* pueden hacerse sobre un mismo conjunto. La manera de *asociar* estos conjuntos es por medio de un conjunto de operaciones con los elementos de A y los elementos de B sobre una matriz denotada por M, estos conjuntos pueden ser de dominio real o entero [24, 25, 26]. En la **Figura 15** se observa un esquema de una Memoria Asociativa Clásica¹ general.



Figura 15. Esquema de una memoria asociativa.

Formalmente un elemento de A se denota por $a \in A$, y un elemento de B se denota por $b \in B$, entonces las asociaciones son denotadas por un conjunto donde cada elemento es una 2 tupla de la forma (a^i, b^i) , donde i representa la asociación i -ésima. El conjunto de relaciones de asociaciones se denota por R . Una MA esta constituida por una matriz M , que

¹ Se entiende por Memoria Asociativa Clásica los modelos basados en la memoria Hopfield.

contiene en ella (bajo algún conjunto de operaciones) codificadas las asociaciones en R , de modo que cuando se opera sobre M , se puede distinguir si el elemento ha sido codificado o no en M . Las MA, se clasifican en dos grupos:

- a) Heteroasociativa: Se dice que una MA es heteroasociativa si para el conjunto de asociaciones R , de los conjuntos A y B, para alguna asociación (a^i, b^i) contenida en R se tiene que $a^i \neq b^i$, donde $a^i \in A$ $b^i \in B$.
- b) Autoasociativas: Se dice que una MA es autoasociativa si para el conjunto de asociaciones R , se tiene que $A = B$ y para toda relación (a^i, b^i) contenida en R donde $a^i \in A$ $b^i \in B$ se tiene que $a^i = b^i$.

Las MA, son empleadas como clasificadores universales, porque se puede ver al conjunto A, como *patrones* de entrada, asignados a una *clase*, dentro del conjunto B, entonces, es posible tener subconjuntos de A, que están asociados a algún elemento en B, donde B es una agrupación que denota al conjunto A.

Los conjuntos A y B usualmente son vistos como algún producto n dimensional del espacio $\{0, 1\}$ [24], entonces los elementos que están asociados son vectores que contienen 0 y 1, algunos modelos expanden a conjuntos mayores como los naturales, o continuos como los reales. La dimensionalidad de A y B se denota por n y m respectivamente. Cada elemento de entrada (del conjunto A) es un vector n dimensional, que está asociado por operaciones sobre la matriz M, al elemento m dimensional (del conjunto B).

Una de las principales ventajas de una MA es que tiene tolerancia a variaciones sobre cada componente de los vectores empleados de entrada, es decir, tiene tolerancias a ruido aditivo o sustractivo y esto queda expresado como:

Si se tiene la MA denotada por M y $(a^i, b^i) \in \mathbb{R}$, y se tiene $a^i' \in A$, que denota un versión alterada por ruido (puede ser sustractivo o aditivo) de a^i , al operar sobre a^i' en M, el resultado obtenido es b^i . Es necesario notar que el nivel de intensidad de ruido que presente a^i' , que pueda tolerar M, depende de cómo fue introducido el vector a^i en M.

Una arquitectura de una MA está dada por la manera de operar sobre los conjuntos A y B. En la literatura se distinguen las varias arquitecturas distintas de memorias asociativas. Estas arquitecturas se observan en **Figura 16** y cada una utiliza diferentes conceptos para codificar los patrones en la matriz M.

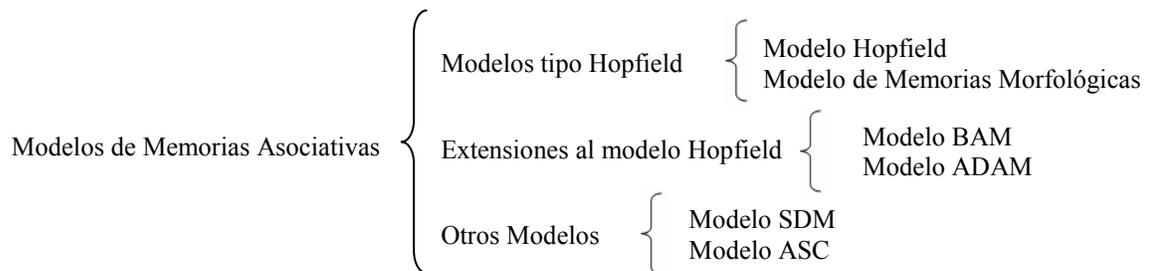


Figura 16. Modelos de memorias asociativas.

1.2.4.1 Modelo Hopfield

Este tipo de modelo de memorias asociativas es la generalización de los trabajos hechos por Steinbuch [27, 28, 29], Anderson [30, 31] y Willshaw [32]. Todos estos trabajos incluyendo al modelo Hopfield [20], se sustentan en la asignación de una clase basándose en la linealidad que puede estar expresada en la matriz M , por operaciones de suma y resta entre los elementos de entrada y los de salida. Tanto la entrada como la salida son considerados vectores en $\{0, 1\}$ con dimensionalidad n y m , respectivamente; los vectores de salida representan las clases a las que se les *asocia* cada vector de entrada, cada *clase* se representa por algún elemento de la base del espacio $\{0, 1\}^m$. Por ejemplo si $m=3$ entonces la base consta de los siguientes elementos: $[1, 0, 0]$, $[0, 1, 0]$ y $[0, 0, 1]$, razón por la que este tipo de clasificadores se apoya en la linealidad que existe entre los elementos de entrada y salida.

La Learn Matrix Steinbuch [27, 28] construye la matriz M de manera que cada elemento es codificado como una fila dentro de M , donde la posición de la fila determina la clase a la que se asocia el elemento, y a cada componente de M se le asigna un valor de la siguiente manera: por cada 1 existente en el vector de entrada se le asigna un +1, y por para cada 0 un -1. Anderson [30] y Willshaw [32] extienden el modelo en donde *mezclan* el vector de entrada con la clase asociada, de forma tal que aprovechan la linealidad en la matriz y la superposición de cada vector de entrada. La matriz generada se obtiene por el producto del vector de entrada por el vector de clase asociado, generando n matrices, una por cada elemento asociado, que al final son sumadas, obteniendo una matriz M que incluye a todos los elementos de entrada con sus clases asociadas. Formalmente si $a^i \in A$ y $b^i \in B$, y representan a la asociación *i-ésima* entonces la matriz M queda denotada por

$$M = \sum_{i=0}^n a^i (b^i)^T$$
. Para la fase de recuperación es necesario iterar gradualmente sobre un

conjunto de operaciones (siempre interviniendo el producto) hasta llegar a un punto de estabilidad que depende del modelo que se emplee entonces, el vector resultante es el vector recuperado dado la entrada empleada.

La contribución de Hopfield consiste en definir un modelo asociativo empleando una matriz, es decir la salida es la misma de la entrada, donde la entrada puede contener ruido o alguna distorsión. Como los elementos de entrada son los mismos que de salida, entonces M es una matriz cuadrada, define la diagonal con ceros de manera que no se asocie estrictamente una clase a una entrada.

Otros trabajos derivados de esta idea han tomado importancia en los últimos años, tal es el caso de las memorias morfológicas [33, 34, 35], que consisten en cambiar el conjunto de operaciones básicas sobre los renglones, pero manteniendo la linealidad que ata al modelo matricial. Pocos modelos (algunos se mencionan a continuación) intentan dar alternativas diferentes al modelo de memoria clásica de Hopfield.

1.2.4.2 Modelo ADAM

ADAM es un acrónimo de Advanced Distributed Associative Memory [36, 37], es una idea para extender el modelo propuesto por Hopfield. La contribución esencial es la introducción ahora de dos matrices para codificar y asociar cada vector, en otras palabras introduce el concepto de indexación de información sobre matrices, añade el concepto de *llave* o *código* sobre el cual se codifica el vector de entrada y sirve de referencia para *asociar* con su correspondiente salida. Este código es un vector también en $\{0, 1\}$. Gráficamente el modelo ADAM se muestra en la **Figura 17**. La fase de recuperación a diferencia que Hopfield, está dada por una iteración sobre la matriz M y el elemento de entrada.

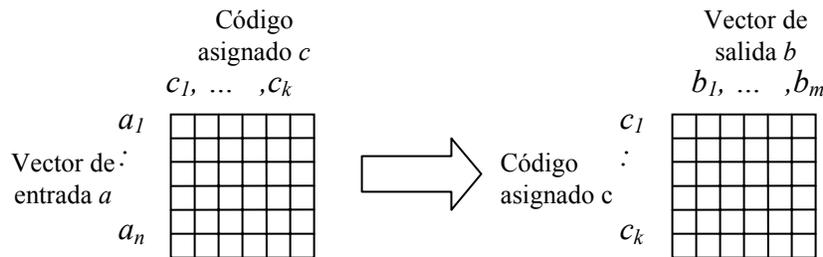


Figura 17. Esquema del modelo de memoria asociativa ADAM

1.2.4.3 Modelo BAM

Las siglas BAM son el acrónimo de Bidireccional Associative Memory [38, 39], y al igual que el modelo ADAM [36, 37], basa su funcionamiento en *indexación* en dos matrices, pero con la diferencia que no hace uso de un código en la fase de aprendizaje, en cambio, construye una matriz a partir del vector de entrada, y el vector de salida (similar a Hopfield), y la segunda matriz es la transpuesta de la primera. La fase de recuperación consiste en un proceso iterativo, de manera que dado un vector de entrada converja a un vector de salida, recuperando la asociación entre la entrada dada y el vector asociado. Este modelo, al igual que ADAM, es un intento de extender las posibilidades del modelo Hopfield.

1.2.4.4 Modelo SDM

SDM es el acrónimo de *Sparse Distributed Memory* (Memoria Distribuida Dispersa), este modelo es propuesto por Pentti Kanerva [6, 13, 40], es un modelo de memoria asociativa diferente al modelo Hopfield. Las principales características que tiene son:

- Emplea una alta dimensionalidad para representar los *patrones*.
- Hace uso de técnicas aleatorias para definir en principio el contenido de la memoria.
- Apoyarse en las características del espacio métrico conocidas como:
 - La relativa cercanía de los elementos distintos.
 - La distribución homogénea de la distancia de los puntos.
 - Una representación simbólica.

El modelo de Kanerva asocia en un espacio métrico $\{0, 1\}^n$ donde un conjunto de elementos de entrada (un conjunto de a^i) se asocia con un elemento de salida (b^i). El elemento de salida, esta en función de una métrica de similitud (para las entradas) y el calculo de “elemento medio” (para las salidas). La métrica de similitud es dada por la distancia de Hamming. Al igual que en el modelo ADAM hace uso de *indexación* entre los elementos que se desean asociar. El espacio de Kanerva es considerado un *anillo* ante el operador \oplus .

El tamaño del espacio empleado en el modelo de Kanerva se denota por n y el tamaño de cada ejemplar se denota por k , de manera que se cumple que $k \ll 2^n$. Entonces antes de describir la fase de aprendizaje y recuperación es necesario definir una matriz Q de dimensiones $k \times n$, que en un principio contiene elementos tomados al azar del espacio $\{0, 1\}^n$.

La fase de aprendizaje consiste en que dado un elemento de entrada $x^i \in \{0,1\}^n$, se asocia a un vector $y^i \in \{0,1\}^k$, en una matriz M . El método consiste en generar un vector b denominado de transición interna, resultado de la multiplicación de la matriz Q por el vector x^i , este vector de transición al multiplicarlo por el elemento a asociar y^i , genera una matriz M , que ha codificado la asociación deseada, entonces para toda entrada x^i , se repite el proceso. Al final se suma cada matriz M^i y la matriz resultante es la memoria que contiene las asociaciones de todos los elementos. Esquemáticamente se observa el proceso en **Figura 18**.

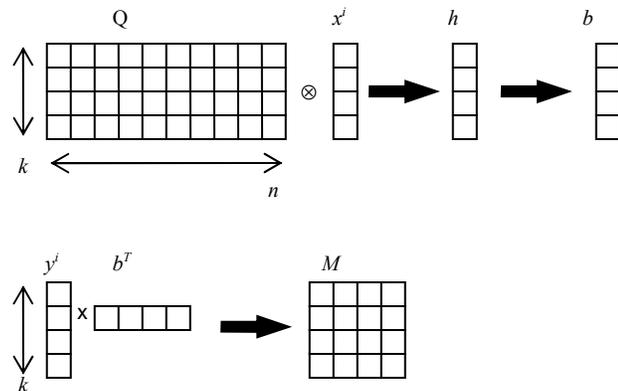


Figura 18. Fase de aprendizaje del modelo de Kanerva.

La fase de recuperación, consiste en que dada una entrada x^i , se determina el elemento y^i asociado. Para la fase de recuperación es necesario determinar el vector de transición b , que se calcula igual que en la fase de aprendizaje. Sobre la matriz M calculada en la etapa de aprendizaje se genera el vector ϕ , que contiene los umbrales para construir el vector a recuperar y^i ; al multiplicar el vector de transición b con la matriz M se obtiene un vector T , que de acuerdo a los umbrales calculados en el vector ϕ , se construye un vector y^i , que es el vector asociado a la entrada x^i . Esquemáticamente la **Figura 19** muestra el proceso.

En la fase de recuperación, el vector y^i , resultante es una variación sobre el vector original; entonces el modelo SDM es un modelo de memoria en donde las asociaciones se obtienen

con ligeras variaciones, es decir, se obtienen las asociaciones de manera *inexacta*, entendiendo por inexactitud el grado de error que existe entre el vector recuperado y el vector asociado originalmente.

El modelo SDM se puede considerar como una memoria asociativa inexacta, esta característica permite codificar cierto tipo de problemas de manera favorable, como es el caso de reconocimiento de patrones; donde la inexactitud se ve como la ambigüedad que puede existir al clasificar elementos en patrones, en donde la información global del evento representado como patrón ayuda a caracterizar a un conjunto de eventos con ligeras variaciones a este. En el siguiente capítulo se detallan las características de este modelo, que es el que se toma como base para la construcción de un método de reconocimiento de caracteres.

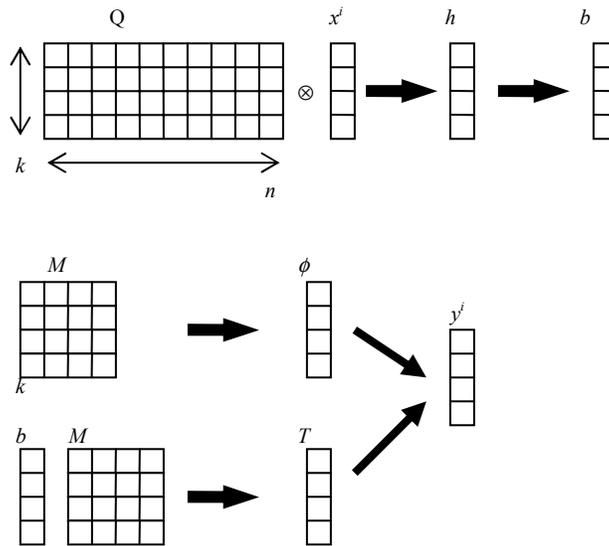


Figura 19. Fase de recuperación del modelo de Kanerva.

1.3.4.5 Modelo de cómputo asociativo ASC

El modelo de cómputo asociativo (conocido como ASC: ASsociative Computing) [41], se considera como un modelo de memorias asociativas, aunque estrictamente es un modelo de programación basado en asociatividad; el modelo lo propuso por Jerry L. Potter en 1992 La importancia de este método consiste en que propone una perspectiva diferente de realizar el cómputo empleando conceptos de *asociatividad*, *paralelismo* y *cómputo celular*. Este modelo hace representaciones simbólicas de los problemas de manera que, mediante un conjunto grande de *procesadores* a los que denomina celdas, se pueden realizar operaciones básicas sobre un conjunto de datos, comenzando a desarrollarse como un sistema de cómputo SIMD (Single Instruction Multiple Data); la característica principal de los modelos SIMD de computación es que a un conjunto grande de información se le aplica un conjunto de operaciones definidas que son implementadas directamente en Hardware. Las unidades de proceso en un modelo SIMD ejecutan sólo operaciones *básicas* sobre los datos, de manera que resulte económica su implementación.

El paralelismo del modelo se da al operar en forma simultánea (pero sincronizada) los procesadores y la información. El concepto de asociatividad, del modelo consiste en la forma de estructurar los datos, de manera que éstos se acomodan de tal forma que grandes volúmenes de información se procesan al mismo tiempo.

El concepto de asociatividad que propone el autor consiste en definir para un conjunto de datos, una estructura tabular de modo que en ella se codifique a cada *descriptor* como si fuera una columna de manera que el renglón completo identifica a un evento con cada uno de sus posibles atributos y valores (ver **Figura 20**).

Código	Atributo 1	Atributo 2	...
:	:	:	:
:	:	:	:
:	:	:	:

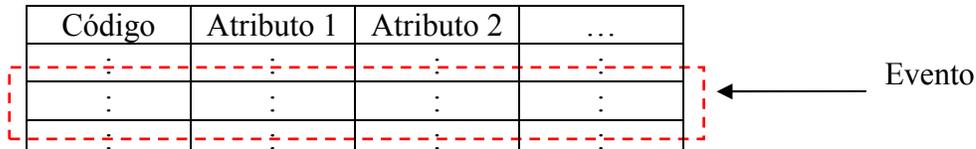


Figura 20. Estructura tabular, para codificar la información donde cada renglón representa un hecho codificado.

Adicionalmente se le añade un código, este código permite estructurar la información almacenada en forma tabular, de manera que puedan expresarse estructuras básicas de cómputo como arreglos, listas, o árboles. Este código también permite efectuar búsquedas sobre los renglones, los cuales se denominan como asociaciones.

El uso de múltiples celdas no sólo permite realizar operaciones sobre esta estructura tabular sino que se realicen en paralelo empleando tantas celdas como atributos contenga la estructura, lo cual permite que en un tiempo lineal se opere sobre los datos (ver **Figura 21**).

Al final cada célula del proceso, se activa o no dependiendo de las operaciones que se deseen realizar sobre la estructura. Por ejemplo, en el caso de una búsqueda la ejecución comienza de manera secuencial sobre la estructura, buscando en forma paralela sobre los atributos, y aquellos atributos que coincidan con el objetivo buscado se activan, de manera que se indica que se ha encontrado información para los atributos particulares que se especificaron.

Este modelo pone de manifiesto dos perspectivas de cómputo:

- Emplear más de una entidad de procesamiento para efectuar operaciones sobre los datos (es decir el concepto de unidad operativa lo extiende a una dimensión –vector de procesadores–)
- Usando estructuras de almacenamiento y procesamiento de información en dos dimensiones, diferentes a las estructuras lineales que se propone en la arquitectura de Von Neuman, permite tener modelos distintos de operación sobre la operación de los datos y modelos distintos de memoria, donde la información no es referida por direcciones físicas, si no que ofrece alternativas a direccionamientos por contenidos.

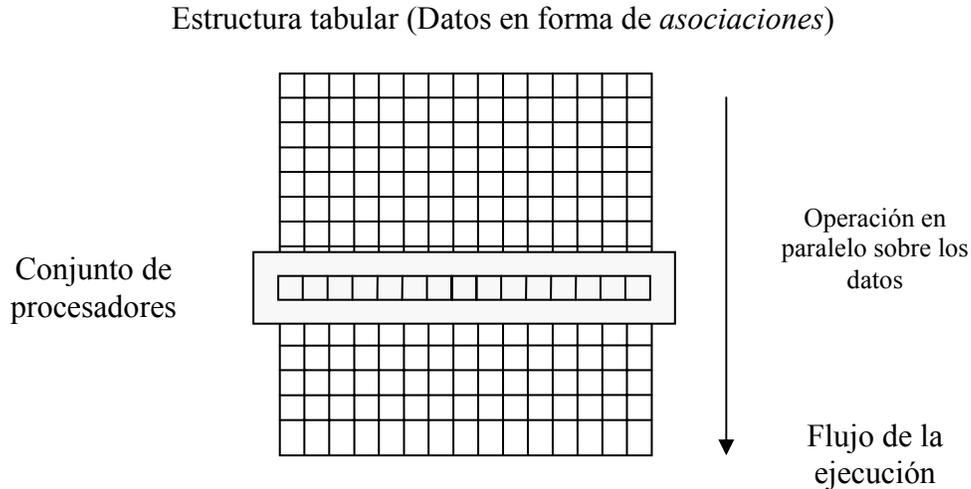


Figura 21. Estructura del modelo ASC.

1.3 Discusión

La diversidad de métodos para el reconocimiento de patrones es grande, cada uno ofrece características particulares que hacen que sea el adecuado para tareas específicas. La mayoría de los modelos codifica los *descriptores* como cantidades numéricas y con el conjunto de éstos se construye algún espacio donde se supone la linealidad de cada descriptor. Las soluciones propuestas son formas de agrupar conjuntos de puntos que representan ejemplares codificados y que se encuentran dispersos en este espacio, la más empleada es la que utiliza aproximaciones lineales. Proponer alguna solución particular empleando estos modelos implica encontrar los descriptores que sean lo más ortogonales entre ellos. Emplear descriptores que tengan una baja ortogonalidad impide generar modelos adecuados para reconocimiento de patrones [1, 2, 3].

En reconocimiento de caracteres, encontrar los descriptores necesarios para poder clasificar y reconocer a cada grafía diferente es un problema que no tiene una solución general [1, 3]. Es necesario buscar formas para codificar la información sobre las grafías de manera que expresen en ellas los descriptores que sean necesarios para su análisis. Estas formas distintas de codificación permite dar aproximaciones a la solución de problemas que son de difícil solución en una máquina de Turing [42, 43].

Uno de los modelos teóricos de decisión para el reconocimiento son las memorias asociativas. Las memorias asociativas clásicas ofrecen *codificar* y *asociar* los eventos basándose en la linealidad que puede ser expresada en una matriz. Son pocos los modelos que ofrecen maneras diferentes de solucionar este problema (por ejemplo los modelos ADAM, BAM y SDM implementan la noción de indexación para aportar maneras no directamente lineales de codificar y clasificar a la información).

La existencia de modelos diferentes al modelo de memoria asociativa de Hopfield permite estructurar de manera distinta la información, realizando el cómputo de manera diferente al

modelo clásico de la Máquina de Turing [15, 44, 45, 46], que puede ayudar a cierta clase de problemas. En este capítulo se han mostrado dos modelos de memoria asociativa: el SDM y el ASC, ambos ofrecen diferentes modelos de memoria, en este caso ambos modelos ofrecen representaciones simbólicas de los datos y ofrecen maneras distintas de direccionamiento y almacenamiento de la información.

En el modelo ASC una asociación se considera como un renglón en una estructura tabular que contiene todos aquellos descriptores sobre el evento que se está codificando además de un descriptor extra denominado *código*, que ayuda a organizar la información para tener un acceso más eficiente.

El modelo de Memoria Distribuida Dispersa de Kanerva ofrece una manera distinta de memoria asociativa. El concepto de *descriptor* como unidad cuantificable (numérica) la desecha en favor de un modelo simbólico de cadenas. En una cadena de longitud definida se codifican los eventos, en donde las operaciones, son simbólicas. La métrica de *similitud* es la distancia de Hamming, donde las cadenas se representa en un espacio métrico $\{0,1\}^n$.

El modelo de Kanerva ofrece una alternativa para el reconocimiento de caracteres en donde la información se codifica como cadenas de dos estados de una alta dimensionalidad. En el siguiente capítulo se describe este modelo y se dan definiciones de patrón y *similitud*.

2 El modelo de memoria distribuida dispersa de Kanerva

Las capacidades de manipulación y codificación están limitadas por la manera de representar las *características* del problema que deseamos analizar. Un modelo de representación es una herramienta conceptual que consiste en *abstraer* los hechos en forma de un conjunto de símbolos y reglas de operación.

El Modelo de Memoria Distribuida Dispersa (SDM) propuesto por Kanerva [6, 13, 40], es un modelo de memoria asociativa que emplea modelos de representación simbólicos con pocos estados², diferente a los modelos aritméticos que emplean la mayoría de los modelos teóricos de clasificación. Las representaciones simbólicas son cadenas de longitud definida. En este capítulo se detalla el modelo SDM de Kanerva y también se da una definición operativa de conceptos de patrón y similitud.

2.1 Patrón y similitud

En el capítulo anterior se comentó acerca de la dificultad de caracterizar a un objeto de acuerdo a sus *características* (descriptores). Si se desean emplear diferentes formas de representar la información [5, 8, 47] es necesario crear conceptos que permitan manipular este modelo de representación. Emplear modelos distintos de representación la información conduce a emplear modelos no aritméticos simbólicos, si estos modelos cuentan con pocos estados las operaciones entre estos símbolos es relativamente fácil de definir, y al usar altas dimensionalidades se permite codificar con mayor detalle ciertos *descriptores* sobre algún evento que se desea representar. Con un conjunto de operaciones sobre el modelo de representación se puede hacer una generalización sobre la información codificada y dar aproximaciones a conceptos como *patrón*, y *similitud* [1, 2].

2.1.1 La representación

La información se representa como cadenas de símbolos, en el que cada símbolo tiene dos estados, (símbolos binarios) resultado de la concatenación de varios símbolos, este tipo de representación es no aritmético y permite codificar de manera diferente los fenómenos que se desean analizar [6, 13]. Algunos ejemplos de estas cadenas se muestran a continuación:

Si se consideran el conjunto $\{a, b\}$, algunas de las cadenas generadas son:

aaabbbbaaabbaabbababbaba
bbbabbbabbbababbaabbabb

² En este caso dos estados.

...

En el caso que se consideren los estados $\{s_0, s_1\}$ se tienen:

$s_1 s_0 s_0 s_0 s_0 s_1 s_0 s_1 s_1 s_0 s_1$
 $s_1 s_0 s_1 s_1 s_1 s_0 s_0 s_1 s_1 s_0 s_0$

...

O en el caso de considerar por comodidad los símbolos $\{0,1\}$:

1100010010100100010010
 0010010100101010100101

...

Emplear distintas formas de representación de la información no puede tener maneras directas de implementación en un cómputo clásico (Máquina de Turing). Para implementar en una arquitectura de cómputo clásica es necesaria hacer una emulación del modelo no clásico auxiliándose de un modelo intermedio que sea implementable. El espacio $\{0,1\}^n$ sirve de intermediario para hacer una emulación de un modelo no clásico basada en cadenas simbólicas. La equivalencia de cadenas simbólicas de dos estados en puntos de este espacio métrico se da por una biyección entre las cadenas simbólicas y los puntos del espacio. Tomando los ejemplos anteriores se tienen las siguientes biyecciones:

En el primer caso

$f : \{a,b\}^n \rightarrow \{0,1\}^n$ tal que para que x_i sea la i -ésima componente de $x \in \{a,b\}^n$ se tiene

$$\text{si } f(x_0 \dots x_i \dots x_n) = y_0 \dots \left(\begin{matrix} a & 0 \\ b & 1 \end{matrix} \right)^n \dots y_n, \text{ para toda componente de } x$$

en el segundo

$f : \{s_0, s_1\}^n \rightarrow \{0,1\}^n$ tal que para que x_i sea la i -ésima componente de $x \in \{s_0, s_1\}^n$ se tiene

$$\text{si } f(x_0 \dots x_i \dots x_n) = y_0 \dots \left(\begin{matrix} s_0 & 0 \\ s_1 & 1 \end{matrix} \right)^n \dots y_n, \text{ para toda componente de } x$$

y por último

$f : \{0,1\}^n \rightarrow \{0,1\}^n$ tal que para que x_i sea la i -ésima componente de $x \in \{0,1\}^n$ se tiene

$$\text{si } f(x_0 \dots x_i \dots x_n) = y_0 \dots \left(\begin{matrix} 0 & 0 \\ 1 & 1 \end{matrix} \right)^n \dots y_n, \text{ para toda componente de } x$$

Obsérvese que esta función f es una biyección sobre el mismo espacio, debido que $\{0,1\}^n$ del dominio corresponde al espacio aritmético $\{0,1\}^n$ y el codominio corresponde a una representación simbólica, donde sólo se tienen los símbolos “0”, “1”.

Asociar a cada elemento de $\{0,1\}^n$ una representación que sea implementable en el modelo de la máquina de Turing, permite emular el comportamiento de este modelo de representación. Las operaciones básicas del modelo son solo dos [7]:

- a) *La igualdad.* Permite decidir si dos cadenas son o no iguales.

- b) *La diferencia entre dos cadenas*. Indica las diferencias sustanciales que existen entre la caracterización de dos cadenas.

Con estas dos operaciones se puede dar una definición operativa de *patrón*, y una métrica de *similitud* entre dos cadenas simbólicas, las cuales se comentan en la siguiente sección.

2.1.2 Definición de patrón y similitud

El problema de decir si x es parecido a y depende de las maneras y los criterios de cómo se maneja la información sobre el fenómeno x y y . Pueden existir fenómenos caracterizados que empleando algunas *suposiciones* o algunos *argumentos* resulten parecidos o no a otros. Definir una manera general de decir que x es parecido a y no es posible por la cantidad de elementos que un fenómeno puede tener. Éstos dependen directamente de la forma de codificar y de representar dicho fenómeno. Se pueden usar definiciones locales al problema ayudando a resolver situaciones específicas.

El término de patrón no es generalizable porque depende de la manera de caracterizar y codificar al conjunto de descriptores que lo conforman, tal como se mencionó en el capítulo 2. Esta codificación puede ser cuantitativa, simbólica, o de algún otro tipo. En este trabajo se considera que bajo un modelo de representación simbólica de dos estados, un patrón de un conjunto de cadenas de símbolos binarios, es una cadena que es *representativa* bajo una *métrica de similitud*.

Se puede dar una aproximación de una definición de patrón acotado al modelo de representación. Esta definición se hace por medio de una función de *similitud*, de manera que permita distinguir cuando un elemento en particular contiene o no las características más significativas de un conjunto de elementos. Si este elemento es el más significativo, entonces se dice que es un patrón de conjunto de datos que representa. Teniendo en cuenta lo anteriormente expresado para un sistema de codificación simbólico, en este caso el sistema basado en $\{0,1\}$ la definición de patrón es la siguiente:

Definición 2

Un patrón es una cadena simbólica $x \in \{0,1\}^n$, de longitud n tal que a un conjunto $C \subset \{0,1\}^n$, y bajo una función de similitud $F: \{0,1\}^n \times \{0,1\}^n \rightarrow \mathbb{N}$, x es una cadena representativa al conjunto C bajo un δ dado.

En esta definición la representatividad está dada por intervalos de elementos contenidos en \mathbb{N} , donde de acuerdo a la naturaleza del problema serán definidos en instancias particulares de F . La similitud o parecido entre elementos de $\{0,1\}^n$, se da por la distancia de Hamming; formalmente queda expresado en Definición 3.

Definición 3

Sea la función de similitud $F: \{0,1\}^n \times \{0,1\}^n \rightarrow \mathbb{N}$, está definida como $F(x, y) = \|x - y\|$, donde “ $\|$ ” es la norma y “ $-$ ” la diferencia entre dos cadenas, se dice que son muy parecidos si y

solo si $F(x, y) < \delta$ donde δ es el radio del círculo $O(x, \delta)$ que es $\delta \ll n$, y recibe el nombre de radio de cercanía³.

Entonces se tiene que $\forall_{y \in C} F(x, y) \leq \delta$ para todo elemento de C , donde \mathbb{N} , representa el conjunto de números naturales.

2.2 El espacio $\{0,1\}^n$

La representación de información en este modelo está basada en el reflejo de estructuras y mecanismos de almacenamiento y transformación de la información. Los datos en comparación sobre modelos tradicionales (aritméticos) son correlacionados. Las principales ideas que gobiernan en este modelo son [7]:

1. La representación es uniforme: Todas las cosas son representadas por puntos en el mismo espacio.
2. El significado es: Los puntos cercanos en la representación en el espacio significan lo mismo (la semántica no está separada de la sintaxis).

La alta dimensionalidad es más importante que la dimensión natural en el caso del uso de cadenas con pocos estados, por ejemplo: la capacidad de la información contenida en un vector simbólico es aproximadamente equivalente a una página de texto, todos los pasos de computación pueden ser tomados como una página de Turing.

En el cómputo tradicional, se emplean palabras o registros, que a su vez están divididos en campos, de manera, que el conjunto de partes más pequeñas representan una abstracción del evento codificado. Con el uso de cadenas simbólicas se puede modelar atendiendo a las definiciones de un patrón, es decir, la manera de hacer abstracción consiste en encontrar elementos representativos, a un grupo que no necesariamente es la suma de las partes.

Estos resultados en una representación distribuida reciben el nombre de holográfico u holístico. Las operaciones permitidas en este espacio se discuten en la siguiente sección.

2.2.1 Conceptos del espacio $\{0,1\}^n$

En un espacio $\{0, 1\}^n$, el número de puntos se encuentra determinado por $N=2^n$, donde n , es el número de dimensión del espacio [13].

Los puntos en N se representan por n tuplas de símbolos '1' y '0' y, pueden ser escritos como *números enteros* de n bits en *representación simbólica*, es decir se puede formalizar una biyección del espacio $\{0,1\}^n$ al $\{0, 1\}^n$, donde este último solo es una representación simbólica, (representados entrecomillados con " de aquí en adelante), la cual queda definida como:

³ Los conceptos de norma y diferencia se detallan más adelante.

$f : \{0,1\}^n \rightarrow \{0,1\}^n$ donde x_i es la i -ésima componente de
de $x \in \{0,1\}^n$ se tiene si $f(x_0 \dots x_i \dots x_n) = y_0 \dots \left(\begin{matrix} 0 & 0 \\ 1 & 1 \end{matrix} \right)^i \dots y_n$, para toda
componente de x

El espacio N en conjunto con las operaciones de tuplas en coordenadas como el *and*, y *or* forman un álgebra booleana. Algunos conceptos importantes dentro del espacio $\{0,1\}^n$ son

Definición 4

Origen 0. Es el punto que tiene todas las componentes en 0, para una n particular; formalmente $x \in \{0,1\}^n$ y $i \in \{1, \dots, n\}$ tal que $\forall x_i, x_i = 0$ por ejemplo:

1. Para $\{0,1\}^3$ se tiene el origen (0, 0, 0), visto como una cadena 000.
2. Para $\{0,1\}^5$ se tiene el origen (0, 0, 0, 0, 0), visto como una cadena 00000.

Definición 5

Complemento $\neg x$: El punto complemento (opuesto) a otro, es aquel punto que tiene 1's en las componentes que el punto original tiene 0, y 0 en donde el original tiene 1's., formalmente se tiene:

$$\neg : \{0,1\}^n \rightarrow \{0,1\}^n \text{ tal que } \neg(x) = (y_1, \dots, y_i, \dots, y_n), \text{ donde cada } y_i = \begin{cases} \text{si } x_i = 0 \text{ entonces } 1 \\ \text{si } x_i = 1 \text{ entonces } 0 \end{cases} \quad (5)$$

donde x_i es la componente i -ésima.

Por ejemplo:

1. Para el punto (1,0,1,0,0,1,1,0) en $\{0,1\}^7$ su complemento es el punto (0,1,0,1,1,0,0,1), visto como la cadenas correspondiente al punto se tiene 10100110 y su complemento 01011001.
2. Para el punto (1,1,1,0,1,1,1,0,1) en $\{0,1\}^9$ su complemento es el punto (0,0,0,1,0,0,0,1,0), visto como la cadenas correspondiente al punto se tiene 111011101 su complemento 000100010.

Definición 6

Norma $|x|$: La norma de un punto x es la cantidad de unos en la representación simbólica. La norma es un escalar. Formalmente se tiene:

$$|| : \{0,1\}^n \rightarrow \mathbb{N}, \text{ tal que } |x| = \sum_{i=1}^n x_i \text{ si } x_i = 1 \quad (6)$$

Ejemplo: Para $\{1,0\}^9$ y los puntos $x_1=(0,0,1,1,0,0,0,1,1)$, $x_2=(1,1,1,1,1,1,1,1,1)$, $x_3=(0,0,0,0,0,0,0,0,0)$ se tiene que $|x_1|$ es 4; $|x_2|$, es 9 y $|x_3|$ es 0.

Definición 7

Diferencia x-y: La diferencia de dos puntos x y y es otro punto que tiene '1's en las componentes donde x y y difieren en contenido y '0's en otro caso. La diferencia es conmutativa. Formalmente se tiene:

$$- : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}^n \text{ tal que } -(x,y) = (z_1, \dots, z_i, \dots, z_n), \text{ donde cada } z_i = \begin{cases} 0 & \text{si } x_i = y_i \\ 1 & \text{en otro caso} \end{cases} \quad (7)$$

Ejemplo: Sea el espacio $\{0,1\}^8$ y los puntos en este espacio $x_1=(1,1,1,0,0,1,1)$, $x_2=(0,1,1,0,0,1,1)$, $x_3=(1,1,0,1,0,1,1)$, $x_4=(0,1,0,1,0,1,0)$.

$$\begin{aligned} x_1 - x_2 &= (1, 0, 0, 0, 0, 0, 0) \\ x_3 - x_4 &= (1, 0, 0, 0, 0, 0, 1) \end{aligned}$$

Definición 8

Distancia d(x, y): La distancia entre dos puntos x y y , es la cantidad de componentes en las cuales difieren. Se le conoce como *distancia de Hamming* y está expresada en bits. Entonces la distancia es la norma de la diferencia. La distancia también al igual que la norma es un escalar. Formalmente queda definida como:

$$d : \{0,1\}^n \times \{0,1\}^n \rightarrow \mathbb{N}, \text{ tal que } d(x, y) = |x - y| \quad (8)$$

donde el símbolo $-$ es la diferencia y $||$ es la norma, los cuales fueron definidas anteriormente.

Dados dos puntos de \mathbb{N} que son muy cercanos bajo alguna cota en proporción a \mathbb{N} , uno con respecto al otro decimos que son similares.

Ejemplo: $d(1110011, 0110011) = 1$
 $d(1101011, 0101010) = 2$

Definición 9

Betweenness x:y:z El punto y está entre el punto x y z si y sólo si la distancia desde x a z es la suma de las distancias de x a y y desde y a z ; esto es:

$$\text{Sean los puntos } x, y, z \in \{0,1\}^n, y \text{ esta entre } x \text{ y } z \text{ si y sólo si } d(x, z) = d(x, y) + d(y, z) \quad (9)$$

Es fácil ver que todo bit de un punto en betweenness es una copia del bit correspondiente al punto final.

$x:y:z$ si y solo si $y_i=x_i$ ó $y_i=z_i$ para $i=0, \dots, n-1$. La relación $x:y:z: \dots$ es definida recursivamente como $w:x:y: \& x:y: \dots$

La relación de *betweenness* es simétrica ($x:y:z: \dots$ si y sólo si $\dots:z:y:x$) pero no transitiva. El conjunto Y esta entre los puntos x y z ($x:y:z$) si y solo si para todos los puntos se esta entre x y z . Todos los elementos de \mathbb{N} están entre un punto y su complemento, es decir $x:\mathbb{N}:\neg x$.

Definición 10

Ortogonalidad $x \perp y$. Se dice que dos puntos x, y son ortogonales entre sí, si son perpendiculares o indiferentes y si y solo si la distancia entre los dos es la mitad del número de dimensiones.

$$\text{Se dice que dos vectores } x, y \in \{0,1\}^n, \text{ son ortogonales sii } d(x, y) = \frac{n}{2} \quad (10)$$

La distancia $n/2$ es llamada la distancia indiferente de $\{0,1\}^n$. Si x es ortogonal a y , es también ortogonal a su complemento y' (x es el punto medio entre y y $\neg y$): $x \perp y$ implica $x \perp \neg y$

Definición 11

Círculo (r, x) : Un círculo con radio r y centro x es el conjunto de puntos que tienen como máximo una distancia r de x , formalmente es expresado como:

$$O : \mathbb{N} \times \{0,1\}^n \rightarrow 2^{\{0,1\}^n} \text{ tal que } O(r, x) = \{y \in \mathbb{N} \mid d(x, y) \leq r\} \quad (11)$$

Por ejemplo, en el espacio $\{0,1\}^5$, se considera un punto $x=(1, 0,0,0,1)$ y un radio $r=1$, se tiene que $O(r, x) = \{(1,0,0,0,1), (0,0,0,0,1), (1,1,0,0,1), (1,0,1,0,1), (1,0,0,1,1)\}$.

El espacio \mathbb{N} puede ser representado por los vértices de un cubo de una unidad en un espacio euclidiano n dimensional. La distancia entre dos puntos de \mathbb{N} es entonces la longitud de la ruta más corta a través de los ejes del cubo que une a los dos puntos, también se llama *distancia city block*. Cuando dos puntos son más distantes que un bit, hay múltiples caminos cortos entre ellos.

Los vértices de una unidad cúbica n dimensional se encuentran en la superficie de una esfera n dimensional con radio $\frac{\sqrt{n}}{2}$. Se observa que el espacio es esférico debido a que:

1. Cualquier punto en x tiene un único opuesto $\neg x$.
2. El espacio entero está contenido entre el punto x y su opuesto $\neg x$.
3. Todos los puntos son iguales (esto es que para cualquier pareja de puntos x, y hay una distancia preservando automorfismo⁴ del espacio que mapea de x a y que desde cualquier punto el espacio se observa igual).

La superficie de la esfera claramente es esférica, de acuerdo a la definición \mathbb{N} es también esférica, desde $y \oplus x \oplus (\dots)$ ⁵ es un automorfismo que mapea de x a y (para cada $n!$ permutaciones de coordenadas hay un automorfismo que mapea de x a y)

Porque \mathbb{N} es esférico se puede pensar como una superficie de una esfera con circunferencia $2n$. Todos los puntos de \mathbb{N} son igualmente cualificados como puntos de origen y un punto y este complemento son como dos polos a una distancia n de cada otro, con el espacio completo entre ellos.

⁴ Ver glosario.

⁵ El símbolo \oplus denota el operador o exclusivo.

2.2.2 Propiedades del espacio $\{0,1\}^n$

Dado que N es esférico [13], puede ser rotado al mover cualquier punto de éste al origen, por lo tanto hablar de un punto arbitrario x es usualmente conveniente pensar en x como el origen $x=0$, porque la distancia desde 0 a algún otro punto y es simplemente la cantidad de unos en y :

$$d(0,y) = |y-0| = |y| \quad (12)$$

El espacio por ser $\{0, 1\}^n$, al ser esférico presenta las siguientes tres características:

1. Cualquier punto x en $\{0, 1\}^n$, tiene un único opuesto \bar{x} .
2. El espacio entero está contenido entre x y \bar{x} .
3. Para dos puntos x y y la distancia preserva un automorfismo del espacio que mapea de x a y , que para cualquier punto del espacio se observa igual.

La Distribución del espacio N . El número de puntos que hay exactamente en d bits desde un punto arbitrario x (decimos que el punto 0) es el número de maneras para seleccionar d coordenadas de un total de n coordenadas y por lo tanto esta dada por el coeficiente binomial.

$$\binom{n}{d} = \frac{n!}{d!(n-d)!} \quad (13)$$

La distribución de N entonces es una distribución binomial [48] con parámetros n y p , donde $p = 1/2$. La media de la distribución binomial es $n/2$, y la varianza es $n/4$ esta función de distribución se denota por $N(d)$. La distribución normal F con media $n/2$ y desviación estándar $\sqrt{n}/2$ es una aproximación a ésta.

$$N(d) = \Pr\left\{\frac{d - n/2}{\sqrt{n/4}}\right\} \quad (14)$$

Muchas de las propiedades de N pueden ser derivadas de la distribución binomial y su aproximación normal.

Tendencia a la ortogonalidad. Una propiedad de N es que muchos de los puntos contenidos en N están aproximadamente a una distancia media de $n/2$ desde un punto, con solo una pequeña fracción cercana por otro punto [13]. En otras palabras, gran parte del espacio tiende a ser ortogonal dado un punto arbitrario y la analogía con a cualquier punto y su complemento, tomados como polos. En la **Figura 22**, se muestra la distribución de la distancia en el espacio $\{0, 1\}^n$, para distintos valores de n .

En otras palabras los puntos en este espacio tienen una distancia definida por las variaciones existentes entre las cadenas y no se tiene una jerarquía de posición como en una representación aritmética: la variación de una componente representa lo mismo que este al principio o en medio de la cadena, mientras que en un sistema aritmético, el valor depende de la posición y el símbolo representado. Por ejemplo:

Sea el punto (1, 1, 1, 0, 0, 0, 1, 1, 1, 0) en $\{0, 1\}^{10}$, los puntos que se encuentran a una distancia de 1 son:

$$\{x / x: d(1110001110, x) = 1\}$$

$$\{1110001111, 1110001100, 1110001010, \dots, 1100001110, 1010001110, 0110001110\}$$

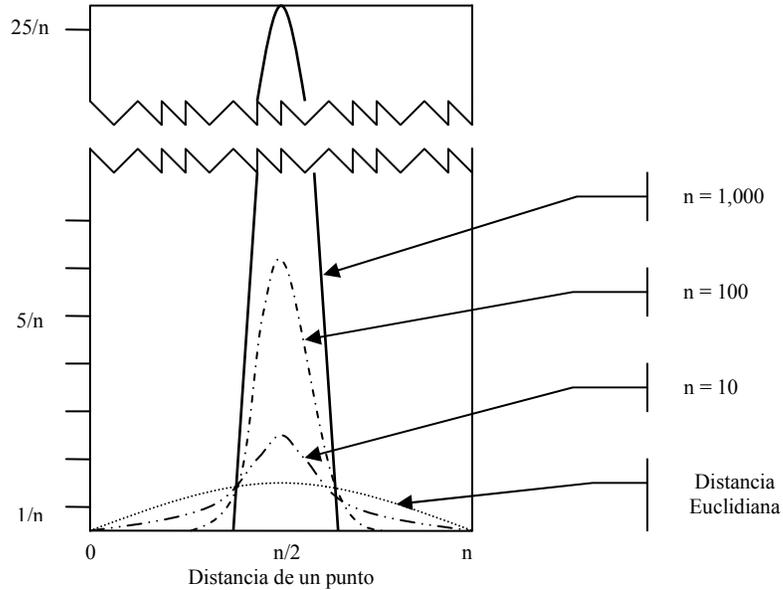


Figura 22. Distribución de la distancia de Hamming $\{0,1\}^n$ y la distancia euclidiana de tres dimensiones.

Distribución del círculo $O(r, x)$. Se observa como el espacio N es distribuido alrededor de un punto.

El área del círculo $O(r, x)$ es el número de puntos que son 0, 1, 2, ..., r bits desde x , por lo tanto es la suma del primer $r+1$ coeficiente binomial

$$|O(r, x)| = \binom{n}{0} + \binom{n}{1} + \dots + \binom{n}{r}, \quad (15)$$

Su aproximación normal es $N \cdot F\{(r-n/2)/\sqrt{n/4}\}$, donde F es la función de distribución normal. El círculo que incluye aproximadamente la mitad del espacio ($r \leq n/2$) está situado a una distancia muy cercana de su periferia. La distribución del círculo que inscribe menos de la mitad del espacio ($r < n/2$) es aproximadamente a una binomial con parámetros n y p , donde p es significativamente menos que $2(r/n) \cdot (1-r/n)$.

Multiconjuntos

Definición 12

Un multiconjunto es una colección de objetos no ordenada, similar a un conjunto excepto que cualquier objeto puede ocurrir más de una vez. Los objetos son llamados elementos. Los multiconjuntos están encerados entre corchetes cuadrados []

Por ejemplo: $[A, A, B, C]$ es un multiconjunto de tamaño 4 y $[A, A, B, C] = [A, B, C, A] \neq [A, B, C]$

Definición 13

Multiunión. La multiunión denotada por \oplus , se define como el multiconjunto constituido de los elementos que contienen dos multiconjuntos A y B .

Por ejemplo: $[A, A, B] \oplus [A, B, C] = [A, A, A, B, B, C]$

El número de elementos de la multiunión es la suma de elementos de sus constituyentes multiconjuntos.

$$|X \oplus Y| = |X| + |Y| \quad (16)$$

2.3 Conceptos de memoria dispersa

Una palabra puede ser almacenada en memoria por escritura en un espacio libre y al mismo tiempo proveer la localidad con la dirección apropiada de decodificación. Las localidades libres y sus direcciones están dadas desde el comienzo y solo el contenido de la posición es modificable. La posición de almacenamiento es muy pequeña en comparación con 2^n (la memoria es escasa).

Las localidades de almacenamiento se distribuyen aleatoriamente en direcciones localizadas en el espacio $\{0,1\}^n$.

2.3.1 Elementos de memoria como puntos en el espacio $\{0,1\}^n$

Los elementos se representan por patrones de n componentes de ceros y unos. La distancia entre dos puntos $\{0, 1\}^n$ representa la *similitud* de dos elementos de memoria y es el número de lugares en cual los dos patrones difieren, es decir entre mas cercanos más parecidos son. Se considera que dos puntos son iguales o parecidos si la distancia entre ellos no excede $n/4$, en otro caso, si la distancia es mayor a $n/4$ y menor a $3n/4$ se consideran indiferentes. Un elemento puede ser considerado como el centro del círculo con un radio dado. Los puntos dentro del círculo se pueden considerar como elementos de memoria.

Si se observa el espacio $\{0, 1\}^n$ en el plano cartesiano, se puede caracterizar como círculos donde el radio determina el umbral para definir si los elementos son suficientemente parecidos o no y el centro es el elemento característico de cada círculo.

2.3.2 Conceptos de memoria en $\{0,1\}^n$ y almacenamiento distribuido

Multiconjuntos N' de localidades fijas. Las localidades de almacenamiento de N' de una memoria dispersa se obtienen de un muestreo aleatorio, con reemplazo de direcciones del espacio N , donde $N' \ll N$. Estas pueden ser llamadas simplemente localidades o localidades fijas. Las localidades de N' son un multiconjunto. El símbolo 'es empleado para indicar si

un elemento pertenece a N' , es decir x' se refiere a una localidad fija en N' . El símbolo N' es usado para el número de localidades fijas dispersas en la memoria.

Distancia a una localidad fija. Las localidades fijas pueden ser referidas por su dirección, que se denota por x' , entonces una distancia a una localidad fija desde un punto arbitrario es la distancia a la localidad x' , esto es la distancia $d(y, x')$ entre dos direcciones. De manera similar, por la distancia entre dos localidades es decir la distancia entre sus direcciones.

Vecinos cercanos a N' . El elemento de N' es muy similar a los elementos de x de N llamados vecinos cercanos de x . Se denota por x' (x' es la localización fija más cercana a X). Si X es un subconjunto de N entonces X' denota el conjunto de vecinos cercanos de X :

$$X' = \{x' | x \in X\} \quad (17)$$

Distancia a una localidad cercana $d(x, x')$. La distribución de la distancia de N a sus más cercanos N' vecinos puede ser derivada de la distribución de N . La distancia en N es distribuida de acuerdo a $N(d)$.

$$N(d) = \Pr \{d(x, y) \leq d | x, y \in N\} \quad (18)$$

La distribución de $d(x, x')$ se denota por $N'(d)$ donde

$$N'(d) = \Pr \{d(x, x') \leq d\} \quad (19)$$

Una aproximación a la expresión (19) se muestra en la expresión (20).

$$N'(d) \approx 1 - \exp \{-N'N(d)\} \quad (20)$$

Radio de Acceso r y Círculo de acceso $O'(x')$. Todas las localidades dentro de una distancia r de x pueden ser almacenadas o provistas de datos, dependiendo de cómo la memoria está siendo escrita o leída. Se define el círculo de acceso $O'(r, x)$ como el conjunto de localidades fijas en el círculo $O(r, x)$. Por notación abreviamos $O(r, x)$ a $O(x)$ y $O'(r, x)$ a $O'(x)$. El círculo de acceso es el multiconjunto de localidades rígidas dado por:

$$O'(x) = N' \cap O(x) \quad (21)$$

Se dice que las localidades fijas y' son accesibles por x si y' no es tan lejano r bits desde x .

Acceso solapado $I'(x, y)$. El conjunto de localidades fijas accesible desde x y y está dado por

$$I'(x, y) = O'(x) \cap O'(y) \quad (22)$$

La media del número de localidades en el acceso solapado depende del círculo de acceso y la distancia $d(x, y)$ entre los centros de los círculos solapados.

Contenido de la localidad $C(x')$. El contenido de la localidad x' se representa por $C(x')$ donde este es un multiconjunto de los objetos (palabras) que han sido escritas en esta localidad.

Escritura en x' . Escribir la palabra η en la localidad fija x' consiste en incluirla en el multiconjunto de palabras $C(x')$ contenidas en x' , es decir

$$C(x') = C(x') \cup [\eta] \quad (23)$$

Escribir en la dirección ξ . Escribir la palabra η en ξ , significa escribir η en todas las localidades rígidas accesibles por ξ .

Dato en x , $D(x)$. El dato en x es el contenido completo de localidades accesibles desde x . Formalmente queda expresado como:

$$D(x) = \bigoplus_{y' \in O'(x)} C(y') \quad (24)$$

Si la palabra η ha sido escrita con la dirección ξ , entonces el multiconjunto $D(x)$ contiene $|O'(x) \cap O'(\xi)|$ copias de η , una para cada localidad accesible desde x y ξ . Las palabras están escritas con una dirección distinta a x .

Leer en x . Leer en x significa tomar un elemento representativo de N del dato en x .

Palabra en x . La palabra en x $W(x)$ es propiamente seleccionar una palabra representativa del dato x .

2.4 Discusión

El modelo de Kanerva es un modelo diferente a los modelos clásicos de memorias asociativas; este modelo se considera asociativo porque se asocia a un patrón de entrada x , con una posición del espacio denotado por y . Los patrones son codificados como puntos en un espacio métrico $\{0, 1\}^n$. Los aspectos de más importancia en este modelo son:

1. Los patrones son *cadena de símbolos* de dos estados.
2. Uso de la aleatoriedad en la construcción de la memoria.
3. Empleo de una dimensionalidad grande para codificar la información.
4. Una métrica distinta de modelos aritméticos, en donde las características de esta métrica ayudan a codificar y agrupar la información. Algunas de estas características son:
 - a. Distribución de los puntos.
 - b. Tendencia a la ortogonalidad de un punto dado.
5. Redundancia de la información.

Al decir que los patrones son cadenas simbólicas, se tiene un comportamiento distinto en operación y manipulación sobre los patrones con respecto a los modelos aritméticos. Las operaciones que se definen en el modelo simbólico pueden tener comportamientos distintos que son aprovechados en la codificación. Estos comportamientos son la distribución del espacio y la tendencia a la ortogonalidad [13] de los elementos codificados, logrando una independencia distinta a una independencia lineal (que es la que intenta explotar Hopfield y otros modelos similares).

Al igual que los modelos BAM y ADAM, se basa en técnicas de indexamiento, para la construcción de las asociaciones, pero a diferencia de ellos emplea métodos aleatorios para establecer lo que denomina posiciones fijas de almacenamiento. El uso de técnicas aleatorias permite proponer características nuevas en modelos asociativos. Las operaciones y procesos que se definen en el modelo van de acuerdo a las características que presenta función de similitud que se ha definido como la distancia de Hamming. Este marco conceptual permite mostrar un modelo distinto de memorias en donde se parte de emplear una representación simbólica no aritmética, sobre el cual se construye el modelo operativo para proporcionar un modelo distinto de memoria asociativa.

Estas características, son las que hacen *útil* el modelo para definir nuevos modelos computacionales para el reconocimiento de patrones, en donde al proponer maneras distintas de representar y operar (basándose en el trabajo de Kanerva) se pueden desarrollar diferentes algoritmos con un costo computacional bajo, para resolver tareas específicas, como el reconocimiento de caracteres.

3 Modelo de muestreo aleatorio de imágenes

Dependiendo de la manera de representar, codificar y operar la información de algún fenómeno es necesario contar con un mecanismo para extraer y evaluar datos que permita caracterizar y analizar al fenómeno. Un método de extracción de información, es el muestreo probabilístico: un muestreo *adecuado* permite la selección de los elementos de la *población* que contienen la información *relevante* [48], de manera que al trabajar con la muestra se trabaja solo con una fracción de la información original.

En el contexto de reconocimiento de caracteres, el proceso de extracción de información permite identificar y extraer el conjunto de *descriptores* útiles para discriminar cada carácter. La información de un carácter se representa como el total de los puntos que conforma la imagen del mismo, el Método de Muestreo tiene el objetivo de extraer un conjunto de puntos que sean representativos, que, como se comentó en el capítulo 2, se presenta un mecanismo de muestreo en imágenes de caracteres con la finalidad de extraer la información que resulte útil para discriminar a cada carácter, codificando a cada una de ellas como una cadena simbólica.

3.1 Muestreo de imágenes

El mecanismo de extracción de información de un fenómeno, parte del hecho de que el fenómeno es medible y puede ser caracterizado por un modelo de representación que emula o imita el comportamiento del mismo. Este modelo de representación codifica de forma parcial la información del fenómeno, en ocasiones limitada por cantidad de espacio, recursos computacionales y de almacenamiento [8, 10, 11]. Para solucionar esto, se emplean mecanismos de extracción de información, los cuales deben ser suficientemente robustos, de manera que la información sea lo más relevante para la tarea a realizar. Esquemáticamente se ve como en la **Figura 23** [47], donde a partir de un fenómeno, se hacen representaciones abstractas de su comportamiento, en que para hacer el análisis general, se extrae la información que sea significativa de este fenómeno.

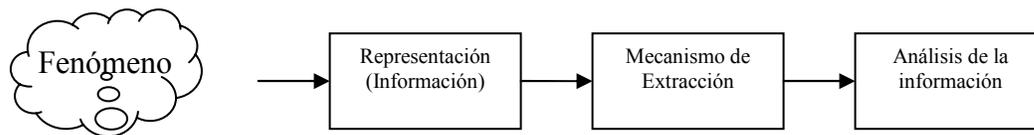


Figura 23. Extracción de información

Una población es el total de individuos que representa un fenómeno. Cuando una población es considerablemente grande, se emplean modelos de extracción de información y el muestreo aleatorio probabilístico es uno de ellos. El muestreo consiste en seleccionar en

una población un conjunto de elementos que describa a la población total. El muestreo si es el representativo, tiene múltiples ventajas, debido a que recoge de manera más compacta los datos (en este caso información de los caracteres). El nivel y calidad de la abstracción dependerá de la calidad de la muestra seleccionada [37]. En la **Figura 24** a través de un mecanismo de muestreo se seleccionan los elementos característicos de la población.⁶

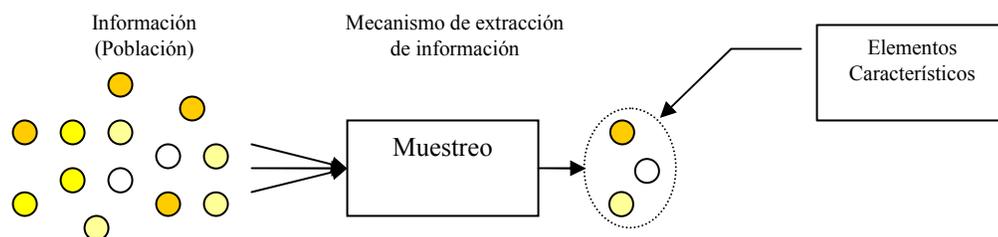


Figura 24. El muestreo como una manera de hacer reducción de información.

La extracción de información de los caracteres en las imágenes en reconocimiento de caracteres puede considerarse como un problema de muestreo, el objetivo es extraer aquellos puntos que conforman la imagen de los caracteres de manera que la identifique. Un muestreo en reconocimiento de caracteres es un subconjunto de puntos de una imagen que contiene la información necesaria para representar un carácter, o, en forma general, a un conjunto de caracteres, entonces un muestreo *adecuado* permite que no se use la totalidad de la información de cada variante de carácter, si no una fracción representativa.

El muestreo de una imagen obtiene un subconjunto de puntos que representa un *patrón* de este carácter. En el caso que se tengan diferentes variantes de los caracteres al aplicar el mismo muestreo de puntos, la información será muy parecida.

Un reconocimiento de caracteres consiste en extraer la información por medio de un muestreo que caracterice a cada símbolo del alfabeto escrito $\{a...z\}$ y un método de clasificación que identifique a cada carácter de acuerdo a las características que fueron codificadas por el muestreo.

Se considera un conjunto de imágenes donde cada imagen representa a un carácter. No se consideran invariantes de rotación, escala o color (ver **Figura 25** y ver **Figura 26**).

a· b· c· d· e· f· ...
a b c d e f...
a b c d e f...

Figura 25. Ejemplo de casos de caracteres que son empleados en el reconocimiento

⁶ Por elemento característico entiéndase el elemento que representa a un conjunto y no necesariamente pertenece a este pero contiene las características más representativas del conjunto.

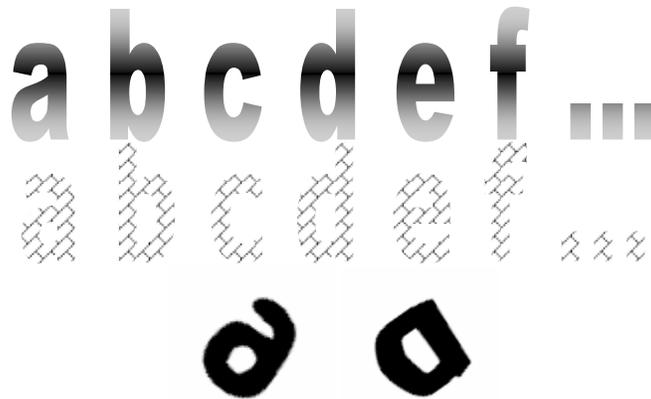


Figura 26. Ejemplo de caracteres no válidos para el análisis.

Cada carácter está estandarizado a un tamaño de $m \times n$ píxeles, en color monocromo. Se consideran 26 símbolos distintos por cada tipo de caracteres que se tenga (Considerando solo letras mayúsculas o solo letras minúsculas).

3.2 Muestreo aleatorio

Un método aleatorio ayuda a encontrar la información que es significativa dentro de las imágenes correspondientes a distintas variantes de caracteres, la codificación en cadenas simbólicas contiene información acerca de cada carácter para clasificar a cada símbolo del alfabeto escrito. El muestreo consiste en encontrar una lista de posiciones que contenga los puntos de muestreo *adecuados* para identificar a cada carácter (ver **Figura 27**).

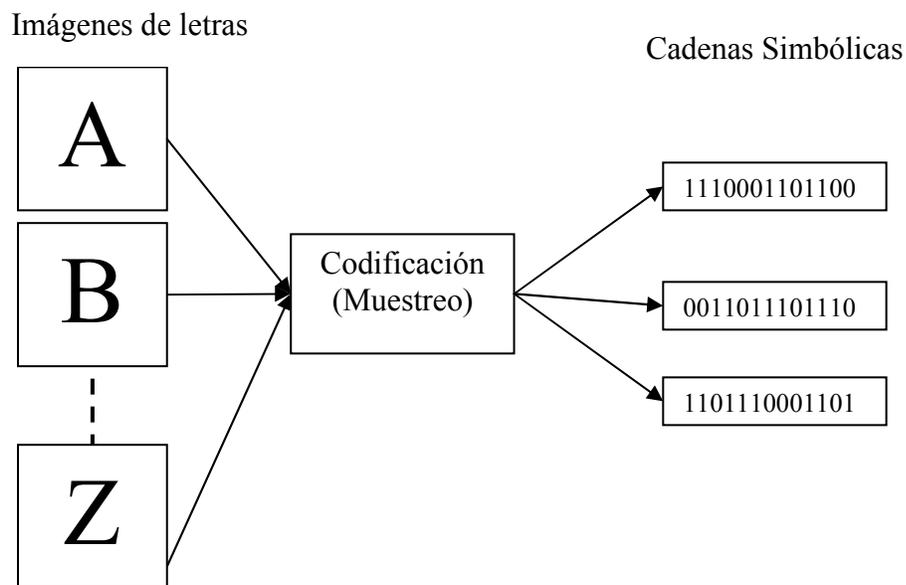


Figura 27. Proceso de generación de cadenas simbólicas sobre caracteres empleando un muestreo.

El uso de métodos aleatorios para codificar la información, no siempre garantiza el uso de la información *adecuada*, es necesario optimizar la muestra para extraer los puntos que sean los más significativos, para poder clasificar en forma eficiente [49]; se retoman los conceptos de similitud y patrón, descritos en el capítulo 2, los cuales sirven de fundamento para la manipulación de cadenas simbólicas.

En la siguiente sección se expone la manera de realizar un muestreo aleatorio sobre imágenes; considerando que cada punto, dentro de una imagen de un carácter, tiene la misma probabilidad para ser seleccionado. Los puntos representativos son almacenados en una lista de puntos, que se utiliza para construir cadenas simbólicas.

3.2.1 Representación

Un modelo de representación simbólica de dos estados no aritmético de gran dimensionalidad ayuda a codificar problemas específicos. Una aplicación particular es el reconocimiento de caracteres en donde se codifican como cadenas los atributos de cada carácter.

El uso de cadenas simbólicas de dos estados para reconocimiento de caracteres se ha venido empleando desde los años de 1960, pero el objetivo de utilizar las cadenas binarias ha sido para la construcción de estructuras indexadas (árboles) que permitan clasificar de acuerdo a las características de un carácter, pero no para representar a un carácter [9].

Las cadenas simbólicas permiten tener una representación en pocos estados de los símbolos de cada carácter, en donde no es necesario definir explícitamente los descriptores a codificar, siempre y cuando se cuente con una técnica de muestreo que pueda reducir la información que se maneja, resultando una manera más compacta y manejable para situaciones en que se utilice una excesiva cantidad de información.

En el reconocimiento de caracteres una representación simbólica con dos estados permite codificar y expresar nociones de *ausencia* o *existencia* de información [4], es decir, en una imagen de carácter dado, la distribución de los píxeles por el carácter, indica el conjunto de píxeles que tienden a ser ocupados y no ocupados por este carácter en particular, de manera que esta distribución de píxeles permite identificar a dicho carácter, la **Figura 28** muestra un ejemplo de esto, en donde se observa que para las grafías de los caracteres de las letras *w* y *r*, existe un área donde se esperaría que hubiese información acerca de los puntos (denotados por píxeles en negro) que conforman a la grafía en particular.

El uso de menos información que el total de la imagen del carácter permite que las *variaciones* pequeñas sean ignoradas, permitiendo así la tolerancia a varios estilos de caracteres, aunque perdiendo detalle de cada símbolo.

El objetivo del muestreo es encontrar los puntos dentro de las imágenes para la generación de cadenas que son codificadas en dos estados. Se consideran imágenes con representación de color binaria, en caso que se tengan imágenes con una representación más amplia del

espectro de colores, se define una función de transformación donde por lo general es de la forma:

$$f: \mathbb{N} \rightarrow \{ '0', '1' \} \tag{25}$$

Lo más común es definir un intervalo tal que f quede definida como :

$$\text{Si } x, y \in \mathbb{N} \Rightarrow f(x) = \begin{cases} x < y & '0' \\ x \geq y & '1' \end{cases} \tag{26}$$

Las cadenas generadas corresponden a la concatenación de símbolos dentro del conjunto $S = \{ '0', '1' \}$:

$$s_1 \cdot s_2 \cdot \dots \cdot s_n \tag{27}$$

tal que cada $s_i \in S$, donde n es el número de puntos muestreados.

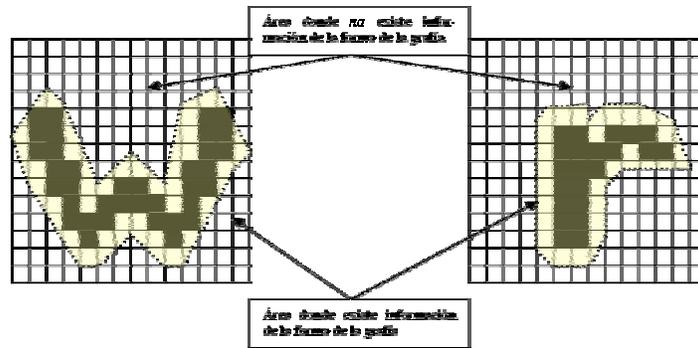


Figura 28. Ejemplos de la información y la no información en las grafías correspondientes a los caracteres de las letras “w” y “r” respectivamente

3.2.2 Codificación

La codificación es la forma de caracterizar el conjunto de información empleando cadenas simbólicas, para el caso particular de reconocimiento de caracteres es la manera de codificar en forma de cadenas simbólicas las características de un carácter. La codificación consiste en generar un conjunto de puntos aleatorios dentro de los rangos de la imagen monocromático, bajo la función de transformación f .

Cada imagen asociada a un carácter se representa como una matriz M de dimensiones $s \times t$, tal que para todo elemento $M_{i,j} \in \mathbb{N}$, el valor del color del punto j , i -ésimo se representa por el contenido de la posición $M_{i,j}$. Este valor es un número natural que representa el color de este punto.

Sea n ⁽⁷⁾, el número de puntos que se desean muestrear de la matriz M , entonces es necesario generar $2n$ puntos aleatorios para conformar las n posiciones a muestrear dentro de la matriz M de la forma (i, j) en el intervalo $(0,0) - (s, t)$, este conjunto de puntos es

⁷ Por lo general $n \ll s \times t$, sin embargo, algunas veces se puede tomar la imagen completa y generar una cadena simbólica, donde el proceso de muestreo puede ser definido como la sucesión de píxeles por línea de la imagen.

almacenado en una lista L en el orden en que son generados, denotado por “<”, por notación $L^<k$ representa el elemento k -ésimo de la lista L , donde el primer elemento es $L^<_1$ y por consecuencia el ultimo es $L^<_n$ ⁽⁸⁾. Para cada tupla contenida (i, j) en $L^<$, se obtiene el valor de la matriz M , aplicando la función de transformación, el resultado es concatenado con el resto de los símbolos de los puntos, quedando expresado como:

$$f(M_{L^<_1}) \cdot f(M_{L^<_2}) \cdot \dots \cdot f(M_{L^<_n}) \tag{28}$$

donde n es el numero de puntos contenidos en L .

En la **Figura 29**, se representa la imagen de un carácter que tiene asociada una matriz M , de dimensiones i, j . Las cadenas generadas a partir de M , representan a su vez un punto p en un espacio $\{0, 1\}^n$, donde n es la longitud de la cadena, es decir, el tamaño de la muestra.

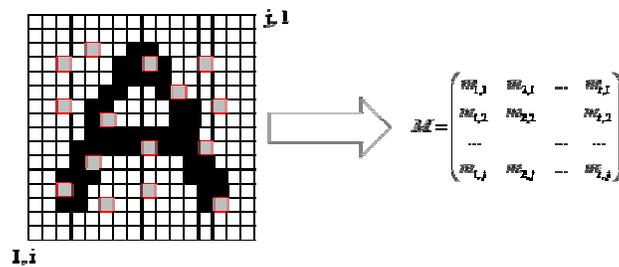


Figura 29. Representación de una imagen

Tomando como población inicial el conjunto de posiciones de la matriz M , que corresponden a los distintos puntos de la imagen, se escogen aleatoriamente un conjunto de n puntos de muestreo en M , de manera que bajo el orden en que son seleccionados, se construye una cadena de longitud n (ver **Figura 30**).

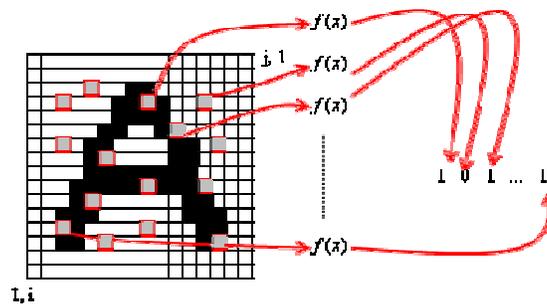


Figura 30. Proceso de generación de una cadena simbólica a partir de una imagen de un carácter.

En el caso de la **Figura 30**, $f(x)$ representa la función de transformación, que codifica un punto de la imagen en un símbolo. Si se emplean los mismos puntos de muestreo, se observa que ante pequeñas variaciones sobre el contenido de la imagen, la cadena simbólica generada sufre cambios mínimos con esta codificación. Los puntos que representan a la cadena c , de una imagen de un carácter M , y los puntos de otra ligeramente diferente, c' , tienen una distancia de Hamming en el espacio $\{0, 1\}^n$ muy pequeña y se dice que se trata del mismo carácter, caso opuesto, si las cadenas varían mucho, corresponden a caracteres

⁸ Consultar sección de optimización del muestreo.

diferentes sobre una misma lista de puntos (ver **Figura 31**), es decir $x, y \in \{0,1\}^n$ $d(x, y) \leq \delta$, donde δ es el umbral de similitud, que se discute más adelante [41]. Los puntos correspondientes a las variaciones de un carácter se encuentran dispersos en un círculo $O(p, r)$ de dimensión n con punto p , radio r y dimensión n (Ver **Figura 32**⁹).

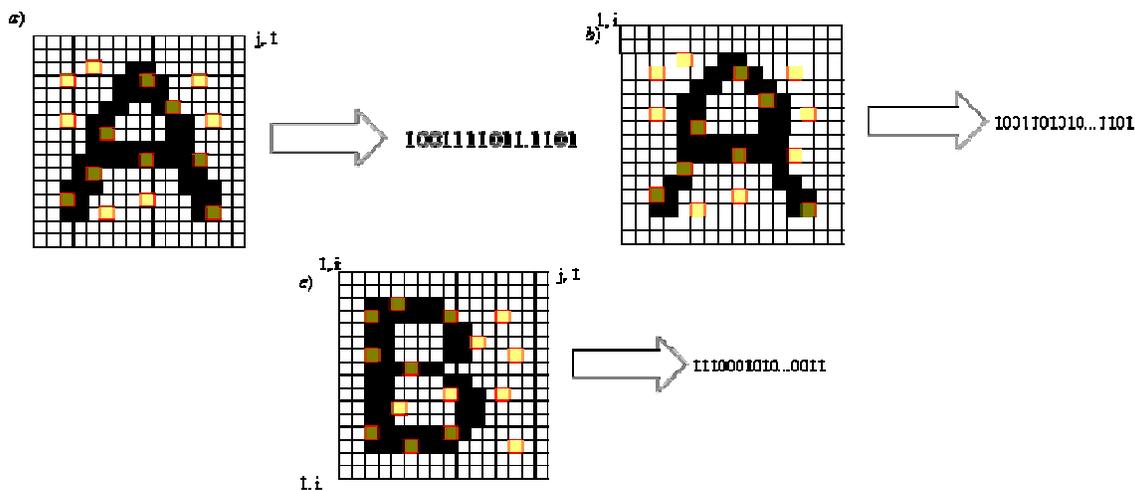


Figura 31. Ejemplos de cadenas generadas a partir de caracteres: ligeramente diferentes (a) y (b) y caracteres diferentes (c).

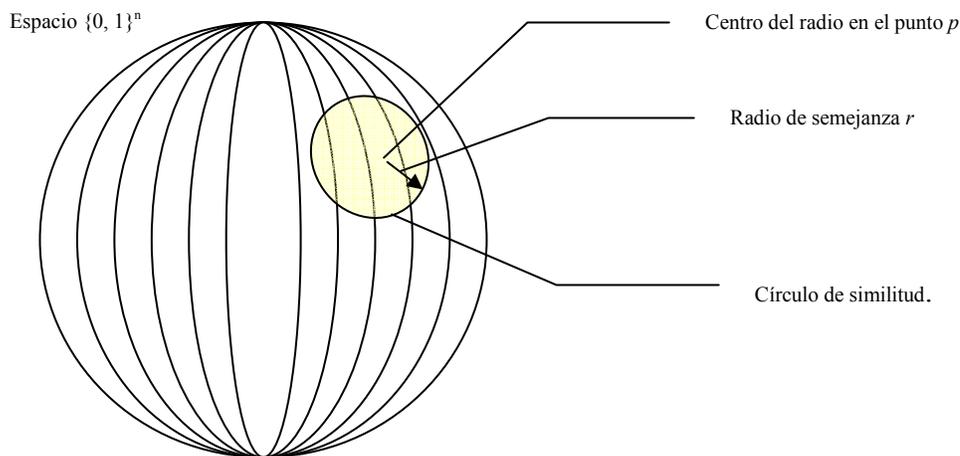


Figura 32. Distribución de los puntos de las cadenas generadas del muestreo a variantes de caracteres.

Una cadena representativa de un conjunto de cadenas correspondientes a las diferentes variaciones de un carácter es la cadena que representa el centro del círculo que circunscribe a todas las variantes y que tiene una distancia mínima a cada punto correspondiente de cada cadena generada. En el siguiente capítulo se muestra cómo encontrar este punto.

⁹ La representación es solo ilustrativa, no quiere decir que el espacio $\{0, 1\}^n$ tenga un aspecto de un espacio euclidiano de tres dimensiones

3.2.3 Cálculo de patrones

Un punto p que corresponde a una cadena significativa de un conjunto de puntos C , en el espacio $\{0, 1\}^n$, se le denomina patrón del conjunto C , siempre que la distancia de Hamming es aquella que tiene la menor distancia en promedio a cada elemento del conjunto C . Es decir si p , es el patrón de C se cumple que:

$$\text{Sea } C \subseteq \{0,1\}^n \text{ y } p \in \{0,1\}^n \text{ entonces } \forall_{c \in C} d(p,c) < \delta, \quad (29)$$

donde d es la distancia de Hamming y δ es la distancia promedio

En donde δ , es un umbral que para que sea el más significativo de C , tendrá que ser mínimo. Gráficamente es representado por el centro de la circunferencia que contiene a los puntos de las cadenas del conjunto C donde p es el centro de la circunferencia y el radio r representa a δ .

Se usa el concepto de *cadena patrón*, para denotar una cadena representativa que puede encontrarse por varios métodos.

Antes de describir la forma en que se obtiene una cadena se introduce el concepto de promedio, se toma como base para generar los patrones correspondientes.

Promedio Simple

El promedio simple de un conjunto $\{x_1, \dots, x_n\}$ de n observaciones dadas por una distribución se define como:

$$\text{promedio} \equiv \frac{1}{n} \sum_{k=1}^n x_k \quad (30)$$

El promedio es una función homogénea que tiene la propiedad de que un conjunto de números satisface la desigualdad:

$$\min(\{x_1, \dots, x_n\}) \leq \text{promedio} \leq \max(\{x_1, \dots, x_n\}) \quad (31)$$

El promedio como estimador, permite encontrar un valor significativo dentro de un conjunto que está dentro del intervalo del elemento más pequeño y el más grande del conjunto (31).

Una manera de obtener un punto significativo es mediante la construcción de un *promedio* sobre los puntos que representan a las posibles variaciones de un carácter, y en el caso más general si $n = ij$, (dimensiones de M) se encuentra el *carácter promedio* de un conjunto de caracteres C .

A continuación se describen dos modelos que permiten obtener un estimador representativo para el caso de cadenas simbólicas codificadas en $\{0, 1\}^n$.

Majority rule (MR)

Consiste en encontrar un punto p de un conjunto C a partir de la frecuencia en que ocurre alguno de los dos símbolos del modelo de representación; para encontrarlo, se toma cada elemento de C , y se calcula la frecuencia de cada símbolo ('0' y '1') que se tiene obteniendo un vector de frecuencias de la misma dimensión que los elementos en C . Se aplica una función de codificación a dos estados, en donde para valores mayores a $|C|/2$ se asigna un 1, caso contrario un 0. Por ejemplo, supóngase que se tiene el conjunto $C \subseteq \{0,1\}^5$ (Figura 33), donde al sumar todos los elementos se obtiene el vector s ,

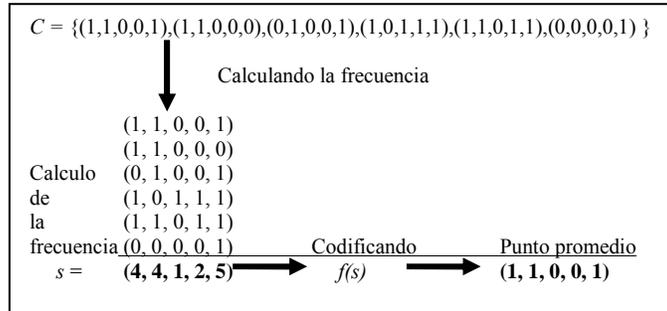


Figura 33. Ejemplo del cálculo del punto promedio por majority rule.

Sea un conjunto $C \subseteq \{0,1\}^n$ y $my \in \{0,1\}^n$, donde el punto significativo determinado por *Majority Rule* esta definido como [10]:

$$\forall_{i \in 1 \dots n} my_i = \begin{cases} 1 \text{ sii } \sum_{c \in C} c_i > \frac{|C|}{2} \\ 0 \text{ sii } \sum_{c \in C} c_i \leq \frac{|C|}{2} \end{cases} \quad (32)$$

donde my_i representa la componente i ésima del punto my y $|C|$ es la cardinalidad de C .

Lecturas aleatorias sucesivas (LA)

Otro método para el cálculo de un punto significativo, consiste en incrementar la información contenida en C auxiliándonos de un multiconjunto. El proceso consiste en lo siguiente:

Del conjunto C se define un número $l \in \mathbb{N}$ que puede ser $|C| \neq l$, y un multiconjunto C' , de manera que se tome al azar un elemento $c \in C$ y se añade a $C' \uplus [c]$, repitiendo sucesivamente hasta tomar l elementos. Obsérvese que más de algún elemento de C , estará contenido más de una vez dentro de C' , entonces, con C' , se calcula el punto significativo por *Majority Rule*.

Formalmente se tiene:

$$C' = \bigoplus_{c \in C} [c]$$

donde c , es cualquier elemento de C seleccionado al azar, repitiendo el proceso l veces. (33)

una vez generado C' por la ecuación (32), se calcula el punto significativo por MR .

Ambos métodos ofrecen una forma de encontrar un punto significativo (patrón) de un conjunto de puntos C .

El conjunto de puntos C representa las posibles variantes de un carácter, la cadena patrón es el punto que tiene una distancia cercana a cada elemento de C , si las variaciones en C , son lo suficientemente significativas a las posibles variaciones de un carácter en particular, entonces el punto significativo o la cadena que representa el punto, es significativa a un grupo mayor de variaciones de un carácter en particular, que no necesariamente han sido consideradas en el análisis; si p es el punto significativo al conjunto C , el círculo $O(p, \delta)$ (ver la **Figura 34**), corresponde a todos aquellos puntos que son parecidos a p , donde p es el centro y δ es el radio de cercanía.

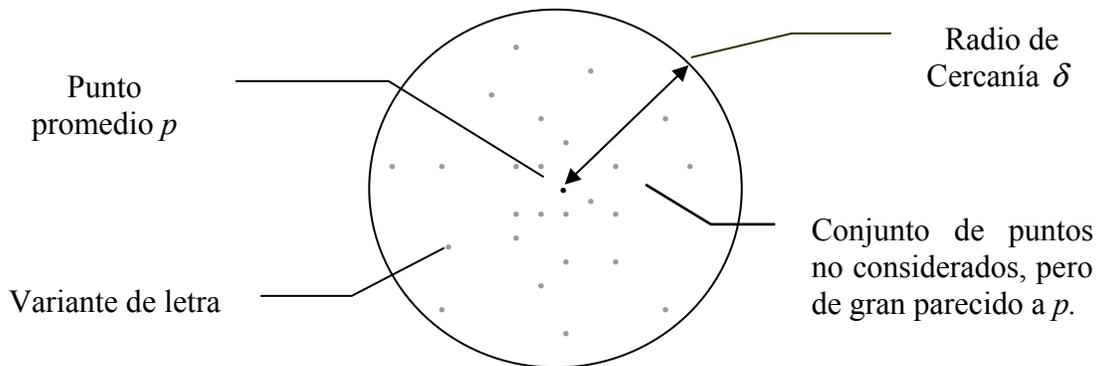


Figura 34. Esquema del radio de cercanía, y el conjunto de puntos correspondientes a caracteres muy parecidas a p .

Si se consideran todos los puntos de la imagen que representan a un carácter para codificarla como una cadena simbólica sobre un orden preestablecido se puede aplicar ambos métodos para encontrar una *cadena patrón* de las distintas *variaciones* de los caracteres¹⁰, obteniendo esquemáticamente lo que sería *un carácter significativo* al conjunto de variantes de caracteres, en este caso, la función de muestreo se cambia por la codificación en orden de todos los puntos de la imagen.

Si se tiene un conjunto de imágenes que contienen un carácter en particular (C), tras calcular el *carácter patrón*, por ambos métodos se obtienen los resultados que se muestran en la **Figura 35**. En el caso del método LA , se ha utilizado $l=200$, los ejemplos del capítulo usan el conjunto de caracteres uno, mostrado en el Apéndice. Variantes de Caracteres de Letras.

¹⁰ La construcción de las cadenas simbólicas emplean el total de puntos de los caracteres, en algunos procesos. descritos.

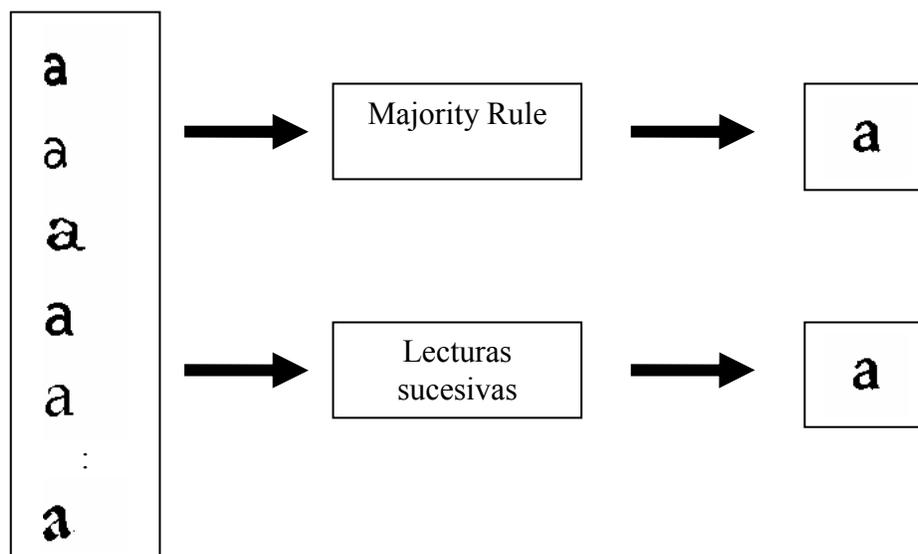


Figura 35. Ejemplo de caracteres promedio por MR y LA.

Obsérvese que en el caso de LA, la imagen correspondiente al carácter (en este caso el carácter *a*) tiene ligeramente más información que con *MR*. Estas diferencias pequeñas son las que sirven de base para establecer un método de optimización de muestreo, que es el tema a tratar en la sección siguiente.

3.2.4 Optimización del muestreo

La representación simbólica empleada para la codificación depende directamente de la calidad del Método de Muestreo, si se tiene un método pobre, la información codificada también lo es, así la búsqueda de un método que garantice la significatividad del muestreo permite que no se generen cadenas iguales o con ligeras diferencias, repercutiendo en la capacidad de eficiencia del reconocimiento.

Un método eficiente de optimización del muestreo probabilístico debe distribuir de manera uniforme la información de cada carácter [49]. El uso de diferentes maneras de encontrar una cadena significativa a partir de un conjunto de caracteres codificados denotados por *C*, genera ligeras variaciones en la *cadena resultante* (**Figura 33**), estas variaciones se toman en consideración para obtener una mejor optimización del conjunto de puntos a muestrear. La dificultad está en el número de caracteres a reconocer. En este trabajo en particular, se toman 26, que son los caracteres de las letras significativas de un alfabeto (mayúsculas o minúsculas).

Análisis de la distribución de los puntos

El muestreo es un muestreo aleatorio que consiste en generar una lista *L* de longitud *n*, donde *n* representa el número de puntos a muestrear de las imágenes, sin embargo, el uso de puntos con una distribución aleatoria uniforme no siempre garantiza encontrar los puntos óptimos para muestrear a las imágenes, se pueden dar casos en que algunos puntos resulten

repetidos, o las partes muestreadas dentro de la imagen no ayuden para discriminar a un carácter. La definición de un método de optimización de muestreo garantiza, en la mayoría de los casos, que se tenga un muestreo eficiente.

La primera parte del proceso consiste en adaptar la distribución de los puntos relevantes. Supóngase que se generan los n puntos, en su mayoría en áreas donde siempre permanece constante el color, estos puntos no tienen información que ayude en la tarea de reconocimiento. En la **Figura 36** se muestra un patrón correspondiente al carácter de la letra “a” (encontrado por alguno de los métodos descritos en la sección anterior). El análisis de los patrones ayuda a no manejar el total de los caracteres, los puntos significativos a muestrear dentro de un patrón p tienen que ser *adecuados* para muestrear todo el conjunto C .

Para un conjunto de caracteres distintos¹¹ $\{\Sigma_1, \dots, \Sigma_n\}$, se construyen los conjuntos C_a, \dots, C_z , que representan a los conjuntos de posibles variantes de cada carácter, denotando como C_i al conjunto de variantes del i -ésimo carácter; de cada conjunto C_i , se puede calcular el patrón p_i con alguno de los métodos descritos. El conjunto de $\{p_a, \dots, p_z\}$ es el conjunto de patrones distintivos de cada carácter que sirve para distribuir de manera *adecuada* los puntos a muestrear. Por cuestiones del análisis y para lograr la optimización de la muestra, se toman todos los puntos de las imágenes, cada punto representa una imagen completa y el patrón promedio p_i , es una imagen promedio al conjunto de variantes C_i .

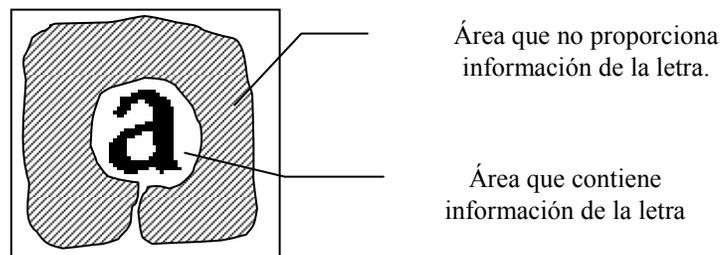


Figura 36. Áreas donde es posible extraer información acerca de cada carácter.

Si cada patrón p_i representa a un carácter en particular, los puntos con 1's en p_i son los puntos significativos al carácter i -ésimo, al empalmar 2 caracteres (utilizando el operador \cap lógico), se tienen los puntos que en conjunto utilizan ambas; los puntos comunes a ambos caracteres se encuentran mediante un \cap lógico de las imágenes; la información usada, sin considerar la información común, es la resta de ambos. En la **Figura 37**, se muestra un ejemplo¹² con los patrones p_a y p_b ; el resultando en c) es una imagen que contiene los puntos significativos a ambas imágenes sin considerar aquellos puntos comunes.

El análisis de puntos a muestrear para ambos caracteres, tiene que tomar en consideración el resultado c) de la **Figura 37**.

¹¹ Se consideran en este caso alfabetos completos, sin embargo, en la práctica pueden ser sólo algunas letras de cada alfabeto.

¹² El grupo de letras empleado para esquematizar el proceso de optimización es el grupo 1 con 80 variantes.

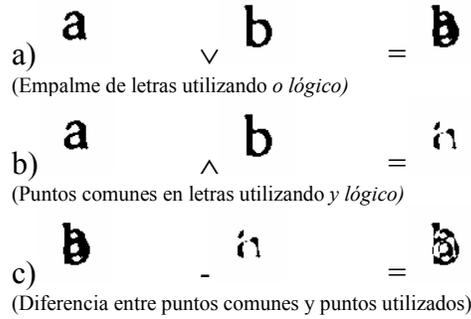


Figura 37. Operaciones de a) empalme b) puntos comunes y c) diferencia entre los patrones.

Para generalizar el proceso con todo el alfabeto (26 letras distintas) se procede de la siguiente forma:

Sean p_a, p_b, \dots, p_z los patrones de los conjuntos C_a, C_b, \dots, C_z que contiene a las variantes del carácter correspondiente al carácter i -ésimo, los puntos utilizados están determinados por la expresión (34).

$$p_a \vee p_b \vee \dots \vee p_z \quad (34)$$

Donde el resultado es un punto (gráficamente una imagen) que contiene el símbolo ‘1’s si el punto es usado por al menos un carácter.

En el caso de que existan puntos comunes al conjunto de patrones p_i 's, se obtiene por la expresión (35).

$$p_a \wedge p_b \wedge \dots \wedge p_z \quad (35)$$

Por último, los puntos que representan a los utilizados por los patrones p_i 's sin considerar a los comunes en ambos están definidos con la resta de las ecuaciones (34) a (35), y queda expresado en (36).

$$p' = (p_a \vee p_b \vee \dots \vee p_z) - (p_a \wedge p_b \wedge \dots \wedge p_z) \quad (36)$$

Entonces p' , contiene a los puntos útiles para definir a L ; se puede considerar a p' como un metapatrón, sobre los patrones p_i 's de cada carácter, el metapatrón en particular, muestra información acerca de qué puntos son usados por los distintos caracteres y sus variantes.

Sea n , el número de puntos que se desean muestrear de la matriz M , que representa la imagen contenida en p' . Es necesario generar $2n$ puntos aleatorios, para conformar las n posiciones a muestrear dentro de la matriz M de la forma (i, j) del intervalo $(0, 0) - (p, q)$, este conjunto de puntos se almacena en una lista L bajo un orden $<$ de acuerdo a como son generados. La notación $L^<_k$ representa el elemento k -ésimo de la lista L , donde el primer elemento es $L^<_1$ y por consecuencia el ultimo será $L^<_n$.

De L se puede generar una cadena de p' que representa un punto l en el espacio $\{0,1\}^n$. El metapatrón p' representa los puntos que se utilizan y los que no sin repetición en las diferentes grafías, entonces la cadena generada de p' es una cadena significativa sobre la

información empleada por todos los patrones de los caracteres. Si los puntos de L , están distribuidos adecuadamente sobre este metapatrón, entonces la cadena generada a partir de p' utilizando L debe tener una norma l aproximadamente igual a $l \approx \frac{n}{2}$. En la **Figura 38** se esquematiza el proceso para el alfabeto completo para un conjunto de caracteres¹³:

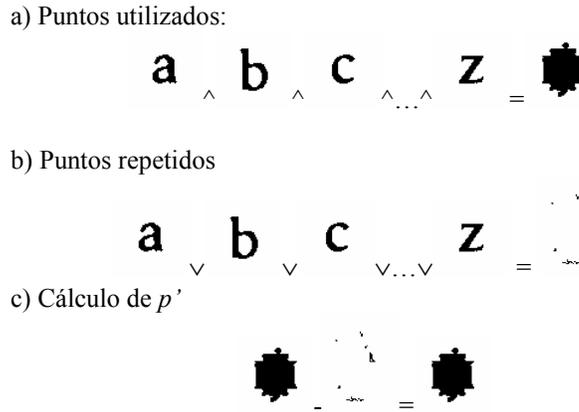


Figura 38. Ejemplo de la obtención de p' , para optimizar el muestreo

El pseudo código queda expresado de la siguiente manera [49]:

1. Sea l_i , tal que $l_i \in L$ tomado al azar de manera que $f(M_{l_i})$ sea 0 si $|cl| < n/2$ ó 1 si $|cl| > n/2$.
2. Generar puntos aleatorios de la forma (i, j) del intervalo $(0,0) - (m, n)$,
3. calcular p'
4. si $|p'| \approx n/2$ terminar
5. si $|p'| < n/2$ encontrar dos valores i, j no repetidos de manera que $f(M_{(i,j)}) = 1$ e ir a 3.
6. encontrar dos valores i, j no repetidos de manera que $f(M_{(i,j)}) = 0$ e ir a 3.

Esquemáticamente el proceso de balanceo se presenta en la **Figura 39**. El proceso de balanceo consiste en seleccionar un punto de manera aleatoria de la lista de puntos de muestreo, basado en el siguiente criterio: si la norma de una cadena c generada por L en p' , tiene $|c| > n/2$, se busca al azar puntos no utilizadas, caso contrario se buscan puntos utilizados dentro de p' . Este proceso se repite sucesivamente hasta que la cadena c tiene una norma aproximadamente a $n/2$.

Análisis de la superposición de puntos

Seleccionar los puntos de muestreo sobre la imagen resultante de p' ayuda a distribuir los puntos de muestreo de manera más eficiente, sin embargo, hay situaciones en que no necesariamente un punto se localiza en todas las p_i 's, pero sí en una gran mayoría de los caracteres, es decir, en p' existen algunos puntos pintados en las imágenes de algunas p_i 's

¹³ Para este ejemplo se toma el primer grupo de caracteres del apéndice.

superpuestas, consecuencia de que los patrones son subconjuntos (ver **Figura 41**) de otros o gran parte del patrón se encuentre superpuesto dentro de otro (ver **Figura 40**); en consecuencia que no se puede determinar de forma adecuada un radio de cercanía para identificar a cada conjunto posible de variantes de caracteres, generando solapamientos o inclusiones de cada uno de los círculos $O(p_i, r)$ y generan falsos reconocimientos para un conjunto particular de puntos dados. En la **Figura 40** se muestran casos particulares para los caracteres e , b y c , donde la parte inferior es común a ambas (la parte remarcada en cada carácter), los puntos muestreados en esta área no permiten clasificar a cada carácter. En la **Figura 41**, se muestra el caso más extremo, en donde un carácter puede estar contenido en otro, de manera que si se toman los puntos dentro del rectángulo indicado para ambos caracteres resulta imposible distinguir entre el carácter i y el carácter j .

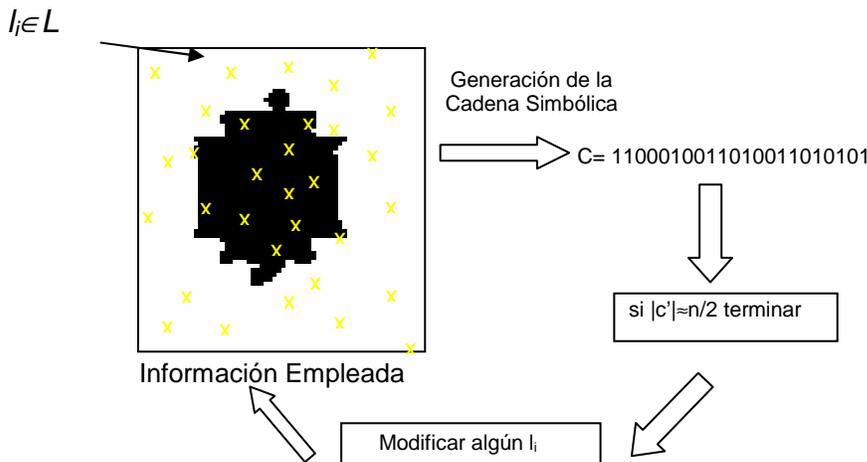


Figura 39. Proceso de balanceo de la lista L.

El problema de superposición no es exclusivo de casos particulares, una gran cantidad de puntos de la imagen p^{i4} muestran superposición con un número considerable de caracteres. En la **Figura 42**, se muestra la frecuencia de superposición al empalmar los patrones p_i 's. Hay puntos utilizados aproximadamente por 16 caracteres, entonces, estos puntos no resultan útiles para la extracción de información, como los puntos en donde la frecuencia mucho menor. Los puntos utilizados por una gran cantidad de patrones p_i , tienden a una frecuencia máxima de 26 (número de caracteres existentes), mientras que los puntos son utilizados por pocos patrones p_i , tienden a 0.

La selección de puntos debe hacerse en áreas intermedias, descartando áreas de alta o baja frecuencia; todo elemento de L debe respetar este criterio. Se define una matriz $Mp^{(15)}$, que contiene la frecuencia de aparición del símbolo '1' de las imágenes patrón p_i . La **Figura 43** muestra las curvas de nivel de la **Figura 42**, donde se muestra el área con mayor frecuencia

¹⁴ Recordar que p^i es un punto en $\{0,1\}^n$, cada componente de p^i representa un punto de una imagen, en este caso, de la imagen del empalme de todos los p^i 's, considerando a $n = p \times q$ codificando a la imagen completa como una cadena simbólica.

¹⁵ La **Figura 42** y la **Figura 43** son generadas a partir de Mp

de los símbolos '1' en los patrones p_i , las regiones delimitadas por las curvas de nivel con altas o bajas frecuencias se omitirán para el muestreo.

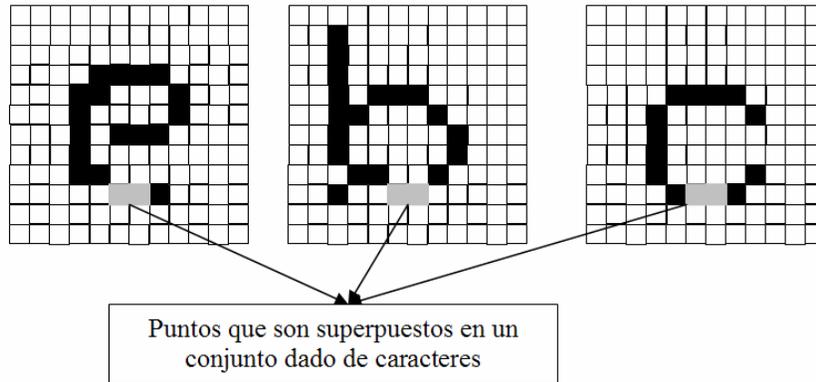


Figura 40. Áreas con información común en más de un carácter.

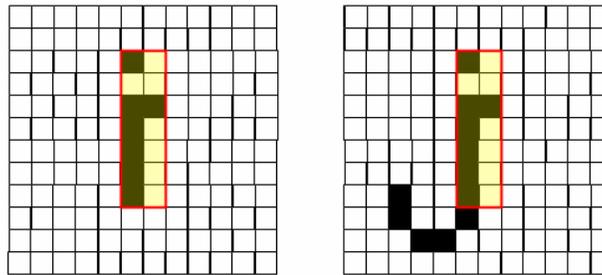


Figura 41. Caracteres contenidos por otros caracteres.

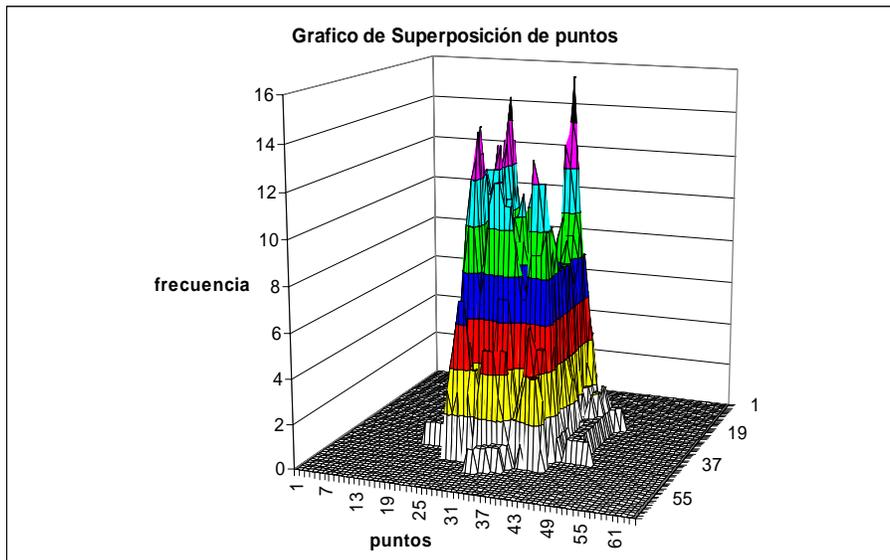


Figura 42. Gráfica de la frecuencia de superposición en un conjunto de patrones p 's.

Para verificar la lista de puntos a muestrear se recorre cada elemento de L , comparando con respecto a Mp , si $Mp(i,j) < Umbral$ donde $(i, j) \in L$, en donde $Umbral$ es un valor entero, que permite definir si el punto es útil o no.

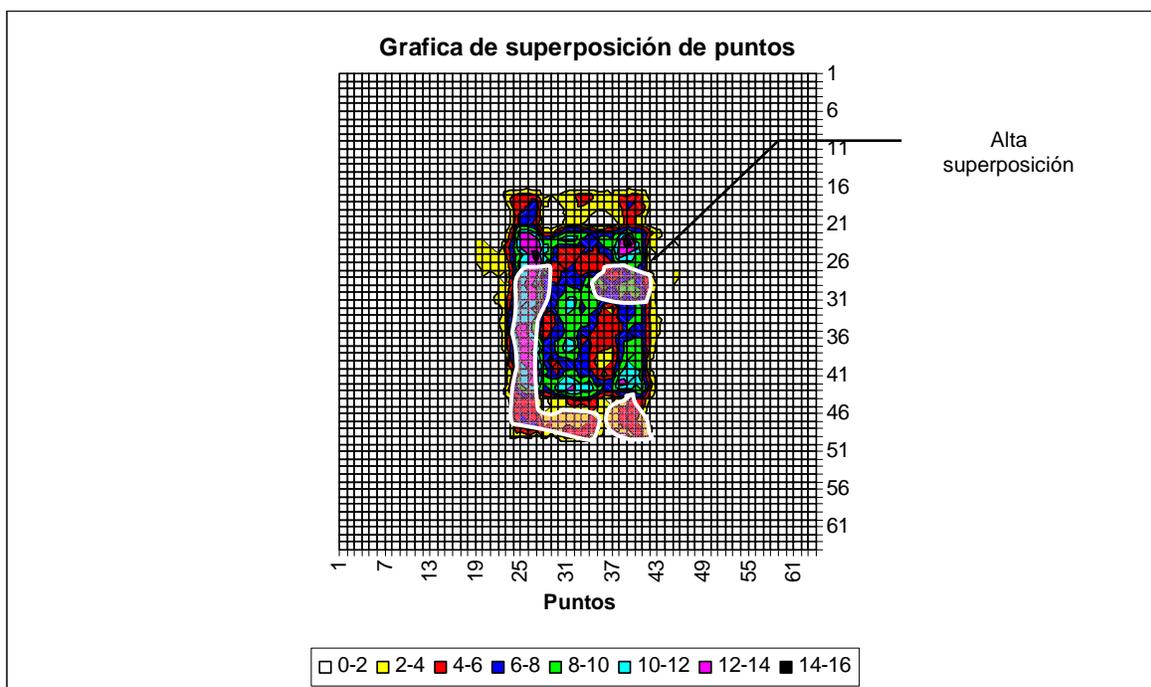


Figura 43. Curvas de nivel de diferentes grados de superposición de puntos en las imágenes de los patrones p_i .

El valor de *umbral* se define como 13, es decir, el número máximo de elementos que pueden estar superpuestos (en este caso 26) dividido entre 2. Se garantiza como máximo que los puntos pueden participar en 13 caracteres, si se consideran caracteres muy parecidos, este *umbral* puede tener valores más pequeños a 13⁽¹⁶⁾, los valores con un *umbral* demasiado pequeño, generan patrones p_i , no representativos para cada carácter. El límite inferior también tiene que estar acotado, si existen puntos etiquetados con 1 que tengan una frecuencia muy cercana a 0, no ofrecen información suficiente, si consideramos el análisis de superposición como aquellos etiquetados en 0, muestra regiones que son empleados por una gran cantidad de patrones con el símbolo 0; haciendo un razonamiento análogo se puede definir un *umbral* inferior mínimo. El *umbral* mínimo ideal tiende a ser cercano a 13, pero el área ocupada por los puntos de caracteres no es exactamente $\frac{1}{2}$ del área de la imagen, por esto se considera un límite cercano a la proporción del área ocupada por el empalme de todas los caracteres, una es $\frac{1}{4}$ proporción del el área ocupada por 1 con respecto al área total de la imagen; una aproximación al límite inferior se define por aproximadamente $\frac{1}{4}$ *FrecuenciaMaxima*, de elementos que pueden ser superpuestos.

Una vez que todos los puntos se encuentran dentro del umbral máximo y mínimo L , contiene a un conjunto de puntos útil para muestrear las imágenes. El pseudocódigo se muestra a continuación [49]:

1. Mientras que $\forall l \in L \wedge M_{p_l} \leq 13$ ir a 2 en caso contrario terminar

¹⁶ Considerando a la mitad del total de puntos que pueden estar empalmados.

2. Encontrar el $l \in L$, tal que : $\max(Mp_l)$
3. generar un par de números aleatorios (i, j)
4. Si $(Mp_{i,j} \leq Mp_l \vee Mp_{i,j} \geq \frac{1}{4} \text{Frecuenciamaxima}) \wedge Mp_{i,j} \neq 0$ sustituir en L a l , por (i, j) y continuar, o si no ir a 3
5. Fin mientras

Al terminar la ejecución del algoritmo, L ya no tendrá símbolos con alta o baja frecuencia de superposición, además que se respeta el balanceo de los puntos en la condición en 4, al no permitir puntos $Mp_{i,j} = 0$, que representan los 0's en la imagen generada de p' .

Se calculan los puntos p_i 's, de longitud n , a partir de los puntos en L , en donde éstos serán los patrones p_i 's de longitud n , que identifican a cada grupo, donde cada C_i contiene las variaciones posibles de algún carácter.

Majority rule modificado (MRM)

Tomando como base el análisis de la frecuencia de los símbolos, se propone una variación a MR para encontrar una p_i significativa a un conjunto C_i , debido a que algunas veces, dependiendo del número de caracteres, al calcular p_i con MR existen áreas con una alta frecuencia, que dificultan encontrar áreas con frecuencias útiles para definir L .

Las variantes de un carácter se diferencian en un porcentaje muy bajo, el empalme de las distintas variantes de un carácter tiene una frecuencia de superposición muy alta, es decir, como las variaciones entre cada versión son mínimas, la gran mayoría de los puntos se mantiene en todas los caracteres que conforman a un C_i .

Una modificación a MR consiste en variar el criterio para asignar el símbolo '1' o '0'. El criterio originalmente está definido por la frecuencia máxima dividida entre dos, es decir, si un elemento (punto), tiene una frecuencia mayor a la mitad de la frecuencia máxima se considera como 1, pero al tratarse de variaciones de un mismo carácter, la frecuencia de ocurrencia de un punto dado en todas las variantes es grande y mayor a un medio de la frecuencia máxima, la definición de un umbral mayor identifica aquellos puntos con una frecuencia mayor, reduciendo el área de puntos utilizados.

La **Figura 44**, muestra la gráfica de la frecuencia al empalmar todas las variantes del carácter "a", obsérvese que el cálculo de la cadena patrón con MR , genera el patrón delimitado entre el color de la segunda y tercera curva de nivel. La cantidad de píxeles es significativamente mayor que aquellos que tienen una frecuencia mayor o igual a 60, en donde además son más significativos (por estar en una mayor cantidad de variantes del carácter), evitando la saturación al empalmar todos los p_i 's, de cada carácter. Al disminuir el número de puntos, se garantiza que el hacer al empalme de todas los caracteres para la construcción de p' , no se saturan, logrando tener puntos más significativos para cada carácter.

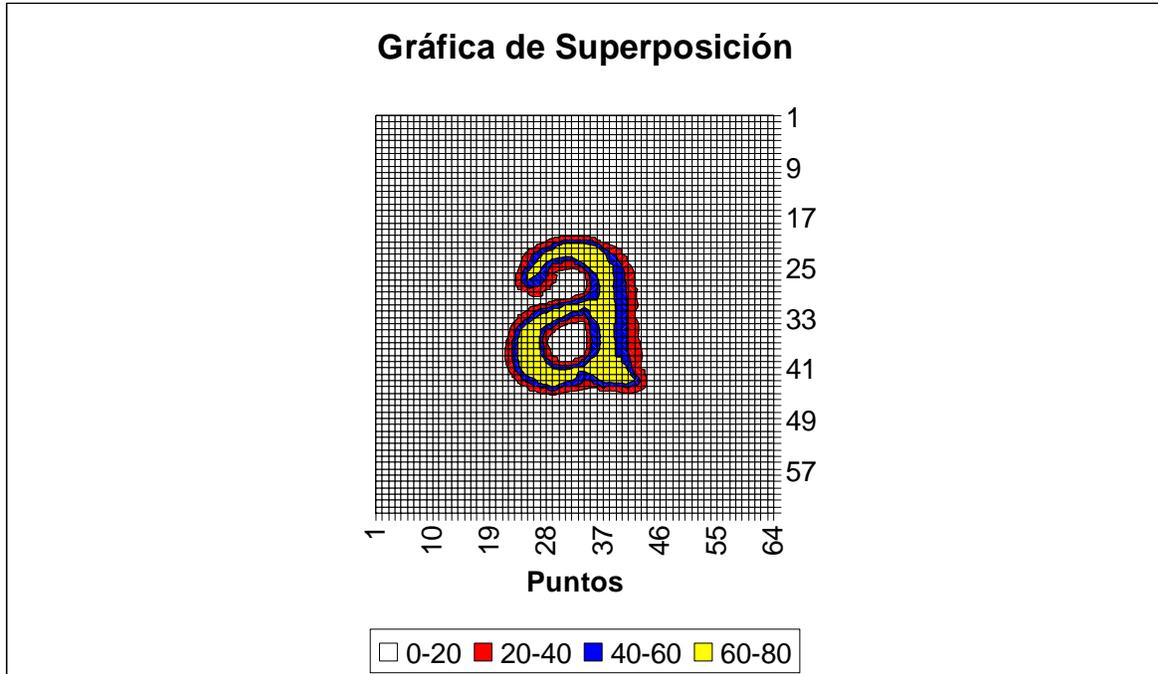


Figura 44. Gráfica de curvas de nivel de la frecuencia de los puntos de los caracteres “a”

El criterio para asignar uno de los posibles símbolos queda abierto a cualquier número dentro del intervalo $[0, |C|]$; notacionalmente queda expresado como:

$$\forall_{i \in 1 \dots n} my_i = \begin{cases} 1 & \text{sii } \sum_{\forall c \in C} c_i > u \\ 0 & \text{sii } \sum_{\forall c \in C} c_i \leq u \end{cases}; \quad (37)$$

donde my_i representa la componente i ésima del punto my y $u \in [0, |C|]$.

el umbral u , se define a partir de la frecuencia de superposición que existe en la variante de los caracteres; si usamos caracteres con una frecuencia grande, es común definir a u , como $\frac{3}{4}|C|$.

Cálculo del radio de similitud

En $\{0, 1\}^n$, la distribución de los puntos es una distribución binomial con media $n/2$, donde se considera que dos puntos a una distancia $n/2$ son ortogonales y el umbral para decidir cuando dejan de ser parecidos uno con otro es $n/4$, en el caso de reconocimiento de caracteres, por la manera de distribuir los puntos, la ortogonalidad entre cada p_i , no necesariamente esta a una distancia $n/2$, donde es necesario explicitar un radio de similitud que permita decidir cuando dos puntos son similares, a esto se le conoce como radio de *similitud* entre los puntos contenidos en C_i , representados por p_i ; el radio de cercanía, es un escalar que representa la distancia de Hamming máxima a la que un punto puede ser considerado igual o no a un patrón p_i . Idealmente, el conjunto C_i , está contenido en el elemento con máxima distancia en C_i , al patrón p_i , el cual se denota como radio de similitud óptimo, sin embargo, si la dispersión de los puntos contenidos en C_i es grande, una buena aproximación es la distancia media más la desviación estándar, que contendrá el mayor

porcentaje de puntos que son similares a p_i ; un radio demasiado grande, tiende a generar más solapamientos entre los círculos generados en el espacio $\{0, 1\}^n$, y el criterio de decisión se dificulta.

Entonces, el radio de cercanía esta definido como:

Sea p_i , el patron i -ésimo, el radio de cercanía es :

$$r = \max \{ |d(p_i, \forall c \in C_i)| \} \quad (38)$$

y una aproximación adecuada es :

$$r = \overline{|d(p_i, \forall c \in C_i)|} + \sigma(\{|d(p_i, \forall c \in C_i)|\}) \quad (39)$$

donde $|d(p_i, C_i)|$, es la norma de la distancia media entre los elementos de C_i al patron p_i y $\sigma(\{|d(p_i, \forall c \in C_i)|\})$ es la desviación estandar.

en este caso $d(p_i, \forall c \in C_i)$, es la distancia de Hamming, que ha sido definida como nuestra métrica de similitud.

Sea entonces, x_i el punto asociado a la cadena p_i , el círculo que contiene a los elementos más parecidos a x_i , esta denotado por:

$$O(x_i, r_i) \quad (40)$$

Esquemáticamente los puntos de las variantes de los caracteres (incluyendo a p_i) están dispersos como nubes en el espacio $\{0, 1\}^n$, el círculo de similitud es la distancia (tomando como referencia a p_i) que abarque el número mayor de variantes de los caracteres codificados como puntos (ver **Figura 45**). El conjunto de todos los radios de similitud de cada C_i , se denota por R . El radio de cercanía permite determinar el conjunto de puntos similares a un punto p_i , pero en la gran mayoría de las situaciones solo es necesario encontrar el p_i con mínima distancia para decidir a que elemento es más parecido; es decir dado $e \in \{0, 1\}^n$ tiene más parecido al elemento $\min(\{d(e, p_i)\})$, por notación se identificará con P , al conjunto de patrones o puntos representativos $P = \{p_a, \dots, p_z\}$.

3.3. Discusión

La manera de codificar la información de un problema, determina las capacidades de operación y manipulación, permitiendo facilitar o limitar la tarea de reconocimiento de caracteres. El uso de una representación simbólica no aritmética de dos estados ofrece un modo de caracterizar la información en estructuras que ayudan a proponer una solución al reconocimiento de caracteres, diferente a los modelos aritméticos tradicionales.

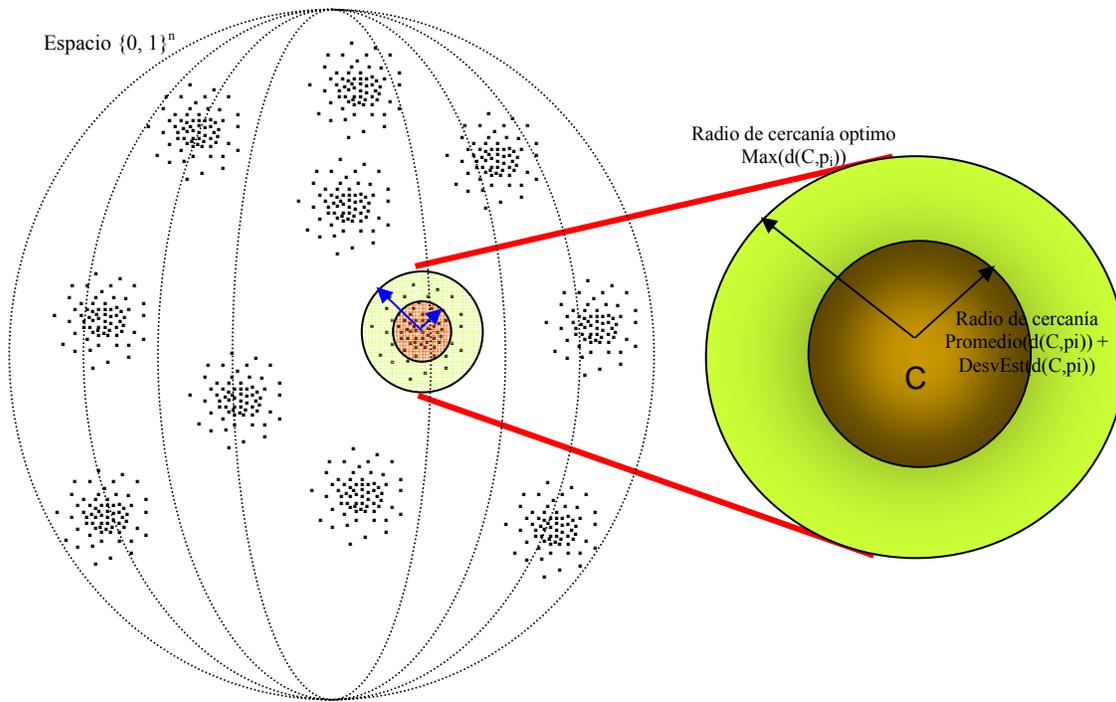


Figura 45. Cálculo del radio de cercanía

Existen varios métodos para obtener la información significativa que caracteriza algún fenómeno; el uso de métodos aleatorios es un modelo práctico por que debido a sus propiedades ayuda a caracterizar la información de una manera más compacta. En el caso particular de reconocimiento de caracteres, los métodos aleatorios ofrecen modelos para encontrar la información, pero es necesario hacer análisis más particulares, optimizando la búsqueda de la información a emplear. Los métodos aleatorios junto con una representación simbólica permiten codificar en una cadena la información de los caracteres expresados como cadenas [4, 5, 49].

A lo largo del capítulo una forma de caracterizar la información de los caracteres es a partir de un muestreo aleatorio y un método para optimizar el muestreo. De cada imagen de un carácter se genera una cadena simbólica, las cadenas son agrupadas en conjuntos de variantes de cada carácter; también se han mostrado tres maneras para encontrar una cadena significativa a un conjunto de variantes de caracteres: *MR*, *LA* y *MRM*.

Una de las ventajas que se tiene al emplear un modelo de muestreo aleatorio y una codificación simbólica, es el tener un modelo de reconocimiento en donde no es necesario tener la totalidad de información. La caracterización de la información como *una sola cadena*, permite representar a *un carácter* como *una cadena*, en donde de acuerdo a los criterios de codificación puede representar a la imagen completa sin la necesidad de tener *un conjunto* de descriptores de *características* específicas sobre el carácter que se esta codificando, ofreciendo alternativas distintas como la de suponer *a priori* los descriptores y el número de estos que son necesarios para poder representar a un carácter.

Este proceso de muestreo, junto con los modelos de optimización y los métodos para encontrar cadenas patrones, permite definir las bases para la construcción de un método de reconocimiento de caracteres.

4 Método propuesto de reconocimiento de caracteres

Al emplear maneras distintas de representar y codificar la información se pueden construir métodos diferentes para resolver problemas particulares, como es el caso de reconocimiento de caracteres. En el capítulo anterior se mostró cómo combinando un modelo de representación simbólico y un modelo de muestreo se pueden establecer las bases suficientes para definir un método de reconocimiento de caracteres; en este capítulo se presenta una propuesta de método para el reconocimiento de caracteres basado en una codificación por un muestreo aleatorio y donde la información se representa como cadenas simbólicas de longitud definida. Para evaluar el método propuesto se ha desarrollado un modelo experimental, donde se mide el porcentaje de reconocimiento, al reconocer un conjunto de diferentes tipos de caracteres.

4.1 Descripción del método propuesto

La representación de la información de los diferentes caracteres se hace mediante cadenas simbólicas de longitud definida. El uso de cadenas simbólicas ofrece diferentes posibilidades de realizar cómputo en comparación a una representación aritmética. Este modelo procesa y opera información en forma de cadenas. Cada cadena se puede representar como un vector, donde cada componente del vector, representa a cada símbolo que contiene la cadena.

Las operaciones, entre las cadenas, al ser consideradas como vectores, se hacen en forma paralela. Un modelo de máquina de Turing no permite ejecución de operaciones paralelas de forma directa. Una implementación en una máquina de Turing [8, 13, 17, 41] emula el comportamiento de procesos paralelos en donde se puede evaluar el funcionamiento de dichos métodos.

En este capítulo se expone una propuesta de método general de reconocimiento de caracteres en base al análisis hecho en el capítulo anterior. La evaluación del método propuesto consiste en medir el porcentaje de aciertos al reconocer un conjunto de distintos tipos de letras con distintas variantes de cada carácter y ajustando los distintos parámetros del método. Este método está formado por dos etapas:

- a) Cálculo de los patrones representativos.
- b) Fase de reconocimiento.

La etapa del *cálculo de los patrones representativos* permite encontrar cadenas significativas, que representan a las distintas letras del abecedario. Esta etapa parte de un conjunto de caracteres, que se agrupan por variantes de cada carácter (C_i), donde a partir de

cada C_i se encuentra la cadena más significativa (p_i); esta etapa requiere mayor poder computacional por los cálculos necesarios para encontrar a cada p_i .

La segunda etapa consiste en identificar a una letra por su parecido, al codificarla como una cadena simbólica, a alguna de las cadenas p_i calculadas en la etapa anterior; en esta etapa se elabora un conjunto de pruebas para evaluar el desempeño basado en el porcentaje de aciertos al clasificar distintos tipos de caracteres.

El método propuesto depende de un conjunto de parámetros, los cuales permiten aumentar la eficiencia del reconocimiento según los tipos de letras que se desean reconocer. A lo largo de esta sección se presenta el pseudocódigo para realizar el reconocimiento de caracteres, donde se explican los parámetros para su funcionamiento. Al final se hace un análisis de la complejidad del algoritmo y las correspondientes pruebas para su validación.

4.1.1 Cálculo de los patrones representativos

El cálculo de los p_i 's, requiere computacionalmente más recursos, estos recursos dependen de un conjunto inicial de diferentes tipos de caracteres. El proceso de construcción de p' se ilustra en la **Figura 46**¹⁷, en la que para cada etapa se detalla el pseudocódigo necesario.

Dados un conjunto de abecedarios (tipos de letra) $\{\Sigma_1, \dots, \Sigma_n\}$ (¹⁸), representados como imágenes de tamaño fijo $p \times q$; se agrupan en conjuntos de las diferentes variaciones de un mismo carácter C_a, \dots, C_z , donde para cada C_i se procede a encontrar un patrón representativo, p_i .

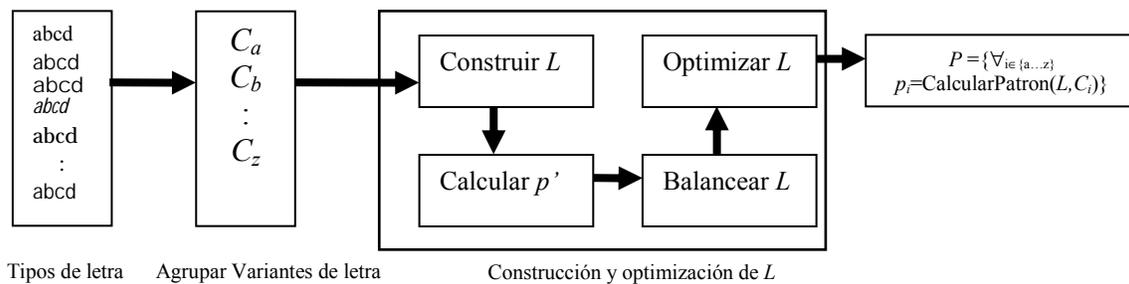


Figura 46. Esquema del proceso de la etapa del cálculo de patrones

Construcción de L

El método propuesto toma inicialmente posiciones aleatorias para definir los puntos de muestreo dentro de los rangos de las imágenes en los distintos caracteres. En esta etapa la generación de puntos de muestreo no está restringida, se consideran números aleatorios con una distribución uniforme. La generación de puntos aleatorios está dada por el **Código 1**.

Función ListaAleatoria
Parámetros de entrada:

¹⁷ El cálculo de p' , es por alguno de los métodos mencionados para el cálculo de un punto representativo.

¹⁸ Los alfabetos no necesariamente contienen todos los símbolos del abecedario en conjunto los alfabetos contienen o solo símbolos para letras mayúsculas o sólo para letras minúsculas.

```

        n, lix, liy, lsy, lsy ∈ N
Inicio
  L Lista
  Desde c←1 hasta c=n
    L←L∪(aleatorio modulo (lsx -lix)+lix, aleatorio modulo (lsy -liy)+liy)
  Regresar L
Fin

```

Código 1. Generación de la lista aleatoria L

Donde n es el número de puntos a generar; lix y lsx , son los límites horizontales para muestrear en tanto que liy y lsy son los límites verticales respectivamente. La función devuelve una lista de puntos L .

Cálculo de p'

Para el cálculo del patrón representativo p' se pueden usar cualesquiera de los tres métodos empleados para la búsqueda de patrones representativos: MR , MRM , LA . El uso de alguno de estos métodos repercute en el índice de reconocimiento. El pseudocódigo queda definido para cada método en **Código 2**.

Función MajorityRule
 Parámetros de entrada:

C_i	Lista de vectores
n	Dimensión del espacio $\{0,1\}^n$

```

Inicio
  Suma ∈ Nn
  pi ∈ {0,1}N
  Lon←|Ci|
  Desde c←1 hasta Lon
    Sea ec, el elemento c--ésimo de Ci
    Suma ← Suma + ec
  Fin
  pi ← NaturalBinario(Suma, Lon/2)
  Regresar pi
Fin

```

Función LecturasAleatorias
 Parámetros de entrada:

C_i	Lista de vectores
n	Dimensión del espacio $\{0,1\}^n$
m	Lecturas aleatorias

```

Inicio
  Suma ∈ Nn
  pi ∈ {0,1}N
  Desde c=1 hasta m
    Sea ec, cualquier elemento al azar de Ci
    Suma ← Suma + ec
  Fin
  pi ← NaturalBinario(Suma, m/2)
  Regresar pi
Fin

```

Función MajorityRuleModificado
 Parámetros de entrada:

L_i	Lista de vectores
n	Dimensión del espacio $\{0,1\}^n$
u	Umbral

```

Inicio
  Suma ∈ Nn
  pi ∈ {0,1}n

```

```

Lon ← |Li|
Desde c ← 1 hasta Lon
  Sea ec, el elemento c--ésimo de Li
  Suma ← Suma + ec
Fin
pi ← NaturalBinario(Suma, u)
Regresar pi
Fin
    
```

Código 2. Métodos para el cálculo de la cadena representativa

Las funciones regresan un patrón p_i , de cada C_i dado; si en la codificación se toma el carácter completo, p_i es el carácter patrón de C_i ; los parámetros que reciben dependen en cada caso y son:

1. *Majority Rule*: El conjunto C_i de variantes de caracteres, y la dimensión del espacio de trabajo n .
2. *Lecturas aleatorias*: El conjunto C_i de variantes de caracteres, la dimensión del espacio de trabajo n y el número de lecturas que se hace sobre C_i .
3. *Majority Rule Modificado*: El conjunto C_i de variantes de caracteres, la dimensión del espacio de trabajo n y el umbral para asignar alguno de los símbolos ‘1’ o ‘0’.

El procedimiento NaturalBinario transforma un vector binario $v \in \mathbb{N}^n$ a un vector $v' \in \{0, 1\}^n$, dando un intervalo para función de transformación $F: \mathbb{N}^n \rightarrow \{0, 1\}^n$ tal que

$$f(x) = \begin{cases} 1 & \text{si } x_i \geq \text{umbral} \\ 0 & \text{si } x_i < \text{umbral} \end{cases} \text{ y queda expresado en Código 3.}$$

Función NaturalBinario
 Parámetros de entrada:
 $Vn \in \mathbb{N}^n$
 Umbral Umbral para determinar 1 ó 0

```

Inicio
  Binario ∈ {0,1}n
  Para toda componente de Vn
    Si Vni < Umbral
      Binarioi ← 0
    O si no
      Binarioi ← 1
  Regresar Binario
Fin
    
```

Código 3. Función para transformar un vector en \mathbb{N}^n a $\{0, 1\}^n$

El cálculo de p' necesita aplicar alguno de los métodos descritos a cada C_i para generar los p_i 's. Este proceso está descrito en Código 4.

Función cálculo p'
 Parámetros de entrada: C_a, \dots, C_z
 Inicio
 Calcular $P = \{p_i / i \in \{a, \dots, z\} \text{ metododepromedio}(p_i, C_a)\}$
 $p', x, y \in \{0, 1\}^n$
 $x \leftarrow p_a \wedge \dots \wedge p_z$
 $y \leftarrow p_a \vee \dots \vee p_z$
 $p' \leftarrow x - y$

Regresar p'
Fin

Código 4. Función para determinar el punto p'

El resultado del cálculo de p' se usa para la optimización de los puntos de muestreo de M .

Balanceo de L

Para el proceso de balanceo se calcula p' con toda la información contenida en las distintas variantes de caracteres (se emplea una dimensión de $n = p \times q$); p' es el meta patrón que indica los puntos utilizados y los no utilizados en las imágenes de los caracteres. El procedimiento de balanceo necesita la imagen de p' , representada por M y la lista de puntos de muestreo L , el proceso queda descrito en el **Código 5**.

Función Balancear

Parámetros de entrada: M matriz de la imagen asociada a p'
L Lista de puntos
n dimensión del espacio $\{0,1\}^n$

Inicio

```
Sea  $l_i \in L$ 
Mientras no  $|L| \approx n/2$ 
  Mientras no existan más de dos elementos  $l_i, l_j \in L$  que  $l_i = l_j$ 
    si  $|cl| < n/2$ 
      mientras no  $M_{li} = 1$ 
         $l_i \leftarrow$  (aleatorio, aleatorio)
      o si no
        mientras no  $M_{li} = 0$ 
           $l_i \leftarrow$  (aleatorio, aleatorio)
    Actualizar L con  $l_i$ 
  Regresar L
```

Fin

Código 5. Código para balancear a L

Este proceso distribuye la cantidad de 0's y 1's que aparecen en el punto generado a partir de L , se considera como función objetivo la norma del punto generado de L , si es menor a un umbral $n/2$ (¹⁹) el valor asignado es 0, en caso contrario 1.

Optimización

El resultado del método *balancear* es una lista ligeramente diferente a L que contiene un conjunto de puntos de muestreo distribuidos de manera uniforme, sobre los puntos utilizados y no utilizados por los distintos caracteres. Para seleccionar los mejores puntos de muestreo es necesario hacer un análisis más detallado de cómo los puntos participan en los distintos los caracteres del abecedario. El método de optimización se presenta en **Código 6**.

Función optimizar

Parámetros de entrada: $M_p \in \mathbb{N}$ Matriz de la suma aritmética de la superposición de cada p_i
L Lista de puntos

Inicio

```
Mientras  $\exists l \in L: \max(M_{p_l}) > 13$ 
   $l \leftarrow$  buscar  $\max(M_{p_l})$ 
```

¹⁹ En situaciones experimentales se puede jugar con este parámetro; lograr que tienda a tener más puntos que $n/2$ repercute en la cantidad de puntos que son significativos y usados por las letras; una densidad muy alta puede saturar el número de puntos a muestrearse, repercutiendo en la capacidad de reconocimiento.

```

repetir
  l'←(aleatorio,aleatorio)
  hasta Mp1.> ¼FrecuenciaMaxima ^Mp1.≤Mp1
l←buscar min(Mp1)
repetir
  l'←(aleatorio,aleatorio)
  hasta Mp1.>¼FrecuenciaMaxima ^Mp1.≤Mp1
fin
regresar L
Fin

```

Código 6. Optimización de L, analizando la superposición de los elementos

La parte de entrenamiento es construido por la conjunción de los procesos descritos y queda expresado en **Código 7**.

```

Función entrenamiento
  Sean Ca,...Cz los conjuntos de variaciones de caracteres
  Sean p1,...pz ∈ {0,1}^n cadenas que denotan los representantes de cada conjunto
  n La dimensión de {0,1}^n
  lix, lsx, liy, lsy ∈ N
  M ∈ N^p x N^q Matriz que contiene la imagen asociada a p', de dimensiones p x q
  L ← ListaAleatoria(n,lix,lsx,liy,lsy)
  P' ← Cálculop'(Ca,...,Cz)
  L ← Balancear(M^20,L,n)
  L ← Optimizar(Ms,L)
  P ← {e}
  Para todo Li calcular
    pi ← Método del Patron Representativo21(Li,n,m)
    P = P ∪ pi
  Regresar P,L
Fin

```

Código 7. Fase de entrenamiento

4.1.2 Fase de reconocimiento

El conjunto de patrones representativos se denota por P ; la representatividad está dada por la métrica de similitud descrita en el capítulo 3. Este proceso consiste en decidir cuando una imagen de un carácter es parecida o no a alguno de los patrones $p_i \in P$. Los criterios de decisión son:

- a) La distancia de alguna cadena c' : si la distancia de un p_i es menor al radio de cercanía i -ésimo, entonces se dice que c' es muy parecida a p_i .

$$\{x/c' \in \{0,1\}^n : \forall_{p_i \in P} \exists_{r_i \in R} d(p_i, c') \leq r_i\} \quad (41)$$

²⁰ No se expone como calcular M y Mp ; en el caso de M es directo si p' es una cadena en $\{0,1\}^{p \times q}$, en caso de Mp , es la suma aritmética de todos los elementos en P , recordar que en esta etapa cada elemento en P representa una imagen completa.

²¹ Puede ser entre los métodos presentados: MR, LA, MRM.

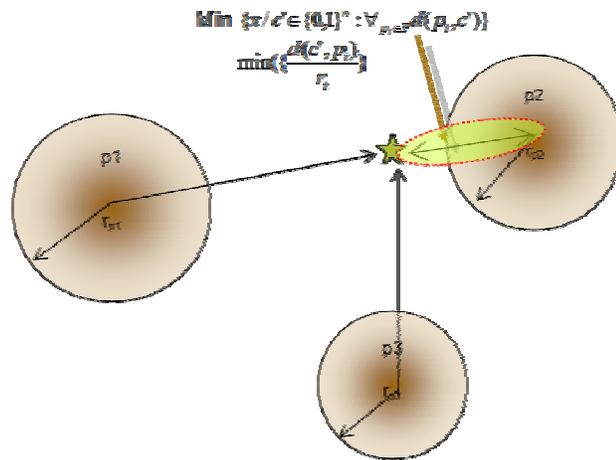
- b) Si el criterio anterior arroja más de una p_i , entonces el elemento de P , que contenga la mínima distancia a c' , es el más parecido, es decir:

$$\text{Min } \{x/c' \in \{0,1\}^n : \forall_{p_i \in P} d(p_i, c')\} \quad (42)$$

- c) Si el criterio anterior arroja empates se considera que la cadena c' es mas parecida a la clase que tiene una relación menor entre la distancia de c' al patrón p_i , dividido por el radio de cercanía.

$$\min\left(\left\{\frac{d(c', p_i)}{r_i}\right\}\right) \quad (43)$$

Si se implementa el criterio c) no es necesario decidir sobre el criterio b). Esquemáticamente el proceso se presenta en **Figura 47**.



Donde r_i es el radio de cercanía del conjunto p_i ésimo

Figura 47. Criterios de decisión para determinar cuando un punto dado p pertenece a una clase p_i

El proceso de prueba queda expresado en el **Código 8**, el resultado de este proceso son las p_i 's con más similitud a una cadena generada de una imagen dada, aquella que tenga la mayor similitud corresponde a la clase más probable o al carácter con más parecido a la imagen codificada.

```

Función generaCadena
Parámetros de entrada: M Matriz que representa a la imagen de dimensiones p x q
                       L Lista de puntos a muestrear
                       n dimensión del espacio {0, 1}^n

Inicio
  Sea p ∈ {0, 1}^n
  Desde i ← 1 hasta n
    p_i ← M_{L,i}
  Regresar p_i
Fin

Función Reconoce
Parámetros de entrada: M Matriz que representa a la imagen de dimensiones p x q

Inicio
  c' ∈ {0, 1}^n
  L ← {e}
    
```

```

c' ← generarCadena(M,L,n)
Por cada ri ∈ R
  Si |pi-c'| < ri
    L ← L ∪ {pi}
Si |L| > 1
  Por cada pi ∈ P
    L ← {min({|pi-c'|})} //Usando criterio b)
    L ← {min({(|pi-c'|)/ri})} //Usando criterio c)
Regresar L
Fin
    
```

Código 8. Pseudo código para el reconocimiento de una imagen M

4.1.3 Análisis de complejidad del algoritmo

El método propuesto ha sido pensado en una máquina no Turing, donde la unidad de operación básica es el vector con una dimensionalidad mayor a uno; las operaciones entre los puntos en esta arquitectura se efectúan en paralelo, sin embargo, el pseudocódigo mostrado es una adaptación al modelo de computación tradicional, la diferencia fundamental está en la métrica de complejidad.

La métrica de complejidad más utilizada en la literatura contabiliza en un algoritmo, el número de operaciones hechas por unidad de tiempo; en el modelo de cadenas simbólicas la complejidad es mucho menor, debido a que muchas operaciones las puede efectuar en paralelo. En la **Tabla 1**, se presenta un análisis de la complejidad²² tomando como referencia el modelo vectorial de longitud n^{23} y el modelo de la máquina de Turing.

Procedimiento	Modelo Clásico	Modelo Cadenas Simbólicas	Modelo Clásico (Notación O(f(n)))	Modelo Cadenas simbólicas (Notación O(f(n)))
Majority Rule	$n \times 26$ ⁽²⁴⁾	26	O(n)	O(1)
Lecturas Aleatorias	$n \times m$ ⁽²⁵⁾	m	O(n ²)	O(n)
Reconocer	$3 \times n$	3	O(n)	O(3)
Entrenamiento	$N \times (n + m)$	$n + m$	O(n ²)	O(2n)
ListaAleatorea	n	1	O(n)	O(1)
Majority Rule Modificado	$n \times 26$ ⁽²⁶⁾	26	O(n)	O(1)
NaturalBinario	n	1	O(n)	O(1)
Calcularp'	$54 \times n + n \times m$ ⁽²⁷⁾	$54 + m$ ⁽²⁷⁾	O(n ²)	O(n)
Balancear	no det.	no det.	no det.	no det.
Optimizar	no det.	no det.	no det.	no det.
Entrenamiento	$87 \times n + m$	$81 + m + 6 \times n$	O(n ²)	O(2n)
GenerarCadena	$2 \times n$	2	O(n)	O(1)

Tabla 1. Análisis de la complejidad del algoritmo.

La complejidad en ambos modelos es distinta, en el caso de cadenas simbólicas, permite operar sobre un conjunto de elementos por ciclo al permitir operaciones sobre las cadenas de longitud n (vistos geoméricamente como puntos en un espacio n dimensional), mientras que en caso del modelo clásico, es necesario efectuar en forma serial las operaciones de

²² Como el número de operaciones a efectuar en el modelo de representación sobre unidad de tiempo.

²³ Muestran el número de operaciones por 1 letra a la vez.

²⁴ Es la longitud de la cadena.

²⁵ Es el número de lecturas aleatorias.

²⁶ Es la longitud de la cadena.

²⁷ Se considera el método con mayor complejidad de los métodos para la búsqueda del patrón representativo.

manera que cuando se requiera operar una cadena de n elementos, es necesario operar sobre cada componente para efectuar cada operación, donde en un modelo de máquinas simbólicas se requiera una operación, en computo clásico requerirá n operaciones, donde n , es el tamaño de las cadenas a operar.

4.2 Prueba y análisis de resultados

En esta sección se describe un conjunto de pruebas donde se evalúa el porcentaje de reconocimiento y cómo los diferentes parámetros del método propuesto afectan este porcentaje, utilizando un conjunto de variantes de caracteres.

4.2.1 Descripción del método experimental

Aproximarse hacia arquitecturas diferentes de memorias es una labor experimental debido a las implicaciones de concepto que se manejan. La manera directa de evaluar el comportamiento es *emulándolo* sobre modelos existentes, la labor experimental es la emulación del modelo de representación de cadenas simbólicas sobre la arquitectura clásica, mostrando sus características y posibilidades.

4.2.1.1 Descripción de pruebas

Para el modelo experimental se ha considerado un conjunto de varios tipos de caracteres²⁸, agrupados en 5 conjuntos diferentes, se tiene cada carácter en una imagen con dimensiones fijas, con diferentes conjuntos de tipos de caracteres. El primer grupo contiene 80 diferentes tipos de caracteres, los otros 4 grupos contienen en promedio de 9 a 14 tipos de caracteres. Sobre el primer grupo se hace un análisis más exhaustivo por contar con una mayor diversidad de letras. El modelo experimental se ilustra en la **Figura 48**.

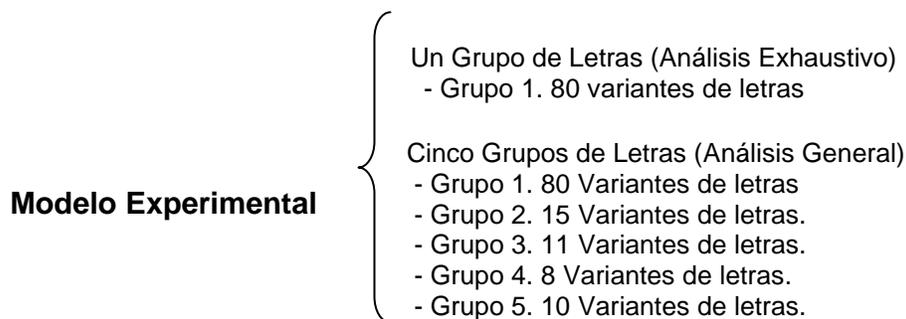


Figura 48. Modelo experimental

Análisis particular al grupo 1 de caracteres. Sobre este conjunto se hace el análisis con mayor detalle. Estos análisis sirven para mostrar el comportamiento de las diferentes etapas del método propuesto. Las etapas a analizar son:

²⁸ Los distintos tipos de letras usados se muestran en el apéndice.

1. Análisis a nivel de patrones p_i 's, donde se muestra el grado de superposición antes y después de aplicar la optimización
2. Consecuencias de emplear alguno de los métodos para la búsqueda de diferentes p_i 's.
3. Eficiencia en el porcentaje de reconocimiento de éstos.

Análisis generales a los 5 grupos de caracteres. A partir de cada grupo de caracteres Σ ²⁹ se generan los C_i 's correspondientes a cada tipo de carácter, para diferentes longitudes de cadenas simbólicas, se calculan las p_i 's bajo los métodos: *MR*, *MRM* y *LA*³⁰.

Una vez que se tienen las cadenas representativas (p_i) se prueba la eficiencia con un grupo de caracteres, por cada prueba se presentan las siguientes tablas:

1. Análisis de la distancia media del patrón p_i , resultado de C_i .
2. Tablas de distancias entre los p_i 's encontrados.
3. Resultados del porcentaje de eficiencia en el reconocimiento.

4.2.1.2 Parametrización

Descripción de las variables

El método experimental consiste en pruebas paramétricas del método. El conjunto de variables parametrizadas son las siguientes:

1. Dimensiones de la imagen: i, j
2. Longitud de la cadena generada: n
3. Cantidad de caracteres a utilizar por experimento: l

En experimentos más particulares existen variables dependientes del método utilizado para el cálculo del patrón representativo:

1. Número de lecturas aleatorias para generar el promedio: nl
2. Umbral para la generación de cadenas con *MR*: $\frac{1}{4}|C_i|$

Definición de los rangos y valores de las variables para el grupo 1 de caracteres

Los valores paramétricos descritos anteriormente adoptan los siguientes valores:

1. Dimensiones de la imagen: $i=64, j=64$
2. Longitud de la cadena usada: $\{25, 30, 40, 50, 100, \dots, 550\}$
3. Cantidad de caracteres a utilizar por experimento: 80 Diferentes tipos
4. Métodos para el cálculo de p_i : *MR*, *MRM*, *LA*.
5. Número de lecturas aleatorias para generar el promedio: $\{150\}$
6. Umbral al emplear *MRM*: $\frac{1}{4}|C_i|$

²⁹ $|\{\Sigma_1, \dots, \Sigma_n\}| = |C_i|$ es decir la cardinalidad de cada C_i será el número distinto de abecedarios, siempre y cuando todo abecedario Σ_i este completo.

³⁰ Ver Optimización del muestreo.

Definición de rangos y valores de las variables de los cuatro grupos de caracteres

Los valores paramétricos antes descritos adoptan los siguientes valores

1. Dimensiones de la imagen: $i=64, j=64$
2. Longitud de la cadena usada: $\{200, 150, 100, 50\}$
3. Cantidad de caracteres a utilizar por experimento: 5 grupos de caracteres.
 - Grupo 1. 80 diferentes variantes de caracteres
 - Grupo 2. 15 diferentes variantes de caracteres
 - Grupo 3. 12 diferentes variantes de caracteres
 - Grupo 4. 8 diferentes variantes de caracteres
 - Grupo 5. 10 diferentes variantes de caracteres
4. Métodos para el cálculo de p_i : *MRM, LA*.
5. Número de lecturas aleatorias para generar el promedio: $\{150\}$
6. Umbral al emplear *MRM*: $\frac{1}{4}|C_i|$

4.2.2 Análisis de resultados

El método propuesto para reconocer caracteres, es un método distinto a los métodos encontrados en la literatura, el método experimental muestra la facilidad de su implementación y los resultados que se obtienen al aplicarlo ante distintas situaciones de reconocimiento de caracteres.

La relación de complejidad y eficiencia del método muestran un comportamiento *adecuado* al resolver la tarea de reconocimiento de caracteres, a pesar que se emula un comportamiento de una máquina abstracta distinta a la clásica donde se identifican los elementos por cadenas simbólicas.

Análisis de los resultados del grupo 1 de caracteres

Análisis de la generación de las cadenas representativas. Esta etapa consiste en generar cadenas representativas al grupo 1 de caracteres empleando los tres métodos descritos. Se construye cada C_i y se calcula cada patrón p_i , construyendo el conjunto P , por cada método. Graficando el nivel de superposición de las cadenas contenidas en P , se obtienen los resultados en la **Figura 49**, la frecuencia de las cadenas es mayor para las cadenas calculadas por *MR* y *LA*.

El análisis de los patrones encontrados muestran que para el caso de *MR* y *LA*, el nivel de superposición es mayor que el de *MRM* (ver **Figura 42**), para umbrales altos en frecuencia tienden los p_i 's a saturarse resultando cadenas con menor significatividad, caso opuesto, frecuencias que estén dentro de un umbral de 13 caracteriza mejor la información de las diferentes caracteres.

Los métodos para encontrar cadenas significativas de *MR* y *LA* ofrecen resultados estadísticos muy similares, (ver **Tabla 2**), en cambio al emplear *MRM* la distancia es distinta, existiendo mayores diferencias conforme disminuye la longitud de las cadenas.

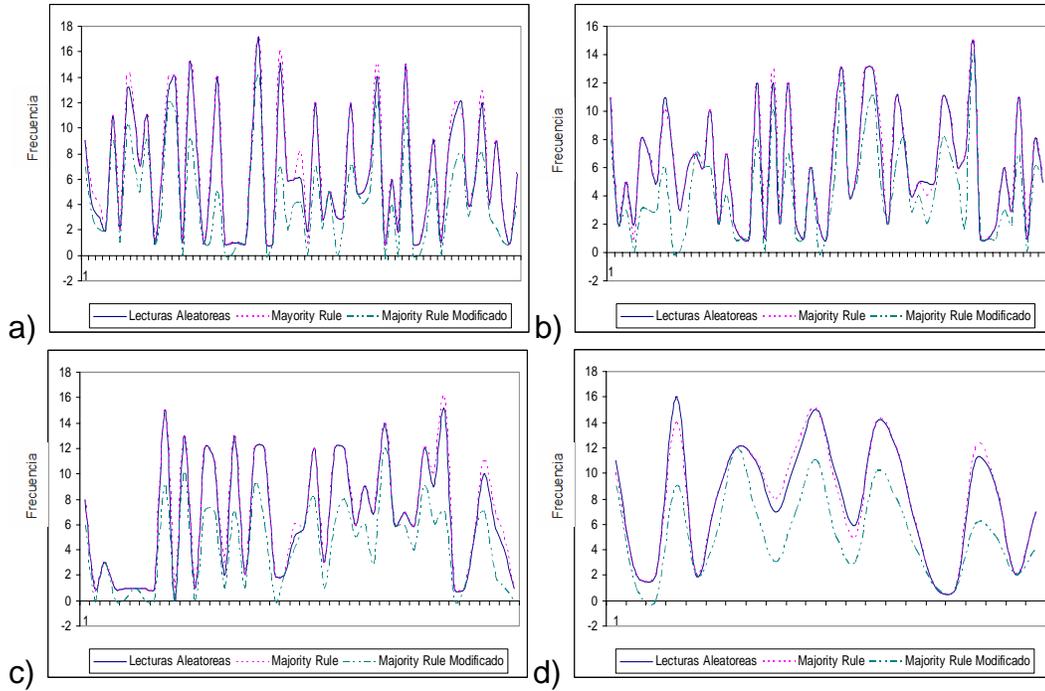


Figura 49. Gráfica de la frecuencia de superposición para los patrones del grupo 1 usando longitudes de a) 200 b) 150 c) 100 y d) 50.

a)

Distancia Media			
Longitud	LA	MR	MRM
200	8.069	8.053	9.496
150	7.014	7.009	8.238
100	5.676	5.662	6.678
50	3.103	3.092	3.643

b)

Desviación Estándar			
Longitud	LA	MR	MRM
200	4.666	4.639	5.314
150	4.593	4.593	5.101
100	3.324	3.291	3.724
50	2.035	2.029	2.375

Tabla 2. Resultados estadísticos obtenidos por cada método: a) Distancia media entre cada patrón; b) Desviación estándar.

Análisis de la etapa de optimización sobre el grupo 1.

Para evaluar el nivel de optimización se efectúa un modelo experimental, en el que, utilizando los caracteres del grupo 1 se construyen dos listas L_1 y L_2 (L_1 sin optimizar y L_2 optimizada), la longitud para cada lista es de $\{200, 150, 100, 50\}$. Se construyen los alfabetos C_a, \dots, C_z y los patrones p_i 's son calculados por los tres métodos descritos.

La **Figura 50** muestra la frecuencia de superposición; L_2 muestra que la frecuencia está dentro de los límites definidos, mientras que el comportamiento de L_1 , no presenta este comportamiento. Los resultados solo se muestran para los casos $n=200$ y $n=100$. La distribución de los puntos sobre MP para un $n=200$ antes y después de la optimización se observa en **Figura 51**, en a) se observa la distribución sin optimización, los puntos están dispersos por toda la figura, en b) notar como ciertas áreas no son muestreadas, debido a un nivel alto de superposición.

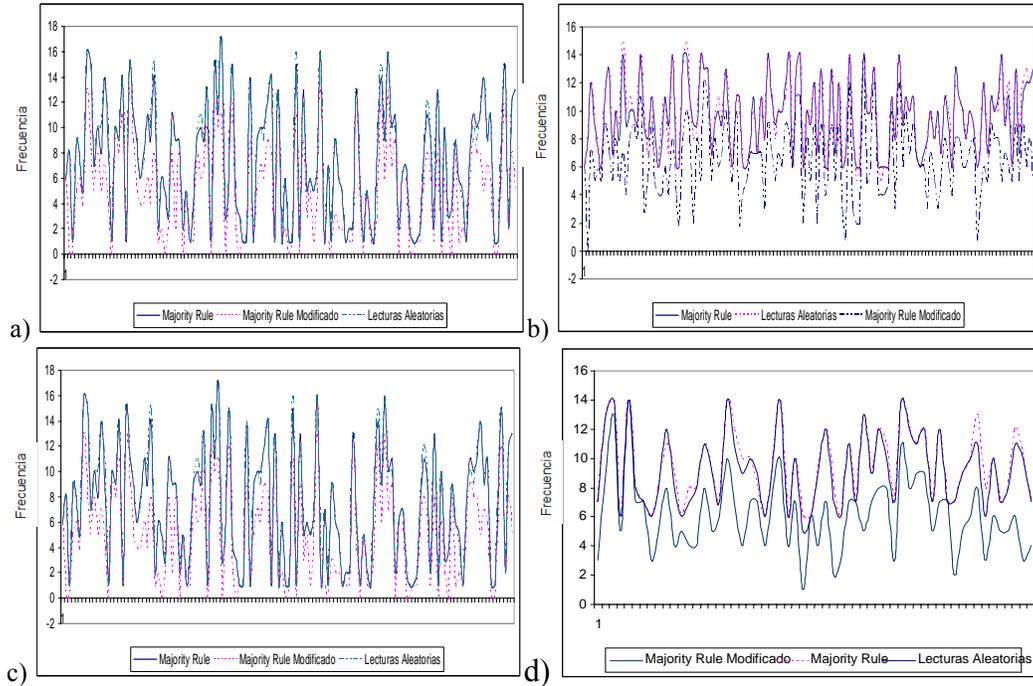


Figura 50. Gráfica de la frecuencia de superposición para a) L_1 con $n=200$ b) L_2 con $n=200$ c) L_1 con $n=100$ d) L_2 con $n=100$

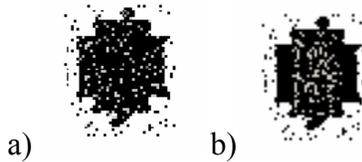


Figura 51. Distribución de los puntos para a) L_1 con $n=200$ b) L_2 con $n=200$.

La **Tabla 3** muestra la eficiencia de reconocimiento para las dos listas (sin optimizar y optimizada), se observa el incremento considerable al aplicar el algoritmo de optimización; la eficiencia de los patrones generados por *MRM* es peor en la lista sin optimizar cuando la longitud tiende a ser pequeña; en longitudes grandes es ligeramente mejor (casos 150 y 200), sin una optimización previa el método de *LA* ofrece mejor eficiencia

Si el muestreo es optimizado la diferencia de eficiencia gradualmente se incrementa para *MRM* y *LA* cuando decrece la longitud de la cadena. Con *MR* y *MRM* la diferencia es menor.

Estadísticamente *MR* ofrece una distancia y una desviación estándar mas pequeña, ofreciendo mejores resultados cuando la longitud de la cadena es pequeña. El método de *LA* ofrece es un buen método cuando se usan cadenas de longitud grande, ofreciendo mejores resultados que *MR*.

MRM estadísticamente tiene una distancia mayor que los otros dos métodos ofreciendo una manera diferente de buscar patrones, pero cuando se trata de un muestreo aleatorio ofrece

peor desempeño, aunque junto con el modelo de optimización ofrece una eficiencia ligeramente mejor que los demás.

N	Sin Optimizar ¹			Optimizado ¹		
	MRM	MR	LA	MRM	RR	LA
200	83.60	82.90	83.10	91.65	90.62	90.10
150	80.60	80.30	81.10	89.65	88.70	87.20
100	69.90	68.20	69.15	85.25	83.00	83.05
50	58.95	59.68	60.65	80.85	77.4	78.4

¹ La eficiencia está medida en el porcentaje de reconocimiento (0-100)

Tabla 3. Tabla de la eficiencia del porcentaje de reconocimiento.

Análisis del rendimiento en función de la longitud de la cadena. El parámetro que tiene mayor repercusión en el índice de reconocimiento es la longitud de cadena. Se evalúa el comportamiento de reconocimiento en función de variar la longitud de cadena. Los valores empleados son 25, 30, 40, 50, 100, 150,... 550.

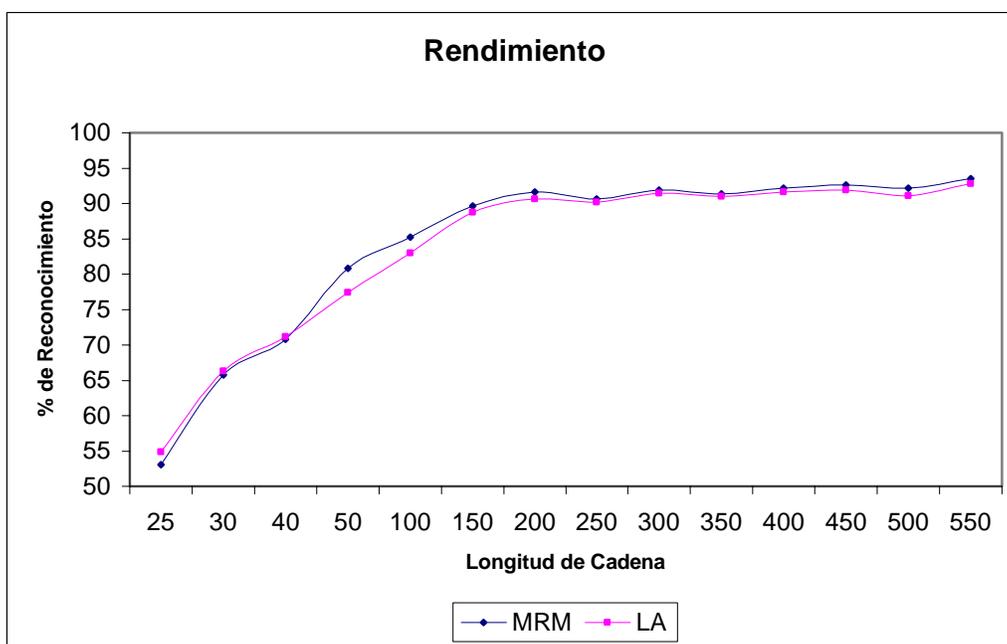


Figura 52. Gráfica de rendimiento de acierto con diferentes longitudes de cadena para métodos para el cálculo de los p_i 's MRM y LA

El rendimiento máximo se estabiliza en aproximadamente un 92 por ciento de eficiencia a partir de longitudes de cadena de 200. El rendimiento es significativamente inferior en longitudes de cadena menores a 100, observándose que la eficiencia depende del método utilizado para calcular los p_i 's. LA tiende a tener un mejor comportamiento en longitudes de cadenas pequeñas, en longitudes de cadenas grandes la diferencia entre MRM y LA es significativa a favor del primero. La **Figura 52**, muestra los resultados de la eficiencia con diferentes longitudes de cadena, el índice de reconocimiento se estabiliza a partir de una longitud de 200, en longitudes menores a 50 decae de manera considerable.

Análisis generales de los 5 grupos

Análisis estadístico. La distancia media entre los patrones en P es pequeña en comparación a la dimensionalidad usada (valor de n). La media de cada grupo depende de la cantidad de caracteres que tenga y las diferencias que existen en cada muestra. Los resultados generales de la distancia media se muestran en la **Tabla 5**. La distancia media es más pequeña para los patrones calculados por MR , por el contrario los resultados mayores se obtienen por MRM .

Experimento	Distancia media											
	MR				LA				MRM			
	200	150	100	50	200	150	100	50	200	150	100	50
1	8.05	7.01	5.66	3.09	8.07	7.01	5.68	3.10	9.22	8.18	6.83	4.26
2	6.39	5.34	4.28	1.57	6.61	5.58	4.52	1.63	7.56	6.51	5.46	2.74
3	8.20	5.10	3.20	2.40	9.00	5.40	3.40	2.60	8.20	5.10	3.20	2.40
4	10.40	7.05	5.00	2.95	10.40	7.20	5.00	3.00	10.40	7.05	5.00	2.95
5	11.60	11.10	6.30	3.30	12.40	11.55	6.60	3.55	11.60	11.10	6.30	3.09

Tabla 4. Distancia media entre las cadenas representativas por cada método.

En la **Tabla 5** se observa cómo MRM en todos los casos tiene una distancia media mayor a MR , pero no necesariamente a LA . Se muestra en **Figura 53** el comportamiento de la distancia media. Este comportamiento se debe a cómo está definido MRM , este método la mayoría de las veces tendrá distancias mayores que MR y en condiciones idealizadas tendrá la misma media que MR , pero no menor. Se observa también cómo el comportamiento de MRM sigue al comportamiento de MR . A pesar que MRM tiene una distancia mayor, para la media en la tabla de eficiencia (ver **Tabla 6**) obtiene mejores resultados que LA . Los resultados obtenidos por MR Normal no se muestran debido a que ha tenido índices más bajos que MRM .

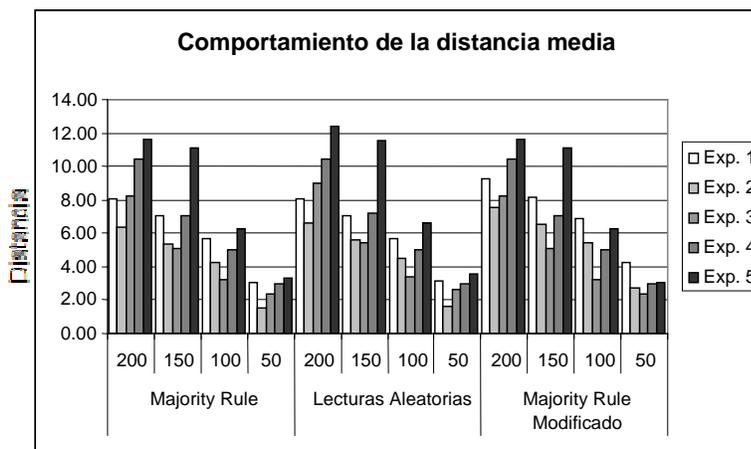


Figura 53. Gráfica del comportamiento de la distancia media por los tres métodos

La proporción de la distancia media con respecto a la longitud de la cadena, va creciendo en la medida que disminuye la longitud de la misma, indicando que a menores longitudes de cadena los puntos que caracterizan a los caracteres se encuentran más dispersos como consecuencia que el índice de reconocimiento baja (en el caso del grupo 1 ver **Figura 52**, en el caso de los demás grupos ver **Figura 54** y **Figura 55**).

Las áreas donde se pueden caracterizar las clases de distintos caracteres cuanto mayor sea la media indican que están más dispersos, llegando situaciones en donde las áreas (radios de cercanía) se solapan haciendo más complicado determinar a que conjunto pertenece un carácter a caracterizar por los parámetros empleados.

Exp.	Relación entre la longitud de cadena y la distancia media											
	MR				LA				MRM			
	200	150	100	50	200	150	100	50	200	150	100	50
Exp.1	0.0403	0.0467	0.0566	0.0618	0.0404	0.0467	0.0568	0.0620	0.0461	0.0545	0.0683	0.0852
Exp.2	0.0320	0.0356	0.0428	0.0314	0.0331	0.0372	0.0452	0.0326	0.0378	0.0434	0.0546	0.0548
Exp.3	0.0410	0.0340	0.0320	0.0480	0.0450	0.0360	0.0340	0.0520	0.0410	0.0340	0.0320	0.0480
Exp.4	0.0520	0.0470	0.0500	0.0590	0.0520	0.0480	0.0500	0.0600	0.0520	0.0470	0.0500	0.0590
Exp.5	0.0580	0.0740	0.0630	0.0660	0.0620	0.0770	0.0660	0.0710	0.0580	0.0740	0.0630	0.0660

Tabla 5. Relación entre la longitud de la cadena y la distancia media de cada experimento.

Análisis de la eficiencia

Generando la cadena a partir de la lista L las pruebas de eficiencia, muestran el porcentaje de aciertos de reconocimiento sobre los caracteres; este se ve influenciado por un conjunto de parámetros.

Cómo afecta cada parámetro en el porcentaje de eficiencia se discute a continuación:

- *La longitud de la cadena:* cuando se consideran bastantes variantes de caracteres, y una longitud *adecuada* de cadena, se obtienen índices de reconocimiento cercanos al 92 por ciento de eficiencia, que se degrada conforme se reduce la longitud de cadena.
- *Reducción de la longitud de la cadena simbólica:* depende en gran medida de la cantidad de estilos de caracteres a reconocer, dando como resultado que a menor grado de tipos de caracteres, la longitud puede reducirse sin comprometer demasiado su eficiencia.
- *Incrementos substanciales de la longitud de cadena n :* tienen un incremento substancial en el porcentaje de reconocimiento. Vía experimental se estabiliza en longitudes de cadena mayores a 150 (Ver **Figura 52**).
- *La manera de generar los tipos de patrones:* repercute en la calidad de la muestra, permitiendo o no tener índices más altos de reconocimiento (ver **Tabla 3** y **Tabla 6**).
- *Métodos alternos para construir a P (MRM):* junto con un modelo de optimización tienen mejor resultado que usar a ambos por separado.

El comportamiento de la **Tabla 6** es: a menor longitud de cadena el reconocimiento es menor, sin embargo, en el experimento cuatro (ver la **Figura 54** y la **Figura 55**) se muestra una disminución casi nula de reconocimiento al disminuir la cadena, entre los rangos de 200 a 100, pero al cambiar a una longitud de 50 sufre una disminución drástica, este

comportamiento se debe en particular al tipo de caracteres que se está empleando, recordar que el grupo 4 está constituido por caracteres de letras en mayúsculas, la longitud necesaria es diferente que en los demás casos donde los caracteres de las letras esta en minúsculas, consecuencia que de acuerdo al tipo de carácter y estilo varían los parámetros para tener altos índices reconocimiento.

Exp	% de reconocimiento							
	MRM				LA			
	200	150	100	50	200	150	100	50
Exp. 1	92.05	89.24	86.34	83.66	91.73	89.43	85.37	83.54
Exp. 2	89.22	94.49	88.43	87.25	90.80	93.63	89.17	85.58
Exp. 3	93.37	90.42	89.99	88.55	92.35	90.87	91.03	88.93
Exp. 4	78.23	77.77	80.71	70.25	77.71	77.77	80.65	71.42
Exp. 5	89.28	88.35	86.93	86.25	90.96	87.55	85.15	84.62
Prom.	88.43	88.05	86.48	83.19	88.71	87.85	86.27	82.82

Tabla 6. Resultados de la eficiencia de reconocimiento.

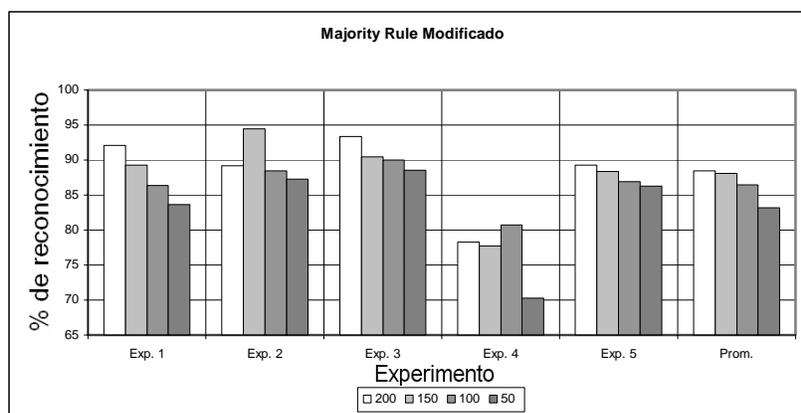


Figura 54. Eficiencia del reconocimiento de cada conjunto de caracteres empleando MRM

Otro caso, es el que ocurre con el experimento 2, en donde uno esperaría que disminuyera el porcentaje de reconocimiento al ir de 200 a 150, al contrario se eleva, esto es indicación que es muy probable que la lista generada L no necesariamente era la óptima, además que la cantidad de símbolos que se tienen en el grupo presentan variaciones muy características, en este caso este grupo el conformado por caracteres muy cuadrados (ver Apéndice, Grupo 2).

De los tres modelos, MRM ofrece mejor rendimiento que MR , en algunas situaciones LA ofrece un rendimiento similar e incluso mejor a MRM , estas situaciones son donde la longitud de cadena es menor, pero se comporta a la inversa con longitudes de cadena mayores. El umbral práctico para un reconocimiento *adecuado* en ambos métodos son longitudes de cadena mayores o iguales a 150, en otras situaciones se va degradando considerablemente el rendimiento (ver **Figura 52**, **Figura 54** y **Figura 55**)

En este trabajo se ha mostrado un reconocedor de caracteres con un grado de eficiencia considerable para realizar dicha tarea, cuya importancia es la idea de buscar nuevos modelos de representación de información que permitan realizar tareas tradicionales en formas distintas y posiblemente más eficientes.

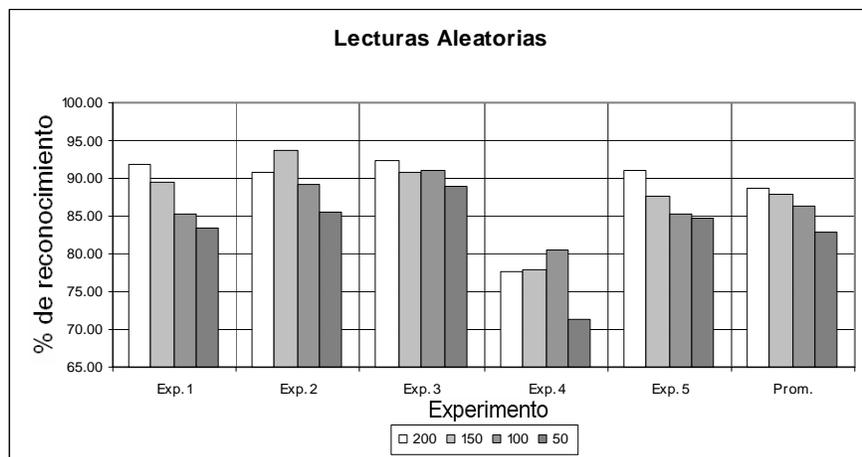


Figura 55. Eficiencia del reconocimiento de cada conjunto de caracteres empleando LA

Se ha dado énfasis en la búsqueda de nuevas formas de representación junto con un conjunto de reglas de operación que permiten definir máquinas abstractas que realizan de forma diferente ciertas tareas que bajo el modelo clásico. Por fines experimentales se ha implementado en una arquitectura de cómputo tradicional, logrando una equivalencia de este modelo emulando, el comportamiento a expensas de perder ciertas características.

4.3 Discusión

Proponer maneras distintas para resolver cierto tipo de problemas, por ejemplo, de reconocimiento de patrones, en el que depende del modelo teórico usado la manera de representar los datos y la manera de operarlos. En la literatura clásica de reconocimiento de patrones [1, 2, 3] los patrones son considerados como cantidades cuantificables que expresa la *ausencia, tipo o carencia* de ciertas características, que en conjunto describen a un objeto, esta idea ha sido tomada para el análisis de reconocimiento de caracteres, las desventajas que tiene usar modelos de cuantificación de *descriptores* es que no se puede aseverar cuantos ni cuáles son los *adecuados* para describir en un contexto determinado un fenómeno. Proporcionar modelos distintos de representación de los *descriptores* permite modelar de manera diferente las situaciones que se hacen mediante cantidades cuantificables, pero es necesario definir operadores y modelos de codificación que ayuden a representar, mediante un modelo abstracto al fenómeno a caracterizar. En este capítulo se han conjuntado las ideas de una representación simbólica y un modelo de codificación basado en muestreo probabilístico; con estos dos conceptos se ha presentado un método de reconocimiento de caracteres. El modelo de representación simbólica permite expresar y codificar de manera distinta los descriptores y el modelo probabilístico, al emplear aleatorios, permite la aproximación a soluciones útiles en problemas NP Completos [8, 9, 10].

El método propuesto ofrece una manera diferente de atacar problemas de reconocimiento de patrones, en donde los *descriptores* se codifican como cadenas simbólicas de dos estados. Proponer maneras diferentes de resolver ciertos problemas resulta útil, cuando

ofrecen soluciones *adecuadas* en el contexto; este método se ha evaluado mediante un modelo experimental con diferentes tipos de caracteres. Los resultados que ofrece son *adecuados* para el tema de reconocimiento de caracteres; estos resultados son dependientes de un conjunto de parámetros que afectan directamente el índice de reconocimiento de caracteres. Algunos de estos parámetros son: la longitud de la cadena simbólica, el método de búsqueda de patrones representativos y las posibles variantes de caracteres.

Contribuciones

En este trabajo se ha presentado una propuesta de método para el reconocimiento de caracteres, que muestra cómo modelos diferentes de información respecto del cómputo clásico³¹, pueden ayudar a proponer nuevas alternativas de solución al problema de reconocimiento de patrones, siendo un caso particular el de reconocimiento de caracteres.

El método propuesto se ha evaluado bajo un modelo experimental con diferentes tipos de caracteres y se han identificado parámetros que modifican el desempeño del método.

Se ha propuesto una representación en cadenas simbólicas no aritméticas, para el reconocimiento de caracteres que evita la necesidad de conocer los descriptores.

Se ha mostrado que es posible usar un Método de Muestreo aleatorio sobre imágenes de caracteres para extraer la información que ayude a su clasificación.

Se ha propuesto un método para buscar patrones simbólicos (*MRM*) en las cadenas que representan a las variantes de los caracteres. El método se basa en dos ya existentes: *MR* y *LA*.

Se ha mostrado que es posible identificar una distribución de puntos de muestreo sobre imágenes para codificar la información de los diferentes caracteres de manera que se obtenga el mayor porcentaje de reconocimiento. Esta distribución se hizo por medio de un método de optimización basado en el análisis de puntos comunes en las distintas imágenes de letras.

³¹ Se entiende por modelo de cómputo clásico, al modelo de cómputo propuesto por Turing.

Conclusiones

En este trabajo se ha probado que es posible realizar el reconocimiento de caracteres, empleando una representación vectorial de imágenes de caracteres (representación simbólica) y un Método de Muestreo aleatorio.

La representación vectorial, en forma de cadenas con longitud fija, permite extraer la información de diferentes caracteres para clasificarlas. Esta forma de representación evita la necesidad de definir los descriptores de los caracteres.

Se ha empleado un Método de Muestreo aleatorio para extraer la información, a partir de un número representativo de puntos de muestreo, en donde, la cantidad de puntos es suficientemente grande para poder clasificar los caracteres de manera eficiente.

Se ha probado, a través de un método experimental, que un modelo de representación simbólica permite hacer reconocimiento de patrones.

Se ha presentado un método de optimización sobre el conjunto de puntos de muestreo, que maximiza la efectividad del reconocimiento.

Se ha observado que la longitud (dimensionalidad del espacio) de las cadenas tiene relación con la cantidad de símbolos codificados y afecta las capacidades de reconocimiento, es decir, que si los símbolos a caracterizar son *demasiados*, en relación a la longitud de las cadenas, se tienen índices de reconocimiento bajos.

Trabajos futuros

Se proyecta explorar estructuras donde se trabajen con hipercubos de cadenas, es decir, de una dimensionalidad mayor, además de hacer un análisis de las operaciones que pueden ser expresadas entre estas estructuras, aprovechando las características que pueda ofrecer espacios en una alta dimensionalidad.

También se puede desarrollar tareas de reconocimiento de patrones, distintas al reconocimiento de caracteres, que puedan ser beneficiadas con el uso de una representación simbólica, de manera que se empleen las capacidades y propiedades que tiene un modelo de representación como este.

Considerando las características del método de reconocimiento de caracteres aquí propuesto, se tiene contemplado desarrollar una versión para computadoras con arquitectura paralela.

Comentarios finales

Tomando como referencia los resultados obtenidos y la manera de caracterizar la información, se puede hablar de modelos distintos de memoria, posiblemente inexactos al modelo tradicional, basados en *patrones* (con la problemática que trae el concepto de patrón), y modelos de direccionamientos basados en contenido.

Este trabajo constituye un avance sustancial para la búsqueda de nuevas formas de cómputo, con modelos abstractos distintos, en donde no se busca una generalidad de las arquitecturas actuales, por el contrario, el desarrollo de nuevas opciones de plantear y resolver problemas que en la arquitectura clásica resulta de suma dificultad.

Las memorias tradicionales de computadora tienen un acceso a la información basado en una dirección dentro de un espacio físico y solo es retenida o almacenada una vez que se contiene esa dirección física; el uso de otro tipo de memorias y otro tipo de modelos estructurales para representar la información conlleva a desarrollar diferentes modos de direccionamiento a los modos clásicos, por ejemplo: basados en el contenido y no en dirección.

Con diferentes maneras de operar y representar la información, el cómputo en estos modelos resultaría distinto a un cómputo tradicional, ya que puede traer beneficios en tareas particulares.

Apéndice. Variantes de Caracteres de Letras

Las variantes de los caracteres de las letras empleadas para el modelo experimental han sido formateadas como imágenes de dimensión de 64 por 64 píxeles en escala de dos colores, donde cada grafía esta centrada; en formato de fuentes tipo True Type u Open Type el equivalente es a un tamaño de 36. Cada imagen de carácter ha sido almacenada en formato bmp (Bit Map Picture), sin compresión.

Conjunto de caracteres utilizadas para el experimento 1.

1. a b c d e f g h i j k l m n o p q r s t u v w x y z
 2. a b c d e f g h i j k l m n o p q r s t u v w x y z
 3. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 4. a b c d e f g h i j k l m n o p q r s t u v w x y z
 5. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 6. a b c d e f g h i j k l m n o p q r s t u v w x y z
 7. a b c d e f g h i j k l m n o p q r s t u v w x y z
 8. a b c d e f g h i j k l m n o p q r s t u v w x y z
 9. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 10. a b c d e f g h i j k l m n o p q r s t u v w x y z
 11. a b c d e f g h i j k l m n o p q r s t u v w x y z
 12. a b c d e f g h i j k l m n o p q r s t u v w x y z
 13. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 14. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 15. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 16. a b c d e f g h i j k l m n o p q r s t u v w x y z
 17. a b c d e f g h i j k l m n o p q r s t u v w x y z
 18. a b c d e f g h i j k l m n o p q r s t u v w x y z
 19. a b c d e f g h i j k l m n o p q r s t u v w x y z
 20. a b c d e f g h i j k l m n o p q r s t u v w x y z
 21. a b c d e f g h i j k l m n o p q r s t u v w x y z
 22. a b c d e f g h i j k l m n o p q r s t u v w x y z
 23. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 24. a b c d e f g h i j k l m n o p q r s t u v w x y z
 25. a b c d e f g h i j k l m n o p q r s t u v w x y z
 26. a b c d e f g h i j k l m n o p q r s t u v w x y z
 27. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 28. a b c d e f g h i j k l m n o p q r s t u v w x y z
 29. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 30. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 31. a b c d e f g h i j k l m n o p q r s t u v w x y z
 32. a b c d e f g h i j k l m n o p q r s t u v w x y z

33. a b c d e f g h i j k l m n o p q r s t u v w x y z
 34. a b c d e f g h i j k l m n o p q r s t u v w x y z
 35. a b c d e f g h i j k l m n o p q r s t u v w x y z
 36. a b c d e f g h i j k l m n o p q r s t u v w x y z
 37. a b c d e f g h i j k l m n o p q r s t u v w x y z
 38. a b c d e f g h i j k l m n o p q r s t u v w x y z
 39. a b c d e f g h i j k l m n o p q r s t u v w x y z
 40. a b c d e f g h i j k l m n o p q r s t u v w x y z
 41. a b c d e f g h i j k l m n o p q r s t u v w x y z
 42. a b c d e f g h i j k l m n o p q r s t u v w x y z
 43. a b c d e f g h i j k l m n o p q r s t u v w x y z
 44. a b c d e f g h i j k l m n o p q r s t u v w x y z
 45. a b c d e f g h i j k l m n o p q r s t u v w x y z
 46. a b c d e f g h i j k l m n o p q r s t u v w x y z
 47. a b c d e f g h i j k l m n o p q r s t u v w x y z
 48. a b c d e f g h i j k l m n o p q r s t u v w x y z
 49. a b c d e f g h i j k l m n o p q r s t u v w x y z
 50. a b c d e f g h i j k l m n o p q r s t u v w x y z
 51. a b c d e f g h i j k l m n o p q r s t u v w x y z
 52. a b c d e f g h i j k l m n o p q r s t u v w x y z
 53. a b c d e f g h i j k l m n o p q r s t u v w x y z
 54. a b c d e f g h i j k l m n o p q r s t u v w x y z
 55. a b c d e f g h i j k l m n o p q r s t u v w x y z
 56. a b c d e f g h i j k l m n o p q r s t u v w x y z
 57. a b c d e f g h i j k l m n o p q r s t u v w x y z
 58. a b c d e f g h i j k l m n o p q r s t u v w x y z
 59. a b c d e f g h i j k l m n o p q r s t u v w x y z
 60. a b c d e f g h i j k l m n o p q r s t u v w x y z
 61. a b c d e f g h i j k l m n o p q r s t u v w x y z
 62. a b c d e f g h i j k l m n o p q r s t u v w x y z
 63. a b c d e f g h i j k l m n o p q r s t u v w x y z
 64. a b c d e f g h i j k l m n o p q r s t u v w x y z
 65. a b c d e f g h i j k l m n o p q r s t u v w x y z
 66. a b c d e f g h i j k l m n o p q r s t u v w x y z
 67. a b c d e f g h i j k l m n o p q r s t u v w x y z
 68. a b c d e f g h i j k l m n o p q r s t u v w x y z
 69. a b c d e f g h i j k l m n o p q r s t u v w x y z
 70. a b c d e f g h i j k l m n o p q r s t u v w x y z
 71. a b c d e f g h i j k l m n o p q r s t u v w x y z
 72. a b c d e f g h i j k l m n o p q r s t u v w x y z
 73. a b c d e f g h i j k l m n o p q r s t u v w x y z
 74. a b c d e f g h i j k l m n o p q r s t u v w x y z
 75. a b c d e f g h i j k l m n o p q r s t u v w x y z
 76. a b c d e f g h i j k l m n o p q r s t u v w x y z
 77. a b c d e f g h i j k l m n o p q r s t u v w x y z

78. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 79. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 80. **a b c d e f g h i j k l m n o p q r s t u v w x y z**

Conjunto de caracteres utilizadas para el experimento 2.

1. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 2. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 3. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 4. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 5. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 6. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 7. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 8. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 9. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 10. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 11. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 12. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 13. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 14. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 15. **a b c d e f g h i j k l m n o p q r s t u v w x y z**

Conjunto de caracteres utilizadas para el experimento 3.

1. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 2. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 3. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 4. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 5. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 6. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 7. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 8. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 9. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 10. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 11. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
 12. **a b c d e f g h i j k l m n o p q r s t u v w x y z**

Conjunto de caracteres utilizadas para el experimento 4.

1. **A B C D E F G H I J K L M N O P Q R S T U V W X Y Z**
 2. **A B C D E F G H I J K L M N O P Q R S T U V W X Y Z**
 3. **A B C D E F G H I J K L M N O P Q R S T U V W X Y Z**
 4. **A B C D E F G H I J K L M N O P Q R S T U V W X Y Z**
 5. **A B C D E F G H I J K L M N O P Q R S T U V W X Y Z**
 6. **A B C D E F G H I J K L M N O P Q R S T U V W X Y Z**

7. **A B C D E F G H I J K L M N O P Q R S T U V W X Y Z**
8. **A B C D E F G H I J K L M N O P Q R S T U V W X Y Z**

Conjunto de caracteres utilizadas para el experimento 5.

1. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
2. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
3. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
4. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
5. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
6. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
7. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
8. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
9. **a b c d e f g h i j k l m n o p q r s t u v w x y z**
10. **a b c d e f g h i j k l m n o p q r s t u v w x y z**

Glosario de Términos

ADAM. Modelo de memoria asociativa, las siglas ADAM tienen por significado Advanced Distributed Associative Memory (Memoria Asociativa Distribuida Avanzada); este modelo fue propuesto por Jim Austin de la *University of Cork* [31], como tema de tesis doctoral, en 1986.

Aleatorio. Se usa para indicar que un fenómeno particular es, desde algún punto de vista, un hecho al azar. Los modelos aleatorios son modelos basados en pruebas al azar, mientras que los modelos determinísticos siempre producen la misma salida dada una condición inicial.

Algoritmo. Sucesión de pasos bien definidos con una estructura propia para lograr un objetivo definido.

Abstracción. Generalización de un hecho o un evento.

Asociación. Es el proceso por el cual se establece una *relación* entre dos entes abstractos, que puede representa algún hecho codificado.

BAM. Son las siglas de Bidirectional Associative Memory [38] (Memoria Asociativa Bidireccional), es un modelo de memoria heteroasociativa que fue propuesto por Bart Kosko

Biyección. Es una relación donde la regla de correspondencia es uno a uno entre dos conjuntos dados.

Carácter. Cualquier símbolo del alfabeto escrito.

Clase. Conjunto de elementos que comparten algunas características *similares*, en base a una o varias métricas definidas.

Codificar. Transformar una representación dada en otra. Esta transformación se hace respetando las reglas individuales de cada representación. Formalmente se representa como una relación de codificación entre dos dominios diferentes $R: A \rightarrow B$; donde R es la relación de transformación, A es el dominio de partida y B el dominio de transformación.

Código. Sistema de símbolos, con operaciones y reglas particulares para representar y manipular la información.

Complejidad. Métrica que muestra la posibilidad de representar y tratar a un hecho mediante un esquema de representación definido; un caso particular es la métrica computacional que mide la cantidad de espacio y tiempo que consume un proceso.

Conjunto. Es una colección de objetos, que puede ser finita o infinita, los cuales no tienen un orden y la multiplicidad es generalmente ignorada. Los miembros de un conjunto son referidos como elementos y su notación $a \in A$ es usada para decir que a es un elemento de A . El estudio de conjuntos y sus propiedades es el objeto de estudio de la teoría de conjuntos.

Descriptor. Es cualquiera de las características medibles de un fenómeno, que está codificada en un modelo de representación.

Distribución Aleatoria. Es la descripción de los números relativos de los tiempos probables de cada posible salida que pueda ocurrir en un número de pruebas. La función que describe la distribución es llamada función probabilidad, y la función que describe la probabilidad acumulada de un valor o cualquier valor pequeño para que este ocurra, se le llama función de distribución.

Grafía. Es el símbolo (imagen) asociado a algún carácter.

Homeomorfismo. Es una relación de equivalencia y una correspondencia uno a uno entre puntos en dos figuras geométricas o espacios topográficos los cuales son continuos en ambas direcciones, también se conoce como transformación continua.

Hiperesfera. La n hiperesfera (comúnmente llamada como n esfera) es una generalización del círculo (a la esfera en dos dimensiones) y una esfera normal (a la esfera en 3 dimensiones) para dimensiones mayores o iguales a 4. La n esfera es por lo tanto definida como el conjunto de n tuplas de puntos (x_1, x_2, \dots, x_n) tal que $x_1^2 + x_2^2 + \dots + x_n^2 = R$, donde R es el radio de esta hiperesfera

Máquina de Turing. Consiste en un modelo de cálculo abstracto que consta de dos elementos:

- a) Una cinta infinita de celdas.
- b) Una cabeza que opera sobre la cinta, modificando el contenido, con desplazamiento de izquierda a derecha.

A estos dos elementos se le suma un conjunto de instrucciones de cómo la cabeza puede modificar a la celda activa y los movimientos que puede presentar en la cinta.

Memoria. Cualquier dispositivo capaz de “almacenar” y “recuperar” información, de acuerdo a una estructura determinada.

Memoria asociativa. Es un dispositivo capaz de recuperar patrones completos a partir de patrones de entrada, que pueden ser alterados mediante distintos tipos de ruido.

Memoria no exacta. Medio de almacenamiento y recuperación de información basado en un modelo de referencia y con un margen de error que establece un grado de inexactitud. En el modelo de cómputo clásico se dice que una memoria es no exacta si, mediante la lectura o mediante la escritura de alguna dirección física se obtiene:

- a) No necesariamente la lectura de la dirección dada (una cota de error de lectura);
- b) La información obtenida es distinta a la almacenada inicialmente.

Método. Modo ordenado de proceder para llegar a un resultado o fin determinado.

Muestreo. Es una técnica estadística que consiste en escoger un subconjunto de elementos que son característicos de un conjunto mayor, de manera que no se tenga que trabajar con el conjunto original. Al subconjunto seleccionado se le denomina muestra, y al conjunto original se le denomina población.

Patrón. Es un elemento distintivo a un conjunto de objetos. El elemento es distintivo al conjunto por un conjunto de métricas dadas. No necesariamente un patrón es un elemento del conjunto.

RAM. Siglas de *Random Access Machine (Máquina de Acceso Aleatorio)*; es un modelo de memoria de acceso aleatorio de una versión simplificada de una computadora tipo Turing, que está provista de un dispositivo de entrada, una unidad de memoria y un medio de almacenamiento separado para el programa. Cada celda en la memoria puede ser accesado arbitrariamente por un entero positivo.

Reconocer. Distinguir un fenómeno por sus características *medibles* codificadas en un modelo de *representación*.

Red Neuronal. Modelo computacional basado en la metáfora biológica de neurona que usan muchas entidades simples interconectadas; de acuerdo al modelo de interconexión es el comportamiento de la red, donde una de sus principales aplicaciones es la de clasificación. Se le conoce también como modelo de caja negra.

SMD. (Sparse Distributed Memory) Modelo de memoria asociativa desarrollado en 1988 por Pentti Kanerva [13], que emplea las características de espacios $\{0, 1\}$ con altas dimensionalidades, y las propiedades de la distancia de Hamming presente en estos espacios.

Superposición de información. Es un estado de la información donde al mismo tiempo se representan a dos o más eventos.

Variante de carácter. Es una alteración de estilo sobre la forma habitual de representar un carácter, por ejemplo, algunas variantes del carácter de la letra “a”, son: a, a, a, a.

Simbología

Símbolo	Significado
Operadores lógicos	
\wedge	y lógico
\vee	ó lógico
\oplus	ó exclusivo
\neg	no lógico
Operadores de Conjuntos	
\cup	Unión de conjuntos
\cap	Intersección de conjuntos
\in	Pertenece a
\notin	No pertenece a
\subset	Subconjunto propio
\subseteq	Subconjunto no propio
E	Elemento vacío
$ $	Cardinalidad de un conjunto ³²
$\{e_1, \dots, e_n\}$	Enumeración de conjuntos
Operadores de Multiconjuntos	
\oplus	Multiunión de multiconjuntos
$[e_1, \dots, e_n]$	Enumeración de multiconjuntos
Operadores Relacionales	
\exists_x	Existe un x
\forall_x	Para toda x
Aritméticas	
$\sum x$	Sumatoria de x
Matriciales	
x^T	Transpuesta de x
Operaciones con cadenas binarias	
$\neg x$	Complemento de x
$ x $	Norma de x
$x-y$	Diferencia de x y y
$d(x,y)$	Distancia de Hamming entre x y y
$O(x,r)$	Círculo con centro en x y radio r

³² El mismo símbolo es usado para la norma de cadenas binarias.

Índice de Figuras

Figura 1. Diagrama de relación de capítulos de la tesis.....	6
Figura 2. Diferentes estilos para representar un mismo carácter o letra.....	8
Figura 3. Variantes de estilo en un mismo carácter.....	9
Figura 4. Diferentes maneras de realizar el reconocimiento de caracteres: a) por descomposición de formas primitivas, líneas, y curvas; b) por áreas del carácter ocupado; c) utilizando redes neuronales; d) construyendo gráficas de distancia a partir del centro de la imagen.....	9
Figura 5. Proceso del reconocimiento de patrones mediante técnicas clásicas.....	10
Figura 6. Representación del espacio de atributos del problema ejemplo.....	11
Figura 7. Codificación de elementos por atributos.....	12
Figura 8. Tipos de clasificación tradicionales.....	12
Figura 9. Esquema del criterio de mínima distancia.....	13
Figura 10. Esquema del funcionamiento de la coincidencia por correlación.....	14
Figura 11. Esquema del proceso de correspondencia por correlación.....	15
Figura 12. Estructura de una RNA.....	17
Figura 13. Estructura de una neurona.....	17
Figura 14. Diferentes arquitecturas de RNA sobre el espacio de soluciones: a) de una capa; b) dos capas; c) multicapa.....	18
Figura 15. Esquema de una memoria asociativa.....	18
Figura 16. Modelos de memorias asociativas.....	19
Figura 17. Esquema del modelo de memoria asociativa ADAM.....	21
Figura 18. Fase de aprendizaje del modelo de Kanerva.....	22
Figura 19. Fase de recuperación del modelo de Kanerva.....	23
Figura 20. Estructura tabular, para codificar la información donde cada renglón representa un hecho codificado.....	24
Figura 21. Estructura del modelo ASC.....	25
Figura 22. Distribución de la distancia de Hamming $\{0,1\}^n$ y la distancia euclidiana de tres dimensiones.....	35
Figura 23. Extracción de información.....	40
Figura 24. El muestreo como una manera de hacer reducción de información.....	41
Figura 25. Ejemplo de casos de caracteres que son empleados en el reconocimiento.....	41
Figura 26. Ejemplo de caracteres no válidos para el análisis.....	42
Figura 27. Proceso de generación de cadenas simbólicas sobre caracteres empleando un muestreo.....	42
Figura 28. Ejemplos de la información y la no información en las grafías correspondientes a los caracteres de las letras “w” y “r” respectivamente.....	44
Figura 29. Representación de una imagen.....	45
Figura 30. Proceso de generación de una cadena simbólica a partir de una imagen de un carácter.....	45
Figura 31. Ejemplos de cadenas generadas a partir de caracteres: ligeramente diferentes (a) y (b) y caracteres diferentes (c).....	46
Figura 32. Distribución de los puntos de las cadenas generadas del muestreo a variantes de caracteres.....	46
Figura 33. Ejemplo del cálculo del punto promedio por majority rule.....	48
Figura 34. Esquema del radio de cercanía, y el conjunto de puntos correspondientes a caracteres muy parecidas a p.....	49
Figura 35. Ejemplo de caracteres promedio por MR y LA.....	50
Figura 36. Áreas donde es posible extraer información acerca de cada carácter.....	51
Figura 37. Operaciones de a) empalme b) puntos comunes y c) diferencia entre los patrones.....	52
Figura 38. Ejemplo de la obtención de p' , para optimizar el muestreo.....	53
Figura 39. Proceso de balanceo de la lista L.....	54
Figura 40. Áreas con información común en más de un carácter.....	55
Figura 41. Caracteres contenidos por otros caracteres.....	55
Figura 42. Gráfica de la frecuencia de superposición en un conjunto de patrones p 's.....	55
Figura 43. Curvas de nivel de diferentes grados de superposición de puntos en las imágenes de los patrones p_i	56

Figura 44. Gráfica de curvas de nivel de la frecuencia de los puntos de los caracteres “a”..... 58
Figura 45. Cálculo del radio de cercanía..... 60
Figura 46. Esquema del proceso de la etapa del cálculo de patrones..... 63
Figura 47. Criterios de decisión para determinar cuando un punto dado p pertenece a una clase p_i 68
Figura 48. Modelo experimental..... 70
Figura 49. Gráfica de la frecuencia de superposición para los patrones del grupo 1 usando longitudes de a) 200 b) 150 c) 100 y d) 50..... 73
Figura 50. Gráfica de la frecuencia de superposición para a) L_1 con $n=200$ b) L_2 con $n=200$ c) L_1 con $n=100$ d) L_2 con $n=100$ 74
Figura 51. Distribución de los puntos para a) L_1 con $n=200$ b) L_2 con $n=200$ 74
Figura 52. Gráfica de rendimiento de acierto con diferentes longitudes de cadena para métodos para el cálculo de los p_i 's MRM y LA..... 75
Figura 53. Gráfica del comportamiento de la distancia media por los tres métodos 76
Figura 54. Eficiencia del reconocimiento de cada conjunto de caracteres empleando MRM..... 78
Figura 55. Eficiencia del reconocimiento de cada conjunto de caracteres empleando LA 79

Índice de Tablas

Tabla 1. <i>Análisis de la complejidad del algoritmo.</i>	69
Tabla 2. <i>Resultados estadísticos obtenidos por cada método: a) Distancia media entre cada patrón; b) Desviación estándar.</i>	73
Tabla 3. <i>Tabla de la eficiencia del porcentaje de reconocimiento.</i>	75
Tabla 4. <i>Distancia media entre las cadenas representativas por cada método.</i>	76
Tabla 5. <i>Relación entre la longitud de la cadena y la distancia media de cada experimento.</i>	77
Tabla 6. <i>Resultados de la eficiencia de reconocimiento.</i>	78

Índice de Códigos

<i>Código 1. Generación de la lista aleatoria L</i>	64
<i>Código 2. Métodos para el cálculo de la cadena representativa</i>	65
<i>Código 3. Función para transformar un vector en \mathbb{N}^n a $\{0,1\}^n$</i>	65
<i>Código 4. Función para determinar el punto p'</i>	66
<i>Código 5. Código para balancear a L</i>	66
<i>Código 6. Optimización de L, analizando la superposición de los elementos</i>	67
<i>Código 7. Fase de entrenamiento</i>	67
<i>Código 8. Pseudo código para el reconocimiento de una imagen M</i>	69

Bibliografía

- [1] R. Duda, M. Hart y D. Stork, *Pattern Classification*, Wiley Interscience; 2nd edition, New York, 2001.
- [2] K. R. Castleman, *Digital Image Processing*, Prentice Hall, New Jersey, 1995.
- [3] R. C. González y R. E. Woods, *Digital Image Processing*, Adison Wesley Publishing Company, Massachusetts, 1995.
- [4] H. Jiménez y J. Figueroa, “Reconocimiento de Caracteres por Cadenas Binarias Aleatorias”, *CNyCIIC2003 Avances en informática y computación Vol. 2*, J. H. Sossa, M. L. Sanchez, J. Polito y S. Fuenlabrada, (editores) XVI Congreso Nacional y II Congreso Internacional de Computación e Informática de la ANIEI, Zacatecas, México, 2003, págs. 125-130.
- [5] F. Zboril, “An Application of Sparse Distributed Memory”, *Proceedings of the ASIS 1997*, 1997, págs. 127-132.
- [6] P. Kanerva, “Encoding Structures in Boolean Space”, *ICAN98: Perspectives in neural computing*, proceedings of the 8th international conference of neural networks, vol. 1, 1998, págs. 387-392.
- [7] Y. Useaka, P. Kanerva y H. Asoh, *Foundations of Real-World Intelligence*, CSL publications, Stanford, 2001.
- [8] G. J. Chaitin y C. J. Van Rijsbergen, *Algorithmic Information Theory*, Cambridge University Press, Cambridge U.K., 1987.
- [9] D. Hochbaum, *Approximation Algorithms for NP- Hard Problems*, PWS Publishing Company, Boston, 1997.
- [10] E. Motwani y P. Raghavay, *Randomized Algorithms*, Cambridge University Press, Cambridge U.K., 1995.
- [11] O. Toby, *Hypercomputation: computing more than the Turing Machine*, Ph. D. Thesis, Department of Philosophy, University of Melbourne, 2000.
- [12] D. M. Jung y G. Nagy, “N Tuple Feature for OCR Revisted”, *IEEE Transactions on pattern analysis and machine intelligence*, vol. 18, no. 7, 1996, págs. 734-745.
- [13] P. Kanerva, *Sparse Distributed Memory*, MIT Press, Cambridge Massachusetts, 1988.
- [14] J. R. Shulcloper, “Formación Integral del Especialista en Reconocimiento de Patrones”, *Reconocimiento de Patrones Avances y Perspectivas, Research on Computer Science, Vol. 1*, J. L. Díaz de León Santiago y C. Yáñez Márquez

- (editores), VII Congreso Iberoamericano de Reconocimiento de Patrones, Centro de Investigación en Computación IPN, México, D.F. 2002, págs. 245-256.
- [15] L. Harry y C. H. Papadimitriou, *Elements of the Theory of Computation*, Prentice Hall, New Jersey, 1988.
- [16] W. S. McCulloch y W. Pitts, “A Logical Calculus of the Ideas Immanent in Nervous Activity”, *Bulletin of mathematical Biophysics*, vol. 5, 1943, págs. 115-133.
- [17] M. Misky y S. Papert, *Perceptrons: an Introduction to Computational Geometry*, 2da edición aumentada, MIT Press, Cambridge Massachusetts, 1972.
- [18] B. M. Del Brío y A. Sanz, *Redes Neuronales y Sistemas Difusos*, Alfaomega-Rama, México, 2002.
- [19] J. R. Hilera y V. J. Martínez, *Redes Neuronales Artificiales, Fundamentos, Modelos y Aplicaciones*, Alfa Omega-Rama, Madrid, 2000.
- [20] A. Velásquez, J. H. Sosa y S. Levachkine, “Reconocimiento Eficiente de Caracteres Alfanuméricos Provenientes de Mapas Ráster por medio de Clasificadores Neuronales”, *Informe Técnico del Centro de Investigación en Computación*, Centro de Investigación en Computación IPN, México, D.F., 2001, serie azul, no. 85.
- [21] A. Velásquez, J. H. Sosa y S. Levachkine, “Estado del Arte sobre la Problemática de la extracción de Texto y de los Componentes contenidos en los Mapas Ráster”, *Informe Técnico del Centro de Investigación en Computación*, Centro de Investigación en Computación IPN, México, D.F., 2001, serie verde, no. 40.
- [22] A. Velásquez, *Localización, Recuperación e Identificación de la Capa de Caracteres, contenida en los Planos Cartográficos*, Tesis Doctoral, Centro de Investigación en Computación IPN, México, D.F. 2002.
- [23] A. Velásquez, H. Sossa y S. Levachkine, “Metodología para el Entrenamiento de un Conjunto de Redes Neuronales para el Reconocimiento de Caracteres Alfanuméricos”, *Memorias del Congreso Internacional de Computación CIC'2000*, 2000, págs. 201-212.
- [24] J. Hopfield, “Neural Networks and Physical Systems with Emergent Collective Computational Abilities”, *Proceedings of the National Academy of Sciences*, vol. 79, 1982, págs. 2554-2558.
- [25] J. Hopfield y W. Tank, “Collective Computation with Continuous Variables”, *Disordered System and Biological Organization*, E. Bienenstock, F. Fogelman Soulie, G. Weisbuch (editores), Springer- Verlag, Berlin, 1985, págs. 150-17.
- [26] J. L. Díaz de León y C. Yáñez, “Memoria Asociativa Hopfield”, *Informe Técnico del Centro de Investigación en Computación*, Centro de Investigación en Computación IPN, México, D.F., 2001, serie verde, no. 52.
- [27] V. K. Steinbuch y F. Helmar, “Die Lernmatrix”, *Kybernetik*, vol. 1, no. 1, 1961, págs. 36-45.

- [28] V. K. Steinbuch y F. Helmar, “Nichtdigitale Lernmatrizen als Perzeptoren”, *Kybernetik*, vol. 1, no. 3, 1961, págs. 117-124.
- [29] J. L. Díaz de León y C. Yáñez, “Lernmatrix de Steinbuch”, *Informe Técnico del Centro de Investigación en Computación*, Centro de Investigación en Computación IPN, México, D.F., 2001, serie verde, no. 48.
- [30] J. A. Anderson, “A Simple Neural Network generating an Interactive Memory”, *Mathematical Biosciences*, vol. 14, 1972, págs. 197-220.
- [31] J. L. Díaz de León y C. Yáñez, “Linear Associator de Anderson Kohonen”, *Informe Técnico del Centro de Investigación en Computación*, Centro de Investigación en Computación IPN, México, D.F., 2001, serie verde, no. 50.
- [32] D. Willshaw, O. Buneman y H. Longuet Higgins, “Non-holographic Associative Memory”, *Nature*, no. 222, 1969, págs. 960-962.
- [33] C. Yáñez, *Memorias Asociativas basadas en Relaciones de Orden y Operadores Binarios*, Tesis Doctoral, Centro de Investigación en Computación IPN, México, D.F., 2000.
- [34] J. L. Díaz de León y C. Yáñez, “Memoria Morfológica Hetero Asociativa”, *Informe Técnico del Centro de Investigación en Computación*, Centro de Investigación en Computación IPN, México, D.F., 2001, serie verde, no. 57.
- [35] J. L. Díaz de León y C. Yáñez, “Memoria Morfológica Auto Asociativa”, *Informe Técnico del Centro de Investigación en Computación*, Centro de Investigación en Computación IPN, México, D.F., 2001, serie verde, no. 58.
- [36] J. Austin, “ADAM: A Distributed Associative Memory for Scene Analysis”, *In Proceedings of First International Conference on Neural Networks*, M. Caudhill y C. Butler (editores), San Diego, 1987, págs. 285-295.
- [37] J. L. Díaz de León y C. Yáñez, “Modelo ADAM”, *Informe Técnico del Centro de Investigación en Computación*, Centro de Investigación en Computación IPN, México, D.F., 2001, serie verde, no. 54.
- [38] B. Kosko, “Bidirectional Associative Memories”, *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 18, no. 1, 1988, págs. 49-60.
- [39] J. L. Díaz de León y C. Yáñez, “Modelo BAM”, *Informe Técnico del Centro de Investigación en Computación*, Centro de Investigación en Computación IPN, México, D.F., 2001, serie verde, no. 55.
- [40] J. L. Díaz de León y C. Yáñez, “Modelo SDM”, *Informe Técnico del Centro de Investigación en Computación*, Centro de Investigación en Computación IPN, México, D.F., 2001, serie verde, no. 56.
- [41] J. L. Potter, *Associative Computing: A programming Paradigm for Massively Parallel computers*, Kluwer Academic/Plenum Publishers, New York, 1992.

- [42] M. Davis, *The Universal Computer: The Road from Leibniz to Turing*, W.W. Norton & Company, New York, 2000.
- [43] J. Copeland y R. Silvan, “Beyond the Universal Turing Machine”, *Australasian Journal of Philosophy*, vol. 77, 1999, págs. 46-66.
- [44] A. M. Turing, “On Computable Numbers: With an Application to the Entscheidungsproblem”, *Proceedings of the London Mathematical Society*, serie 2, vol. 42, 1936, págs. 230 - 265.
- [45] P. Odifreddi, *Classical Recursion Theory*, North-Holland, Amsterdam, 1992.
- [46] R. Hartley, *Theory of Recursive Functions and Effective Computability*, MIT Press, Cambridge Massachusetts, 1987.
- [47] R. B. Ash, *Information Theory*, Dover publications, New York, 1990.
- [48] R.E. Walpole, R.H. Myers y S.L. Myers, *Probabilidad y Estadística para Ingeniería*, Prentice Hall Hispanoamericana, México, 1999.
- [49] H. Jiménez y J. Figueroa, “Optimization of Random Sampling for Character Recognition using Larges Binaries Strings”, *Proceedings World Multiconference on Systemics, Cybernetics and Informatics Vol. 2.*, 8th World Multi-Conference on Systemics, Cybernetics and Informatics, Orlando, Florida, 2004, págs. 468-473.