

**Centro de Investigación y Estudios Avanzados del
Instituto Politécnico Nacional**

Unidad Zacatenco

Departamento de Ingeniería Eléctrica

Sección de Computación

**Simulación paralela de redes neuronales
aplicadas al proceso de visión temprana ¹**

Tesis que presenta el

Lic. Eloy Juan Cessa López

Para obtener el grado de

Maestro en Ciencias

En la especialidad de

Ingeniería Eléctrica

Opción

Computación

Director de tesis:

Dr. Arturo Díaz Pérez

Ciudad de México, México.

¹Este trabajo fue parcialmente financiado mediante el proyecto CONACyT 31892-A: Algoritmos y arquitecturas de computadoras con dispositivos programables.

Agradecimientos

A **CONACyT**, por haberme apoyado otorgándome la beca de maestría para alcanzar esta meta de mi vida.

A **CINVESTAV**, por brindarme la oportunidad y las facilidades para realizar esta investigación que culmina mis estudios de maestría.

A **mi asesor**, el Dr. Arturo Díaz, quien me guió durante el desarrollo de esta tesis con disciplina, dedicación, constancia y esmero. A quien agradezco por impulsarme a mejorar mi nivel profesional y personal persiguiendo este objetivo mutuo.

A **mis sinodales**, los Drs. Gerardo de la Fraga y Guillermo Morales, por colaborar en beneficio de la calidad de este reporte.

A **mis padres**, Eloy y María de la Luz, quienes una vez más fueron parte fundamental en el cumplimiento de una meta más. Sin duda alguna, gracias a ellos soy lo que soy y vivo lo que vivo.

A **mis hermanos**, por mostrarme de forma continua su apoyo y aprecio. Por el ejemplo que son cada uno de ellos para mí, a pesar de nuestros estilos diferentes de vida.

A **mis amigos**, quienes hombro con hombro nos mantuvimos firmes, día y noche, días hábiles y festivos, luchando por completar este trayecto en común que todos escogimos. Gracias por darle alegría, emoción, distracción, compañía, apoyo, lealtad, etc. a estos dos años de esfuerzo, entrega, lucha y obsesión por concluir nuestros estudios de maestría.

Resumen

El proceso de visión humano se puede dividir en dos etapas: *visión temprana* y *reconocimiento*. En la primera etapa del proceso de visión se generan algunas ilusiones ópticas en algunas imágenes, como consecuencia de la forma en que están organizadas las células receptoras de la retina humana y la interacción que existe entre ellas. Una de estas ilusiones es la aparición de manchas en el área delimitada por cuatro cuadros negros. En este trabajo se presenta un simulador del proceso de visión temprana capaz de simular una ilusión óptica formada por la interacción de células receptoras en la retina. Para simular este efecto, el proceso de visión temprana se dividió en dos grandes partes: *recepción de escenas visuales* e *interacción de las células receptoras*. Para la recepción de escenas visuales se requieren dos módulos: 1) un generador de arreglos hexagonales de receptores, para producir arreglos artificiales apegados a la estructura de los arreglos de células receptoras de la retina humana; y 2) un proyector de imágenes a arreglos hexagonales, para distribuir la imagen de entrada sobre los arreglos hexagonales, pretendiendo simular la diseminación de las escenas visuales que llegan al ojo sobre los arreglos de células receptoras. Al ser recibida una escena visual por la retina, ésta la transforma en una señal eléctrica que se envía al cerebro. Durante este proceso de transformación, las células de la retina interactúan entre sí provocando la aparición de ilusiones. Para la simulación de la retina, cada célula de visión, un bastón, se simula con una célula artificial llamada perceptrón. Para organizar la interacción entre perceptrones, se aplicaron tres vecindades diferentes correspondientes a 3 modelos propuestos con los que se pretende explicar la ilusión mencionada. Estos modelos son: 1) vecindad hexagonal de 7 elementos, 2) vecindad circular con radio arbitrario y 3) vecindad elipsoidal. Durante la interacción de las células, algunas producen una influencia excitatoria y otras inhibitoria, este fenómeno es conocido como inhibición lateral, el cual se reproduce con la aplicación de 4 funciones: exponencial, Sinc, Bessel j_0 y Bessel j_1 , por su comportamiento similar al del fenómeno. Para una simulación efectiva del proceso de visión temprana se requieren manejar arreglos muy grandes de perceptrones (del orden de 10^8 perceptrones, aproximadamente), por lo cual se consume una gran cantidad de tiempo. Para obtener un simulador eficiente de éste proceso, se desarrolló un simulador paralelo con una estrategia de particionamiento por bloques, con el cual se reduce el tiempo de respuesta.

Abstract

Human vision can be divided in two main stages: *early vision*, and *recognition*. In the first stage some optical illusions are generated as a consequence of how receptor cells of the retina are structured and how they interact each other. One of the most studied illusion is that which generates some dark spots in regions among four equidistant dark rectangles. In this report an early vision simulator is being described which is able to reproduce the optical illusion mentioned before by simulating the interaction among light receptors in the retina. Two main parts are included in the early vision simulator: image acquisition and formation, and inhibition and excitation receptor interaction. For the image acquisition and formation two parts were developed: 1) a generator of hexagon receptor arrays, which generates artificial arrays with a similar structure to the receptors in the human retina; and 2) an image projector to hexagonal arrays, which distributes an input image over receptor hexagonal arrays. Once an input image is obtained by the retina, this is transformed to electrical impulses what are propagated to the brain. Receptors in the retina interact among them and this process produce optical illusions. Each receptor in the retina, a rod, is simulated with an artificial cell that is called "perceptron". Three different types of local regions were used in which interactive perceptrons are searched: 1) hexagonal vicinity, 2) circular vicinity, and 3) ellipsoid vicinity. In each kind of region, four transfer functions can be applied to simulate the inhibitory-excitatory phenomenon: exponential, Sinc, Bessel J_0 , and Bessel J_1 . Finally, due the large number of perceptrons (10^8 approximately) required to perform an effective simulation, and therefore a high computational time we developed a simulator's parallel parallel with a block partition strategy which effectively reduces the calculation times.

Índice General

Introducción	xvii
1 Proceso de Visión Temprana	1
1.1 La visión humana	1
1.2 Visión temprana	2
1.2.1 La retina humana	2
1.2.2 Inhibición lateral	3
1.3 Efectos visuales	6
1.4 Un modelo simple	9
2 Redes Neuronales para la Simulación Paralela de la Retina Humana	13
2.1 Redes neuronales	13
2.1.1 La estructura del sistema nervioso central	14
2.1.2 Introducción a las redes neuronales artificiales	16
2.2 Modelos neuronales para la simulación de la retina	18
2.3 Redes neuronales paralelas	25
3 Recepción de Escenas Visuales	27
3.1 Generador de arreglos hexagonales de receptores	27
3.1.1 Forma 1: arreglos regulares con círculos simétricos y puntos regulares	28
3.1.2 Forma 2: arreglos regulares con círculos simétricos y puntos irregulares	30
3.1.3 Forma 3: arreglos concéntricos con círculos y puntos regulares	30
3.1.4 Forma 4: arreglos concéntricos con círculos perturbados y puntos regulares	31
3.2 Proyección de la imagen de entrada	32
3.3 Imagen artificial	35

4 Modelos de Interacción de Perceptrones	37
4.1 Convolución entre funciones	37
4.2 Modelo de red neuronal propuesto	38
4.3 Vecindad hexagonal de 7 elementos	43
4.4 Vecindad circular	44
4.4.1 Circunferencia: conceptos y aplicación	44
4.4.2 Cálculo del área de intersección entre dos circunferencias	46
4.4.3 Aplicación de la función de transferencia	51
4.5 Vecindad elipsoidal	52
4.5.1 Elipse: conceptos y aplicación	52
4.5.2 Aplicación de la función de transferencia	54
5 Paralelización del Sistema	57
5.1 Análisis del programa secuencial	57
5.2 Estrategia de particionamiento	59
5.3 Orquestación	61
5.3.1 Intercambio de mensajes	62
5.3.2 Modelo maestro-esclavo	63
5.3.3 Sincronización	63
5.3.4 Comunicación	65
5.4 Mapeo	65
5.5 Algoritmo paralelo general	66
6 Análisis y Evaluación de Resultados	69
6.1 Patrones de prueba	69
6.2 Resultados sobre el tipo de vecindad	71
6.2.1 Vecindad hexagonal	73
6.2.2 Vecindad circular	74
6.2.3 Vecindad elipsoidal	75
6.3 Comparación de los tres tipos de vecindades	77
6.4 Pruebas de los diferentes elementos del simulador	77
6.4.1 Aplicación de las cuatro funciones de transferencia	78
6.4.2 Evaluación de las formas de los arreglos de perceptrones	79
6.4.3 Efecto de la resolución en el proceso de simulación	80
6.4.4 Variación de los valores de los elementos de las elipses	81

6.5	Efecto de la rotación de la imagen de entrada	82
6.6	Tiempos de ejecución	83
	Conclusiones	87
	Referencias	91

Índice de Figuras

1	La ilusión de cuadros negros caracterizada por la aparición de manchas en el área delimitada por cuatro cuadros negros.	xviii
1.1	Fases del proceso de visión humano.	2
1.2	Microfotografía de células fotorreceptoras de un humano, vistas desde arriba. Aproximadamente, 125 millones de celdas en cada ojo.	2
1.3	Estructuras del ojo y la retina.	3
1.4	Conexión de un conjunto de receptores a una célula ganglionar.	4
1.5	Respuesta de una célula ganglionar por excitación de luz.	5
1.6	Experimento con una luz que se enciende sobre un receptor conectado a una célula ganglionar, se apaga y se enciende sobre otro.	5
1.7	Experimento con una luz que se enciende sobre un receptor conectado a una célula ganglionar, sin apagarla se enciende otra luz sobre otro receptor.	6
1.8	Organización de una vecindad de receptores, conectada a una célula ganglionar, en unidades referidas como on-center y off-center.	7
1.9	Ejemplo de las ilusiones producidas en el cerebro. Para observar la imagen del perro dálmata es necesario agregar la información no presentada.	8
1.10	Ejemplo de las ilusiones producidas en la retina. Los cuadros internos de la figura aparecen con tonos de gris diferentes, cuando tienen el mismo.	8
1.11	Ilusión generada en la etapa de visión temprana. Se perciben manchas en el área localizada entre cuatro cuadros negros, las cuales no existen en la imagen original.	8
1.12	Ejemplos de la ilusión de Hermann.	9
1.13	Respuesta a la luz de un receptor influenciado por sus vecinos.	10
1.14	Caso más simple de vecindad de perceptrones.	10
1.15	Arreglo más real para los receptores.	11
2.1	Dibujo de dos tipos de neuronas mostrando los diferentes tipos de empalmes synápticos.	14

2.2	Perceptrón simple, consiste de dos capas de neuronas. Las neuronas de entrada alimentan a las neuronas de las capas de salida, pero no viceversa.	17
2.3	Una correspondencia uno a uno existe entre los enrejados de spins magnéticos con dos orientaciones diferentes y redes neuronales de McCulloch-Pitts. Un spin apuntando hacia arriba es identificado como una neurona activa.	17
2.4	Una red profunda con una capa interna (oculta) de neuronas.	18
2.5	Modelo del circuito neuronal de la retina externa.	21
2.6	Ilusiones geométricas famosas. (a) Ilusión de Muller-Lyer, (b) ilusión de Hering, (c) ilusión de Zeollnor.. . . .	22
2.7	Modelo propuesto.	22
2.8	Función DOG definida por el ángulo θ	23
2.9	Procedimiento de expansión de ángulos agudos del modelo propuesto. (a) Figura de entrada dividida en sub-imágenes, (b) líneas rotadas en la sub-imagen con la intersección, (c) líneas trasladadas por la unificación.	24
2.10	Imagen de entrada y sub-imágenes divididas en la simulación.	24
2.11	Resultados de la simulación de la ilusión de Hering. (a) Imagen de entrada, (b) imagen de salida.. . . .	24
3.1	Arreglos hexagonales de receptores para simular la retina humana.	28
3.2	Vecindad hexagonal: Un elemento central con 6 vecinos, dos arriba, dos abajo y uno en cada costado.	29
3.3	Especificación de las formas para los arreglos de receptores.	29
3.4	Perspectivas lógica y física de la forma 1.	29
3.5	Perspectivas lógica y física de la forma 4 de los arreglos hexagonales.	32
3.6	Casos de la proyección de una imagen de entrada a un arreglo hexagonal de receptores. Arriba, caso 1, la resolución de la imagen de entrada es menor que la resolución del arreglo de receptores. Abajo, caso 2, la resolución del arreglo de receptores es mayor que la de la imagen de entrada.	32
3.7	Generación de la imagen artificial a partir del arreglo de receptores.	36
4.1	Ilustración gráfica de la convolución. Las áreas intersectadas indican las regiones donde el producto no es <i>cero</i>	39
4.2	Modelo de red neuronal propuesto.	39
4.3	Gráficas descriptivas de las funciones utilizadas por los modelos de interacción de perceptrones.	41

4.4	Vecindad de 3×3 de un punto (x, y) en una imagen.	42
4.5	Modelo matemático para agrupar los perceptrones: vecindad hexagonal de 7 elementos, p_0 es el perceptrón central y p_{1-6} los vecinos.	43
4.6	Modalidades de la vecindad hexagonal de 7 elementos para la determinación de los pesos de los perceptrones.	43
4.7	Modelo matemático para agrupar los perceptrones: vecindad circular, un perceptrón central y un variado número de vecinos.	44
4.8	Elementos de una circunferencia.	45
4.9	Área de intersección entre dos circunferencias.	46
4.10	Casos para el cálculo del área de intersección entre dos circunferencias.	46
4.11	Cálculo del área de intersección entre dos circunferencias.	47
4.12	Elementos requeridos para el cálculo del área de intersección de dos circunferencias.	48
4.13	Traslación del eje de coordenadas al centro de C_1 y Rotación del eje de coordenadas para alinear los centros de las circunferencias al eje de las Xs.	50
4.14	Modelo matemático para agrupar los perceptrones: vecindad elipsoidal. Un perceptrón central y un variado número de vecinos. a) Elipse a 45^0 . b) Elipse a 135^0 . c) Elipse doble.	52
4.15	Elementos de una elipse.	53
4.16	Rotación del eje coordenado para la rotación de una elipse.	54
4.17	Condiciones para la selección de vecinos contenidos en la elipse.	54
4.18	Área de búsqueda para la vecindad elipsoidal.	55
4.19	División de la elipse en tres circunferencias para calcular el porcentaje de contención de un perceptrón por la elipse de la vecindad elipsoidal.	55
5.1	Gráfica de tiempos de ejecución del simulador secuencial.	58
5.2	Estrategia de particionamiento para la etapa de interacción de los perceptrones. Particionamiento por Bloques.	60
5.3	Código para encontrar los índices físicos inicial y final de cada bloque.	60
5.4	Grupos de listas agregadas por la posición de los bloques. A la izquierda, los bloques externos sólo agregan un grupo de listas. A la derecha, los bloques intermedios agregan dos grupos de listas.	61
5.5	Comunicación entre las particiones del arreglo hexagonal de perceptrones.	63
5.6	Comunicación entre las particiones del arreglo hexagonal de perceptrones.	65
6.1	Formato de la imagen de entrada utilizada para las pruebas del modelo propuesto.	70

6.2	Imágenes de salida al convolucionar las 7 modalidades de la vecindad hexagonal de 7 elementos con la forma 1 de los arreglos.	73
6.3	Gráfica resultante al aplicar la modalidad f) de la vecindad hexagonal de 7 elementos sobre la forma 1 de los arreglos.	74
6.4	Imágenes de salida al aplicar la vecindad circular sobre la forma 4 de los arreglos hexagonales de perceptrones y aplicando la función exponencial.	75
6.5	Gráfica resultante al aplicar la vecindad circular con la estrategia de selección de vecinos proporcionales sobre la forma 4 de los arreglos y aplicando la función exponencial.	75
6.6	Imágenes de salida al aplicar la vecindad elipsoidal con la Forma 4 de los arreglos, utilizando la función exponencial y utilizando la estrategia de selección vecinos completos	76
6.7	Gráfica resultante al promediar los resultados de aplicar las elipses rotadas a 45^0 y 135^0 utilizando la función exponencial y la estrategia de selección vecinos completos	76
6.8	Comparación de las imágenes de salida de aplicar las vecindades hexagonal, circular y elipsoidal.	77
6.9	Imágenes de salida al aplicar la vecindad elipsoidal sobre la forma 4 de los arreglos, aplicando la estrategia de selección vecinos completos y utilizando a) la función exponencial; b) la función Sinc; c) la función Bessel j_0 ; y d) la función Bessel j_1 . . .	78
6.10	Imágenes de salida al aplicar la vecindad elipsoidal sobre las cuatro formas de los arreglos, aplicando la estrategia de selección vecinos completos y utilizando la función Bessel j_0	79
6.11	Imágenes de salida al aplicar la vecindad elipsoidal doble , utilizando la función Bessel j_0 , la estrategia vecinos completos , y resoluciones de la imagen de entrada igual y mayores a la resolución de arreglo de perceptrones.	80
6.12	Imágenes de salida al aplicar la vecindad elipsoidal doble sobre la forma 4 de los arreglos, utilizando la función Bessel j_0 , la estrategia de selección vecinos completos y tres combinaciones diferentes para la amplitud de la función y los ejes mayor y menor de la elipse.	81
6.13	Gráfica resultante al aplicar la vecindad elipsoidal doble , utilizando la función Bessel j_0 , la estrategia vecinos completos y la combinación $a = 0.45d$, $b = 0.06d$ y $\sigma = 0.01d$	82

6.14	Imágenes de salida al aplicar la vecindad elipsoidal doble sobre la forma 4 de los arreglos, utilizando la función Bessel j_0 , la estrategia de selección vecinos completos , resolución de la imagen de entrada de 4000×4000 pero, rotada a diferentes grados.	83
6.15	Gráfica de tiempos de ejecución. Secuencial vs. Paralelos.	85
6.16	Gráfica de aceleración. Ideal vs. Normalizada.	86

Índice de Tablas

4.1	Funciones de transferencia utilizadas por los modelos de agrupación de perceptrones.	40
5.1	Tiempos de ejecuciones secuenciales de las dos etapas del simulador: recepción de escenas visuales e interacción de los receptores.	58
6.1	Tabla de conversiones de las medidas utilizadas en las imágenes de entrada y las vecindades de los modelos de agrupación.	71
6.2	Tabla de conversiones de las medidas utilizadas en los elementos de las las vecindades de los modelos de agrupación.	71
6.3	Elementos de prueba del simulador.	72
6.4	Equivalencias de los valores de los elementos de las elipses (a , b) y la amplitud de la función (σ) para probar el efecto de la resolución.	80
6.5	Características a considerar para la ejecución paralela del simulador.	84
6.6	Tiempos de la ejecución paralela y su aceleración.	84

Introducción

El cuerpo humano, su composición, comportamiento y evolución, es un campo de estudio amplio y vigente. La necesidad del ser humano por conocer el funcionamiento de su organismo ha producido diferentes y grandes líneas de investigación que progresivamente van explicando, modelando o descubriendo los diferentes procesos que utiliza el cuerpo humano para su funcionamiento.

Un proceso interesante del cuerpo humano es su proceso de visión. A pesar de los muchos años invertidos en su estudio, existen mecanismos y funciones que aún son un misterio. Reiteradamente, el sistema de percepción humano comete errores: algunas veces produce ilusiones, algunas veces el sistema de percepción requiere de tiempo para recuperarse de una estimulación prolongada, y algunas veces requiere tiempo para interpretar la imagen que se le presenta. El estudio de estos fenómenos es importante porque los mecanismos de un sistema, con frecuencia, son revelados principalmente a través de sus errores y distorsiones [1].

Las ilusiones perceptuales son robustas e inevitables. Uno puede aprender a “compensar” algunas de ellas pero no puede aprender a dejar de verlas. Esta inevitabilidad sugiere que algunas ilusiones se originan en la “primera etapa” del procesamiento visual [2].

El proceso de visión humano se divide en dos etapas: la *visión inicial*, es el proceso que recibe la imagen de entrada y disemina las intensidades de luz recibidas de dicha imagen al arreglo de células receptoras (retina humana) del ojo humano para transformarlas en una señal eléctrica y enviarla al cerebro; el *reconocimiento*, es el proceso que recibe la información transmitida por el ojo humano y utilizando la memoria responde con un significado de la imagen de entrada.

En ambas etapas del proceso visual se generan ilusiones. Las ilusiones del cerebro se producen por condiciones de experiencia y conocimiento; las ilusiones producidas en la retina se producen por la organización de los elementos de procesamiento y su interacción obligada por su estructura [2].

En la actualidad existen diversos trabajos que proponen modelos y estructuras para la simulación de la retina humana y el proceso de visión. Por ejemplo, en [3], se presenta un circuito VLSI analógico para emular la visión biológica, para lo cual se propuso un modelo de red analógico lineal para describir el circuito neuronal de la retina externa. En [4], se presenta un estudio interesante sobre la estructura y comportamiento de la retina de mosquitos (dípteros) proponiendo un modelo

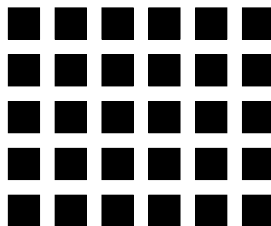


Figura 1: La ilusión de cuadros negros caracterizada por la aparición de manchas en el área delimitada por cuatro cuadros negros.

para la obtención de altas resoluciones basándose en las características de las retinas de estos animales. Otro trabajo interesante se presenta en [5], donde se pretende generar ilusiones geométricas con base en la expansión de ángulos agudos realizada en las células de orientación de la corteza cerebral.

Estos trabajos muestran las tendencias y el interés para explicar el proceso de visión humano. Sin embargo, a pesar de la gran diversidad de estudios realizados aún existen fenómenos de este proceso que no se pueden explicar con los modelos y/o estructuras propuestos. Una de estas ilusiones es generada en la retina, y se identifica por la aparición de manchas en el área delimitada por cuatro cuadros negros, la cual se denominó “ilusión de cuadros negros”. Esta ilusión aparece debido a la forma en que están organizadas las células receptoras (bastones y conos) de la retina humana y la interacción que existe entre ellas.

El objetivo de esta tesis es desarrollar un modelo de red neuronal artificial para explicar la ilusión de cuadros negros (ver Figura 1). Para esto es necesario alcanzar los siguientes objetivos particulares:

- Diseñar e implantar un algoritmo que simule la organización y el comportamiento de la retina del ojo humano mediante el uso de arreglos de perceptrones.
- Desarrollar algunos modelos matemáticos para la organización de perceptrones que expliquen la formación del efecto visual generado por la interacción entre sí de los bastones de la retina humana que resulta en la aparición de manchas inexistentes en una imagen de entrada.
- Aplicar la estrategia de particionamiento por bloques para los cálculos que realiza el simulador de redes neuronales para reducir su tiempo de respuesta al utilizar arreglos de perceptrones muy grandes.

El simulador está dividido en dos módulos principales: la *recepción de escenas visuales* y la *interacción de las células receptoras*.

En el módulo de recepción de escenas visuales se generan los arreglos hexagonales de receptores que reproducen la retina humana, los cuales reciben la imagen de entrada a través de una proyección de los pixels de la imagen. Este módulo tiene como salida un arreglo de receptores con tonos de gris determinados por las intensidades de los pixels de la imagen de entrada.

Cada célula receptora de la retina, un bastón, se simuló con una célula artificial llamada perceptrón. Para simular la interacción entre las células receptoras, los perceptrones se agruparon a través de tres modelos: 1) Vecindad hexagonal de 7 elementos, 2) Vecindad circular, y 3) Vecindad elipsoidal. Con estos modelos se pretende generar la ilusión óptica planteada mediante la aplicación de las vecindades de estos modelos sobre los arreglos hexagonales de receptores excitados con los tonos de gris de la imagen de entrada.

La generación de las vecindades de los modelos de agrupación de perceptrones está determinada por la inhibición lateral la cual fue simulada con cuatro funciones de transferencia: exponencial, Sinc, Bessel j_0 y $Besselj_1$.

La simulación de la retina humana a través de redes neuronales con arreglos muy grandes de perceptrones, del orden de 10^8 perceptrones, requiere un tiempo de procesamiento muy grande. Para reducir el tiempo de esta simulación se aplicaron herramientas de programación paralela para resolver el problema en varios procesadores aplicando la estrategia de particionamiento por bloques para mejorar el rendimiento del simulador.

Este reporte de tesis está organizado de la siguiente manera:

Capítulo 1. Proceso de visión temprana. Se presenta un estudio sobre el proceso de visión humano, donde se describen conceptos básicos y se plantea el fenómeno a simular.

Capítulo 2. Redes neuronales para la simulación de la retina humana. Se presenta un estudio sobre las redes neuronales y su simulación en paralelo. Se describen conceptos básicos y se presentan algunos trabajos actuales relacionados con el tema.

Capítulo 3. Recepción de escenas visuales. Se muestra el *generador de arreglos hexagonales de receptores*, los arreglos utilizados para la simulación de la retina y su generación. Además, se presenta el *proyector de imágenes sobre arreglos hexagonales de receptores*, explicando el proceso de proyección de los elementos de una imagen de entrada sobre diferentes áreas de un arreglo hexagonal.

Capítulo 4. Modelos de interacción de perceptrones. En esta sección se presentan los diferentes modelos utilizados para la generación de las ilusiones buscadas.

Capítulo 5. Paralelización del simulador. Se describe la paralelización del sistema y la estrategia de paralelización utilizada.

Capítulo 6. Análisis y evaluación de resultados. Aquí se presentan los resultados de las pruebas realizadas con las diferentes formas de arreglos y modelos.

Capítulo 1

Proceso de Visión Temprana

Introducción

La visión humana es un proceso complejo. A pesar de los numerosos estudios realizados en esta área aún hay muchas incógnitas. En este capítulo se presentan los mecanismos de percepción del ojo humano, se describen sus elementos, y se hace un estudio de los diferentes efectos visuales en imágenes arregladas especialmente de tal manera que se perciban aquellos no existentes en la imagen de entrada pero que el ojo humano, por su comportamiento, los observa.

1.1 La visión humana

El ojo humano es la antesala de la visión humana. La información obtenida en la retina ocular es transmitida al cerebro para ser interpretada pero, ¿cómo se realiza la percepción de alguna escena visual? ¿Cómo podemos decir después de observar una imagen, que dicha imagen está compuesta por alguna forma en particular? Empezaremos por conocer el proceso de visión humano.

El proceso de visión humano se divide en dos etapas (ver Figura 1.1). La primera etapa, llamada *Visión Temprana* o (Primary Vision), se lleva a cabo en la retina humana, y es donde se realiza el primer proceso de visión; y la segunda etapa, llamada *Reconocimiento* o *Interpretación*, se lleva a cabo en el cerebro, y es donde el cerebro utiliza información almacenada en la memoria para procesar la información recibida de la retina y darle algún significado [1].

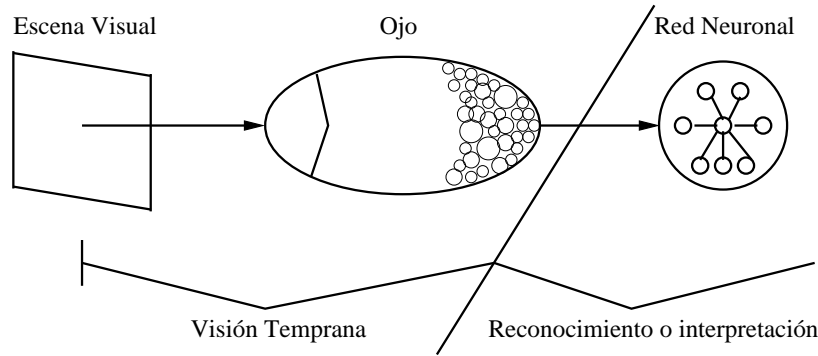


Figura 1.1: Fases del proceso de visión humano.

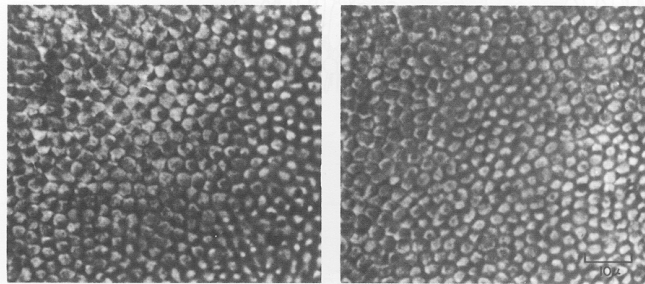


Figura 1.2: Microfotografía de células fotorreceptoras de un humano, vistas desde arriba. Aproximadamente, 125 millones de celdas en cada ojo.

1.2 Visión temprana

La visión temprana o primer proceso parte desde el momento que la luz es reflejada sobre la escena visual y es recibida por un gran número de células fotorreceptoras (bastones y conos) ubicadas en la capa externa de la retina, sobre las cuales la luz se dispersa. En la Figura 1.2 se muestran en una microfotografía receptores humanos y la dispersión de las intensidades de luz sobre ellos.

Las células fotorreceptoras (denominadas receptores) interactúan entre ellas al recibir la luz y afectan a sus células vecinas con la respuesta que producen a la intensidad de luz que las excita.

Para explicar con mayor profundidad el fenómeno de influencia entre receptores, es conveniente estudiar los elementos que conforman la retina humana y su relación entre sí.

1.2.1 La retina humana

La *retina* es inicialmente una parte del cerebro. Llega a ser separada del cerebro durante el desarrollo fetal, pero permanece conectada a él a través del nervio óptico. La retina consiste de tres capas de células nerviosas. La capa exterior consiste de fotorreceptores, conos y bastones, los cuales convierten la luz en señales eléctricas. Los *bastones* nos permiten ver a media luz, mientras los

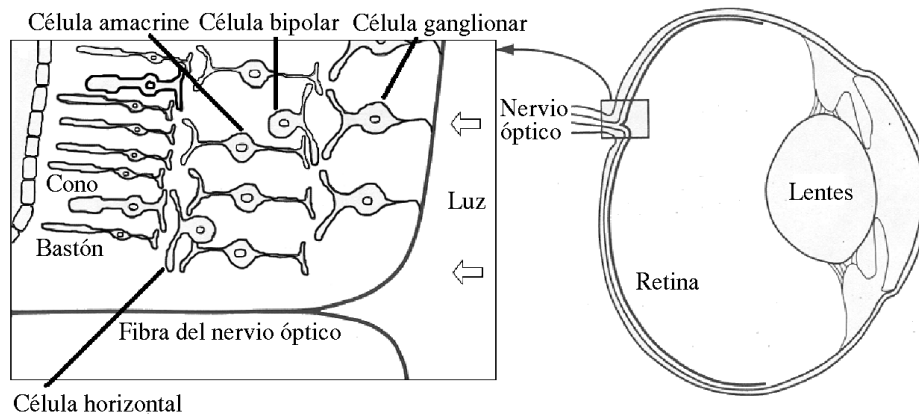


Figura 1.3: Estructuras del ojo y la retina.

conos nos permiten ver detalles finos y color. Por razones no completamente comprendidas la luz debe pasar por otras dos capas de la retina en orden para estimular los bastones y los conos.

La capa intermedia de la retina consiste de tres tipos de células: células bipolares, células horizontales y células amacrines. Las células bipolares reciben la entrada de los receptores y la envían a la tercer capa de la retina. Las células horizontales ligan los receptores y las células bipolares, y las células amacrines ligan las células bipolares con las células ganglionares.

La capa final de la retina contiene las células ganglionares. Las axons (fibras nerviosas de conexión entre células nerviosas) de las células ganglionares pasan a través de la superficie de la retina y se empalman unas con otras para formar el nervio óptico. Es interesante notar que cada ojo contiene aproximadamente 125 millones de receptores, pero sólo un millón de células ganglionares. Claramente hay un procesamiento significativo en la retina para realizar la reducción de datos. En la Figura 1.3 se puede observar esta estructura de la retina.

1.2.2 Inhibición lateral

Las células receptoras y las células ganglionares son elementos importantes para el procesamiento en la visión temprana. Estos dos elementos no son independientes ni opuestos, sino que mantienen una estrecha relación, de tal manera que, el funcionamiento de uno depende del funcionamiento del otro y viceversa. Su existencia está ligada a la interacción de ambas células.

Uno de los fenómenos que aparecen en la etapa de la visión temprana, es la obtención de ilusiones que no existen en la imagen original pero que el ojo observa debido a la interacción entre las células receptoras. La interacción entre las células receptoras se genera por la diferencia que existe entre la cantidad de estas células y las células ganglionares, como las receptoras superan por mucho a las ganglionares, muchas de ellas se conectan a una sola célula ganglionar, por ello, la entrada que

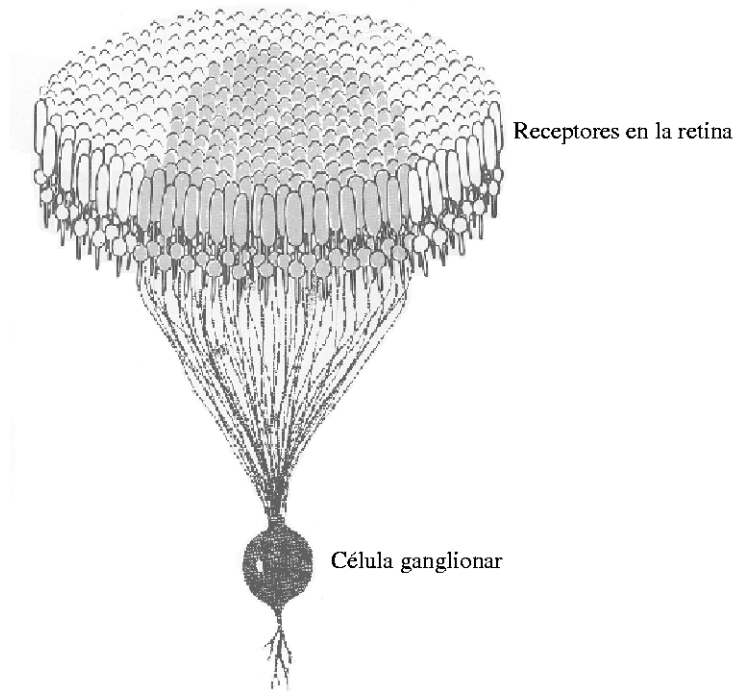


Figura 1.4: Conexión de un conjunto de receptores a una célula ganglionar.

esta última recibe es una combinación de las respuestas de los receptores conectados a ella (ver Figura 1.4). Para poder explicar o entender este fenómeno es necesario comprender la interacción entre ambos tipos de células, aunque no se pretenda explicar el mecanismo en el cual el cerebro utiliza las células ganglionares para sus fines de reconocimiento o comprensión.

P. H. Lindsay y D. A. Norman, en [1], presentan un experimento realizado sobre la retina de un *Limulus* o cangrejo de herradura, el cual tiene un ojo fácilmente accesible con fibras nerviosas largas y fáciles de disecar. Con este experimento, Lindsay y Norman muestran la interacción entre sí de las células receptoras y la interacción de éstas con las ganglionares.

El experimento se basa en la grabación de las respuestas de una célula ganglionar, que tiene como entrada las respuestas de varios receptores, a través de un electrodo. Las respuestas del electrodo son conectadas a la entrada de un osciloscopio para graficarlas.

El primer paso del experimento es disecar el ojo de tal forma que los receptores individuales puedan ser estimulados directamente sin usar lentes ni materiales biológicos. Después, el electrodo se coloca en una célula ganglionar y se enciende una luz.

El resto del experimento consta de la grabación de las respuestas de la célula ganglionar al estimular sus receptores. Tal como lo presentan los autores: Supongamos que se quiere observar la respuesta de una célula neuronal cuando la luz es dirigida hacia el receptor etiquetado con A en la

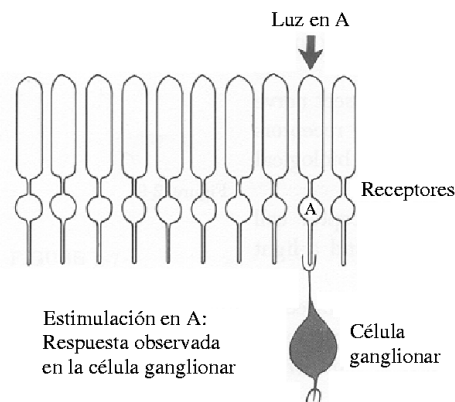


Figura 1.5: Respuesta de una célula ganglionar por excitación de luz.

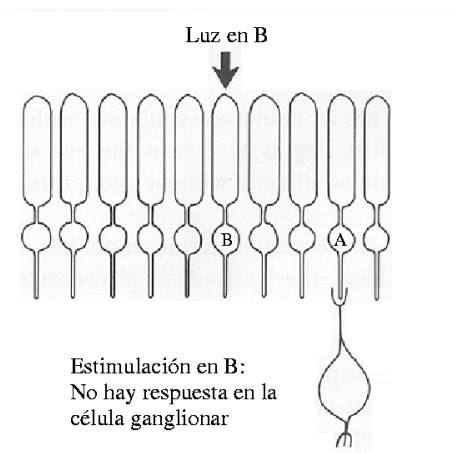


Figura 1.6: Experimento con una luz que se enciende sobre un receptor conectado a una célula ganglionar, se apaga y se enciende sobre otro.

Figura 1.5. Si la respuesta neuronal de la célula neuronal se incrementa, entonces aparentemente esta célula está monitoreando la actividad en el receptor *A*.

Supongamos que la luz es movida al receptor *B* y se enciende (ver Figura 1.6). Esta vez, la grabación del electrodo no cambia en la actividad de la célula ganglionar. Aparentemente, el receptor *B* no está influenciando a la célula neuronal en *A*.

Consideremos qué sucede con una combinación de luces. Comencemos de nuevo con una luz en *A*. Luego encendemos una segunda luz en *B*. La respuesta neuronal de la célula ganglionar **decrece**. Ver Figura 1.7.

Este experimento simple demuestra uno de los mecanismos más importantes del procesamiento de información sensorial: la *inhibición lateral*. La actividad en un receptor es modificada por la actividad de sus vecinos.

Para más información sobre el uso de electrodos para la grabación de las respuestas de las células

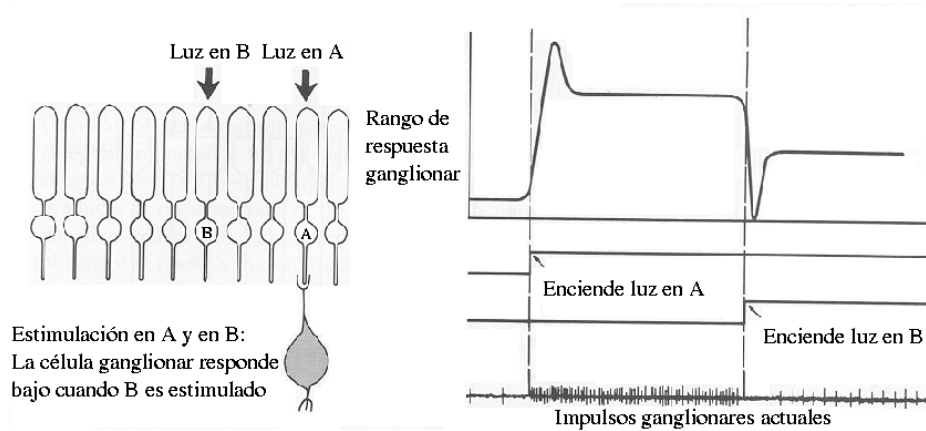


Figura 1.7: Experimento con una luz que se enciende sobre un receptor conectado a una célula ganglionar, sin apagarla se enciende otra luz sobre otro receptor.

ganglionares se recomienda acudir a la bibliografía referida [1].

En organismos más complejos, tales como mamíferos, las operaciones retinales difieren un poco de aquellas de un cangrejo, pero las características básicas del análisis son similares, es decir, las células vecinas pueden incrementar o decrementar la respuesta de una célula ganglionar.

Con un experimento usando la retina de un gato, Lindsay y Norman describen una vecindad de receptores conectadas a una célula ganglionar como la interrelación de dos campos receptores referidos como unidades *on-center* y *off-center*, las cuales corresponden a formas circulares ubicadas en el centro y en el contorno de la vecindad, tal como se muestra en la Figura 1.8.

La luz en la región central causa un incremento en la respuesta del ganglio, región excitatoria; la luz en el contorno causa un decremento en el rango de respuesta con un breve arranque de actividad cuando la luz se apaga, región inhibitoria. Una luz difusa no provocaría respuesta alguna. En general, el sistema nervioso es completamente simétrico.

1.3 Efectos visuales

En la naturaleza, como en el arte y la ciencia existen muchas escenas visuales (o imágenes) que forman efectos que realmente no existen. Las cosas no siempre aparecen como realmente son. Lo que vemos u oímos no es siempre lo que es. Continuamente, los sistemas de percepción cometen errores: algunas veces vemos ilusiones, algunas veces el sistema de percepción requiere de tiempo para recuperarse de una estimulación prolongada, y algunas veces requiere tiempo para interpretar la imagen que se le presenta. Los mecanismos de un sistema son frecuentemente revelados principalmente a través de sus errores y distorsiones. Existen muchos tipos de ilusiones, algunas de

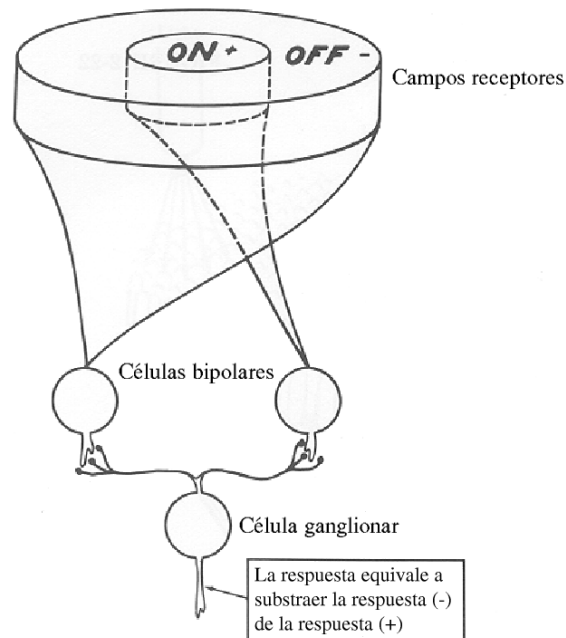


Figura 1.8: Organización de una vecindad de receptores, conectada a una célula ganglionar, en unidades referidas como on-center y off-center.

ellas se forman en la retina y otras más, se forman en el cerebro. Un ejemplo de los efectos que se producen en el cerebro, es la capacidad de éste para organizar imágenes degradadas. Vea la Figura 1.9. Para ver que es una figura de un perro dalmata, la información no presentada en la figura debe ser agregada. El conocimiento de que la figura es de un perro parece acelerar el proceso interpretativo completo: cuando se sabe qué ver, es más fácil verlo.

En la Figura 1.10 se presenta una ilusión que es generada en la retina. Los cuadros internos de la figura aparecen con tonos de gris diferentes: De hecho, todos tienen el mismo tono. Así como estas ilusiones existen muchas otras.

En este trabajo, específicamente, se estudiará una ilusión en particular formada en la retina, aquella que se produce por la interacción entre las células fotorreceptoras y que favorece el apareamiento de sombras. En la Figura 1.11 se presenta la ilusión que se quiere reproducir con este sistema: la aparición de sombras en el área delimitada por cuatro cuadros negros.

Este efecto fue estudiado por Hermann [2], quien dio una explicación convencional basada en **campos concéntricos oponentes**, es decir, elementos circulares con el centro más pequeño que el contorno y ubicados en posición contraria unos con otros. Sin embargo: (1) la intensidad de la ilusión depende de la orientación del patrón, (2) cuando se presenta a 45° , se perciben franjas horizontales y verticales, siguiendo las esquinas de los cuadros, no explicadas por los campos perceptuales concéntricos oponentes. Además, en una variante del patrón original (cuadros vacíos):



Figura 1.9: Ejemplo de las ilusiones producidas en el cerebro. Para observar la imagen del perro dálmata es necesario agregar la información no presentada.

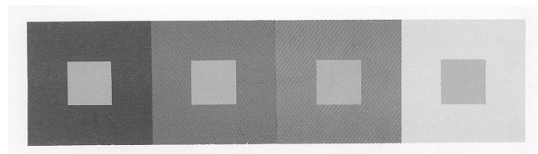


Figura 1.10: Ejemplo de las ilusiones producidas en la retina. Los cuadros internos de la figura aparecen con tonos de gris diferentes, cuando tienen el mismo.

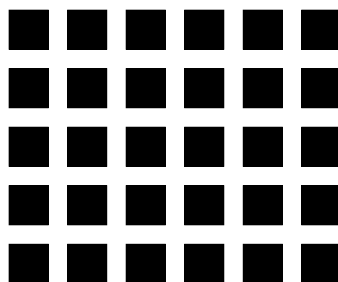


Figura 1.11: Ilusión generada en la etapa de visión temprana. Se perciben manchas en el área localizada entre cuatro cuadros negros, las cuales no existen en la imagen original.

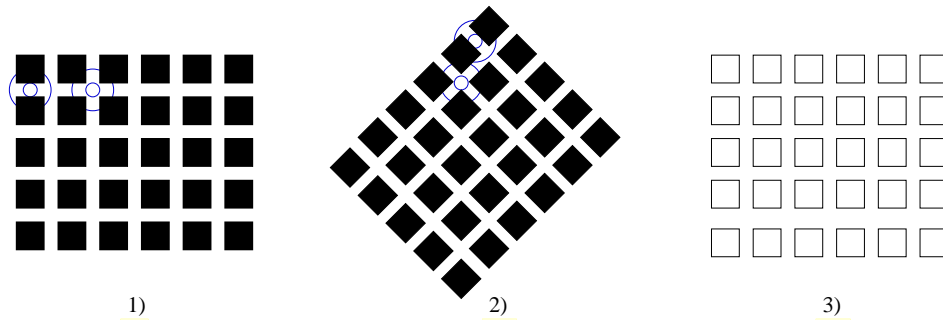


Figura 1.12: Ejemplos de la ilusión de Hermann.

(3) las sombras percibidas son claramente distintas a las del patrón con cuadros llenos, y (4) estas sombras son muy sensibles a la orientación (desaparecen al rotar el patrón unos quince grados). Ver Figura 1.12.

Hermann no ha sido el único que ha estudiado este tipo de fenómeno. En la literatura técnica existen diferentes trabajos con los que se han pretendido explicar características o fenómenos diversos del proceso de visión humano, por ejemplo, T. Yagi, Y. Hayashida y S. Kameda, presentaron un circuito VLSI analógico para emular la visión biológica (descrito en [3]), para lo cual se propuso un modelo de red analógico lineal para describir el circuito neuronal de la retina externa. Otro trabajo es el de M. J. Wilcox y D. C. Thelen, quienes realizaron un estudio interesante sobre la estructura y el comportamiento de la retina de mosquitos (dípteros), proponiendo un modelo para la obtención de altas resoluciones basándose en las características de las retinas de estos animales [4]. T. Ogawa, T. Monohara, H. Kanada y Y. Kosugi desarrollaron un modelo de redes neuronales para generar ilusiones con base en la expansión de ángulos agudos de las células de orientación de la corteza cerebral [5].

En la sección siguiente se presenta un modelo simple para la simulación de la retina humana, como antecedente de este trabajo.

1.4 Un modelo simple

Modelar el comportamiento de la retina humana requiere simular un gran número de receptores visuales, cuya respuesta a la luz está en función tanto de la luz de entrada como de la respuesta de sus receptores vecinos como se presenta en la Figura 1.13.

En el modelo más simple, los receptores se ordenan en arreglos bidimensionales y cada arreglo se organiza en una capa. Una simulación efectiva de la retina requiere simular $10^4 \times 10^4$ receptores y 10 capas de estos arreglos bidimensionales. Esta cantidad de receptores es considerada a partir del conocimiento de que la retina cuenta con 125 millones de receptores, aproximadamente [6]. En

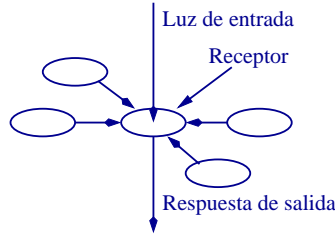


Figura 1.13: Respuesta a la luz de un receptor influenciado por sus vecinos.

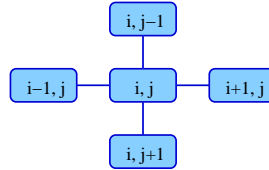


Figura 1.14: Caso más simple de vecindad de perceptrones.

este modelo la respuesta de salida de un receptor es obtenida mediante la suma ponderada de las respuestas de salida de sus vecinos y de la luz de entrada. Por esta razón, es necesario realizar iterativamente los cálculos correspondientes a cada perceptrón hasta que se llegue a una condición de estabilidad.

Los receptores se organizan por niveles, y el valor obtenido en el receptor (i, j) se puede representar por la ecuación (1.1).

$$P_{i,j}^k = F^k(L, V) \quad (1.1)$$

donde L - es la intensidad de la luz de entrada.

V - es la vecindad considerada.

k - es el nivel en el que se está procesando la imagen.

La forma de F se representa por un modelo matemático cuyos parámetros se determinan de acuerdo al observador y al efecto visual que se quiere explicar. Estos modelos matemáticos se aplican sobre arreglos de vecindades de perceptrones. Estas vecindades de perceptrones pueden ser variadas, el caso más simple es la vecindad-4, la cual implica 4 vecinos para un perceptrón central, tal como se muestra en la Figura 1.14. Otro tipo de vecindades es la vecindad-8 y la vecindad hexagonal.

En un caso simple, el valor de un perceptrón se calcula como la suma ponderada de los valores de sus vecinos, como en la Ecuación (1.2).

$$P_{i,j}^k = W_{i,j}P_{i,j}^k + W_{i-1,j}P_{i-1,j}^k + W_{i,j-1}P_{i,j-1}^k + W_{i+1,j}P_{i+1,j}^k + W_{i,j+1}P_{i,j+1}^k \quad (1.2)$$

En este caso, la red neuronal se itera hasta que se alcanza una condición de estabilidad para lo

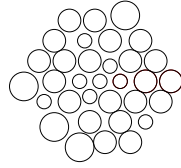


Figura 1.15: Arreglo más real para los receptores.

cual se puede usar un umbral que determina cuando el proceso iterativo debe detenerse.

En situaciones reales, los arreglos de perceptrones no son rectangulares ni completamente regulares, como puede observarse en la microfotografía de la Figura 1.2. Una estructura más real puede observarse en la Figura 1.15, en donde algunos parámetros tales como diámetro de las células, posición del centro y radio, varían de acuerdo a su ubicación en el ojo.

Con este tipo de arreglos, que están más apegados a la realidad, se presenta un problema a resolver: la proyección de una imagen de entrada bidimensional sobre arreglos hexagonales irregulares. En el Capítulo 3 se plantea más a profundidad la complejidad del problema y su solución.

Por otro lado, la vecindad que se utiliza en este modelo simple es muy pequeña, en el Capítulo 4 se proponen vecindades de mayor tamaño y más aún, con preferencias sobre la orientación.

En este modelo simple, la interacción entre los receptores de la retina se representa a través de la Ecuación 1.2, la cual corresponde a un proceso sencillo de asignación arbitraria de pesos y una ponderación. En el Capítulo 4 se muestra el uso de funciones de transferencia y modelos de interacción que simulan los fenómenos excitatorios e inhibitorios de los receptores de la retina humana.

Resueltos los tres puntos anteriores se tiene un arreglo que sigue la estructura de los receptores de la retina humana, pero esto no lo es todo. Con el fin de tener un arreglo artificial cercano a su contraparte biológica (la retina humana), es conveniente simular una gran cantidad de células receptoras, 10^8 aproximadamente. Una simulación como esta consume mucho tiempo, por lo cual, en el Capítulo 5 se propone una estrategia de particionamiento para la ejecución en paralelo de este simulador, buscando obtener el menor consumo de tiempo con el uso de varios procesadores.

Capítulo 2

Redes Neuronales para la Simulación Paralela de la Retina Humana

Introducción

Históricamente, el interés en redes neuronales tiene dos raíces: (1) el deseo de entender los principios sobre los cuales el cerebro humano trabaja, y (2) el deseo de construir máquinas que sean capaces de desarrollar tareas complejas para las cuales la operación secuencial de las computadoras programables concebidas por Babbage y von Neumann no están capacitadas. Desde el reciente surgimiento del interés sobre redes neuronales y “computación neuronal” estos se han basado en el segundo aspecto. Hoy día, muchas tareas se realizan con redes neuronales artificiales, en este capítulo se introducen los conceptos básicos sobre redes neuronales y su surgimiento, para entender su estructura, tendencia y modelos.

2.1 Redes neuronales

A pesar de que el sistema nervioso central humano ha sido estudiado por doctores médicos desde la edad media, su estructura detallada comenzó a ser desenmarañada sólo hasta hace un siglo. En la segunda mitad del siglo diecinueve existían claramente dos escuelas contenidas por su prevalencia científica: *los reticularistas* declararon que el sistema nervioso estaba formado por una red continua e ininterrumpida de fibras nerviosas, mientras que los *neuronalistas* declararon que esta red neuronal estaba compuesta de un vasto número de unidades celulares simples interconectadas, *las neuronas*. Siguiendo el curso de la ciencia, la lucha entre estas dos doctrinas fue decidida por el advenimiento de una nueva técnica, inventada por Camilo Golgi alrededor de 1880, el tñido de fibras nerviosas

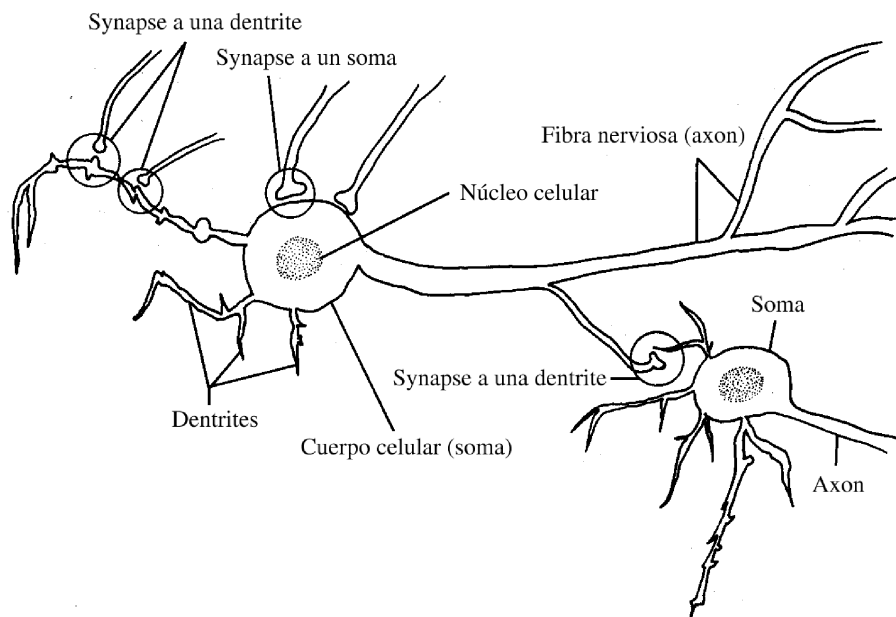


Figura 2.1: Dibujo de dos tipos de neuronas mostrando los diferentes tipos de empalmes sinápticos.

por medio de una reacción del bicromato de plata. Esta técnica fue ingeniosamente aplicada por el doctor español Santiago Ramón y Cajal en 1888 para refutar la doctrina del reticularismo exhibiendo las pequeñas brechas entre neuronas individuales [7].

2.1.1 La estructura del sistema nervioso central

La neurona

El bloque de construcción básico del sistema nervioso es la neurona, la célula que maneja la intercomunicación de información a lo largo de las distintas partes del cuerpo. Una unidad neuronal consiste de varias partes (ver Figura 2.1): un *cuerpo de célula* (también llamado *soma*), una *fibra nerviosa* (también llamada *axon*) que lleva la información de una neurona a otra, y una terminal de *empalme* (una *synapse*) a la cual la actividad de una neurona influencia las características eléctricas del cuerpo de la célula de otra. Los empalmes entre neuronas ocurren en cualquier parte de la célula o en pequeñas extensiones del cuerpo de la célula llamadas dendrites.

Las fibras nerviosas y las dendrites pueden ser tratadas como conductores aislados para transmitir las señales básicas del sistema nervioso - impulsos eléctricos. Estas entradas eléctricas son generadas en el cuerpo de la célula en respuesta a la actividad de la synapse de entrada. En la synapse misma, un impulso de entrada dispara la liberación de una *substancia transmisora química*, la cual viaja a través de las pequeñas brechas entre el axon de la neurona entrante y la locación

receptora de la nueva neurona.

Los empalmes synápticos son de dos formas: *excitatorios*, en los cuales los efectos de los impulsos que llegan incrementan la posibilidad de que la célula responda, o *inhibitorio*, en el cual el impulso que llega reduce la posibilidad de que la célula receptora responda.

El tamaño del cuerpo celular (soma) de una neurona típica es de 10-80 micrones, mientras que las dendritas y los axons tienen un diámetro de unos cuantos micrones. Mientras que las dendritas sirven como receptores de señales de neuronas adyacentes, el propósito de los axons es la transmisión de la actividad neuronal generada a otra célula nerviosa o a otras fibras musculares. En el primer caso el término *interneurona* se usa continuamente, mientras que la neurona es llamada una *neurona motriz* en el segundo caso. Un tercer tipo de neurona, el cual recibe información de los músculos u órganos sensoriales, tales como el ojo o el oído, es llamado una *neurona receptora*.

La corteza cerebral

La complejidad increíble del sistema nervioso central, especialmente del cerebro humano, no resta mucho en complejidad y diversidad de la célula nerviosa simple, la cual está completamente limitada, como en el vasto número de sus unidades constituyentes, es decir, de las neuronas y sus conexiones mutuas. Sin embargo, podría ser engañoso asumir que todas las neuronas en el cerebro lucen igual. Existen varios tipos muy diferentes, distinguidos por su tamaño y grado de ramas de su árbol dendrítico, la longitud de su axon, y otros detalles estructurales. Estos tipos ocurren en diferentes partes o capas de la corteza y su cooperación aparece para ser esencial para la solución de tareas cognitivas complejas.

La conectividad en el sistema nervioso central está caracterizada por las propiedades complementarias de *convergencia* y *divergencia*. En la corteza humana cada neurona se estima que recibe entradas convergentes en la media de 10,000 synapses. Por otro lado, cada celda alimenta su salida a muchos cientos de otras neuronas, continuamente a través de un número largo de synapses tocando una célula nerviosa simple.

El número total de neuronas en la corteza humana es inmensa y sólo se puede estimar. Cuentas completas en varias regiones pequeñas han producido una cantidad virtualmente constante de 1,500,000 neuronas por mm^2 , las cuales son distribuidas sobre la profundidad completa del tejido cortical. Considerando que el área total de la corteza doblada pesadamente mide alrededor de 200,000 mm^2 , uno obtiene un número de al menos 3×10^{10} neuronas [7].

2.1.2 Introducción a las redes neuronales artificiales

Las redes neuronales artificiales (RNA) se han utilizado para tareas cognitivas, tales como el aprendizaje y la optimización, las cuales están en un amplio sentido basadas en conceptos derivados de la investigación hacia la naturaleza del cerebro. En términos matemáticos *un modelo de red neuronal* está definido como *una gráfica dirigida* con las siguientes propiedades:

1. Una variable de estado n_i está asociada con cada nodo i .
2. Un valor real como peso $w_{ik} \in \mathbb{R}$, está asociado con cada liga (ik) entre dos nodos i y k .
3. Un valor real como *polarización* (inclinación) ϑ_i está asociado con cada nodo i .
4. Una función de transferencia $f_i[n_k, w_{ik}, \vartheta_i, (k \neq i)]$ está definida para cada nodo i , la cual determina el estado del nodo como una función de su polarización, del peso de sus ligas de entrada, y de los estados de los nodos conectados a él por estas ligas.

En la terminología estándar, los nodos son llamados *neuronas*, las ligas son llamadas *conexiones sinápticas*, y la polarización es conocida como el *umbral de activación*. La función de transferencia es usualmente tomada de la forma

$$f\left(\sum_k w_{ik}n_k - \vartheta_i\right), \quad (2.1)$$

donde $f(x)$ es cualquier función de pasos discontinuos o su equivalente conocida como función sigmoideal. Los nodos sin ligas hacia ellos son llamados neuronas *entrantes*; la neuronas *salientes* son aquellas que no tienen ligas que se tiendan desde ellos [7].

Una historia breve de los modelos de redes neuronales

En 1943 Warren McCulloch y Walter Pitts propusieron una teoría general de procesamiento de información basada en redes de alternancia binaria o elementos de decisión, los cuales son un poco eufemísticamente llamados “neuronas”, aunque sean mucho más simples que su contraparte biológica real. Cada uno de estos elementos $i = 1, \dots, n$ sólo puede tomar los valores de salida $n_i = 0, 1$, donde $n_i = 0$ representa el estado restante y $n_i = 1$ el estado activo de la unidad elemental. Para la simulación del período de regeneración finita de neuronas reales, los cambios en el estado de la red neuronal son supuestos para que ocurran en pasos de tiempo discreto $t = 0, 1, 2, \dots$. El nuevo estado de una cierta unidad neuronal está determinado por la influencia de todas las otras neuronas, como se expresa por una combinación *lineal* de sus valores de salida:

$$h_i(t) = \sum_j w_{ij}n_j(t). \quad (2.2)$$

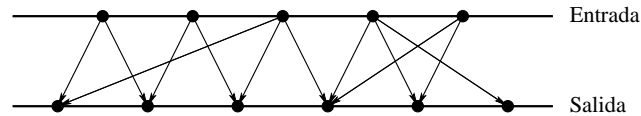


Figura 2.2: Perceptrón simple, consiste de dos capas de neuronas. Las neuronas de entrada alimentan a las neuronas de las capas de salida, pero no viceversa.

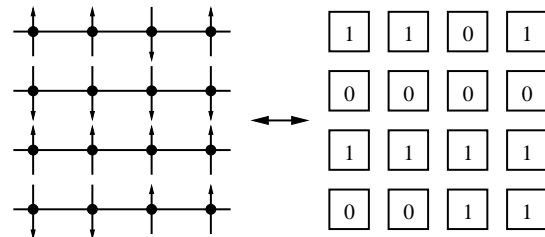


Figura 2.3: Una correspondencia uno a uno existe entre los enrejados de spins magnéticos con dos orientaciones diferentes y redes neuronales de McCulloch-Pitts. Un spin apuntando hacia arriba es identificado como una neurona activa.

Aquí, la matriz w_{ij} representa la fuerza de emparejamiento sináptico entre las neuronas j e i , mientras que $h_i(t)$ modela el total del potencial de polarización postsináptica en la neurona i causado por la acción de todas las otras neuronas.

Alrededor de 1960 Frank Rosenblatt [7] y sus colaboradores estudiaron extensamente un tipo específico de redes neuronales, a la cual llamaron *perceptrón*, porque lo consideraron un modelo simplificado del mecanismo biológico del procesamiento de información sensorial, es decir, la percepción. En su forma más simple, un perceptrón consiste de dos capas separadas de neuronas representando la capa de entrada y salida, respectivamente, como se ilustra en la Figura 2.2. Las neuronas de la capa de salida reciben señales sinápticas desde aquellas de la capa de entrada, pero no viceversa, y además, las neuronas dentro de una capa no se comunican entre ellas.

Otro desarrollo fructificante comenzó cuando William Little [7] denotó la similitud entre una red neuronal de los tipos propuestos por McCulloch y Pitts y los sistemas de momentos magnéticos elementales o *spins*, ver Figura 2.3. En estos sistemas, llamados *modelos Ising*, el spin s_i en el enrejado i puede tomar sólo dos orientaciones diferentes, arriba o abajo, denotado por $s_i = +1$ (arriba) y $s_i = -1$ (abajo). La analogía con la red neuronal se realiza identificando cada spin con una neurona y asociando la dirección hacia arriba $s_i = +1$ con el estado activo $n_i = 1$ y la orientación hacia abajo con el estado restante $n_i = 0$.

Estas ideas fueron fuertemente desarrolladas por Little y Gordon Shaw y por John Hopfield [7] quienes estudiaron cómo una red neuronal o un spin puede almacenar y recuperar información. Los modelos de Little y Hopfield difieren en la manera en que los estados del sistema son actualizados. En el modelo de Little todas las neuronas (spins) son actualizadas síncronamente de acuerdo a la

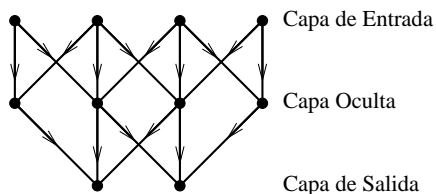


Figura 2.4: Una red profunda con una capa interna (oculta) de neuronas.

Ecuación (2.1), mientras que en el modelo Hopfield las neuronas son actualizadas secuencialmente una a la vez.

En los años recientes se ha revivido el interés sobre redes profundas en capas (perceptrones). Este desarrollo fue iniciado por el (re)descubridor de un algoritmo eficiente para la determinación de un emparejamiento sináptico estrecho en redes multicapas con *capas ocultas*, ver Figura 2.4. Este método fue inicialmente sugerido por Werbos y ahora es conocido como *error de propagación hacia atrás*. Este algoritmo de aprendizaje está basado en un principio simple pero efectivo: las fuerzas sinápticas w_{ik} son modificadas iterativamente hasta que la señal de salida difiera tan poco como sea posible de la deseada. Otros conceptos de “aprendizaje”, es decir, ajustes a la fuerza sináptica, son estudiados intensamente, tales como estrategias basadas en recompensa y penalización, evolución y selección, y otras.

2.2 Modelos neuronales para la simulación de la retina

El estudio del cuerpo humano, su estructura y funcionamiento siempre ha sido un área de gran interés para la sociedad. A pesar de los avances de la ciencia y la tecnología existen procesos biológicos humanos que aún son un misterio, como es el caso del proceso de visión humano. Cada vez, estudios fisiológicos arrojan nuevas tendencias, líneas de investigación, con las cuales se busca explicar el funcionamiento del ojo. Muchas de las soluciones propuestas son modelos que pretenden explicar algún fenómeno en particular, como es el caso de las ilusiones ópticas. Estos modelos están basados en los conceptos y conocimientos sobre redes neuronales biológicas, proponiendo uno nuevo para desarrollarse utilizando redes neuronales artificiales.

En 1998, T. Yagi, Y. Hayashida y S. Kameda desarrollaron un VLSI análogo para emular la visión biológica, presentado en [3], para el cual se propuso un modelo de red analógico lineal para describir el circuito neuronal de la retina externa que consiste de fotorreceptores, células horizontales y células bipolares.

A decir de los autores, algunos problemas de la visión temprana, por ejemplo, detección de bordes o *stero visión*, son formulados en términos de la teoría de regularización estándar, en la cual,

estos problemas son atribuidos a la minimización de una función de costos,

$$J(y) = \|y - d\|^2 + \lambda_r \int \sum_{r=1}^p \left(\frac{d^r y(\xi)}{d\xi^r} \right)^2 d\xi, \quad \lambda_r > 0. \quad (2.3)$$

Aquí, d y y corresponden a una imagen sin refinar (bruta) y una imagen filtrada para interpolarlas, respectivamente. El segundo término de la función de costos asegura unicidad y convergencia de la solución.

Como los circuitos retinales consisten de neuronas, las cuales son elementos discretos, los autores convienen en reemplazar las operaciones de derivada en 2.3 por operaciones de diferencia, es decir,

$$\frac{dy(\xi)}{d\xi} \rightarrow y_k - y_{k-1},$$

$$\frac{d^2y(\xi)}{d\xi^2} \rightarrow y_{k-1} - 2y_k + y_{k+1}.$$

Aquí $k = 1 \dots n$. Estas operaciones pueden ser expresadas por

$$\frac{dy(\xi)}{d\xi} \rightarrow \mathbf{D}\mathbf{y}, \quad \frac{d^2y(\xi)}{d\xi^2} \rightarrow \mathbf{L}\mathbf{y}$$

Donde

$$\mathbf{D} = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ -1 & 1 & 0 & 0 & \dots \\ 0 & -1 & 1 & 0 & \dots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & -1 & 1 \end{pmatrix},$$

$$\mathbf{L} = \begin{pmatrix} -2 & 1 & 0 & \dots & 0 \\ 1 & -2 & 1 & 0 & \dots \\ 0 & 1 & -2 & 1 & \dots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & 1 & -2 \end{pmatrix},$$

y

$$\mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}.$$

Note que

$$\mathbf{D}^T \mathbf{D} = -\mathbf{L} \quad (2.4)$$

Aquí T denota la transpuesta. Consecuentemente, el problema de regularización es formalizado en el caso discreto por la minimización de

$$J(y) = \|y - d\|^2 + \sum_{r=1}^p \lambda_r \begin{cases} \|\mathbf{L}^{\frac{r}{2}} \mathbf{y}\|^2 & r : \text{even} \\ \|\mathbf{DL}^{\frac{r-1}{2}} \mathbf{y}\|^2 & r : \text{odd}, \end{cases}$$

Donde es referido como el problema de regularización de orden p en el caso discreto. La solución única para minimizar $J(y)$ es obtenida por la ecuación de Euler,

$$\mathbf{y} - \mathbf{d} + \sum_{r=1}^p (-1)^r \lambda_r \mathbf{L}^r \mathbf{y} = 0. \quad (2.5)$$

Aquí se usó 2.4. Para el problema de regularización de segundo orden, la función de costos se expresa como

$$J(y) = (\mathbf{y} - \mathbf{d})^T (\mathbf{y} - \mathbf{d}) - \lambda_1 (\mathbf{D}\mathbf{y})^T (\mathbf{D}\mathbf{y}) + \lambda_2 (\mathbf{L}\mathbf{y})^T (\mathbf{L}\mathbf{y}).$$

Y la ecuación de Euler correspondiente es

$$\mathbf{y} - \mathbf{d} - \lambda_1 \mathbf{L}\mathbf{y} + \lambda_2 \mathbf{L}^2 \mathbf{y} = 0. \quad (2.6)$$

Después de plantear esta teoría computacional de la visión temprana, los autores describen la estructura de la red que utilizaron como base para su circuito retinal. En esta estructura de red, se mencionan las siguientes características:

- La retina consiste de varias capas de redes neuronales.
- Las neuronas del mismo tipo y los fotorreceptores son arreglados en arreglos bidimensionales para incluirlos en capas separadas.
- La información visual es procesada en etapas sucesivas.
- La interacción entre estas capas incluye conexiones de realimentación (feedback) y de avance (feedforward).
- Este alambrado se puede alcanzar debido a la estructura de capas.
- La estructura de capas se conserva en todas la retinas de vertebrados, desde peces hasta mamíferos.

Finalmente, los autores presentan la teoría de regularización empleada y el circuito retinal. Este último puede observarse en la Figura 2.5.

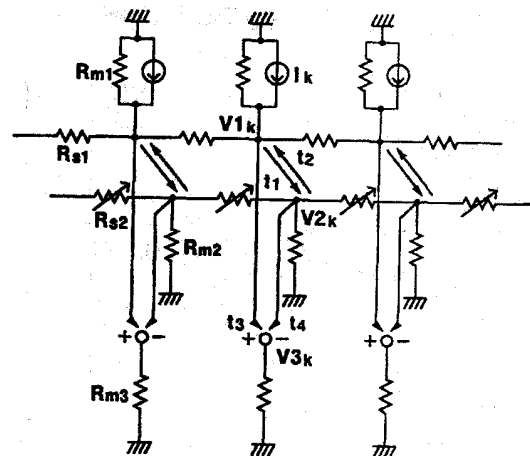


Figura 2.5: Modelo del circuito neuronal de la retina externa.

Las similitudes de este trabajo con el nuestro son interesantes, T. Yagi, Y. Hayashida y S. Kameda [3] presentan un estudio amplio sobre la visión temprana orientado a su solución por medio de una minimización de una función de costos. Por nuestro lado, el proceso de visión temprana se está determinando como el resultado de aplicar una suma ponderada a un conjunto de perceptrones organizados por una figura geométrica.

Además de la visión temprana, utilizan una estructura de la retina similar a la nuestra: a través de capas, la diferencia estriba en el número de capas utilizadas.

La revisión de este modelo fue interesante pues presenta una forma clara de desarrollar un modelo de red neuronal artificial a partir de uno biológico.

Otro trabajo notorio sobre modelos de redes neuronales artificiales aplicadas a la simulación de la retina fue desarrollado en 1999 por M. J. Wilcox and D. C. Thelen, Jr., presentado en [4]. En este trabajo se presenta un estudio interesante sobre la estructura y comportamiento de la retina de dípteros (insectos con dos alas, mosquitos). A partir del estudio realizado sobre las capacidades de los mosquitos se deja notar la inferioridad de muchos modelos aplicados en máquinas convencionales.

La capacidad de los mosquitos para extraer información de alta resolución es atribuida a un fenómeno llamado hiperagudeza. Este concepto es aplicado para desarrollar un modelo neuronal para implementar un sistema donde la resolución más alta no implique la carga más alta de una máquina convencional, como sucede en la visión de máquinas convencionales.

La forma que proponen los autores para obtener una súper-resolución es proveer la razón-ruido-a-señal con la cual una imagen es calificada es suficientemente alta. En este punto, los autores se apoyan para indicar que mucho del procesamiento temprano de información en una retina es dedicado a elevar el contraste de la imagen.

Este trabajo, aunque no está orientado a la retina humana, es interesante por la profundización

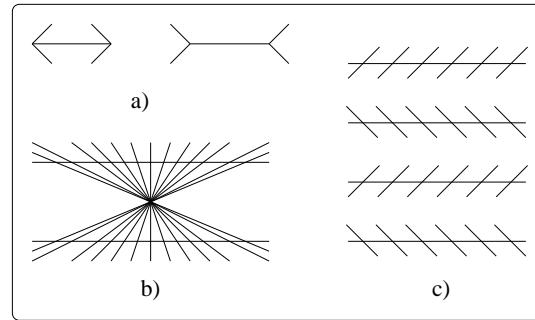


Figura 2.6: Ilusiones geométricas famosas. (a) Ilusión de Muller-Lyer, (b) ilusión de Hering, (c) ilusión de Zeollnor..

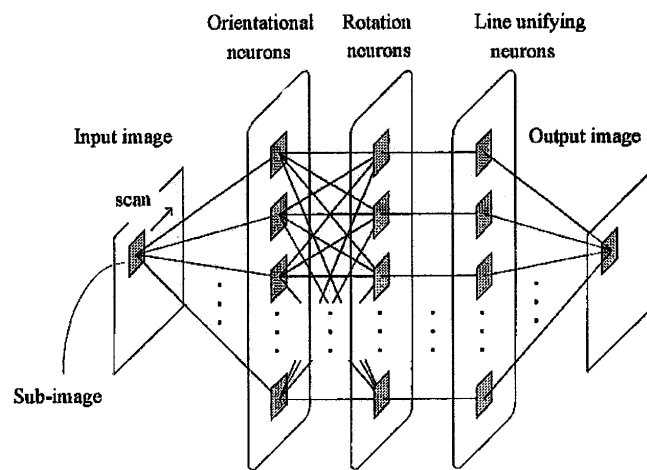


Figura 2.7: Modelo propuesto.

acerca de la estructura y la conceptualización de una retina animal (la de los mosquitos).

En el mismo año, 1999, T. Ogawa, T. Monohara, H. Kanada y Y. Kosugi, propusieron un modelo de red neuronal, presentado en [5], el cual pretende explicar el fenómeno de reconocimiento de ángulos entre líneas cruzadas por la inhibición lateral entre las neuronas de orientación en la corteza cerebral.

En este estudio, los autores trabajaron con ilusiones geométricas que están compuestas por la naturaleza geométrica de la longitud, dirección y ángulo de los segmentos de línea. Estas ilusiones se ilustran en la Figura 2.6.

El modelo propuesto por los autores consiste de tres grupos de neuronas; las de orientación, las de expansión de ángulos agudos y las células de unificación de líneas, ver Figura 2.7. Este modelo maneja la imagen de entrada después de dividirla en algunas sub-imágenes y las integra después de todo el procesamiento. El tamaño de la sub-imagen se decide por el tamaño de la imagen de entrada y la complejidad o finura de la figura.

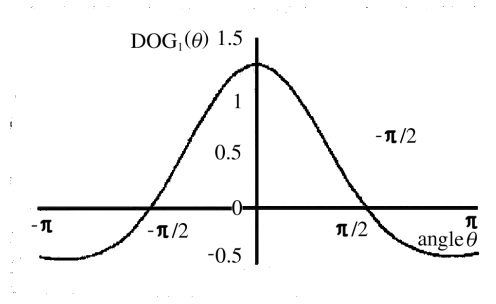


Figura 2.8: Función DOG definida por el ángulo θ .

Para los autores, las células de orientación extraen los segmentos de línea que pertenecen a la dirección preferida de cada célula de la sub-imagen usando el filtro DOG bidimensional, definido como

$$G_{\sigma}(x) = \exp\left(\frac{-x^2}{\sigma^2}\right) \quad (2.7)$$

$$DOG_2(x, y) = AG_{\sigma_{env}}(y)[G_{\sigma_{ex}}(x) - k\sigma_{ex}/\sigma_{in}G_{\sigma_{in}}(x)] \quad (2.8)$$

En las células de expansión de ángulos agudos, se aplicó la función DOG unidimensional para el ángulo entre los segmentos de línea.

$$DOG_1(x) = A[G_{\sigma_{ex}}(x) - k\sigma_{ex}/\sigma_{in}G_{\sigma_{in}}(x)] \quad (2.9)$$

donde $G_{\sigma}(x)$ es definida en (2).

Las células de expansión de ángulos agudos calculan la cantidad de la rotación con el ángulo entre los segmentos de línea usando el valor de la función DOG y rota la parte de los segmentos de línea dentro de la sub-imagen. Las células de unificación de línea reunifica cada segmento de línea roto, asumiendo una condición *a priori* dada.

Las partes de los segmentos de línea rotos se trasladan por unificación, esto es, se buscan los bordes rotos, como se muestra en la Figura 2.9. Esta operación se ejecuta para cada segmento de línea de cada dirección.

En la Figura 2.9 se muestra el proceso de expansión de ángulos fase a fase del modelo propuesto.

Los autores examinaron la validez de su modelo a través de una simulación computacional, para la cual utilizaron imágenes de entrada de las figuras en escala de grises de 8 bits con resolución de 64×64 pixels o 128×128 pixels. Las imágenes de entrada son binarias. La imagen de entrada de 64×64 pixels es usada para obtener las 16 sub-imágenes de 16×16 pixels (ver Figura 2.10. Análogamente para 128×128 .

En la Figura 2.11 se muestra una imagen de entrada y la imagen de salida producida por la simulación del modelo propuesto.

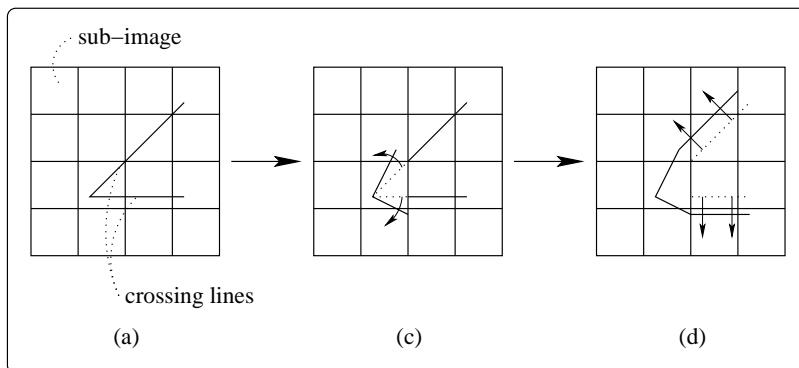


Figura 2.9: Procedimiento de expansión de ángulos agudos del modelo propuesto. (a) Figura de entrada dividida en sub-imágenes, (b) líneas rotadas en la sub-imagen con la intersección, (c) líneas trasladadas por la unificación.

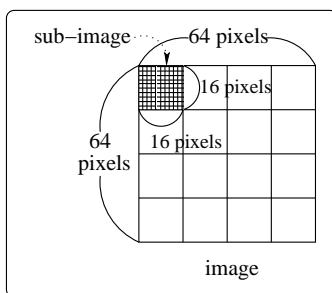


Figura 2.10: Imagen de entrada y sub-imágenes divididas en la simulación.

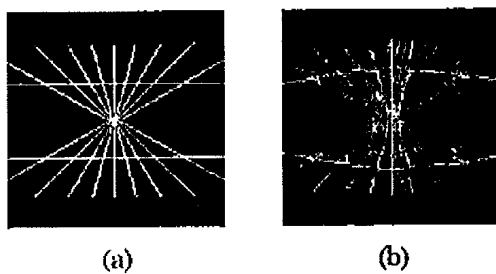


Figura 2.11: Resultados de la simulación de la ilusión de Hering. (a) Imagen de entrada, (b) imagen de salida..

Este trabajo tiene una similitud importante con el nuestro, la idea principal. El modelo propuesto por T. Ogawa, T. Monohara, H. Kanada y Y. Kosugi, toma una imagen de entrada, y al procesarla produce una ilusión óptica, la cual es, esencialmente, la idea de esta tesis. La diferencia principal es la ilusión a explicar.

Continuando con las ideas principales, otra en común es la consideración de la inhibición lateral como un factor importante en la generación de ilusiones, y su simulación a través de Gaussianas. Aquí, la diferencia es que los autores proponen que su ilusión es generada en la corteza cerebral y en esta tesis se propone que es en la retina.

Estos trabajos tienen una aportación importante al estado del arte de las redes neuronales aplicadas al proceso de visión. Con la realización de este trabajo se pretende también aportar un nuevo modelo neuronal para explicar un fenómeno específico del proceso de visión humano.

2.3 Redes neuronales paralelas

Se han propuesto varios tipos de redes neuronales y sus investigaciones y análisis fundamentales se han hecho por años. Al mismo tiempo, la tecnología de las redes neuronales también se ha promovido vigorosamente en campos de aplicación amplios tales como reconocimiento de patrones, procesamiento de imágenes, control. En el estudio de las redes neuronales, las técnicas de simulación usando computadoras digitales es una de las estrategias de investigación más importantes porque el análisis matemático parece ser muy difícil. Sin embargo, debido a que las redes neuronales requieren una enorme cantidad de tiempo para el aprendizaje, se necesitan simuladores más rápidos, especialmente para redes neuronales de gran escala. Para alcanzar rangos de procesamiento de alta velocidad se han desarrollado computadoras paralelas de propósito general/especial, por ejemplo la máquina de conexión, la máquina n-cubo, y otras. Pero, no se puede aprovechar al máximo estos equipos si no se cuenta con algoritmos eficientes que minimicen la cantidad de trabajo adicional en la transmisión y sincronización de los procesadores.

En la literatura técnica existen varios algoritmos paralelos realizados sobre algunos modelos específicos de redes neuronales, en esta sección se presentan algunos de ellos.

1. T. Toshiro, T. Akiyoshi y T. Kunibobu, en [8]. Describen un algoritmo de cómputo paralelo para simular el modelo de retro-propagación (back propagation “BP”) y el mapa auto-organizativo de características de Kohonen (Kohonen’s self-organizing feature map “SOFM”) sobre un arreglo de procesadores 8-vecinos. Tomando en cuenta al paralelismo intrínseco encontrado en las redes neuronales, Toshiro y Kunibobu desarrollaron algoritmos que minimizan la sobrecarga en la transmisión entre procesadores.

2. D. Timothy J y M.-T. Evangelia, en [9]. El proceso ALOPEX es un procedimiento de optimización definido el cual varía, simultáneamente, varios parámetros basados en la retroalimentación de valores simples y aplicación de ruido. D. Timothy J y M.-T. Evangelia usaron ALOPEX para resolver un problema de reconocimiento de imágenes usando algunas plantillas basadas en hexágonos. Se analizan dos formas de la función de “costos” y se adjuntó un sistema jerárquico para obtener más rápido soluciones “buenas”.
3. N. Funabiki y Y. Takefuji, en [10]. Presentan un algoritmo paralelo para los problemas de asignación de canales en redes radio celulares. El problema de asignación de canales envuelve no sólo asignación de canales o frecuencias a cada célula radial, también satisface los contrastes de frecuencia dados por una matriz de compatibilidad. El algoritmo paralelo propuesto está basado en una red neuronal artificial compuesta de $n \times m$ elementos procesables para un problema de n -células y m -frecuencias.

Estos trabajos refieren a modelos de redes neuronales paralelas donde la paralelización está considerada dentro del modelo neuronal. En nuestro caso, la paralelización del modelo está basada en una estructura de particionamiento sencilla, debido a la estructura de almacenamiento propuesta, pero aún así, la mejoría esperada es cuantiosa.

Capítulo 3

Recepción de Escenas Visuales

Introducción

En este capítulo se presenta el generador de arreglos hexagonales de receptores y el proyector de imágenes sobre arreglos hexagonales de receptores. Los arreglos que el generador produce son representativos de la retina humana, los elementos de estos arreglos procuran la forma y estructura de las células de visión arregladas en la retina. El proceso de proyección realiza la diseminación de una escena visual sobre el arreglo de células receptoras de la retina. En este capítulo se ilustra la forma para generar los arreglos y su estructura, además, se muestra la generación y el uso de la imagen de salida denominada *imagen artificial*.

3.1 Generador de arreglos hexagonales de receptores

Para simular el proceso de visión temprana es necesario contar con un arreglo que represente a la retina humana siguiendo sus características principales. Los arreglos de células receptoras no son rectangulares ni completamente regulares, pretenden ser un arreglo concéntrico de receptores de diferentes tamaños organizados siguiendo una vecindad hexagonal. En la Figura 1.2 se muestra una microfotografía de receptores humanos, en ella se pueden notar las características de los arreglos de receptores humanos.

Para observar las diferencias de los arreglos regulares y los más irregulares se crearon 4 formas distintas de arreglos de receptores: a) Forma 1: arreglo regular con círculos simétricos y puntos regulares, b) Forma 2: arreglo regular con círculos simétricos y puntos irregulares, c) Forma 3: arreglo concéntrico con círculos y puntos regulares, y d) Forma 4: arreglo concéntrico con círculos perturbados y puntos regulares. Estas formas pueden observarse en la Figura 3.1.

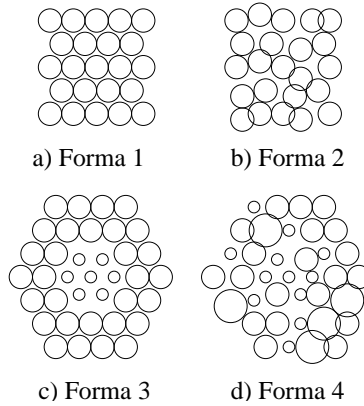


Figura 3.1: Arreglos hexagonales de receptores para simular la retina humana.

Las 4 formas de arreglos hexagonales de receptores cuentan con 2 perspectivas, una lógica y una física. La perspectiva lógica se refiere a la forma en que podemos observar los receptores como círculos formando un arreglo hexagonal similar a la retina humana, tal como se pueden observar en la Figura 3.1. La perspectiva física se refiere a la forma real como los receptores son almacenados en la estructura de datos que los contendrá para su recuperación y manipulación.

La estructura de datos utilizada para representar a las diferentes formas de los arreglos es una lista de listas, en la cual se van almacenando los elementos que definen a cada receptor (los receptores son representados a través de círculos), estos elementos son: coordenadas x, y de la ubicación lógica del receptor en el arreglo, radio del círculo y tono de gris inicial.

3.1.1 Forma 1: arreglos regulares con círculos simétricos y puntos regulares

La forma 1 consta de un arreglo regular (una matriz de $n \times n$) en la cual se indican las coordenadas donde se encuentran los centros de cada círculo (receptor) para después generar los círculos con un radio fijo (*círculos simétricos*). La forma de determinar cada centro dentro del arreglo inicia con la introducción del radio deseado por el usuario. A partir de este radio r se determina la distancia que existirá entre dos centros consecutivos horizontalmente, tomando d_h como la distancia horizontal tenemos que $d_h = 2 \cdot r$ y la distancia entre dos centros consecutivos verticalmente, d_v , equivalente a la distancia vertical, se obtiene con $d_v = 1.5 \cdot r$. En la vecindad hexagonal, cada elemento central tiene 6 vecinos, dos arriba, dos abajo y uno en cada costado, de tal forma que los centros de los vecinos formen un hexágono (ver Figura 3.2); para cumplir con esta regla es necesario alterar la posición horizontal de los centros de las líneas impares con la distancia de un radio, esto es, los renglones **pares** inician en la coordenada $x = 0$ y los **impares** en $x = r$. En la Figura 3.3.a se

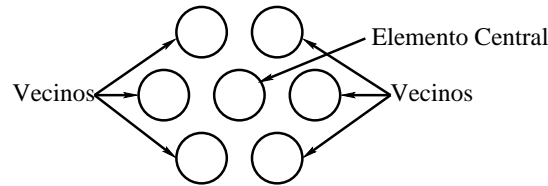


Figura 3.2: Vecindad hexagonal: Un elemento central con 6 vecinos, dos arriba, dos abajo y uno en cada costado.

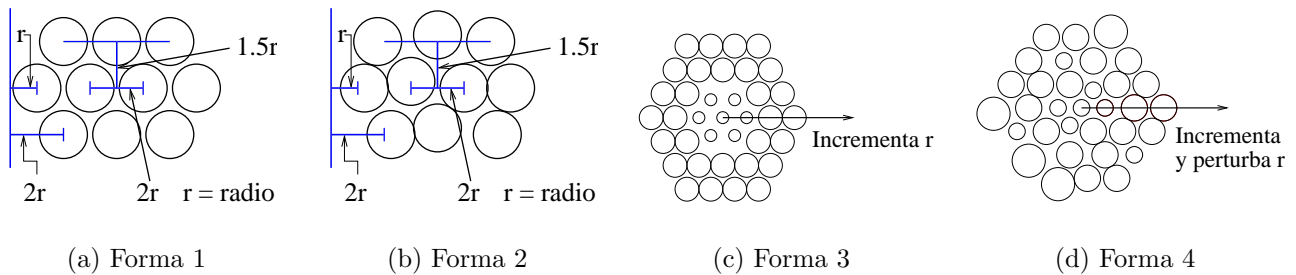


Figura 3.3: Especificación de las formas para los arreglos de receptores.

ilustran las características de la estructura para arreglar los receptores.

Al proceso de determinar las coordenadas para los centros de los receptores y almacenarlos en la matriz se le denomina generación del arreglo hexagonal. Para esta forma, la generación inicia en la esquina superior derecha de la matriz con las coordenadas $(0, d_v)$.

El almacenamiento de la forma 1 es completamente regular; los receptores son generados por un ciclo anidado que permite almacenar de manera secuencial cada elemento. En la Figura 3.4 se puede observar la perspectiva lógica de esta forma y su perspectiva física para su almacenamiento.

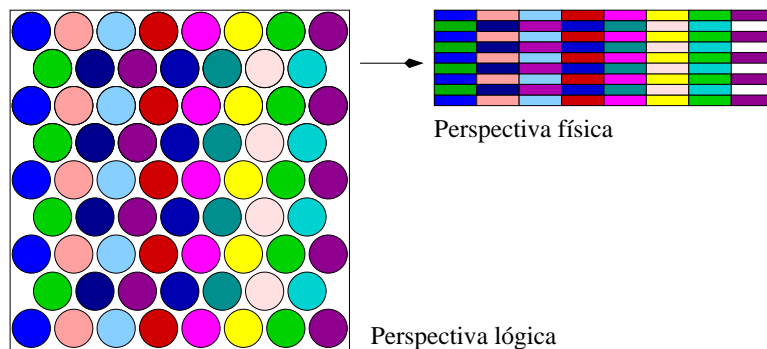


Figura 3.4: Perspectivas lógica y física de la forma 1.

3.1.2 Forma 2: arreglos regulares con círculos simétricos y puntos irregulares

El mecanismo de generación de los centros de cada receptor para la forma 2 es análogo al de la forma 1, la diferencia estriba en la perturbación de las coordenadas correspondientes a cada centro (de ahí el por qué de *puntos irregulares*). Esta perturbación se realiza al agregar un número aleatorio entre $[-1, 1]$ para provocar un ligero ruido en las coordenadas de los centros de los receptores. En la Figura 3.3.b se muestran gráficamente los elementos de esta forma. Para la forma 2, la generación también inicia en la esquina superior derecha de la matriz con las coordenadas $(0, d_v)$, como en la forma 1.

La perspectiva lógica de esta forma es similar a la de la forma 1, mientras que la física es igual, debido a que la generación de los elementos también es completamente regular, la diferencia es que las coordenadas a almacenar en la estructura de datos están perturbadas.

3.1.3 Forma 3: arreglos concéntricos con círculos y puntos regulares

En la forma 3, la generación del arreglo hexagonal inicia con la determinación del centro de un receptor (x, y) , denominado receptor central. Este receptor se encontrará ubicado al centro de la matriz de receptores y servirá de base para determinar las coordenadas del resto de los receptores. Las coordenadas de los receptores restantes se encontrarán a través de las ecuaciones

$$x' = r \cdot \cos(\theta) + x \quad (3.1)$$

$$y' = r \cdot \sin(\theta) + y \quad (3.2)$$

donde θ es el ángulo al que se encuentra alineado el centro del receptor vecino, este ángulo puede ser 0, 60, 120, 180, 250 ó 300 grados. El radio para cada receptor depende de la distancia de su centro al centro del receptor central. Esto es debido al deseo de producir un **arreglo concéntrico**, en el cual, se tiene un elemento central con un radio inicial r_i , y a partir de éste se generan los demás, con radios r_k asignados de forma creciente. Así que se requiere de un radio inicial r_i , y un radio final r_f , donde $r_i \leq r_k \leq r_f$ y de los cuales se obtiene el incremento $incre = r_f - r_i$. $incre$ es utilizado para encontrar los radios de los receptores correspondientes a las diferentes distancias de los receptores del contorno al receptor central, es decir, para los receptores alejados por la distancia $dist_k$ del central, su radio se obtiene por la ecuación

$$r_k = r_i + dist_k \cdot \left(\frac{incre}{\left(\frac{resolucionX}{2} \right)} \right). \quad (3.3)$$

Donde $resolucionX$, es la resolución del arreglo de receptores. Las distancias que establecen los radios de los perceptrones pueden verse como el radio de una circunferencia, donde todos los perceptrones cuyos centros se intersecten con dicha circunferencia tendrán el mismo radio.

Este crecimiento del radio de los receptores puede verse como un crecimiento regular, por ello el término de *círculos regulares* del nombre de esta forma. El término *puntos regulares* se estableció debido a que las coordenadas de los receptores obtenidas por las ecuaciones (3.1) y (3.2) no son alteradas, sino que se mantienen tal cual fueron generadas. En la Figura 3.3.c se muestra la estructura de la forma 3.

El almacenamiento (perspectiva física) de esta forma es bastante irregular. Cada vez que se genera un elemento, éste se almacena en la lista que corresponde a su coordenada y . Los elementos de cada lista, se van almacenando de acuerdo a como son generados, es decir, el primer elemento generado ocupará la posición $x = 0$, aunque ésta no sea su coordenada en la perspectiva lógica de la forma y los demás se almacenan consecutivamente detrás del anterior. Debido a este sistema de almacenamiento, al terminar la generación de receptores, quedan listas sin elementos y listas con elementos desordenados; por ello, después del almacenamiento de los receptores la lista de listas se compacta para eliminar aquellas listas sin elementos y cada lista se ordena de acuerdo a la coordenada x de los receptores para facilitar la manipulación de los datos.

3.1.4 Forma 4: arreglos concéntricos con círculos perturbados y puntos regulares

El mecanismo de generación de los centros de cada receptor para la forma 4 es análogo al mecanismo de la forma 3, la diferencia estriba en la perturbación de los radios de cada receptor. Esta perturbación consta de un número aleatorio entre $[-1, 1]$ para provocar un ligero ruido en los radios de los receptores, es decir, al obtener el radio de un receptor se perturba, entonces dicho receptor tendrá nuevo radio, más pequeño o más grande: perturbado. El crecimiento de los radios del centro hacia afuera ya no se conserva regular sino que algunos receptores al centro serán más grandes que otros en el contorno y viceversa. En la Figura 3.3.d se muestra la arquitectura de la forma 4.

La perspectiva lógica de esta forma es similar a la de la forma 3, la física es igual: una lista de listas con longitudes irregulares de listas. En la Figura 3.5 se muestran estas dos perspectivas de la forma 4; la más irregular.

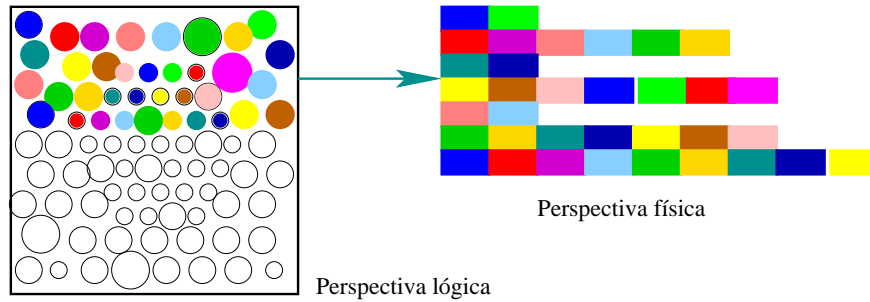


Figura 3.5: Perspectivas lógica y física de la forma 4 de los arreglos hexagonales.

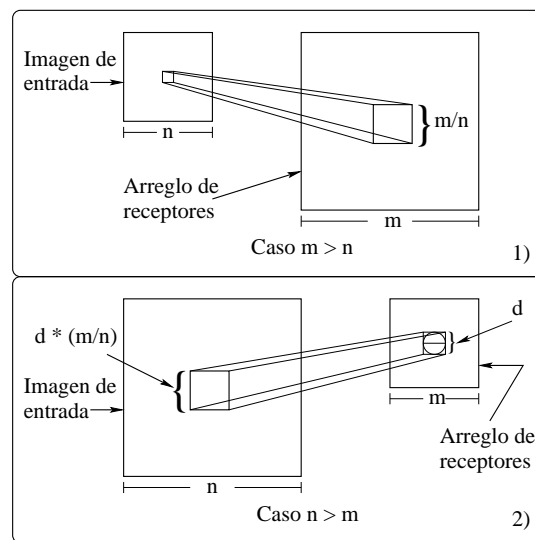


Figura 3.6: Casos de la proyección de una imagen de entrada a un arreglo hexagonal de receptores. Arriba, caso 1, la resolución de la imagen de entrada es menor que la resolución del arreglo de receptores. Abajo, caso 2, la resolución del arreglo de receptores es mayor que la de la imagen de entrada.

3.2 Proyección de la imagen de entrada

El uso de arreglos hexagonales para la representación de la retina trae un problema implícito: la proyección de la imagen de entrada a este tipo de arreglos. Esta proyección se refiere al proceso de diseminación de la luz sobre la retina humana, por lo tanto, en la proyección debe determinarse cuáles receptores son excitados por un haz de luz de la imagen de entrada.

La proyección la imagen de entrada al arreglo de receptores tiene dos casos, 1) la resolución de la imagen de entrada es menor que la resolución del arreglo de receptores, y 2) la resolución del arreglo es mayor que la resolución de la imagen. En la Figura 3.6 se ilustran estos casos.

Caso 1: la resolución de la imagen de entrada es menor que la resolución del arreglo de receptores.

La necesidad de utilizar un sistema de proyección para el caso 1 se debe a la irregularidad de la estructura de datos donde se almacenan los receptores, la cual consta de 4 campos para cada receptor: coordenada x y coordenada y de su posición en la forma lógica, radio del receptor y nivel de gris inicial. Dado que el número de receptores en cada lista no es regular, la estructura no tiene una forma rectangular, por tal motivo debemos determinar en una superficie irregular cuales son los elementos contenidos en una forma regular.

La proyección consiste de tomar una matriz de haces de luz (pixels) que forman la imagen de entrada de $p \times q$, denominada IMAGEN y la estructura de datos donde está almacenado el arreglo de receptores con una resolución de $s \times t$, denominada ARREGLO, donde t corresponde al número máximo de elementos almacenados en las listas y no a las posiciones que ocupan. Cada elemento de IMAGEN es proyectado a ARREGLO, esta proyección consiste en tomar un pixel pix_i , con coordenadas (x, y) de su posición en IMAGEN, y a través de una escalación [11], las coordenadas k, l , correspondientes a pix_i en ARREGLO, se obtienen mediante el producto de las matrices

$$\begin{bmatrix} k & l & 1 \end{bmatrix} = \begin{bmatrix} x & y & 1 \end{bmatrix} \begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (3.4)$$

Las posiciones coordenada bidimensionales (x, y) , (k, l) se representan como las triadas $\begin{bmatrix} k & l & 1 \end{bmatrix}$ y $\begin{bmatrix} x & y & 1 \end{bmatrix}$, donde el 1 se utiliza para operaciones bidimensionales. S_x y S_y son los factores de escalación. Cualquier número positivo puede asignarse a los factores de escalación. Los valores menores que 1 reducen el tamaño de los objetos; los valores mayores que 1 producen un agrandamiento. Para este caso, S_x y S_y son determinados mediante las ecuaciones

$$S_x = \frac{p}{s} \quad S_y = \frac{q}{t}. \quad (3.5)$$

Como se puede observar en el caso 1 de la Figura 3.6, la proyección de un elemento de IMAGEN a un área de ARREGLO consiste en relacionar un pixel con un cuadrado. Con la Ecuación 3.4 sólo se ha obtenido el primer punto con el que se puede formar un cuadrado (el punto superior izquierdo), para obtener el segundo punto (el punto inferior derecho), además de aplicar una escalación de las coordenadas (x, y) de pix_i , se debe aplicar una traslación [11]. Las operaciones para realizar una traslación se representan con el producto de las matrices

$$\begin{bmatrix} m & n & 1 \end{bmatrix} = \begin{bmatrix} x & y & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ T_x & T_y & 1 \end{bmatrix}, \quad (3.6)$$

donde, T_x y T_y son las distancias de traslación de las coordenadas originales x , y . Para este caso, T_x y T_y son determinados mediante las ecuaciones

$$T_x = \frac{p}{s} \quad T_y = \frac{q}{t}. \quad (3.7)$$

Para unificar estas operaciones basta con multiplicar la matriz de escalación por la matriz de traslación, para obtener la matriz:

$$\begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ T_x & T_y & 1 \end{bmatrix}. \quad (3.8)$$

De esta manera, para obtener el punto superior izquierdo del cuadrado se utilizaría $T_x = T_y = 0$, mientras que para el punto inferior derecho se aplicarían las Ecuaciones 3.7. La operación de esta transformación compuesta, finalmente sería

$$\begin{bmatrix} m & n & 1 \end{bmatrix} = \begin{bmatrix} x & y & 1 \end{bmatrix} \begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ T_x & T_y & 1 \end{bmatrix}. \quad (3.9)$$

Ya encontrados los dos puntos del cuadrado de proyección para el pixel pix_i de IMAGEN con posición en las coordenadas (x, y) , su área relacionada será

$$a_i = [(k, l), (m, n)].$$

Los receptores contenidos dentro del área a_i asociada al pixel pix_i deben ser excitados. Para determinar si un receptor rec_j , con coordenadas (u, v) está contenido en el área a_i , correspondiente a un pixel pix_i , se evalúa

$$(k < u < m) \wedge (l < v < n).$$

Si el resultado es verdadero, al receptor rec_j se le asigna el tono de gris del pixel pix_i . Esto se considera como la excitación del receptor pix_j .

Caso 2: la resolución del arreglo de receptores es mayor que la resolución de la imagen de entrada.

Para el caso 2, a cada receptor se le asocia un área dentro de la imagen de entrada. Tomando un receptor rec_j con coordenadas (x, y) y radio rad_j , para encontrar el punto superior izquierdo se aplica la Ecuación 3.9 con $T_x = T_y = 0$ y S_x y S_y se obtienen mediante las Ecuaciones 3.5. Para el punto inferior derecho, se aplica la Ecuación 3.9 con

$$T_x = \frac{p}{s} \cdot rad_j \quad T_y = \frac{q}{t} \cdot rad_j. \quad (3.10)$$

Con rad_j se especifica que el área de un receptor es proyectada a una área correspondiente en IMAGEN, tal como se ilustra en el caso 2 de la Figura 3.6. Así, a un receptor rec_j se le asocia un área

$$a_j = [(k, l), (m, n)].$$

El receptor rec_j es excitado con el promedio de los tonos de gris de los pixels p_i contenidos en a_j ,

$$\frac{\sum d_i}{(p \cdot rad)^2},$$

donde $i = 0, 1, \dots, (p \cdot rad)^2$, correspondiente a los pixels contenidos en a_j

3.3 Imagen artificial

Después del proceso de proyección se tiene un arreglo hexagonal de receptores “excitados”. A partir de este arreglo puede generarse una imagen que muestre el efecto de la proyección de la imagen de entrada sobre el arreglo de receptores, esta imagen se denomina “imagen artificial”.

La imagen artificial se produce con una proyección de los receptores sobre la matriz que contendrá a la imagen. Esta proyección consiste de tomar el centro de un receptor al cual se le asocia un punto en la matriz de la imagen artificial (ver Figura 3.7). Después, conociendo el radio del receptor proyectado, se genera un círculo en la matriz de la imagen artificial con el radio reducido del receptor. Como la generación de un círculo se entiende la asignación del tono de gris del receptor proyectado a todas las posiciones de la matriz de la imagen que estén contenidas en la circunferencia delimitada por el radio reducido del receptor proyectado. El radio del receptor proyectado se reduce en proporción a las resoluciones del arreglo de receptores y de la imagen artificial, es decir, siendo $r = u \times v$ la resolución de la imagen artificial y $R = s \times t$ la resolución del arreglo de receptores, la proporción con la que se reduce el radio correspondiente a cada centro proyectado a la imagen

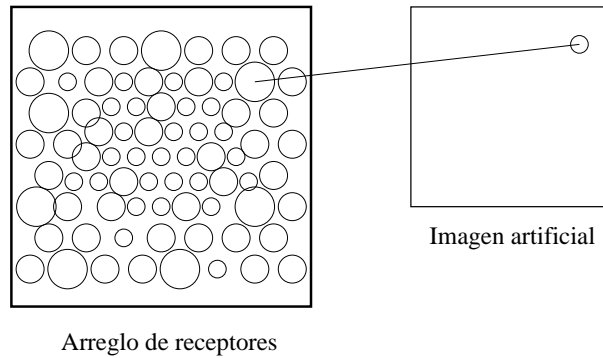


Figura 3.7: Generación de la imagen artificial a partir del arreglo de receptores.

artificial será

$$p = \frac{r}{R}.$$

Así, para un receptor en ARREGLO con radio Rad , en la imagen artificial su radio será

$$rad = p \cdot Rad.$$

Con la imagen artificial se puede observar la imagen que recibiría el cerebro a partir del arreglo de receptores. Por ello, esta imagen artificial, además de crearse para ver el resultado de la proyección, también se utiliza para observar el resultado de la interacción de los perceptrones, es decir, con este procedimiento se genera la imagen resultante de la simulación.

Capítulo 4

Modelos de Interacción de Perceptrones

Introducción

La interacción entre las células receptoras de la retina humana se pretende simular a través de tres modelos de interacción de perceptrones. En este capítulo se presentan estos modelos, las vecindades y sus modalidades que los conforman, cómo se aplican y las características de los arreglos necesarias para una convolución apropiada.

Durante la transformación de la imagen de entrada en una señal eléctrica, las células receptoras de la retina interactúan entre sí; debido a esta interacción, la respuesta de cada célula se ve afectada por la respuesta de sus vecinas. En este proceso de interacción, el fenómeno de la *inhibición lateral* toma lugar. Este fenómeno genera dos tipos de influencias entre las células: *inhibitoria* y *excitatoria*. Con la influencia inhibitoria se reduce o se cancela la respuesta de la célula influenciada, mientras que con la excitatoria se amplifica o se mantiene.

4.1 Convolución entre funciones

El proceso de influencia e interacción puede ser representado por una convolución de dos funciones: $f(x)$ y $g(x)$, denotada por $f(x) \cdot g(x)$, la cual es definida de la manera siguiente:

$$f(x) \cdot g(x) = \int_{-\infty}^{\infty} f(\alpha)g(x - \alpha)d\alpha. \quad (4.1)$$

Para entender la operación de convolución, en la Figura 4.1 se presenta un ejemplo que muestra la convolución de las funciones $f(x)$ y $g(x)$ ilustradas en las Figuras 4.1.a y 4.1.b, respectivamente [12]. Antes de integrar, se debe formar la función $g(x - \alpha)$. Para hacerlo se requiere de dos pasos

mostrados en las Figuras 4.1.c y 4.1.d. Esta operación es simplemente de invertir $g(\alpha)$ sobre el origen para obtener $g(-\alpha)$ y entonces desplazar esta función con x . Luego, por cualquier valor de x , se multiplica $f(\alpha)$ por el correspondiente $g(x - \alpha)$ y se integra el producto desde $-\infty$ a ∞ . El producto de $f(\alpha)$ y $g(x - \alpha)$ es la porción intersectada de la Figura 4.1.e, para el intervalo $[0, 1]$. El producto es 0 para valores de α fuera del intervalo $[0, x]$, así que $f(x) \cdot g(x) = x/2$, lo cual es simplemente el área de la región intersectada en la Figura 4.1.e. Para x en el intervalo $[1, 2]$, se aplica la Figura 4.1.f, y $f(x) \cdot g(x) = (1 - x/2)$. Esto porque $f(\alpha)g(x - \alpha)$ es cero para valores de x fuera del intervalo $[0, 2]$, teniendo finalmente

$$f(x) \cdot g(x) = \begin{cases} x/2 & 0 \leq x \leq 1 \\ 1 - x/2 & 1 \leq x \leq 2 \\ 0 & \text{de otra forma.} \end{cases}$$

La Figura 4.1.e) muestra el resultado.

Para este proceso, f representa al arreglo de perceptrones excitados por la imagen de entrada y g representa un filtro (función de transferencia) que se aplica sobre el arreglo de perceptrones para producir el efecto deseado.

La operación de convolución se realizará sobre una imagen de entrada y una función de transferencia, ambas integradas como funciones en 2 dimensiones. La operación de convolución para las funciones $f(x, y)$ y $g(x, y)$ está definida como

$$f(x, y) * g(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\alpha, \beta)g(x - \alpha, y - \beta)d\alpha d\beta \quad (4.2)$$

La función de transferencia pretende modelar los fenómenos inhibitorio y excitatorio de la interacción entre perceptrones. La región sobre la cual se define una función de transferencia es la vecindad definida sobre un perceptrón central.

4.2 Modelo de red neuronal propuesto

Cada célula de visión de la retina se simula con una célula artificial llamada *perceptrón*, por ello, a partir de la excitación de los receptores, éstos se convierten en perceptrones, los cuales interactuarán unos con otros al producir su respuesta de salida. A cada perceptrón se le asocia un peso de acuerdo a la función de transferencia utilizada en cada modelo de interacción de perceptrones.

El modelo de red neuronal propuesto consiste de 2 grupos de neuronas: una capa de receptores y diez capas de perceptrones, como se muestra en la Figura 4.2. Este modelo maneja una imagen de

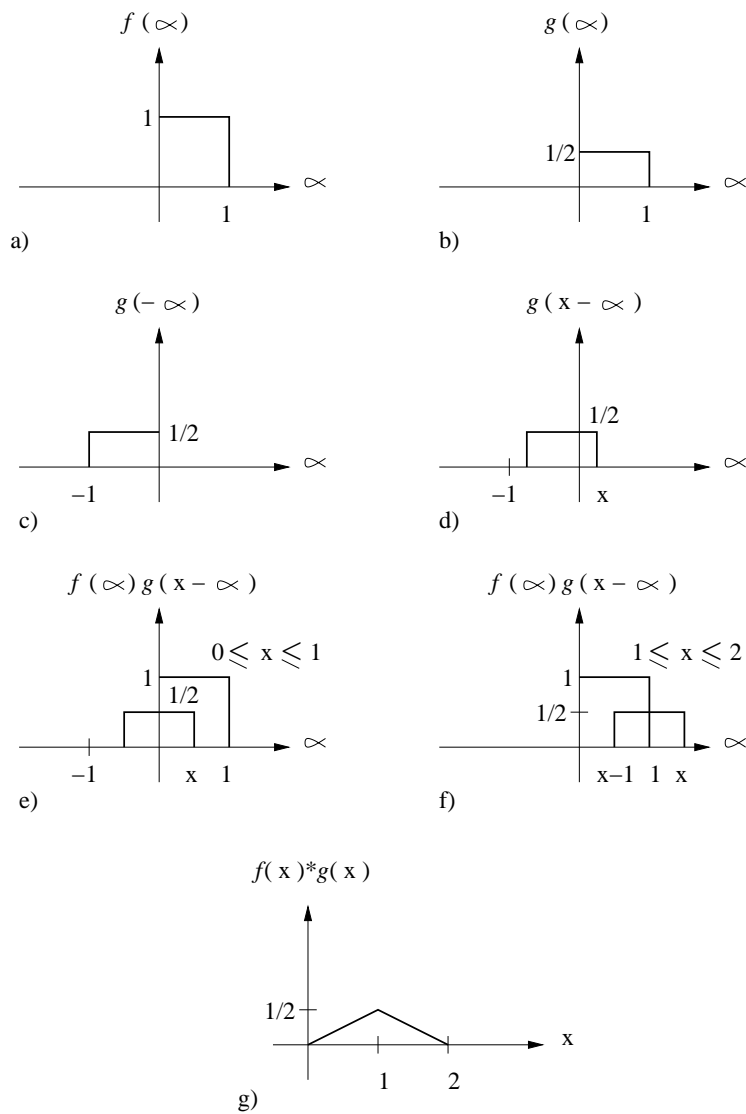


Figura 4.1: Ilustración gráfica de la convolución. Las áreas intersectadas indican las regiones donde el producto no es *ceró*.

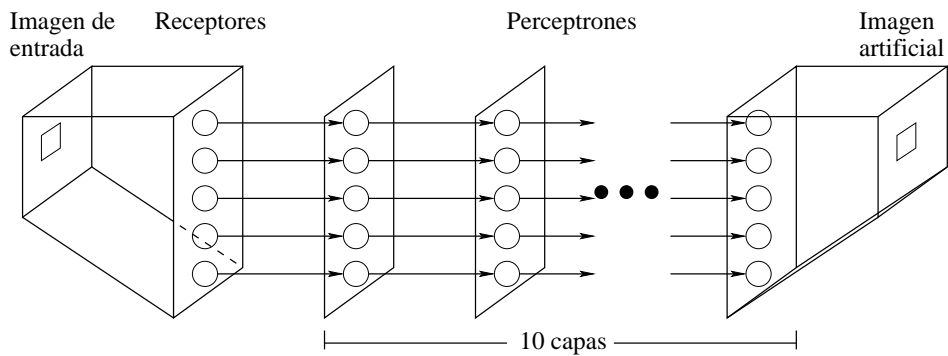


Figura 4.2: Modelo de red neuronal propuesto.

Función	Descripción
$G_1(x, y) = \exp\left(\left[\frac{(x-u)^2}{\sigma} + \frac{(y-v)^2}{\sigma}\right]\right)$	Función exponencial, e^{-x^2}
$G_2(x, y) = \text{sinc}\left(\left[\frac{(x-u)^2}{\sigma} + \frac{(y-v)^2}{\sigma}\right]\right)$	Función sinc, $\frac{\sin(x)}{x}$
$G_3(x, y) = j_0\left(\left[\frac{(x-u)^2}{\sigma} + \frac{(y-v)^2}{\sigma}\right]\right)$	Función Bessel j_0
$G_4(x, y) = j_1\left(\left[\frac{(x-u)^2}{\sigma} + \frac{(y-v)^2}{\sigma}\right]\right)$	Función Bessel j_1

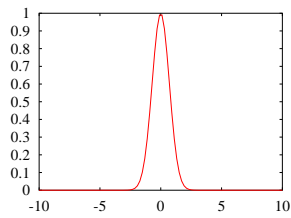
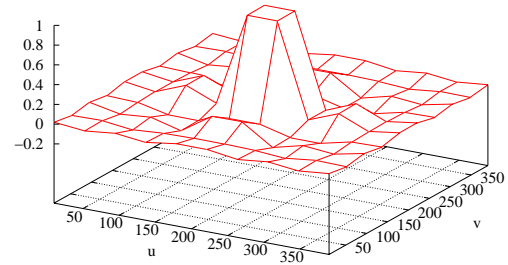
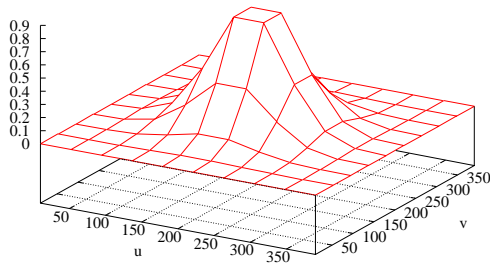
Tabla 4.1: Funciones de transferencia utilizadas por los modelos de agrupación de perceptrones.

entrada en escala de grises de 8 bits de $n \times n$ pixels. Los receptores se excitan con los tonos de gris de la imagen de entrada y transmiten su valor a la primera capa de perceptrones. Cada perceptrón toma el valor obtenido de un único receptor, produce una salida con la aplicación de alguno de los modelos de interacción de perceptrones y la trasmite a su similar en la siguiente capa.

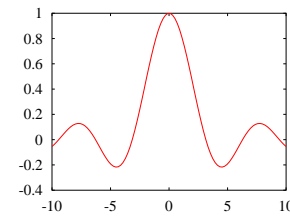
Durante el procesamiento de la información los perceptrones pueden tener dos estados, *centro* o *vecino*, que representan su comportamiento en cada unidad de procesamiento. Por consideraciones fisiológicas y resultados experimentales, la red neuronal cuenta con dos condiciones de paro: un número máximo de capas y un umbral de activación [6]. Se manejan a lo más 10 capas, por lo establecido en el proceso biológico de la retina y un umbral del orden de 1×10^{-5} donde se alcanza la estabilidad de la red.

En los modelos de interacción de perceptrones la respuesta de cada perceptrón está en función de la luz de entrada y de las respuestas de los perceptrones vecinos. La influencia de los perceptrones es definida por la inhibición lateral representada por diferentes funciones (funciones de transferencia). Esta influencia es considerada el peso del perceptrón. En la Tabla 4.1 se presentan las funciones de transferencia usadas por los modelos de agrupación, donde u, v , son las coordenadas del perceptrón central; x, y , las coordenadas de los perceptrones vecinos; y σ , la amplitud del desplazamiento. En la Figura 4.3 se puede observar la forma de cada función en una gráfica representativa.

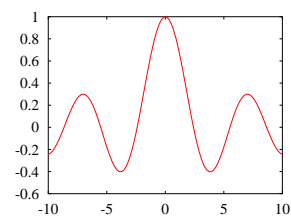
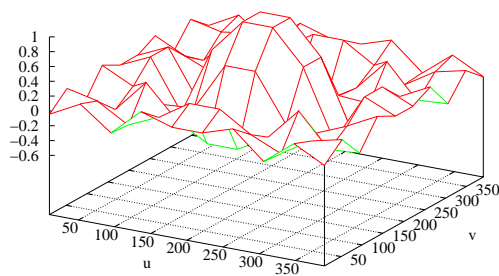
Con la función exponencial, a los elementos del dominio cercanos a un elemento central se les asigna valores mayores que a los elementos más alejados. Los valores del contradominio de esta función se encuentran en el rango $[0, 1]$. La función Sinc tiene un comportamiento similar al de la función exponencial, los valores cercanos al centro son más altos que los más alejados pero, la función Sinc puede asignar valores negativos para los valores más alejados. El rango para el



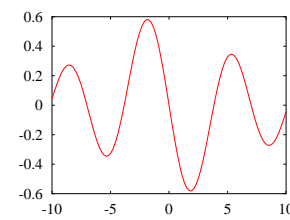
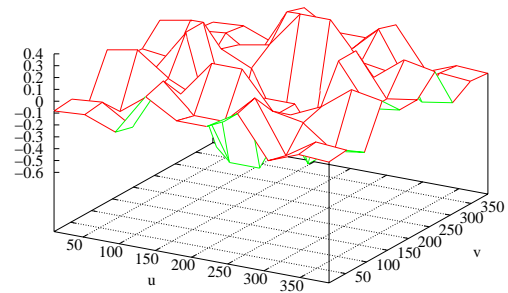
(c) Exponencial



(d) SINC



(g) Bessel j_0



(h) Bessel j_1

Figura 4.3: Gráficas descriptivas de las funciones utilizadas por los modelos de interacción de perceptrones.

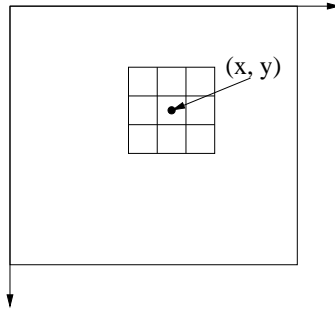


Figura 4.4: Vecindad de 3×3 de un punto (x, y) en una imagen.

contradominio de la función Sinc es $[-1, 1]$. La función Bessel j_0 tiene un comportamiento similar al de la función Sinc, sólo que más abrupto. Los valores del contradominio para esta función se encuentran entre $[-1, 1]$ pero, su asignación ya no es tan regular, los elementos más alejados no son necesariamente más pequeños que los más cercanos al centro. La función Bessel j_1 tiene un comportamiento bastante irregular, esta función no conserva la forma de las funciones anteriores, genera números positivos y negativos dentro del rango $[-1, 1]$ pero el patrón que sigue es bastante irregular.

Para determinar los perceptrones vecinos de un perceptrón central se definieron tres tipos de vecindades espaciales para la agrupación de perceptrones: a) vecindad hexagonal de 7 elementos, b) vecindad circular, y c) vecindad elipsoidal, sobre las cuales se aplica una función de transferencia [12]. El dominio espacial de una imagen se refiere al plano de la imagen misma. Otro dominio de las imágenes es el dominio de la frecuencia. Las funciones de procesamiento de imágenes en el dominio espacial pueden ser expresadas como

$$g(x, y) = T[f(x, y)], \quad (4.3)$$

donde $f(x, y)$ es la imagen de entrada, $g(x, y)$ es la imagen procesada, y T es un operador de f , definido sobre alguna vecindad de (x, y) . La forma básica para definir una vecindad cerca de (x, y) es usando una región cuadrada o rectangular centrada en (x, y) , como muestra la Figura 4.4. El operador aplicado aquí es el de convolución y éste de calcula moviendo el centro de la vecindad sobre cada pixel de la imagen. El centro de la vecindad se mueve pixel a pixel, comenzando en la esquina superior izquierda y aplicando el operador a cada localidad (x, y) del campo g de esa localidad. Es común en procesamiento digital de imágenes utilizar vecindades cuadradas, rectangulares o circulares. Sin embargo, en el caso del trabajo aquí reportado se utilizaron las vecindades que se describen a continuación.

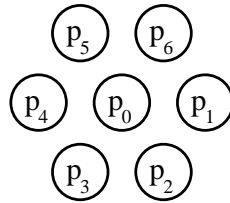


Figura 4.5: Modelo matemático para agrupar los perceptrones: vecindad hexagonal de 7 elementos, p_0 es el perceptrón central y p_{1-6} los vecinos.

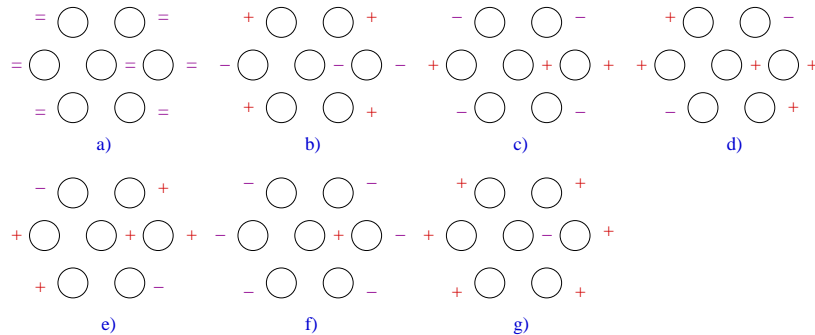


Figura 4.6: Modalidades de la vecindad hexagonal de 7 elementos para la determinación de los pesos de los perceptrones.

4.3 Vecindad hexagonal de 7 elementos

El primer modelo para agrupar los perceptrones es la vecindad hexagonal de 7 elementos, la cual se compone de un perceptrón central (p_0) y 6 vecinos (p_{1-6}) definidos como se ilustra en la Figura 4.5. A cada elemento vecino se le asocia un peso correspondiente a la influencia que ejercerá sobre el perceptrón central, así también, al perceptrón central se le asigna un peso correspondiente a su respuesta a la luz de entrada. Para la determinación de los pesos de los elementos de la vecindad se tienen 7 modalidades (ver Figura 4.6), entre las cuales se encuentran: a) promediación de pesos, b) y c) detectores de bordes horizontales, d) y e) detectores de bordes verticales, f) y g) promediación de pesos utilizando las funciones *exp* y su inversa.

Para la conservación de la escala de grises de 8 bits (de 0 a 255), la suma de los pesos de los perceptrones debe ser igual a 1. El valor del perceptrón siguiente se obtiene mediante la suma ponderada de los elementos de la vecindad, tal como en la ecuación

$$P_0^{k+1} = \sum_{i=0}^n w_i P_i^k, \tag{4.4}$$

donde $i = 0, 1, \dots, 6$; k , representa el nivel o capa de los perceptrones; w_i , el peso o influencia de cada perceptrón; y P , su tono de gris.

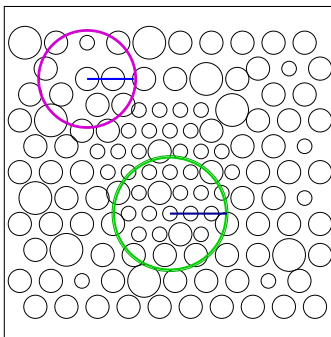


Figura 4.7: Modelo matemático para agrupar los perceptrones: vecindad circular, un perceptrón central y un variado número de vecinos.

La vecindad hexagonal de 7 elementos sólo se puede aplicar a las Formas 1 y 2 de los arreglos hexagonales de perceptrones (ver Figura 3.1), debido a que la convolución de la vecindad con el arreglo de perceptrones es uno a uno, de tal forma que se requiere una estructura de almacenamiento regular como la mostrada en la Figura 3.4.

4.4 Vecindad circular

La vecindad circular es una generalización de la vecindad hexagonal de 7 elementos, extendiendo el número de perceptrones participantes a todos aquellos que se encuentren contenidos en una circunferencia que la delimita (ver Fig. 4.7). El radio de la vecindad es arbitrario. Para generar la vecindad circular se determina un perceptrón como central y a partir de él se genera la circunferencia.

4.4.1 Circunferencia: conceptos y aplicación

Una circunferencia es el lugar geométrico de un punto que se mueve en un plano de tal manera que se conserva siempre a una distancia constante de un punto fijo de ese plano [13]. El punto fijo se llama *centro* de la circunferencia, y la distancia constante se llama *radio*, estos elementos pueden observarse en la Figura 4.8.

La circunferencia cuyo centro es el punto (h, k) y cuyo *radio* es la constante r , tiene por ecuación

$$(x - h)^2 + (y - k)^2 = r^2. \quad (4.5)$$

Para determinar si un punto P está contenido dentro de la región delimitada por una circunferencia, se utiliza una propiedad de la circunferencia: sea $C(h, k)$ el centro de la circunferencia y r

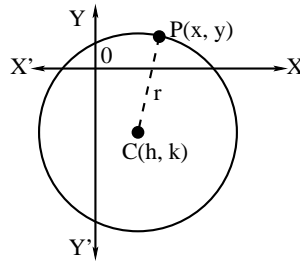


Figura 4.8: Elementos de una circunferencia.

su radio, si P está contenido se debe satisfacer la condición geométrica

$$|\overline{CP}| \leq r. \quad (4.6)$$

Para aplicar una función de transferencia con vecindad circular, es necesario determinar los perceptrones contenidos dentro de la circunferencia centrada en un perceptrón. Como puede verse en la Figura 4.7, el número de perceptrones vecinos de un perceptrón central no es siempre el mismo, varía dependiendo del radio de la vecindad, del radio y la separación de los perceptrones, etc. Se aplicaron tres modalidades de la vecindad circular determinadas por las siguientes estrategias para la selección de vecinos:

1. **Vecinos completos.** Se consideran como vecinos sólo aquellos perceptrones que están completamente contenidos en la circunferencia. Es decir, si el radio del perceptrón i más la distancia entre los centros del perceptrón i y de la vecindad es menor que el radio de la vecindad, entonces se considera como vecino al perceptrón i .
2. **Vecinos parciales.** Se consideran como vecinos aquellos perceptrones que están contenidos el 50% o más de su área en la circunferencia de la vecindad. Es decir, si el área de intersección entre las circunferencias del perceptrón i y de la vecindad es mayor o igual a la mitad del área del perceptrón i , entonces se considera como vecino al perceptrón i y su influencia se considerará como la de un perceptrón contenido completamente.
3. **Vecinos proporcionales.** Se consideran como vecinos aquellos que se encuentren contenidos a partir del 1% de su área en la circunferencia de la vecindad, pero su influencia es proporcional al porcentaje contenido. Es decir, si el área de intersección entre las circunferencias del perceptrón i y de la vecindad es mayor o igual al 1% del área del perceptrón i , entonces se considera como vecino al perceptrón i , pero su influencia se verá reducida en proporción al porcentaje de área contenida en la vecindad.

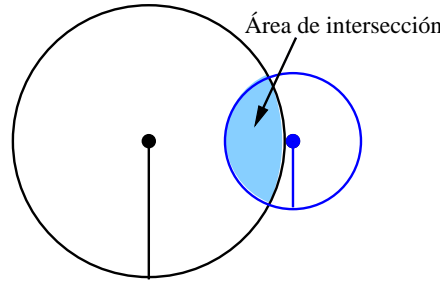


Figura 4.9: Área de intersección entre dos circunferencias.

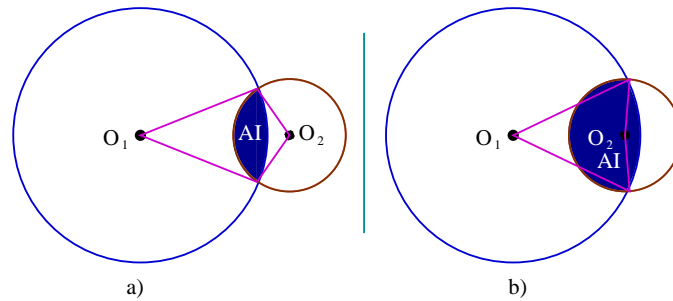


Figura 4.10: Casos para el cálculo del área de intersección entre dos circunferencias.

Para los casos 2 y 3, es necesario calcular el área de intersección entre las circunferencias del perceptrón y de la vecindad circular (ver Figura 4.9); es importante resaltar que se asume que el radio de la circunferencia de la vecindad debe ser mayor que el radio de la circunferencia del perceptrón.

4.4.2 Cálculo del área de intersección entre dos circunferencias

El cálculo del área de intersección de dos circunferencias tiene dos casos: 1) el centro de la circunferencia del perceptrón no está contenido en la circunferencia de la vecindad; y 2) la circunferencia de la vecindad contiene el centro de la circunferencia del perceptrón (ver Figura 4.10). A continuación se describen las operaciones para el primer caso; las operaciones para el segundo caso son similares al caso anterior.

Círculos alineados

Considere por simplicidad que los centros de las dos circunferencias están alineadas con el eje de las X s como se muestra en la Figura 4.11.

Sean C_1 , C_2 las circunferencias intersectadas, $C_1 > C_2$; $A = (x_A, y_A)$, $B = (x_B, y_B)$ los puntos de intersección de C_1 y C_2 ; $O_1 = (x_1, y_1)$, $O_2 = (x_2, y_2)$ los centros de C_1 y C_2 , respectivamente; r_1 , el radio de C_1 ; r_2 , el radio de C_2 .

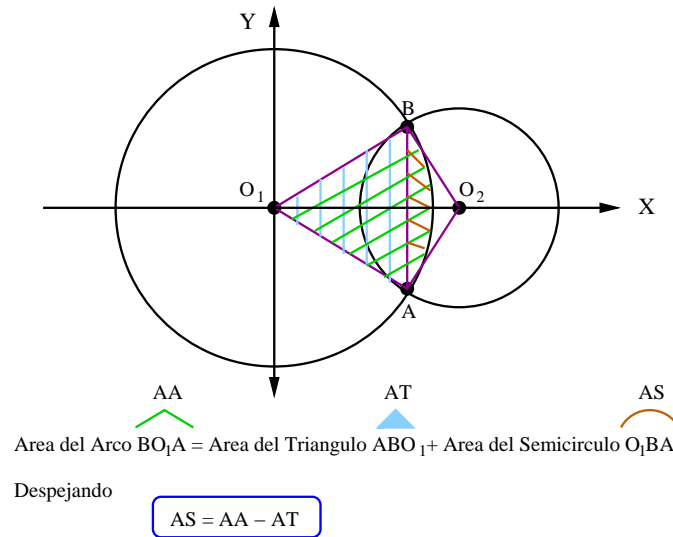


Figura 4.11: Cálculo del área de intersección entre dos circunferencias.

El área de intersección entre los dos círculos está delimitado por: 1) el área del arco que se forma por los dos puntos de intersección de los círculos A y B , 2) los triángulos formados por las líneas que unen el centro de cada círculo a los puntos de intersección (de longitud igual al radio del círculo) y la línea que une los dos puntos de intersección (O_1AB, O_2AB). El área delimitada por estas dos áreas forma un semicírculo, cuya área se puede obtener como se explica a continuación.

Sean $T_1 = \triangle O_1AB$, el triángulo O_1AB ; $T_2 = \triangle O_2AB$, el triángulo O_2AB ; A_1 , el arco $\cap AB$ con centro O_1 ; A_2 , arco $\cap AB$ con centro O_2 ; $\alpha = \angle O_1AB$, ángulo formado entre O_1, A y B ; $\beta = \angle O_2AB$, ángulo formado entre O_2, A y B ; AA_1 , área del arco A_1 ; AT_1 , área del triángulo T_1 ; y AS_1 , el área del semicírculo O_1AB ; AA_2 , área del arco A_2 ; AT_2 , área del triángulo T_2 ; y AS_2 , el área del semicírculo O_2AB (ver Figura 4.11). AA_1 contiene las áreas AT_1 y AS_1 , es decir,

$$AA_1 = AT_1 + AS_1, \tag{4.7}$$

entonces el área del semicírculo AS_1 se puede obtener despejando de 4.7,

$$AS_1 = AA_1 - AT_1, \tag{4.8}$$

el área del semicírculo AS_2 se obtiene de forma análoga

$$AS_2 = AA_2 - AT_2. \tag{4.9}$$

Así, conociendo las áreas de los semicírculos AS_1 y AS_2 formados en C_1 y C_2 , se puede conocer el

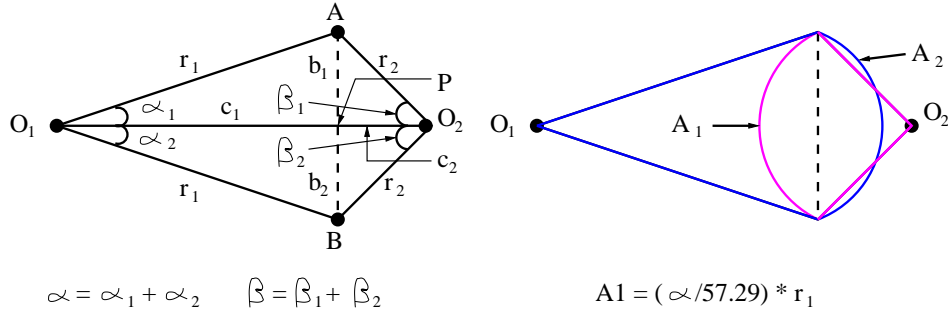


Figura 4.12: Elementos requeridos para el cálculo del área de intersección de dos circunferencias.

área de intersección de las circunferencias con la Ecuación 4.10,

$$AS_1 + AS_2 = AA_1 + AA_2 - AT_1 - AT_2. \quad (4.10)$$

Cálculo de los puntos de intersección A, B

Los primeros elementos que se requieren calcular son los puntos de intersección (A, B) de las dos circunferencias, mostrados en la Figura 4.12.

Sean E_1 y E_2 las ecuaciones de las circunferencias C_1 y C_2 , con centros alineados con el eje de las X s y con el centro del eje de coordenadas en O_1 .

$$\begin{aligned}
 E_1 : x^2 + y^2 &= (r_1)^2 \\
 E_2 : x^2 + y^2 - (2x_2)x - (2y_2)y + (x_2)^2 + (y_2)^2 &= (r_2)^2,
 \end{aligned}$$

Las coordenadas de los puntos $A, (x_A, y_A)$ y $B, (x_B, y_B)$, se obtienen al solucionar el sistema de ecuaciones formado por las ecuaciones de ambas circunferencias: E_1 y E_2 . Para solucionar este sistema es suficiente aplicar el método de eliminación, empleado sólo un paso: $-E_1 + E_2$

$$\begin{aligned}
 E_1 : x^2 + y^2 - (r_1)^2 &= 0 \\
 E'_2 : -(2x_2)x - (2y_2)y + (x_2)^2 + (y_2)^2 - (r_2)^2 + (r_1)^2 &= 0.
 \end{aligned}$$

De la ecuación resultante E'_2 se despeja x

$$x = -\frac{y_2}{x_2}y + \frac{x_2}{2} + \frac{y_2}{2x_2} - \frac{(r_2)^2}{2x_2} + \frac{(r_1)^2}{x_2}, \quad (4.11)$$

y tomando $\frac{x_2}{2} + \frac{y_2}{2x_2} - \frac{(r_2)^2}{2x_2} + \frac{(r_1)^2}{x_2}$ como la constante c , se sustituye el equivalente de x en E_1 ,

obteniendo E'_1

$$E'_1 : \left(-\frac{y_2}{x_2}y + c\right)^2 + y^2 - (r_1)^2 = 0.$$

Esta ecuación queda como una ecuación de segundo grado en términos de y , así que al resolverla se obtienen los valores de y : y_A y y_B . Sustituyendo estos valores en la Ecuación 4.11 se obtiene la solución para x : x_A y x_B .

Cálculo del área de los triángulos AT_1 y AT_2

Para encontrar las áreas de los triángulos T_1 y T_2 (AT_1 y AT_2), mostrados en la Figura 4.11, se requieren conocer su base y altura, para aplicar la fórmula del área de un triángulo.

La base de los triángulos se obtiene calculando la distancia entre los puntos A y B . La altura corresponde a la distancia entre el punto P (con coordenadas $(x_A, 0)$) y el centro de C_1 (O_1). El cálculo del área del triángulo T_2 (AT_2) es análogo.

Cálculo del área de los arcos A_1 y A_2

La fórmula para encontrar el área de un arco es

$$\frac{\text{larco} \cdot \text{radio}}{2}$$

donde *larco*, es la longitud del arco, y *radio*, el radio de la circunferencia que lo contiene. Por tanto, para encontrar el área del arco A_1 (AA_1) se necesita conocer su longitud. El radio del círculo es conocido. La longitud de un arco se calcula con la fórmula

$$\text{larco} = \frac{\alpha}{57.29} \cdot \text{radio}$$

Así, para obtener la longitud del arco A_1 se requiere del ángulo α (ver Figura 4.12). Este se obtiene de la suma de los ángulos α_1 y α_2 . Estos ángulos se calculan con la fórmula

$$\alpha_1 = \arcsin\left(\frac{\text{cateto_opuesto}}{\text{hipotenusa}}\right)$$

La *hipotenusa* es el radio de la circunferencia, mientras que los catetos opuestos (b_1, b_2) equivalen

$$b_1 = b_2 = \frac{y_A + y_B}{2}$$

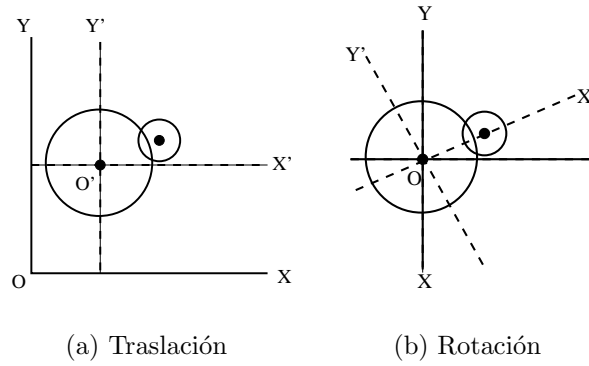


Figura 4.13: Traslación del eje de coordenadas al centro de C_1 y Rotación del eje de coordenadas para alinear los centros de las circunferencias al eje de las Xs.

Con estas fórmulas se puede encontrar el área correspondiente a la fracción de la circunferencia delimitada por el arco A_1 . El cálculo del área del arco A_2 es análogo.

Conociendo las áreas de los triángulos T_1 y T_2 y de los arcos A_1 y A_2 se obtiene el área de la intersección.

Alineación de círculos

Cuando los centros no están alineados o el centro del eje de coordenadas no está ubicado en uno de ellos, se puede aplicar una traslación y una rotación al eje de coordenadas, según se requiera para cumplir estas condiciones (ver la Figura 4.13). Las operaciones de traslación y rotación se pueden representar mediante la siguiente transformación

$$\begin{bmatrix} x' & y' & 1 \end{bmatrix} = \begin{bmatrix} x & y & 1 \end{bmatrix} \begin{bmatrix} \cos(\theta) & \sin(\theta) & 0 \\ -\sin(\theta) & \cos(\theta) & 0 \\ T_x & T_y & 1 \end{bmatrix}, \quad (4.12)$$

donde, T_x y T_y son las distancias de traslación, las cuales son obtenidas mediante $T_x = -x_1$ y $T_y = -y_1$; θ es el ángulo de rotación, el cual se obtiene con la ecuación

$$\theta = \arctan\left(\frac{x_2 - x_1}{y_2 - y_1}\right)$$

donde, $x_2 - x_1$ corresponde al cateto adyacente y $y_2 - y_1$ al cateto opuesto.

4.4.3 Aplicación de la función de transferencia

Como se mencionó al inicio de la sección, el radio de la vecindad circular es arbitrario, por tanto, el número de perceptrones que la vecindad contendrá será variable, además, la influencia de cada perceptrón se verá disminuida conforme aumente la distancia de la posición de su centro con respecto al centro de la vecindad. Este efecto está demostrado según pruebas fisiológicas: entre más cerca estén los receptores más fuerte se influyen y mientras más alejados estén menor será su influencia, propiedades “on-” y “off-center” [1]. Para simular esto, se aplica una de las funciones de transferencia descritas en la Tabla 4.1 y graficadas en la Figura 4.3: la función G_1 , la cual por su comportamiento proporcionará un valor pequeño para los elementos más alejados del centro y un valor grande para los más cercanos.

Sean (x, y) las coordenadas del perceptrón central y r_c su radio. Sea $P = \{p_0, p_1, p_2, \dots, p_i, \dots, p_n\}$ el conjunto de perceptrones contenidos en la vecindad circular.

$$\forall p \in P, p_k = \langle u_k, v_k, r_k, c_k, a_k \rangle$$

donde, u_k es la coordenada en X del perceptrón k ; v_k , es la coordenada en Y ; r_k , es el radio; c_k , es el color; y a_k , es un número real entre 0 y 1, el cual determina el porcentaje del área del perceptrón k contenida en la vecindad.

Para el cálculo de los pesos de los perceptrones se requieren 4 pasos:

- **Paso 1.** Cálculo de la función de transferencia centrada en (x, y) .
- **Paso 2.** Obtención del área de cada perceptrón

$$p_area_k = (\pi \cdot r_k^2) \cdot a_k.$$

- **Paso 3.** Obtención de la contribución proporcional al área y a la distancia entre los centros de la vecindad y el del perceptrón vecino

$$p_contri_k = G_1(x, y) \cdot p_area_k.$$

- **Paso 4.** Normalización de la contribución de todos los perceptrones en la vecindad.

$$p_contri_n_k = \frac{p_contri_k}{\sum_{j=0}^n p_contri_k}.$$

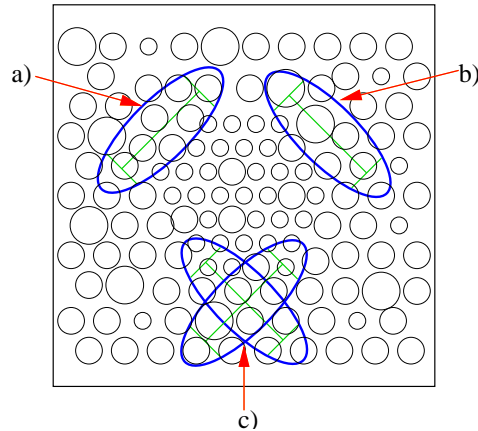


Figura 4.14: Modelo matemático para agrupar los perceptrones: vecindad elipsoidal. Un perceptrón central y un variado número de vecinos. a) Elipse a 45° . b) Elipse a 135° . c) Elipse doble.

4.5 Vecindad elipsoidal

La vecindad circular permite aplicar funciones de transferencia con simetría radial sobre regiones que son igualmente simétricas. Esta simetría hace que la contribución de un perceptrón sea independiente de su orientación y sólo dependa de su distancia a otro perceptrón.

En algunos estudios fisiológicos se ha mostrado que la interacción de una célula receptora depende también de la orientación en la cual se encuentra con respecto a otras. Para mostrar lo anterior, basta con aplicar una rotación al patrón de cuadros negros (objeto de estudio de este trabajo de tesis), para mostrar que en un momento dado las manchas desaparecen.

Una manera simple de modelar este efecto es eliminar parcialmente la simetría completamente radial de las vecindades circulares y aplicar vecindades elipsoidales cuyo eje mayor determinaría la preferencia de la orientación de las células receptoras.

La vecindad elipsoidal puede verse como una especialización de la vecindad circular, reduciendo el número de perceptrones participantes a aquellos contenidos dentro del perímetro de una elipse. Se tienen 3 modalidades: a) la elipse se rota 45° , para obtener un sesgo de la esquina inferior izquierda a la superior derecha; b) la elipse se rota 135° , para obtener un sesgo de la esquina inferior derecha a la superior izquierda; y c) una elipse doble, que es el promedio de los resultados de las elipses rotadas a 45 y 135 grados. En la Figura 4.14 se presentan las tres modalidades de esta vecindad.

4.5.1 Elipse: conceptos y aplicación

Una elipse es el lugar geométrico de un punto que se mueve en un plano de tal manera que la suma de sus distancias a dos puntos fijos de ese plano es siempre igual a una constante, mayor que la

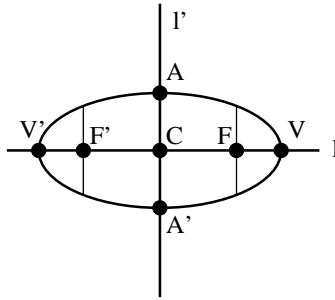


Figura 4.15: Elementos de una elipse.

distancia entre los dos puntos [13].

Los dos puntos fijos se llaman *focos* de la elipse. Designados por F y F' (ver Figura 4.15) los focos de una elipse. La recta l que pasa por los focos se llama *eje focal*. El eje focal corta a la elipse en dos puntos, V y V' , llamados *vértices*. La porción del eje focal comprendida entre los vértices, el segmento VV' , se llama *eje mayor*. El punto C del eje focal, punto medio del segmento que une los dos focos, se llama *centro*. La recta l' que pasa por C y es perpendicular al eje focal l se llama *eje normal*. El eje normal l' corta a la elipse en dos puntos, A y A' , y el segmento AA' se llama *eje menor*. La ecuación para una elipse con centro en el origen y cuyo eje focal coincide con el eje X se muestra en 4.13,

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1, \quad (4.13)$$

donde a y b son constantes.

Para la rotación de la elipse a 45 y 135 grados, para las modalidades a) y b) de la vecindad elipsoidal, se rota el eje coordenado: Si los ejes coordenados giran con un ángulo ϕ en torno de su origen como centro de rotación, y las coordenadas de un punto cualquiera P antes y después de la rotación son (x, y) y (x', y') , respectivamente, las ecuaciones de transformación del sistema original al nuevo sistema de coordenadas están dadas por

$$\begin{aligned} x' &= x \cos \theta - y \sin \theta, \\ y' &= x \sin \theta + y \cos \theta. \end{aligned}$$

En la Figura 4.16 se puede observar el tipo de rotación que se utiliza para rotar la elipse.

Con este sistema, las coordenadas de los vértices V y V' , $(a, 0)$ y $(-a, 0)$, se transforman en (l, k) y $(-l, -k)$, respectivamente, donde $l = a \cos(\theta)$ y $k = a \sin(\theta)$. Así también, las coordenadas de A y A' , $(0, b)$ y $(0, -b)$, se transforman en $(-r, s)$ y $(r, -s)$, donde $r = b \sin(\theta)$ y $s = b \cos(\theta)$, respectivamente. Las constantes a y b son utilizadas por el sistema para determinar la dimensión de la elipse a través de su ecuación. Para determinar si un elemento está contenido en la elipse, se

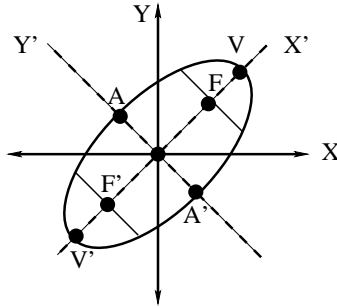


Figura 4.16: Rotación del eje coordenado para la rotación de una elipse.

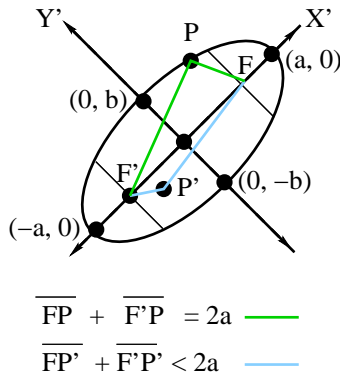


Figura 4.17: Condiciones para la selección de vecinos contenidos en la elipse.

utiliza una de las propiedades de las elipses: sea $P(x, y)$ un punto cualquiera de la elipse. Por la definición de la curva, el punto P debe satisfacer la condición geométrica

$$|\overline{FP}| + |\overline{F'P}| \leq 2a. \tag{4.14}$$

4.5.2 Aplicación de la función de transferencia

Para la aplicación de la función de transferencia a la vecindad elipsoidal se determina un perceptrón como central y sus coordenadas (x, y) son consideradas como el centro de la elipse (punto C). A partir de este punto se genera un área de búsqueda para localizar los perceptrones vecinos del perceptrón central. El área de búsqueda se puede ver como un cuadrado que contiene a la elipse, (ver Figura 4.18). Las coordenadas del cuadrado se determinan utilizando las coordenadas del perceptrón central y las coordenadas (l, k) de V (vértice de la elipse). Así, las coordenadas para el cuadrado estarán definidas por (x_1, y_1) y (x_2, y_2) , donde, $x_1 = x - l$, $y_1 = y - k$, $x_2 = x + l$ y $y_2 = y + k$. La distancia entre el centro de la elipse y el punto V , r , se considera el radio de la elipse.

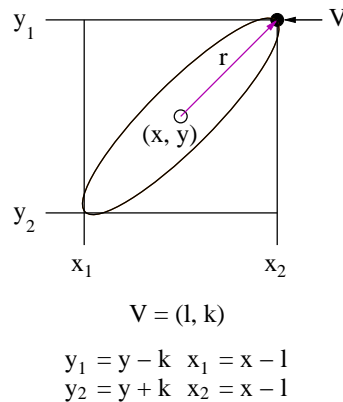


Figura 4.18: Área de búsqueda para la vecindad elipsoidal.

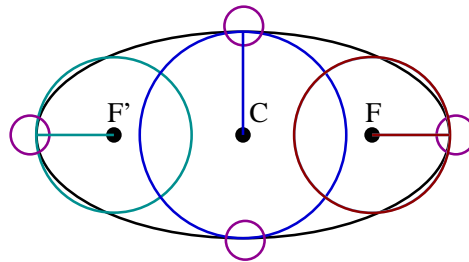


Figura 4.19: División de la elipse en tres circunferencias para calcular el porcentaje de contención de un perceptrón por la elipse de la vecindad elipsoidal.

En esta área de búsqueda se aplican las estrategias de selección de vecinos de la vecindad elipsoidal. Estas estrategias son similares a las estrategias de la vecindad circular, sólo que ahora el contenedor es una elipse.

Para la estrategia **vecinos completos**, se consideran como vecinos sólo aquellos perceptrones que están completamente contenidos en la elipse. En este caso, se utiliza una variante de la Ecuación 4.14, ya que esta indica si el centro del perceptrón está contenido en la elipse. Para saber si un perceptrón p_i está completamente contenido en la elipse, se utiliza

$$|\overline{FP}| + |\overline{F'P}| + r_i < 2a, \quad (4.15)$$

donde r_i , es el radio del perceptrón p_i .

Las estrategias **vecinos parciales** y **vecinos proporcionales** utilizan el mismo criterio de evaluación, sólo que en este caso se evalúa el área del perceptrón contenida en la elipse, y esto implica calcular el área de intersección de la circunferencia del perceptrón con la elipse de la vecindad. Para simplificar este cálculo, la elipse se aproxima como la unión de tres circunferencias como se muestra en la Figura 4.19. Esta división de la elipse genera tres casos:

1. El perceptrón se intersecta con la circunferencia de centro en F : la distancia entre F y el centro del perceptrón es menor o igual que el radio de la circunferencia de centro en F .
2. El perceptrón se intersecta con la circunferencia de centro en F' : la distancia entre F' y el centro del perceptrón es menor o igual que el radio de la circunferencia de centro en F' .
3. El perceptrón se intersecta con la circunferencia de centro en C , o visto de otra manera, si no se intersecta con las circunferencia de centro en F y F' , el perceptrón se evaluará con la circunferencia con centro en C .

Después de determinar con qué circunferencia se intersecta el perceptrón, se procede a calcular el área de intersección entre la circunferencia seleccionada y la circunferencia del perceptrón. Este cálculo se realiza con el mismo procedimiento utilizado en el modelo de la vecindad circular.

Teniendo ya identificados los perceptrones vecinos de un perceptrón central se obtienen los pesos de los perceptrones. Para la generación de los pesos de los perceptrones de la vecindad elipsoidal, se utiliza el mismo procedimiento manejado en la vecindad circular. Se aplica cualquiera de las funciones de transferencia G_{1-4} descritas en la Tabla 4.1.

Este mecanismo de aplicación de la función de transferencia a la vecindad elipsoidal funciona para la elipse rotada a 45^0 y la rotada a 135^0 , esto incluye implícitamente la tercera modalidad de la vecindad, la vecindad elipsoidal doble, la cual calcula el promedio de los resultados de las modalidades de las elipses rotadas.

Capítulo 5

Paralelización del Sistema

Introducción

El campo de aplicación de las redes neuronales se ha extendido a diferentes áreas de la computación, como reconocimiento de patrones, procesamiento de imágenes, control, etc. El número de neuronas a simular, así como la cantidad de synapses para conectarlas varía dependiendo de la aplicación, algunas requieren una cantidad pequeña, como es el caso de un reconocedor de caracteres alfanuméricos pero otro tipo de aplicaciones requiere una cantidad muy grande, como la simulación de la retina humana ($10^4 \times 10^4$, aproximadamente). En este capítulo se presenta la estrategia de particionamiento por bloques aplicada para la paralelización del simulador, como una necesidad de mejorar su rendimiento.

5.1 Análisis del programa secuencial

La cantidad de neuronas a simular es un factor importante para el rendimiento de un simulador. En la Figura 5.1 se muestra una gráfica de tiempos obtenidos con el simulador secuencial de redes neuronales aplicadas al proceso de visión temprana considerando diversos tamaños de problema (resolución del arreglo de perceptrones).

En esta gráfica se puede apreciar lo costoso que resulta simular un gran número de neuronas. Aprovechando el paralelismo implícito que tienen las redes neuronales, se pueden usar computadoras paralelas como alternativa para hacer una simulación eficiente, es decir, reducir el tiempo de cómputo a valores razonables.

Para encontrar la mejor manera de paralelizar un programa se debe elegir apropiadamente la sección de éste que mayor tiempo consuma en cómputo. El programa del simulador de redes neuro-

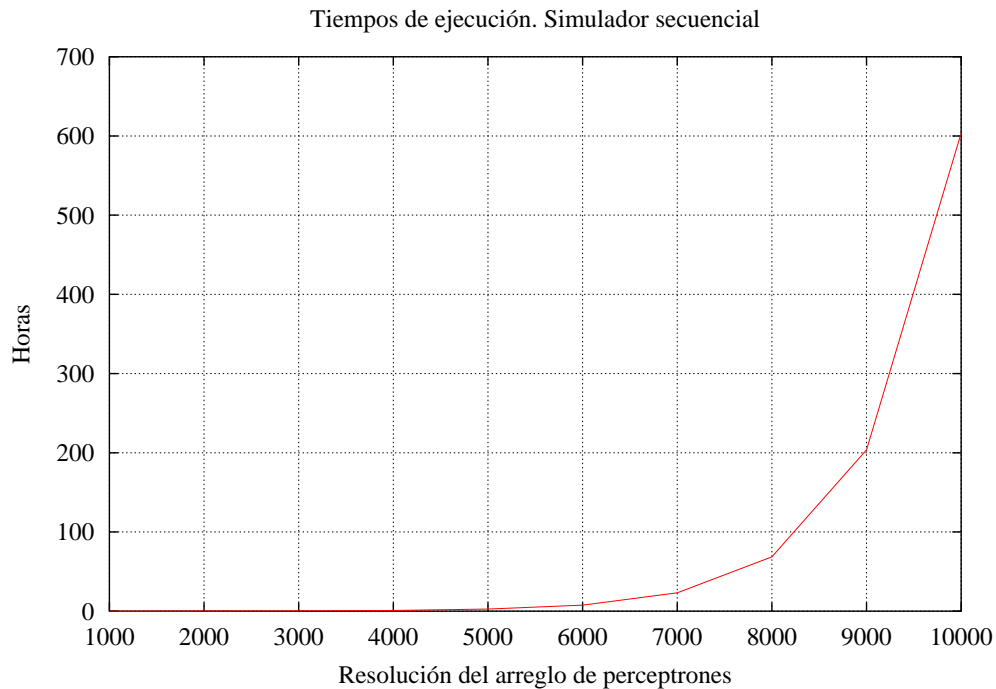


Figura 5.1: Gráfica de tiempos de ejecución del simulador secuencial.

nales aplicado a la visión temprana, aquí presentado, se divide en dos etapas: *recepción de escenas visuales* e *interacción de perceptrones*. Ejecutando el programa secuencial y cuantificando el tiempo de cómputo consumido por cada etapa se determinó que la etapa de **recepción** consume apenas el 1% del tiempo total, mientras que la etapa de **interacción** consume el 99% restante. Estos resultados se pueden observar en la Tabla 5.1, donde se presenta el tiempo total y el consumido por cada etapa para diferentes tamaños de problema. Con estos resultados se decidió que la paralelización era conveniente en la etapa de interacción de perceptrones, pues minimizar el tiempo que consume reduciría significativamente el tiempo total del simulador.

Tamaño del problema	Tiempo en minutos		
	Tiempo de recepción	Tiempo de interacción	Tiempo total
1000	0.097	9.873	9.97
2000	0.675	65.615	66.29
3000	2.116	199.504	201.62
4000	4.494	444.956	449.45

Tabla 5.1: Tiempos de ejecuciones secuenciales de las dos etapas del simulador: recepción de escenas visuales e interacción de los receptores.

5.2 Estrategia de particionamiento

Una de las primeras acciones en el diseño de un algoritmo paralelo es establecer una estrategia de particionamiento del código a paralelizar. Para tener un tiempo razonable al ejecutar el simulador en diferentes procesadores, se debe considerar el tiempo de comunicación entre los procesadores. Una buena estrategia de particionamiento es aquella que minimiza la comunicación y maximiza el cómputo [14].

Para la paralelización de la etapa de interacción de perceptrones, se realizó una partición por bloques de la estructura de almacenamiento de los perceptrones. Esta estrategia consta de dividir en secciones el problema a resolver. En este caso las secciones aplicadas son de forma rectangular y su tamaño depende del número de procesadores (o procesos) que intervengan en el procesamiento. En la Figura 5.2 se ilustra esta estrategia sobre una estructura completamente rectangular y regular. El particionamiento por bloques fue elegido por la forma en que se encuentran almacenados los perceptrones, aunque ésta no sea completamente regular. A pesar de su irregularidad, las perspectivas físicas de las cuatro formas se acercan a una estructura rectangular.

A pesar de las características de las estructuras de almacenamiento de las diferentes formas, la estrategia de particionamiento por bloques funciona sobre las cuatro formas de los arreglos de perceptrones y su aplicación es la siguiente. El particionamiento de la estructura en bloques, consiste en dividir la lista de listas en grupos del mismo número de listas. Una forma simple sería dividir el número de listas entre el número de procesadores, pero, tomando en cuenta que la estructura está compactada, se deben considerar los siguientes puntos: 1) existe una longitud lógica para la estructura de almacenamiento, *olineas*, que corresponde al número de líneas que originalmente se crearon para la estructura; y 2) la longitud real de la estructura, *rlineas*, es diferente de *olineas* y corresponde al número final de líneas utilizadas. Estos dos puntos dan lugar las longitudes *lbloque*, longitud lógica del bloque y *rbloque*, longitud real del bloque. Tomando P , como el número de procesos (o procesadores) que participaran en el procesamiento en paralelo, la longitud lógica del bloque se obtiene mediante la ecuación

$$lbloque = \frac{olineas}{P}.$$

Los índices lógicos inicial, *lini*, y final, *lfin*, de cada bloque lógico, se obtienen mediante las ecuaciones

$$lini = procs \cdot lbloque, \tag{5.1}$$

$$lfin = (procs + 1) \cdot lbloque, \tag{5.2}$$



Figura 5.2: Estrategia de particionamiento para la etapa de interacción de los perceptrones. Particionamiento por Bloques.

```

i = 0;
while( i < rlineas && perceptrones[i][1] < lini)
    i++;
ini = i;
while( i < rlineas && perceptrones[i][1] < lfin)
    i++;
fin = i;

```

Figura 5.3: Código para encontrar los índices físicos inicial y final de cada bloque.

donde *procs*, correspondería al identificador del procesador al que se enviaría el bloque.

Para encontrar los índices físicos inicial, *ini*, y final, *fin*, de cada bloque se hace un recorrido sobre la estructura de almacenamiento. Este recorrido consiste en encontrar las listas que la coordenada lógica *y* de sus perceptrones sea el último índice menor que *lini*, para el índice inicial y el último índice menor que *lfin* para el índice final. La manera de realizar este recorrido se presenta en la Figura 5.3, en donde se muestra el código del programa.

Con *fin* e *ini*, se determinan las secciones de procesamiento sobre las cuales cada procesador aplicará la vecindad seleccionada, pero, como se puede observar por las estrategias de selección de vecinos del capítulo anterior, las vecindades requieren de un área de influencia para determinar los elementos vecinos de cada perceptrón central, independientemente de la vecindad utilizada. Esta área de influencia debe agregarse al bloque determinado para cada procesador. El número de listas que se agregan a cada bloque está determinado por el tamaño de la máscara a aplicar y la posición del bloque en el arreglo de perceptrones.

La posición de cada bloque influye en la determinación del número de listas que se deben agregar de la siguiente manera. Los bloques de los extremos sólo agregarán un grupo de listas: el bloque superior sólo agregará un grupo abajo de él; y el bloque inferior sólo agregará un grupo arriba de él. Los bloques intermedios incluirán dos grupos de listas, uno arriba y otro abajo de ellos. En la Figura 5.4, se presenta un ejemplo de una estructura particionada en cuatro bloques y se muestran

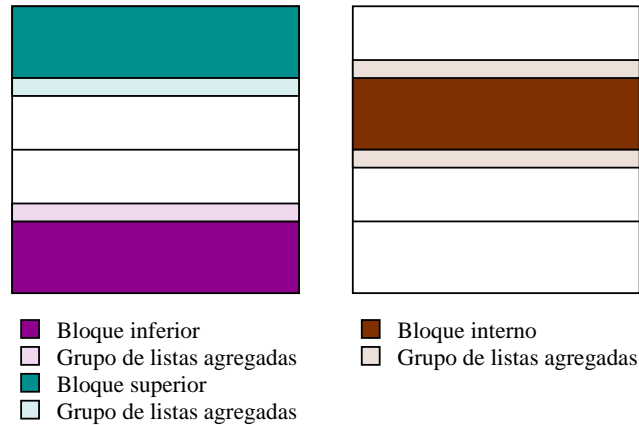


Figura 5.4: Grupos de listas agregadas por la posición de los bloques. A la izquierda, los bloques externos sólo agregan un grupo de listas. A la derecha, los bloques intermedios agregan dos grupos de listas.

los grupos de listas agregados por los bloques internos y externos.

Para determinar el número de listas que se deben agregar a cada bloque se considera el rango de influencia de la máscara a aplicarse: La máscara hexagonal de 7 elementos, tiene un rango de influencia de 1, es decir, un grupo de listas agregadas contendrá solamente una lista. Las máscaras circular y elipsoidal tienen un rango de influencia variable. El rango de influencia de la máscara circular lo determina el radio de la circunferencia que delimita la máscara y el rango de la máscara elipsoidal lo determina el eje mayor de la elipse. Para encontrar el número de listas que se agregarán en cada grupo para estas dos vecindades se realiza un procedimiento análogo al que se utiliza para encontrar los índices reales de los bloques.

5.3 Orquestación

La ejecución de un programa en varios procesadores requiere de la generación de varias tareas o procesos que serán ejecutados en los procesadores designados. La orquestación es la etapa en la que se establece qué proceso se ejecutará en algún procesador específico y la interacción entre ellos. Es decir, en la etapa de orquestación los procesos son mapeados a los procesadores y además, en son sincronizados para regular su comunicación y procesamiento.

En la orquestación, la arquitectura y el modelo de programación y el lenguaje de programación juegan un papel importante. Para ejecutar sus tareas asignadas, los procesos necesitan mecanismos para nombrar (identificar) y acceder los datos, intercambiar datos entre procesos (comunicación) y sincronizar los procesos unos con otros. Algunas de los cuestionamientos que se hacen en la orquestación son: cómo organizar las estructuras de datos, cómo calendarizar las tareas asignadas

a un proceso temporalmente para explotar la localidad de datos, la comunicación debe ser explícita o implícita y en mensajes grandes o pequeños, y cómo exactamente se organizan y expresan la comunicación entre procesos y la sincronización que resulta de la asignación [15].

Las principales metas de la orquestación son reducir el costo de la comunicación y sincronización, la calendarización adecuada de tareas para evitar tiempos latentes, reducir el trabajo adicional para controlar el paralelismo, etc.

A continuación se describen los mecanismos principales que se utilizaron en el desarrollo del simulador **paralelo** de redes neuronales aplicadas al proceso de visión temprana.

5.3.1 Intercambio de mensajes

El método de programación que comúnmente se usa para sistemas de memoria compartida es el intercambio de mensajes. En un intercambio de mensajes básico, los procesos coordinan sus actividades explícitamente enviando y recibiendo mensajes. Para esto se requieren de funciones que permitan el envío y la recepción de mensajes.

Se asume que los procesos están localizados estáticamente, es decir, el número de procesos se establece al inicio de la ejecución, y no se crean procesos adicionales durante la ejecución. A cada proceso se le asigna un único identificador entero *rank*, en el rango $0, 1, \dots, p - 1$, donde p es el número de procesos.

Existen dos formas de comunicación entre procesos, *comunicación bloqueante* y *comunicación no-bloqueante*. En la forma bloqueante, cada tarea que envía información, suspende su ejecución en espera de la respuesta de recepción por parte de la tarea que recibe, de la misma forma, la tarea que recibe, suspende su ejecución hasta recibir la información esperada. En la forma no-bloqueante, una tarea que envía información continúa su ejecución sin detenerse a esperar la confirmación de recepción por parte de la tarea receptora, así también, la tarea que recibe puede continuar ejecutándose mientras llega la información esperada.

Además de estos dos tipos de comunicación 1 a 1 entre procesos, existe un patrón de comunicación que incluye a todos los procesos en una sola instrucción, esta es la **comunicación colectiva**. Como una consecuencia, una comunicación colectiva usualmente incluye más de dos procesos.

Para paralelizar el simulador de redes neuronales, se utilizaron bibliotecas conocidas para programación con intercambio de mensajes, como es MPI (Message Passing Interface). MPI es una definición estándar de bibliotecas de paso de mensajes, desarrollada en 1993 por un grupo de vendedores de computadoras, escritores de software, y desarrolladores de aplicaciones, cuyo objetivo era establecer un estándar de uso amplio para escribir programas basados en el intercambio de mensajes [16].

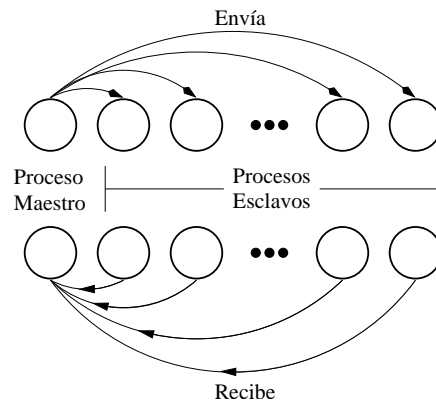


Figura 5.5: Comunicación entre las particiones del arreglo hexagonal de perceptrones.

Las bibliotecas de MPI pueden ser incrustadas en código del lenguaje C, con lo cual permite una gran flexibilidad para desarrollar programas paralelos. La plataforma de trabajo de MPI es un sistema de memoria distribuida, donde todo intercambio de información entre tareas se realiza a través del intercambio de mensajes. MPI implementa las dos formas de comunicación entre tareas y el patrón de comunicación colectiva.

Para el desarrollo de este simulador, se aplicaron la forma de comunicación no-bloqueante y el patrón de comunicación colectiva para agilizar el procesamiento y reducir el tiempo de comunicación al distribuir e integrar la carga de trabajo entre los procesadores.

5.3.2 Modelo maestro-esclavo

Para la distribución del trabajo entre los procesadores se utilizó la estrategia Maestro-Esclavo, ilustrada en la Figura 5.5. Esta estrategia consta de un proceso denominado *maestro*, el cual es responsable de distribuir la información entre los demás procesos, estos últimos denominados esclavos. Debido al uso de la comunicación no-bloqueante, al repartirse la carga de trabajo en los diferentes procesos, todos los procesos, incluyendo al maestro, trabajan en paralelo, es decir, el proceso maestro también conserva una carga de trabajo para él. Al terminar el procesamiento, los procesos esclavos envían sus resultados al proceso maestro, quien los conjunta con los propios y presenta un resultado general.

5.3.3 Sincronización

La base de la sincronización del simulador es el modelo maestro-esclavo. En la creación de los procesos que se utilizarán para el procesamiento, MPI asigna un identificador entero único numérico a cada proceso: *rank*, también llamado “ID del proceso”. *rank* es continuo y comienza en cero. Al

proceso identificado con $rank = 0$ es considerado como el proceso maestro, los demás son esclavos.

La sección secuencial del programa es realizada por el proceso maestro ($rank = 0$), mientras tanto, los demás procesos se encuentran suspendidos en espera de la notificación de activación. Cuando el proceso maestro llega a la sección paralela del programa, despierta a los esclavos a través de un mensaje tipo *broadcast* de datos requeridos por todos los procesos. Un broadcast es una comunicación colectiva en la cual un proceso particular envía el mismo dato a todos los procesos. En la función `MPI_Bcast` de MPI se especifica el dato o un arreglo de ellos que se enviarán, cuantos datos serán enviados, su tipo y el identificados del proceso que envía la información.

Después de recibir este broadcast de activación los esclavos inician los arreglos donde recibirán los datos que les enviará el maestro. La repartición de la carga de trabajo entre todos los procesos se realiza a través de la función `MPI_Scatter`. Esta función permite el envío de bloques completos a cada proceso. La asignación de los bloques es de forma ordenada, es decir, el primer bloque se le envía al proceso 0, el segundo al proceso 1, etc. La función Scatter de MPI también se especifica el arreglo o matriz que se enviará en bloques, la cantidad de datos enviados y su tipo, pero además se debe especificar la localidad que almacenará los datos recibidos, cuántos datos se van a recibir y su tipo. Además de indicar cual es el proceso que envía la información.

Para la transferencia de datos entre dos procesos al final de cada interacción se utiliza la comunicación no-bloqueante `MPI_Isend` que, como se dijo anteriormente, envía la información mas no espera a la confirmación de que el mensaje ha sido recibido por el proceso destino. De la misma forma que las funciones anteriores, en la función `Isend` se especifica el dato o buffer que será enviado, el número de elementos y su tipo, además del identificador del proceso destino, entre otros.

Para la recepción no-bloqueante, MPI utiliza la función `MPI_Irecv`, la cual indica que ya está listo el proceso para recibir la información mas no se detiene a esperarla. En esta función también se especifica el dato o buffer a recibir, la cantidad de los elementos y su tipo, además de indicar cual es el proceso origen.

Para asegurar que los procesos destino esperen los datos que se les envíe durante la comunicación no-bloqueante se utiliza la instrucción `wait` de MPI, la cual no permitirá que el proceso destino continúe hasta que haya recibido el mensaje indicado.

Para la reintegración de la matriz de vértices, los procesos esclavos envían sus bloques modificados al proceso maestro, esta operación se realiza mediante la instrucción `MPI_Gather`, la cual colecta los datos especificados de todos los procesos y los reintegra en el proceso destino. La reintegración de los datos, de la misma manera que la distribución, es de forma ordenada. El primer bloque se toma del proceso 0, el segundo del proceso 1, etc. En la función `Gather` de MPI se hacen las mismas especificaciones que en la función `Scater`, el buffer enviado, el número de elementos y su

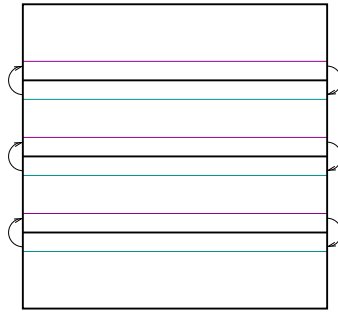


Figura 5.6: Comunicación entre las particiones del arreglo hexagonal de perceptrones.

tipo, el buffer o matriz de recepción, el número de elementos y su tipo, además de la especificación del proceso destino.

5.3.4 Comunicación

La simulación de la visión temprana es un proceso iterativo, requiere de 10 etapas de procesamiento de los perceptrones. Esto trae consigo un problema importante: se incrementa la comunicación. Al final de cada etapa, todo bloque debe transmitir la sección que sus vecinos requieran para actualizar los valores de los perceptrones que corresponden a su área de influencia. La sección que los bloques enviarán a sus vecinos se determina de la misma forma que los grupos de listas agregadas.

La cantidad de secciones que cada bloque transmitirá se determina también por la posición del bloque, es decir, depende del número de bloques contiguos que este tenga. En la Figura 5.6 se muestra un ejemplo de este tipo de partición con 4 procesadores y la comunicación realizada por los bloques al terminar cada etapa.

Las secciones que cada proceso intercambia son denominadas sub-bloques. Los sub-bloques de un proceso en particular, se definen como *sub-bloque superior* y *sub-bloque inferior*. El sub-bloque superior es aquél que será enviado a un proceso con identificador menor al del proceso origen, mientras que el sub-bloque inferior se envía a un proceso con identificador mayor.

5.4 Mapeo

Las bibliotecas de MPI proporcionan independencia a la cantidad de procesadores, es decir, el número de tareas o procesos a crear no depende del número de procesadores disponibles. Dada esta característica, el número de procesos posibles para la ejecución de este simulador es arbitraria.

A pesar de la independencia que MPI proporciona para la generación de procesos, es conveniente considerar la plataforma disponible para la ejecución en paralelo de los programas. La plataforma

con la que se cuenta para este simulador son dos computadoras de multiprocesamiento: Elvis y Presley, del laboratorio de paralelismo:

- Computadora de multiprocesamiento Elvis:
 - con 4 procesadores Intel Xeon de 750 MHz
 - Disco Duro SCSI de 28 GB
 - 1 GB de Memoria RAM
- Computadoras de multiprocesamiento Presley:
 - con 4 procesadores Intel Xeon de 550 MHz
 - Disco Duro SCSI de 28 GB
 - 1 GB de Memoria RAM
- Sistema operativo Linux Red Hat Versión 7.1

Con esta plataforma se cuenta con 8 procesadores para la ejecución del simulador paralelo. Dada esta limitante, la ejecución en paralelo de este simulador se propone con un máximo de 8 procesos, de tal manera que se genere un proceso por cada procesador.

El mapeo de los procesos a los procesadores se deja al criterio del sistema operativo, no es importante ejecutar un proceso en un procesador particular, si la ejecución del programa es con 8 procesos, a cada procesador se le asignará 1 proceso.

5.5 Algoritmo paralelo general

Un programa paralelo en MPI que utilice el modelo maestro-esclavo consta de un solo programa en el cual, a través del identificador de cada proceso, *rank*, se determina qué parte del código ejecutará el proceso maestro ($rank == 0$) y los procesos esclavos ($rank! = 0$).

En el Algoritmo 1 se presenta el pseudocódigo de la sección paralela del programa. Este pseudocódigo consta de 4 operaciones generales: en el *Broadcast* se envían datos generales desde el maestro a todos los esclavos para activar a los procesos que están en espera. Con la operación *Scatter* se distribuyen los datos a procesar entre los procesos participantes a partir del proceso maestro. La función `calculaBloques()` es donde se aplican los modelos de interacción sobre los arreglos de perceptrones. Con la operación *Gather* se recolectan los datos procesados por los procesos para su reintegración en el proceso maestro.

Algorithm 1 Función `simulaInteraccion()`

Require: *vecindad*, *** perceptrones***Ensure:** *imagenArtificial*

```

Broadcast      \* Activación de procesos esclavos */
Scatter        \* Distribución de datos */
calculaBloques() \* Procesamiento independiente de los procesos */
Gather         \* Recolección de datos */

```

En el Algoritmo 2 se describe la función `calculaBloques()`, la cual contiene un ciclo condicionado a 10 iteraciones y al umbral de estabilidad: *error*. Dentro de este ciclo se aplica la modalidad de la vecindad especificada, se cuenta el número de iteraciones realizadas y se calcula el error entre la iteración actual y la anterior. Si se tiene más de un proceso y no se ha llegado al máximo de iteraciones ni a la estabilidad de la red se aplica la función `comunicaSubBloques` para actualizar los datos que utilizará la vecindad en la interacción siguiente y que dichos datos fueron procesados en otro proceso.

Algorithm 2 Función `calculaBloques()`

Require: *ini*, *fin*, *vecindad***Ensure:** *datosProcesados*

```

while iteracion < 10  $\wedge$  error > 0.00001 do
  switch( vecindad ) \* Vecindad a aplicar */
    case 0: simulaVH( modalidad ) \* Aplica vecindad hexagonal */
    case 1: simulaVC( modalidad ) \* Aplica vecindad circular */
    case 2: simulaVE( modalidad ) \* Aplica vecindad elipsoidal */
  iteracion ++
  calculaError(&error)
  if numtasks > 1  $\wedge$  iteracion + 1 < 10  $\wedge$  error > 0.00001 then
    comunicaSubBloques()
  end if
end while

```

En el Algoritmo 3 se describe el intercambio de datos entre los procesos al concluir una interacción. Para comunicar un sub-bloque, cada proceso debe identificarlo para después enviarlo al proceso destino. Además, deberá esperar el sub-bloque que le enviará otro proceso, al recibirlo deberá actualizar sus datos. La comunicación se realiza de forma lineal, es decir, cada proceso se comunica con su vecino inferior y superior, con excepción de los extremos. Los procesos 0 y $p - 1$ sólo tienen un vecino, con el cual intercambiarán sus mensajes.

Algorithm 3 Función comunicaSubBloques()

Require: l_{ini}, l_{fin}

if $rank = 0$ **then**

 identificaSubBloque(*inferior*) * Bloque a enviar al proceso $rank + 1$ */
 enviaSubBloque($rank + 1$) * Envía el sub-bloque al proceso 1 */
 recibeSubBloque($rank + 1$) * Espera el sub-bloque del proceso 1 */
 actualizaSubBloque(*inferior*)

end if

if $rank > 0 \wedge rank + 1 < numtasks$ **then**

 recibeSubBloque($rank - 1$) * Espera el sub-bloque del proceso $rank - 1$ */
 actualizaSubBloque(*superior*)
 identificaSubBloque(*superior*) * Bloque a enviar al proceso $rank - 1$ */
 enviaSubBloque($rank - 1$) * Envía el sub-bloque al proceso $rank - 1$ */
 recibeSubBloque($rank + 1$) * Espera el sub-bloque del proceso $rank + 1$ */
 identificaSubBloque(*inferior*) * Bloque a enviar al proceso $rank + 1$ */
 enviaSubBloque($rank + 1$) * Envía el sub-bloque al proceso $rank + 1$ */
 actualizaSubBloque(*inferior*)

end if

if $rank + 1 = numtasks$ **then**

 recibeSubBloque($rank - 1$) * Espera el sub-bloque del proceso $rank - 1$ */
 identificaSubBloque(*superior*) * Bloque a enviar al proceso $rank - 1$ */
 enviaSubBloque($rank - 1$) * Envía el sub-bloque al proceso $rank - 1$ */
 actualizaSubBloque(*superior*)

end if

Capítulo 6

Análisis y Evaluación de Resultados

Introducción

El simulador de redes neuronales aplicadas al proceso de visión temprana, fue creado para examinar el modelo propuesto para explicar la generación de la ilusión caracterizada por la aparición de manchas en el área delimitada por cuatro cuadros negros. En este capítulo se presentan los resultados de las pruebas del simulador de redes neuronales aplicadas al proceso de visión temprana. Las pruebas comprenden la aplicación de la generación de arreglos hexagonales de receptores, la proyección de imágenes sobre arreglos hexagonales, la aplicación de los tres modelos de interacción de perceptrones y la aplicación de la estrategia de particionamiento por bloques para la ejecución del simulador en paralelo sobre varios procesadores.

6.1 Patrones de prueba

Para realizar las pruebas al modelo propuesto de redes neuronales se establecieron algunos parámetros sobre las características de la imagen de entrada, sobre los cuales se basa toda imagen de entrada utilizada en el sistema. Este formato se puede observar en la Figura 6.1, en el cual los parámetros l y d se utilizan proporcionalmente para cualquier resolución de la imagen de entrada a utilizar. d , es la distancia entre dos cuadros negros; r , la resolución de la imagen de entrada; y l , la longitud de los lados de los cuadros negros. La base de estas medidas es la resolución de la imagen: r , a partir de la cual se generan d y l . La relación que existe entre l y d es $l = 2d$. Ya que el patrón de prueba siempre es de 4×4 cuadros negros, la relación que existe entre las medidas de las imágenes

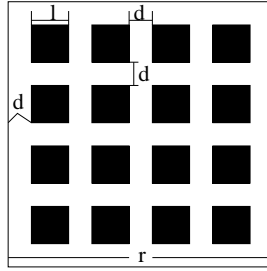


Figura 6.1: Formato de la imagen de entrada utilizada para las pruebas del modelo propuesto.

de entrada se presenta en las siguientes ecuaciones,

$$\begin{aligned}
 5d + 4l &= r \\
 l &= 2d \\
 5d + 8d &= r \\
 d &= \frac{r}{13},
 \end{aligned} \tag{6.1}$$

Los valores que r , d y l representan, son los números de píxeles de las distancias. Estas medidas se ven incrementadas o decrementadas cuando se proyecta la imagen de entrada al arreglo de perceptrones. La proporción de decremento o incremento se calcula mediante la ecuación

$$P = \frac{p \cdot q}{s \cdot t}, \tag{6.2}$$

donde, p y q , corresponden al número de líneas y columnas de la matriz que contiene la imagen de entrada; s y t , corresponden a las dimensiones de la estructura de datos que almacena los perceptrones.

Al proyectar los puntos de la imagen de entrada sobre el arreglo de perceptrones de resolución r' , se obtienen d' y l' , equivalentes a las dimensiones de la imagen de entrada proyectadas sobre un arreglo de perceptrones, estas dimensiones se obtienen incrementadas o decrementadas por P , de la siguiente manera:

$$d' = d \cdot P, \quad l' = l \cdot P, \tag{6.3}$$

Estas medidas y operaciones son aplicadas para la imagen de entrada y su proyección, pero estos parámetros también se utilizan para establecer una relación entre los parámetros aplicados a las funciones de transferencia y las diversas vecindades. Los parámetros usados son: $radioC$, radio de la vecindad circular; a , eje mayor de la vecindad elipsoidal; b , eje menor de la vecindad elipsoidal; σ , amplitud de la función de transferencia. Tales parámetros se basan en las medidas de la imagen de

r	$d = \frac{r}{13}$	$l = 2 * d$	r'	$d' = d * P$	$l' = l * P$
200×200	15	30	2000×2000	150	300
1000×1000	76	152	2000×2000	760	1520
2000×2000	153	306	2000×2000	1530	3060
4000×4000	307	614	2000×2000	3070	6140

Tabla 6.1: Tabla de conversiones de las medidas utilizadas en las imágenes de entrada y las vecindades de los modelos de agrupación.

r	radio	$radioC$	eje mayor	a	eje menor	b	amplitud	σ
200×200	30 pixels	$1.95d$	100 pixels	$6.5d$	15 pixels	$0.97d$	100 pixels	$6.5d$
1000×1000	30 pixels	$0.39d$	200 pixels	$2.6d$	10 pixels	$0.13d$	66 pixels	$0.86d$
2000×2000	50 pixels	$0.32d$	150 pixels	$0.97d$	5 pixels	$0.03d$	75 pixels	$0.48d$
4000×4000	100 pixels	$0.32d$	100 pixels	$0.32d$	15 pixels	$0.04d$	100 pixels	$0.32d$

Tabla 6.2: Tabla de conversiones de las medidas utilizadas en los elementos de las las vecindades de los modelos de agrupación.

entrada. Al especificar que la vecindad circular tiene un radio de e pixels, se indica que $radioC = \frac{e}{d}$. Análogamente se puede indicar que el eje mayor de la vecindad elipsoidal mide $a = 5d$. En la Tabla 6.1 se presentan las resoluciones más usadas para las imágenes de entrada, sus medidas y su efecto sobre el arreglo de perceptrones. En la Tabla 6.2 se presentan los valores y su representación de los elementos principales de las vecindades de los modelos de interacción de perceptrones.

6.2 Resultados sobre el tipo de vecindad

Los resultados de las ejecuciones del simulador al aplicar los diferentes modelos, se presentan de dos formas: la primera, en forma de imagen, para apreciar directamente el efecto producido por la aplicación de las vecindades sobre el arreglo hexagonal de perceptrones; y la segunda, en forma de gráfica de superficie, para eliminar la posible ambigüedad entre las ilusiones creadas por los ojos al observar las imágenes resultantes y las ilusiones generadas por el simulador. Por cuestiones de espacio y para evitar confusión, los resultados presentados en este capítulo son los mejores de todas las pruebas realizadas¹. Para facilitar la comprensión de los resultados de las pruebas realizadas con los diferentes elementos del simulador, presentados en este capítulo, en la Tabla 6.3 se presenta un recordatorio de lo que cada elemento del simulador significa.

¹En el disco compacto que acompaña a esta tesis se incluyen los resultados de todas las pruebas realizadas

Elementos	Descripción
Forma 1 de los arreglos	Arreglo regular de perceptrones con radios simétricos y puntos regulares.
Forma 2 de los arreglos	Arreglo regular de perceptrones con radio simétricos y puntos irregulares.
Forma 3 de los arreglos	Arreglo concéntrico de perceptrones con radios y puntos regulares.
Forma 4 de los arreglos	Arreglo concéntrico de perceptrones con radios perturbados y puntos regulares.
Vecindad hexagonal	Filtro de 7 elementos: 1 cenral y 6 vecinos generados de siguiendo la vecindad hexagonal.
Modalidades de la vecindad hexagonal	Modalidad a) promediación de los elementos, modalidades b) y c) detectores de bordes verticales, d) y e) detectores de bordes horizontales, e) y f) función Exponencial y su inversa.
Vecindad circular	Filtro de un variado número de elementos concéntricos a un elemento central, delimitados por una circunferencia.
Estrategias de selección de vecinos	Vecinos completos: se consideran vecinos los contenidos completamente en la circunferencia de la vecindad; vecinos parciales: se consideran vecinos aquellos contenidos el 50% o más de su área en la circunferencia de la vecindad; vecinos proporcionales: se toma una influencia proporcional al porcentaje de área incluida en la circunferencia.
Vecindad Elipsoidal	Filtro de un variado número de elementos concéntricos a un elemento central, delimitados por una elipse.
Modalidades de la vecindad elipsoidal	Las modalidades están definidas por el tipo de elipse usado: una elipse rotada a 45^0 , una elipse rotada a 135^0 y el uso de una elipse doble: unión de las dos elipses anteriores.
Funciones de transferencia	Se tienen cuatro funciones de transferencia: Exponencial, e^{-x^2} ; Sinc, $\frac{\sin(x)}{x}$; Bessel j_0 y Bessel j_1 .

Tabla 6.3: Elementos de prueba del simulador.

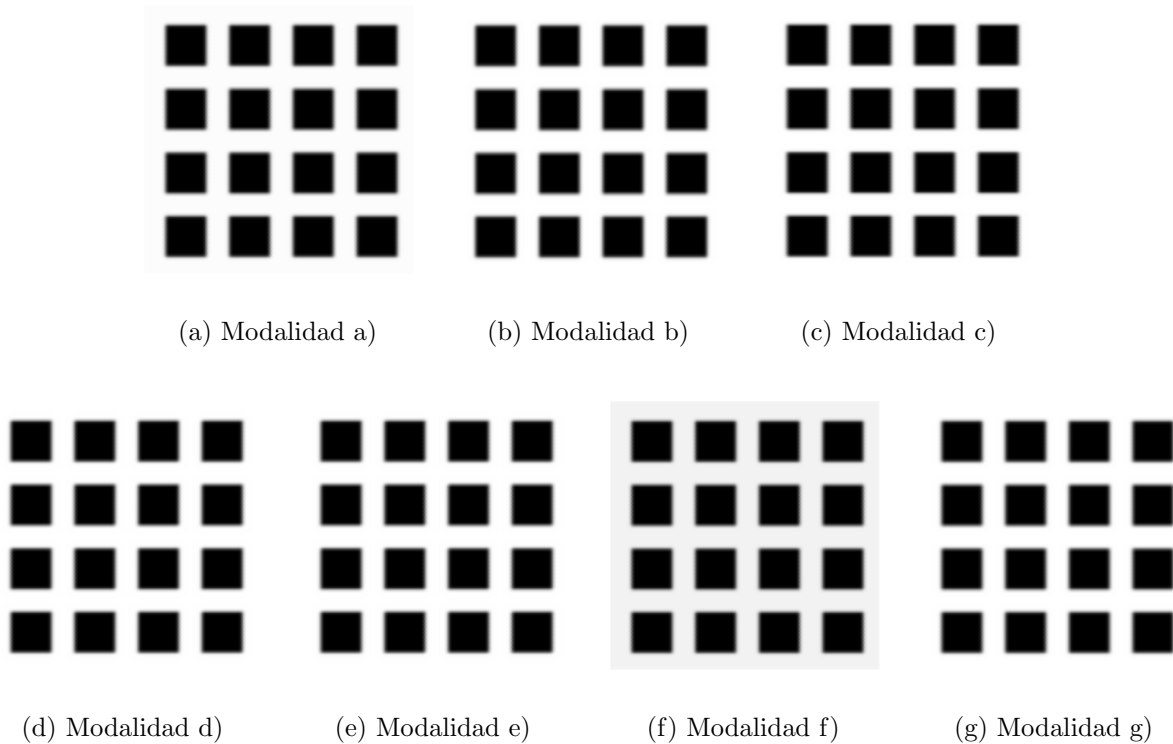


Figura 6.2: Imágenes de salida al convolucionar las 7 modalidades de la **vecindad hexagonal de 7 elementos** con la **forma 1** de los arreglos.

6.2.1 Vecindad hexagonal

Las primeras pruebas se realizaron con el modelo **vecindad hexagonal de 7 elementos** sobre la forma 1 de los arreglos hexagonales. Para estas pruebas, la resolución de la imagen de entrada fue de 200×200 . La resolución de los arreglos utilizados fue de 2000×2000 . Estas resoluciones fueron convenientes por cuestiones del tiempo de ejecución de todos los experimentos realizados. La imagen artificial producida por el simulador se determinó con una resolución de 200×200 , esto debido a que con diferentes pruebas se encontró que para obtener una imagen resultante con buena definición, la resolución de la imagen artificial debía ser 10 veces menor que la resolución del arreglo de perceptrones.

La Figura 6.2 corresponde a las imágenes de salida de la aplicación de la vecindad hexagonal de 7 elementos en sus 7 modalidades (ver Figura 4.6) sobre los arreglos de Forma 1 (ver Figura 3.1). En la Figura 6.2.a), se observa un suavizado de los bordes de los cuadros negros; en las Figuras 6.2.b) y 6.2.c), donde se aplican los detectores de bordes horizontales, se puede observar que los bordes horizontales son resaltados mientras que los verticales son ligeramente suavizados. En las Figuras 6.2.d) y 6.2.e), donde se aplican los detectores de bordes diagonales, se puede observar el

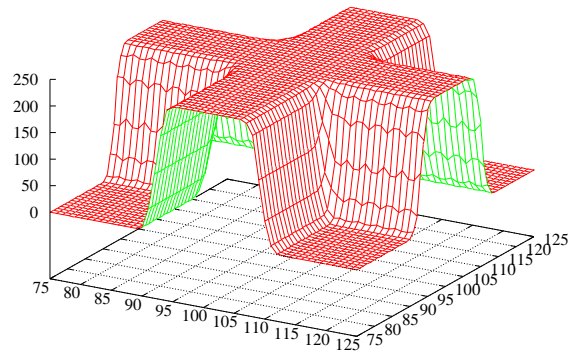


Figura 6.3: Gráfica resultante al aplicar la **modalidad f)** de la **vecindad hexagonal de 7 elementos** sobre la **forma 1** de los arreglos.

acentuado de los bordes verticales y la degradación de los horizontales. En las Figuras 6.2.f) y 6.2.g), donde se aplican las funciones exponencial y su inversa, se puede apreciar en la primera un obscurecimiento general de la imagen y en la segunda, una degradación ligera de la imagen. Al final de estas imágenes de salida se presenta en la Figura 6.3, la gráfica que corresponde a la modalidad f) de la vecindad hexagonal, que se considera como el mejor resultado de las 7 modalidades.

Claramente, ninguna de las modalidades de esta vecindad logra obtener resultados cercanos al efecto deseado; esto se debe a que la región de interacción es demasiado pequeña. Sin embargo, mediante este tipo de vecindad es posible simular muchos de los filtros usados en procesamiento digital de imágenes pero sobre arreglos hexagonales.

6.2.2 Vecindad circular

Para las pruebas de la **vecindad circular** se utilizaron las mismas resoluciones para la imagen de entrada y el arreglo de perceptrones que en la vecindad hexagonal. El radio para estas pruebas fue de 30 pixels, es decir, $1.95d$. Las imágenes de la Figura 6.4 corresponden a las imágenes de salida de aplicar la vecindad circular con sus 3 estrategias para seleccionar perceptrones vecinos sobre arreglos de la forma 4 (ver Figura 3.1). En estas imágenes se puede observar que no existe una diferencia visible entre ellas, al aplicar cualquiera de las tres estrategias de selección de vecinos el resultado visualmente aparece muy similar. Por tal motivo, a partir de estas pruebas se define la estrategia de selección de vecinos **vecinos completos**, como la estrategia a utilizar en el resto de las pruebas del sistema, dado que es la que consume menor tiempo ofreciendo el mismo efecto.

La Figura 6.5, muestra la gráfica correspondiente a la aplicación de la vecindad circular con la

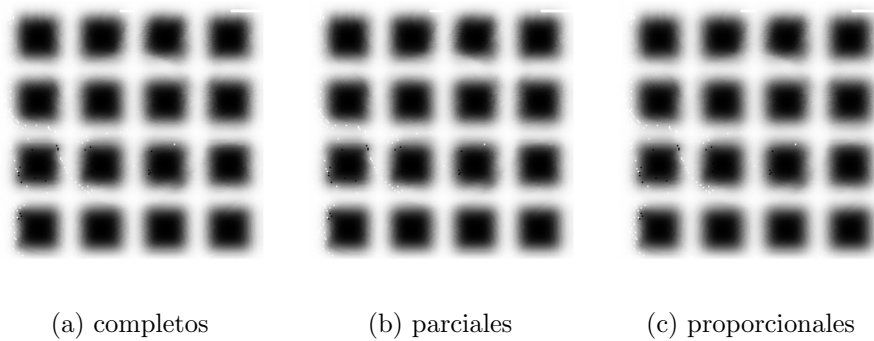


Figura 6.4: Imágenes de salida al aplicar la **vecindad circular** sobre la **forma 4** de los arreglos hexagonales de perceptrones y aplicando la función exponencial.

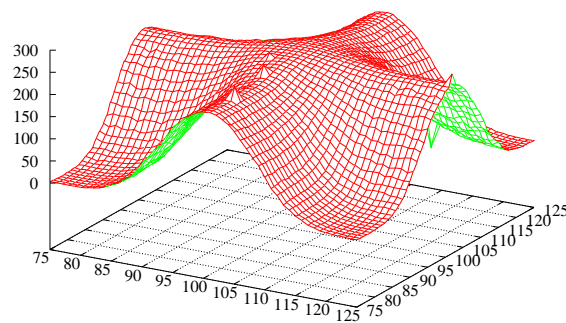


Figura 6.5: Gráfica resultante al aplicar la **vecindad circular** con la estrategia de selección de vecinos proporcionales sobre la **forma 4** de los arreglos y aplicando la función exponencial.

estrategia de selección de **vecinos completos**, para clarificar el efecto obtenido con esta vecindad.

6.2.3 Vecindad elipsoidal

En las pruebas de la **vecindad elipsoidal** se emplearon las mismas resoluciones de imagen de entrada y arreglo de perceptrones que en la vecindad circular. Los resultados de estas pruebas fueron obtenidos con la estrategia de selección de vecinos: **vecinos completos**. El eje mayor de la elipse aplicada fue de 100 pixels ó $6.5d$ y el eje menor de 10 pixels ó $0.65d$. La amplitud de la función de transferencia está determinada por σ (ver Tabla 4.1), la cual corresponde al radio de la elipse. Esta amplitud se utiliza para especificar el área de mayor influencia excitatoria de la función de transferencia, con $\sigma = 100$ pixels o $6.5d$ se especifica que el área de influencia mayor abarca toda el área de la elipse. Las imágenes de la Figura 6.6, corresponden a las imágenes de salida de aplicar

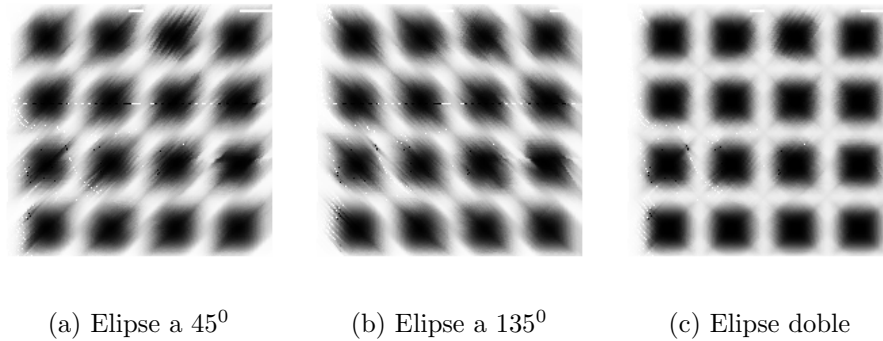


Figura 6.6: Imágenes de salida al aplicar la **vecindad elipsoidal** con la **Forma 4** de los arreglos, utilizando la función **exponencial** y utilizando la estrategia de selección **vecinos completos**.

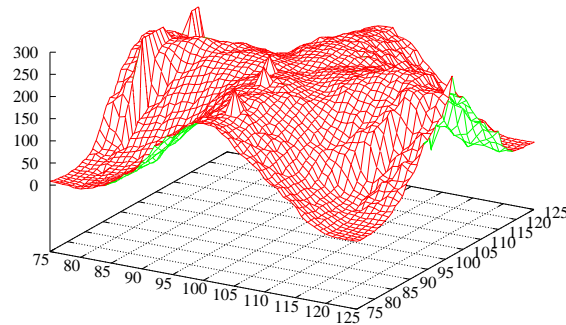


Figura 6.7: Gráfica resultante al promediar los resultados de aplicar las elipses rotadas a 45° y 135° utilizando la función **exponencial** y la estrategia de selección **vecinos completos**.

la vecindad elipsoidal con la función de transferencia exponencial (ver Tabla 4.1) sobre los arreglos de forma 4 (ver Figura 3.1).

Para obtener la imagen de la Figura 6.6.a se utilizó la elipse rotada a 45° , por lo cual, se pueden notar en la imagen las manchas generadas de la esquina inferior izquierda a la esquina superior derecha; para la Figura 6.6.b se utilizó la elipse rotada a 135° , produciendo una imagen con manchas de la esquina inferior derecha a la esquina superior izquierda; y la imagen de la Figura 6.6.c corresponde al promedio de las elipses rotadas a 45 y 135 grados, dando el efecto de una imagen con manchas en forma de cruces entre las áreas delimitadas por cuatro cuadros negros. La Figura 6.7, muestra la gráfica correspondiente al resultado de aplicar la vecindad elipsoidal doble.

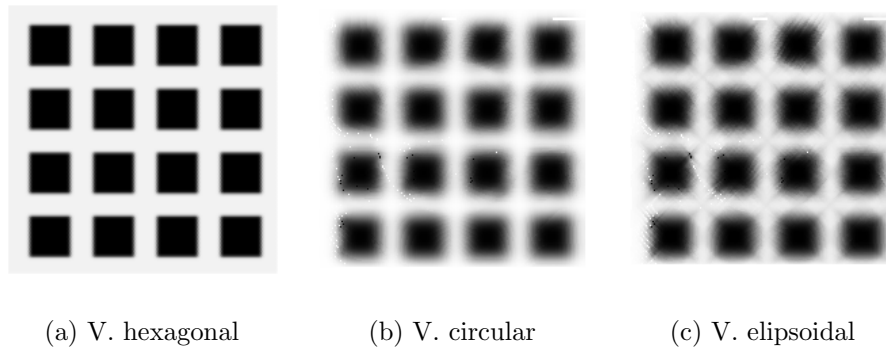


Figura 6.8: Comparación de las imágenes de salida de aplicar las vecindades hexagonal, circular y elipsoidal.

6.3 Comparación de los tres tipos de vecindades

Teniendo las cuatro imágenes resultantes de las vecindades hexagonal, circular y elipsoidal, se hizo una comparación cualitativa de los mejores resultados. Para la generación de estas imágenes se utilizó la función exponencial como función de transferencia.

Esta comparación se presenta en la Figura 6.8, en la cual se pueden observar las imágenes resultantes de aplicar: la modalidad f de la vecindad hexagonal, Figura 6.8.a; la vecindad circular con la estrategia de selección de vecinos proporcionales, Figura 6.8.b; y la vecindad elipsoidal doble, Figura 6.8.c.

Al observar las imágenes de la Figura 6.8, fácilmente se puede concluir que la vecindad que produce mejores resultados es la vecindad elipsoidal, la cual genera manchas alargadas en las áreas delimitadas por cuatro cuadros negros. Con esta comparación se selecciona la vecindad elipsoidal como la vecindad a utilizar para las pruebas restantes de los diferentes elementos del simulador.

6.4 Pruebas de los diferentes elementos del simulador

Se aplicó la vecindad elipsoidal para probar los efectos producidos por otros elementos del simulador, por ejemplo: la resolución de la imagen de entrada, las funciones de transferencia, las formas de los arreglos de perceptrones, las estrategias de selección de vecinos y, finalmente, diferentes valores para los elementos de las elipses de la vecindad elipsoidal. Todo esto, con el fin de encontrar la mejor combinación de elementos para producir el efecto más cercano a la ilusión real formada en el ojo humano.

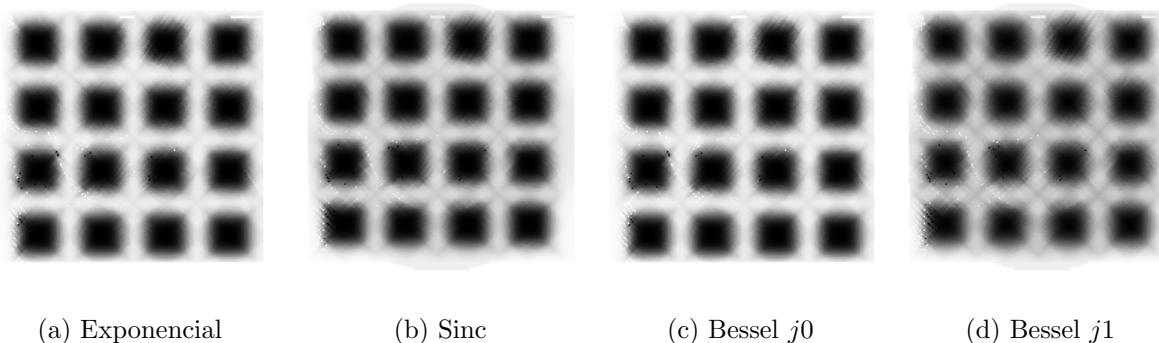


Figura 6.9: Imágenes de salida al aplicar la **vecindad elipsoidal** sobre la **forma 4** de los arreglos, aplicando la estrategia de selección **vecinos completos** y utilizando a) la función exponencial; b) la función Sinc; c) la función Bessel j_0 ; y d) la función Bessel j_1 .

6.4.1 Aplicación de las cuatro funciones de transferencia

El primer elemento a evaluar es la **función de transferencia**, con lo cual se pretende observar qué función, de las cuatro funciones de transferencia con las que cuenta el simulador (mostradas en la Tabla 4.1), presenta un mejor desempeño con relación al fenómeno deseado. Para estas pruebas se utilizó la misma imagen de entrada con resolución de 200×200 pixels y la resolución para el arreglo de perceptrones fue de 2000×2000 elementos.

En la Figura 6.9 se presentan las imágenes resultantes al aplicar la vecindad elipsoidal doble sobre la forma 4 de los arreglos, la estrategia de selección de vecinos completos y las cuatro funciones de transferencia. De la aplicación de la función de transferencia (Figura 6.9.a) se puede apreciar que el fenómeno de inhibición lateral no es simulado. Los valores de esta función son todos positivos. Esto es, asigna influencia excitatoria a todos los perceptrones, a unos mayor y a otros menor, pero a ninguno influencia inhibitoria. Este efecto se observa en la degradación de los bordes de los cuadros negros, puesto que no hay influencia inhibitoria, los bordes entre negro y blanco sólo se promedian, no se cancelan. En la Figura 6.9.b se presenta la imagen obtenida con la función Sinc, esta función simula a la inhibición lateral asignando valores de influencia positivos y negativos, es decir, influencia excitatoria e inhibitoria. Como puede verse en la imagen, los cuadros tienden a conservar su forma y las manchas generadas están mejor definidas. Sin embargo, la contribución inhibitoria no es lo suficientemente grande como para delinear adecuadamente las manchas, sin degradar los bordes de los cuadros. La Figura 6.9.c muestra la imagen resultante de aplicar la función de transferencia Bessel j_0 , la cual tiene un rango de asignación de valores positivos y negativos mayor que la función Sinc, por lo cual se genera una imagen de salida con cuadros negros y manchas mejor definidos. En la Figura 6.9.d se presenta la imagen resultante de aplicar la función Bessel j_1 , con esta función la

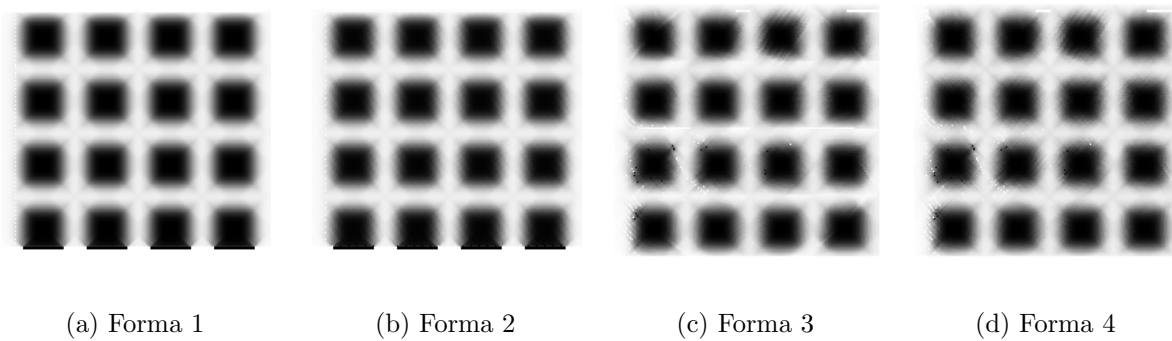


Figura 6.10: Imágenes de salida al aplicar la **vecindad elipsoidal** sobre las **cuatro formas** de los arreglos, aplicando la estrategia de selección **vecinos completos** y utilizando la función Bessel j_0 .

contribución inhibitoria es aun mayor que el caso de la función Bessel j_0 . El efecto resultante es que las manchas se acentúan pero se degradan los bordes de los cuadrados.

A partir de este análisis de las imágenes resultantes al aplicar las cuatro funciones de transferencia propuestas en el sistema, se selecciona la función **Bessel** j_0 como la función de transferencia a utilizar en el resto de las pruebas.

6.4.2 Evaluación de las formas de los arreglos de perceptrones

Para las pruebas de la vecindad hexagonal se utilizó la forma 1 de los arreglos dado que por su estructura de almacenamiento es la más apropiada para aplicar esta vecindad. Para las pruebas de las vecindades circular y elipsoidal se utilizó la forma 4, considerando la búsqueda de elementos contenidos en una forma geométrica regular estando almacenados en una estructura irregular.

Habiendo seleccionado la vecindad elipsoidal y la función de transferencia Bessel j_0 , se realizaron pruebas con las cuatro formas de los arreglos de perceptrones y se evaluaron sus resultados. En la Figura 6.10 se presentan las imágenes resultantes de aplicar la vecindad elipsoidal con la función Bessel j_0 sobre las cuatro formas de los arreglos de perceptrones.

En las Figuras 6.10.a y 6.10.b se presentan las imágenes de salida obtenidas con las formas 1 y 2 de los arreglos, respectivamente; en éstas se puede apreciar una definición pobre de las manchas generadas. En la Figura 6.10.c se presenta la imagen que se obtuvo al utilizar la forma 3, en ésta se tiene una mejor definición de las manchas pero, aún es mejor el efecto mostrado en la Figura 6.10.d, correspondiente a la imagen de salida al aplicar la forma 4 de los arreglos.

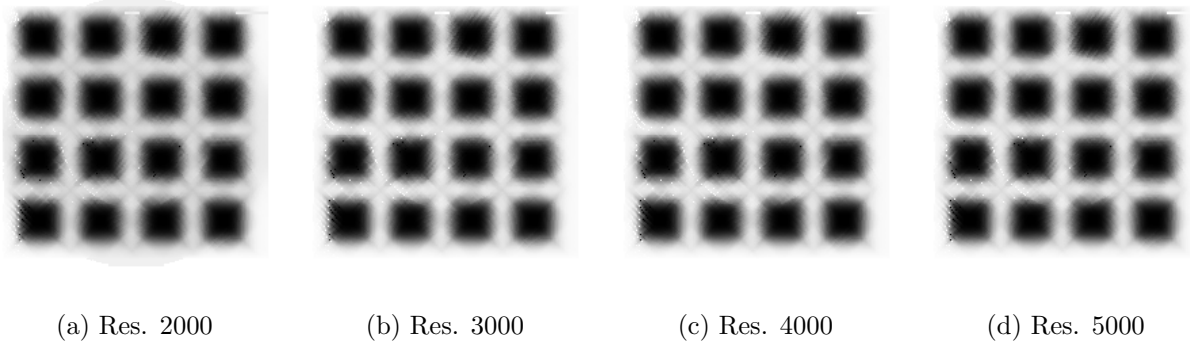


Figura 6.11: Imágenes de salida al aplicar la **vecindad elipsoidal doble**, utilizando la función **Bessel j_0** , la estrategia **vecinos completos**, y resoluciones de la imagen de entrada igual y mayores a la resolución de arreglo de perceptrones.

Resolución	a	b	σ
2000×2000	$0.65d$	$0.06d$	$0.65d$
3000×3000	$0.43d$	$0.04d$	$0.43d$
4000×4000	$0.32d$	$0.03d$	$0.32d$
5000×5000	$0.26d$	$0.02d$	$0.26d$

Tabla 6.4: Equivalencias de los valores de los elementos de las elipses (a , b) y la amplitud de la función (σ) para probar el efecto de la resolución.

6.4.3 Efecto de la resolución en el proceso de simulación

Hasta ahora, la resolución utilizada para las imágenes de entrada ha sido 200×200 pixels pero, con esta resolución se está suponiendo que las escenas visuales observadas por el ojo son de resolución menor al arreglo de receptores de la retina humana, lo cual no es cierto en la realidad. Por tanto se realizaron experimentos con diferentes resoluciones para la imagen de entrada, tomando en cuenta que la resolución de los arreglos de perceptrones que se ha definido para las pruebas del sistema es 2000×2000 pixels.

En las imágenes de la Figura 6.11 se presentan las imágenes de salida obtenidas utilizando diferentes resoluciones para la imagen de entrada. Los elementos de la elipse y de la función de transferencia en pixels son: eje mayor de la elipse de 100 pixels, eje menor 10 pixels, amplitud de la función de transferencia $\sigma = 100$ pixels. Pero, estos valores no son equivalentes para diferentes resoluciones, conforme la resolución crece, el área que abarca la elipse y la función de transferencia se reduce con respecto a d , la unidad básica de medida de estos experimentos. En la Tabla 6.4 se presentan las equivalencias de los valores de los elementos de las elipses y la amplitud de las

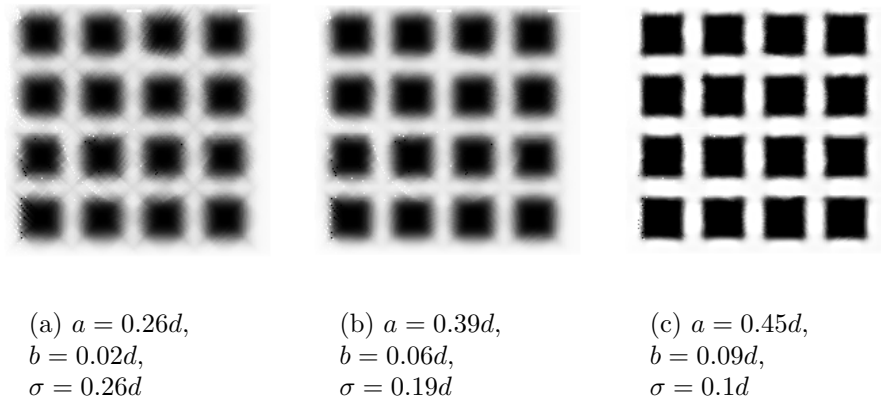


Figura 6.12: Imágenes de salida al aplicar la **vecindad elipsoidal doble** sobre la **forma 4** de los arreglos, utilizando la función **Bessel j_0** , la estrategia de selección **vecinos completos** y tres combinaciones diferentes para la amplitud de la función y los ejes mayor y menor de la elipse.

funciones de transferencia utilizadas.

En las cuatro imágenes de la Figura 6.11 puede observarse el fenómeno deseado: la aparición de manchas en el área delimitada por cuatro cuadros negros, pero, conforme se va incrementando la resolución de la imagen de entrada se obtienen imágenes de salida con mejor definición, es decir, menor distorsión de los cuadros negros y mejor nitidez de las manchas.

El hecho de usar imágenes de entrada con una resolución mayor al arreglo de perceptrones simula de alguna manera el fenómeno integrador que se realiza en las células receptoras de la retina. De ahí que la calidad de la imagen que se forma se ve mejorada.

6.4.4 Variación de los valores de los elementos de las elipses

En las pruebas anteriores, a pesar de las manchas persiste el problema de la degradación de bordes y oscurecimiento de las regiones claras de la imagen. Para mejorar la calidad de la imagen resultante se propuso experimentar con diferentes valores de los elementos estructurales de las elipses y amplitudes diferentes de las funciones.

En la Figura 6.12 se presentan las imágenes resultantes al aplicar la vecindad elipsoidal doble, con la función de transferencia Bessel j_0 e imágenes de entrada con resoluciones mayores a los arreglos de perceptrones. Además, se varía la elongación de las elipses variando los ejes mayor y menor de las mismas.

Estas pruebas se realizaron sólo con la función Bessel j_0 , la cual, en las pruebas anteriores produjo mejores resultados. La estrategia de selección de vecinos aplicada es la de vecinos completos sobre la forma 4 de los arreglos y se utilizaron imágenes de entrada con una resolución de 4000×4000 .

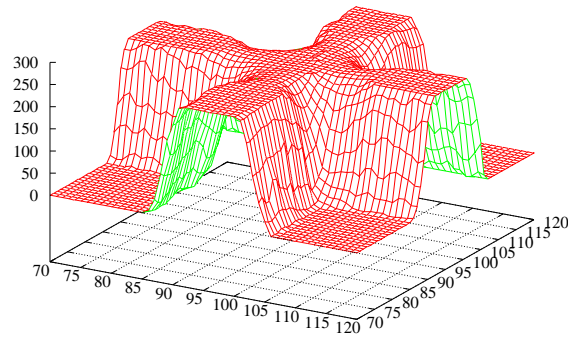


Figura 6.13: Gráfica resultante al aplicar la **vecindad elipsoidal doble**, utilizando la función **Bessel j_0** , la estrategia **vecinos completos** y la combinación $a = 0.45d$, $b = 0.06d$ y $\sigma = 0.01d$.

La imagen de la Figura 6.12.a corresponde a la combinación: eje mayor, a , 100 pixels o $0.26d$, eje menor, b , 10 pixels o $0.02d$, y amplitud, am , 1 o $\sigma = 0.26d$, en la cual puede observarse la generación de las manchas pero con los cuadros negros distorsionados. Para la imagen de la Figura 6.12.b se utilizó la combinación: $a = 0.39d$ o 120 pixels, $b = 0.06d$ o 20 pixels, y $\sigma = 0.19d$ o 60 pixels, en esta imagen se presentan las manchas con mejor definición y los cuadros negros tienen una distorsión menor. Y para la imagen de la Figura 6.12.c la combinación fue: $a = 0.45d$ o 100 pixels, $b = 0.09d$ o 30 pixels, y $\sigma = 0.1d$ o 33 pixels, en la cual se generaron las manchas con una forma muy similar a la producida por el ojo humano y muy buena definición, más aún, la distorsión de los cuadros negros es mínima. La Figura 6.13, muestra la gráfica correspondiente al resultado de aplicar la vecindad elipsoidal doble con la combinación $a = 0.45d$, $b = 0.09d$ y $\sigma = 0.1d$, en la cual se puede comprobar que las ilusiones observadas en la Figura 6.12.c son producidas por el simulador y no por el ojo humano al observar la imagen.

6.5 Efecto de la rotación de la imagen de entrada

Como se presentó en la Sección 1.3 del Capítulo 1, las ilusiones sobre la imagen de entrada varían con la rotación, al grado de que tienden a desaparecer.

En la Figura 6.14 se presentan las imágenes resultantes de aplicar la mejor combinación descrita hasta el momento, sobre 4 imágenes de entrada rotadas a diferentes grados. En estas imágenes se puede observar como las manchas tienden a desaparecer con la rotación de la imagen de entrada, es decir, entre mayor sea el ángulo de rotación (α) de la imagen de entrada, las manchas se generan más ligeras, tal como se produce este efecto en el ojo humano.

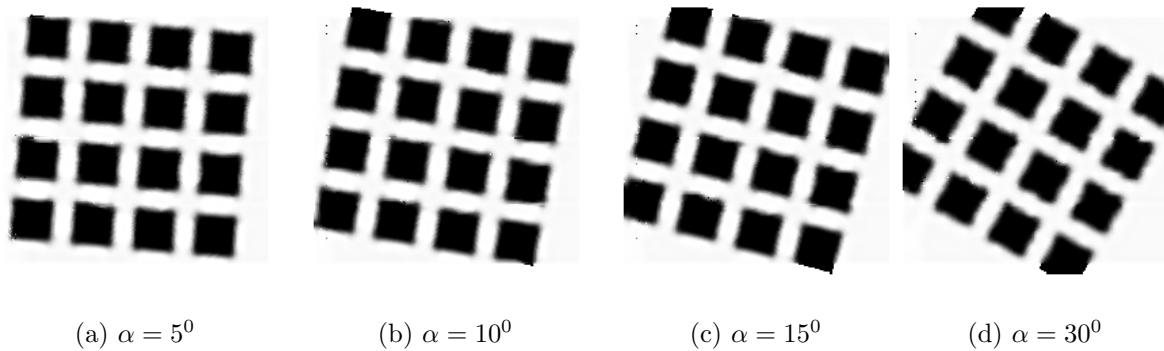


Figura 6.14: Imágenes de salida al aplicar la **vecindad elipsoidal doble** sobre la **forma 4** de los arreglos, utilizando la función **Bessel j_0** , la estrategia de selección **vecinos completos**, resolución de la imagen de entrada de 4000×4000 pero, rotada a diferentes grados.

6.6 Tiempos de ejecución

Los resultados obtenidos de las pruebas realizadas con el simulador de redes neuronales aplicadas al proceso de visión temprana permiten observar su efectividad en la simulación del fenómeno deseado. Ahora corresponde evaluar su rendimiento. Las pruebas del simulador paralelo se aplicaron a la misma resolución de los arreglos de perceptrones utilizados en las pruebas anteriores 2000×2000 , tomando los resultados como una muestra a escala de lo que sucedería el ejecutar el simulador con las resoluciones máximas de los arreglos de perceptrones. A pesar de que la resolución de los arreglos hexagonales de perceptrones es relativamente pequeña, el tiempo consumido por el simulador es considerable. En la Tabla 6.5 se presentan la plataforma de experimentación (infraestructura) y las características del sistema que se emplearon para la obtención de los resultados de las pruebas de rendimiento.

Es importante hacer notar que las características de infraestructura y del sistema son relevantes para el rendimiento, ya que si se varía alguna de las características, los tiempos de ejecución registrados no coincidirían. Los resultados de la ejecución paralela del simulador, aplicando la estrategia de particionamiento por bloques en la etapa de interacción de los perceptrones, se muestran en la Tabla 6.6; gráficamente se presentan en la Figura 6.15, donde se puede observar el beneficio obtenido al resolver el problema en varios procesadores.

En la gráfica de la Figura 6.15 podemos observar la notable reducción del tiempo de ejecución que se obtiene al incrementar el número de procesadores. Con dos procesadores el tiempo consumido por la simulación se reduce el 50%, aproximadamente. Con tres procesadores se reduce alrededor del 64%, con cuatro la reducción es casi del 70%, a partir del uso de 5 procesadores la reducción del tiempo es muy ligera, esto es porque se agrega una segunda computadora para el procesamiento,

Características de las máquinas	
Máquina de multiprocesamiento	Presley
Procesadores	4 procesadores Intel Xeon de 550 MHz
Disco Duro	SCSI de 28 GB
Memoria	1 GB de memoria RAM
Máquina de multiprocesamiento	Elvis
Procesadores	4 procesadores Intel Xeon de 750 MHz
Disco Duro	SCSI de 28 GB
Memoria	1 GB de memoria RAM
Características del sistema	
Resolución del arreglo de perceptrones	2000 × 2000
Modelo	Vecindad elipsoidal doble
Estrategia de selección de vecinos	Vecinos completos
Número de iteraciones	10
Eje mayor de la elipse	140 \approx 0.45 <i>d</i>
Eje menor de la elipse	30 \approx 0.09 <i>d</i>
Resolución de la imagen de entrada	4000 × 4000
Resolución de la imagen de salida	200 × 200

Tabla 6.5: Características a considerar para la ejecución paralela del simulador.

Procesadores	Tiempo en Minutos	Aceleración
1	66.290	1.000
2	33.406	1.984
3	23.507	2.820
4	20.083	3.300
5	19.440	3.409
6	16.441	4.031
7	14.835	4.468
8	13.589	4.878

Tabla 6.6: Tiempos de la ejecución paralela y su aceleración.

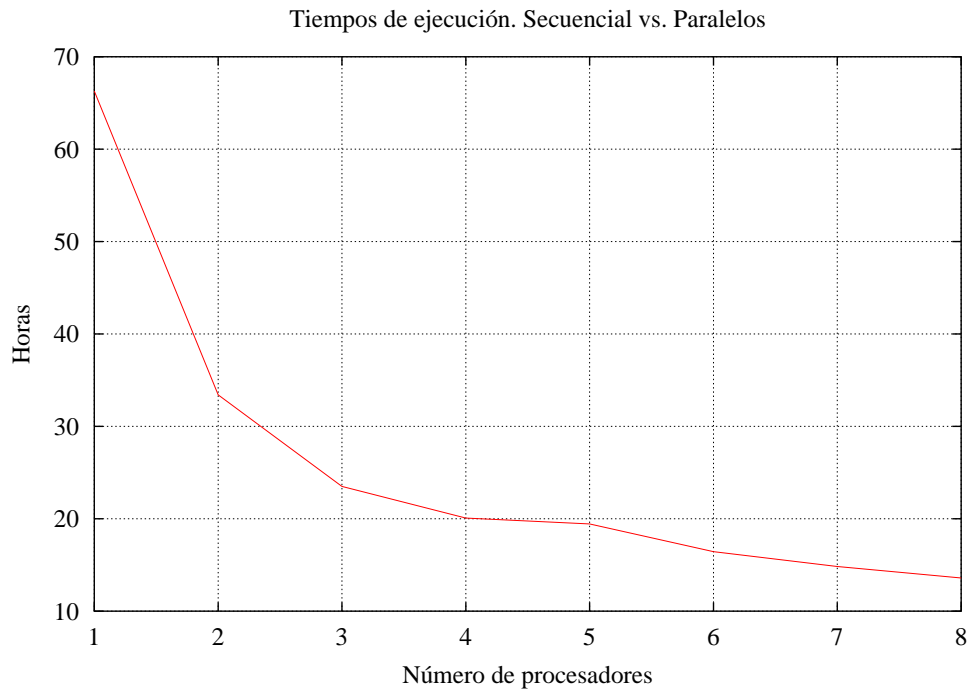


Figura 6.15: Gráfica de tiempos de ejecución. Secuencial vs. Paralelos.

entonces la comunicación ya no es sólo entre los procesadores de la máquina “presley”, sino que ahora tiene que comunicarse con los procesadores de la máquina “elvis”, pero aún así el tiempo se reduce. El menor tiempo se presenta con 8 procesadores, con lo cual, el tiempo se reduce alrededor del 80%.

En la gráfica de la Figura 6.16, se presenta la aceleración del tiempo de ejecución del simulador al agregar procesadores. En esta gráfica se puede observar que existe una buena aceleración al incrementar el número de procesadores hasta 4, con más de cuatro sigue habiendo aceleración, pero la tendencia de reducción del tiempo cambia.

Los resultados mostrados en estas gráficas muestran que la estrategia de particionamiento por bloques es conveniente para este simulador de redes neuronales y la reducción en tiempo es notable.

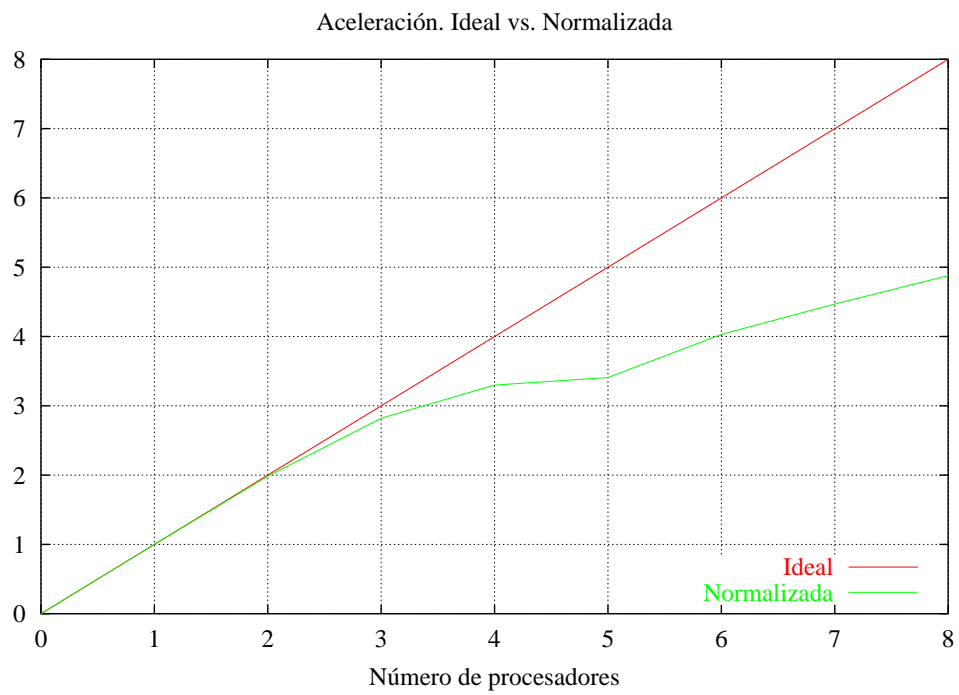


Figura 6.16: Gráfica de aceleración. Ideal vs. Normalizada.

Conclusiones

La simulación de la retina humana es un problema amplio, aún en estudio. Los resultados que en este trabajo se han expuesto presentan un avance en la resolución de incógnitas que se presentan al estudiar el ojo humano.

Dentro de los estudios sobre el comportamiento del ojo humano, el área de la generación de ilusiones está en constante evolución, como se observa en la revisión de los trabajos presentados en el Capítulo 2. El simulador objeto de esta tesis permite simular uno de los fenómenos que han captado la atención de diversos investigadores.

Para simular la visión temprana del proceso de visión humano se reprodujo el arreglo de células fotorreceptoras que forma la retina humana. Este arreglo artificial fue construido por una estructura hexagonal compuesta por círculos, los cuales fueron interpretados como receptores, en la etapa de absorción de escenas visuales y, como perceptrones, en la etapa de interacción de perceptrones para producir la ilusión buscada.

De estos arreglos hexagonales se generaron cuatro formas distintas, de la forma más regular (arreglo regular con círculos simétricos y puntos regulares), hasta la forma más irregular (arreglo concéntrico con círculos perturbados y puntos regulares). Estos diferentes arreglos fueron creados para ilustrar la diferencia en los resultados entre un arreglo que no sigue la estructura de la retina humana y aquellos que se apegan lo más posible a ésta. Como se puede observar en las Figuras del capítulo 6, la forma de los arreglos que mejores resultados arrojó es la **forma 4**, la forma más irregular. Sin embargo, la forma de los arreglos no es el único factor que influyó a la obtención de los resultados buscados. Otra parte importante para obtener estos resultados fueron los diferentes modelos para la interacción entre perceptrones, al aplicarlos sobre los arreglos hexagonales de perceptrones generaron las ilusiones esperadas.

Las diferentes vecindades de los modelos para la obtención de ilusiones fueron creadas para probar diversas formas como se cree que interactúan las células fotorreceptoras de la retina humana.

El primer modelo probado fue la vecindad hexagonal de 7 elementos. Esta vecindad, en algunas de sus modalidades, produjo un efecto ligero de obscurecimiento de la imagen, en otras, se resaltaron ligeramente los bordes de la imagen de entrada, dependiendo de la modalidad. Los resul-

tados obtenidos sugirieron que las vecindades de interacción deberían incorporar perceptrones más alejados.

Mediante la aplicación de una vecindad circular con sus diversas estrategias de elegir los perceptrones vecinos contenidos en la circunferencia circunscrita en un perceptrón central, se obtuvieron manchas circulares por toda la imagen. Este efecto se puede apreciar como un oscurecimiento general de la imagen de entrada, además de suavizar sus bordes. Al aplicar las diferentes estrategias para seleccionar los perceptrones vecinos, no se encontraron diferencias notables. Al aplicar la selección **vecinos completos** el tiempo es el más pequeño en comparación con **vecinos parciales** y **vecinos proporcionales**. El efecto producido por la vecindad circular indicó el camino para probar la vecindad elipsoidal, la cual, con una elipse lo suficientemente delgada, generaría sombras sesgadas en la imagen de entrada con lo que se obtendría la ilusión buscada.

Al aplicar el tercer modelo, la vecindad elipsoidal, se encontraron efectos muy cercanos a los esperados. En un inicio se hicieron pruebas con una sola elipse rotada a 45° . El resultado fue la aparición de sombras alargadas y sesgadas a 45° , formando diagonales discontinuas. Después, se aplicó de forma independiente una elipse rotada a 135° , con la cual también se obtuvieron sombras alargadas pero ahora sesgadas a 135° . Para obtener el efecto deseado se probó la promediación de ambas elipses (como una elipse doble), una rotada a 45° y la otra a 135° . Se obtuvo un ensombrecimiento en el área donde apuntan las esquinas de 4 cuadros negros, cercano a las ilusiones que el ojo produce al observar una escena visual como la imagen de entrada utilizada.

Los primeros resultados obtenidos fueron generados con la función de transferencia exponencial. Los resultados demostraron que era conveniente probar otras funciones con comportamientos similares, con el fin de acentuar el fenómeno de inhibición lateral.

Se realizaron pruebas con las funciones Sinc, Bessel j_0 y Bessel j_1 . Las funciones Sinc y *Bessel* j_0 tienen un comportamiento similar entre ellas, el resultado obtenido al aplicarlas generó sombras mejor definidas que las obtenidas con la exponencial. Esto debido a que los valores de las funciones Sinc y *Bessel* j_0 son positivos y negativos, situación que se apega más a las influencias inhibitoria y excitatoria que genera la inhibición lateral, mientras que los valores de la función exponencial son todos positivos, lo cual provoca que los perceptrones sean poco sensibles. Al aplicar la función Bessel j_1 , se apreció una distorsión muy grande de la imagen, debido a la irregularidad de sus valores producidos, aunque se producen las manchas, el resultado se alejó de lo esperado.

Los resultados obtenidos con la vecindad elipsoidal doble y las cuatro funciones fueron generados al aplicar elipses con las siguientes características: eje mayor = $6.5d$, eje menor = $0.65d$ y una amplitud regular de la función, es decir $\sigma = 6.5d$. Estos datos, aunque generan las manchas como se esperaba, los cuadros negros de la imagen se degradan, efecto que no produce el ojo humano. A

partir de esta observación se procedió a probar diferentes valores para el eje mayor, a , eje menor, b , de las elipses y la amplitud, σ , de las funciones de transferencia.

Al utilizar diferentes valores para a , b y σ , se obtuvieron resultados que mejoraban los anteriores. Con la función Bessel J_0 , los mejores resultados se obtuvieron con $a = 0.45d$, $b = 0.09d$ y $\sigma = 0.1$. Con estos valores, las sombras aparecieron más cercanas a las producidas por el ojo y mejor aún, la distorsión de los cuadros negros es mínima.

Todas estas pruebas fueron realizadas con imágenes de entrada de resolución menor a la resolución del arreglo de receptores, cuestión que en la realidad no se define del todo cierta. Por este motivo se optó por probar con imágenes de entrada de resolución igual o mayor a la resolución del arreglo de receptores. Estas pruebas se hicieron con imágenes de entrada de resolución 2000×2000 , 3000×3000 , \dots , y 5000×5000 ; y con arreglos de receptores de resolución 2000×2000 . Con estas características de entrada, las manchas siguieron ahí, pero los cuadros mejoraron su definición conforme se aumentó la resolución de las imágenes de entrada.

La vecindad elipsoidal, en sus tres modalidades, fue la que produjo mejores resultados, pero también es la que consume mayor cantidad de tiempo. La aplicación de esta vecindad y de la otras dos, está comprendida en la etapa de interacción de perceptrones que, en general, es la que consume un 99% del tiempo de ejecución del simulador. Tomando en cuenta estos dos factores, se procedió a la paralelización de esta etapa, es decir, de la aplicación de las vecindades. Para la ejecución del simulador en varios procesadores, se aplicó la estrategia de particionamiento por bloques a los arreglos de perceptrones. Los resultados de aplicar esta estrategia de particionamiento sobre esta etapa del sistema fueron contundentes. El tiempo de ejecución se redujo casi un 80% al utilizar ocho procesadores. Es de esperarse que al incrementar las resoluciones de la imagen de entrada y del arreglo de perceptrones, se mantenga la proporción en que se reduce el tiempo de respuesta. Se puede decir que la estrategia de particionamiento por bloques es suficiente para obtener tiempos de ejecución razonables con este simulador paralelo de redes neuronales.

Con los resultados mostrados podemos concluir que el simulador de redes neuronales aplicado al proceso de visión temprana tuvo éxito. Se lograron generar las ilusiones presentadas en este trabajo como efecto de la interacción y la inhibición lateral entre los perceptrones y su organización que simula la retina humana, con lo cual se explica la aparición de esta ilusión, haciendo incapié en los siguientes puntos:

1. La estructura de la retina humana mantiene un arreglo concéntrico hexagonal con células fotorreceptoras de radios no simétricos.
2. Las células fotorreceptoras de la retina humana interactúan entre sí con un comportamiento similar a una elipse doble, con elipses delgadas y alargadas.

3. La inhibición lateral entre estas células no tiene un comportamiento regular, tampoco tan suave como una función Sinc ni tan abrupto como una función Bessel j_1 , sino en un nivel intermedio como una función Bessel j_0 .

En este trabajo se propusieron diferentes modelos y formas para la estructura de la retina humana y su comportamiento. No es el último paso. Este simulador solo produce una de las muchas ilusiones que son generadas en el ojo humano en su etapa de visión temprana, además, hay otro tipo de efectos que se producen en el cerebro. El camino para comprender completamente al ojo humano y su comportamiento es aún vasto y muy amplio; el funcionamiento de varios de sus procesos es aún un misterio.

Este trabajo no estaría completo si no se mencionaran las mejoras posibles a este sistema.

- Generalizar el sistema para un mayor número de ilusiones a producir. Es decir, implementar más modelos de interacción de perceptrones para simular la aparición de ilusiones con otras características.
- Agregar al sistema un ambiente gráfico donde se permita observar la imagen de entrada y el efecto que está sufriendo al aplicar los modelos de interacción de perceptrones.
- Adicionar un editor gráfico de imágenes donde se puedan producir las imágenes de entrada para diferentes ilusiones.
- Implantar un lector de imágenes que permita abrir y cargar imágenes en diferentes formatos para introducir escenas visuales reales.

Bibliografía

- [1] P. H. Lindsay and D. A. Norman, *Human Information Processing: An Introduction to Psychology*, Academic Press, 1972.
- [2] O. Ruiz, *Estudio de algunas Variantes de la Ilusión de Hermann*. Tesis Doctoral del Departamento de Fisiología, Biofísica y Neurociencias del Centro de Investigaciones y Estudios Avanzados del I. P. N. México D. F.
- [3] T. Yagi, Y. Hayashida and S. Kameda, "An Analog VLSI Which Emulates Biological Vision", *Second International Conference on Knowledge-Based Intelligent Electronic Systems*, April 21-23, 1998, pp. 454-460.
- [4] M. J. Wilcox and D. C. Thelen, Jr., "A Retina with Parallel Input and Pulsed Output, Extracting High-Resolution Information", *IEEE Transactions on Neural Networks*, Vol. 10, No. 3, MAY 1999.
- [5] T. Ogawa, T. Monohara, H. Kanada and Y. Kosugi, "A Neural Network Model for Realizing Geometric Illusions Based on Acute-Angled Expansion", *Sixth International Conference on Neural Information Processing 1999. Proceedings. ICONIP'99*, November 16-20, 1999, pp. 550-555 vol. 2.
- [6] M. T. Hagan, H. B. Demuth and M. Beale, *Neural Networks Design*, PWS Publishing Company, 1996.
- [7] B. Müller and J. Reinhardt, *Neural Networks. An Introduction*, Springer-Verlag, 1991.
- [8] T. Toshiro, T. Akiyoshi and T. Kunibobu, "Parallel Computing Algorithm of Neural Networks on an Eighth-Neighbor Processor Array", *Twelfth Annual International Phoenix Conference on Computers and Communicatios*, March 23-26, 1993, Phoenix, pp.559-564.

-
- [9] D. Timothy J and M.-T. Evangelia, “A Pattern Recognition Application of the ALOPEX Process with Hexagonal Arrays”, *International Joint Conference on Neural Networks*, June 18-22, 1989, pp.119-125.
- [10] N. Funabiki and Y. Takefuji, “A Neural Network Parallel Algorithm for Channel Assignment Problems in Cellular Radio Networks”, *IEEE Trans. on Vehicular Technology*. Vol. 41. No. 4, November 1992.
- [11] J. D. Foley, A. van Dam, S. K. Feiner, J. F. Hughes and R. L. Phillips, *Introduction to Computer Graphics*, Addison-Wesley, 1997.
- [12] R. C. González and R. E. Woods, *Digital Image Processing*, Addison-Wesley, 1992.
- [13] C. H. Lehmann, *Geometría Analítica*, Ed. Limusa, 1980.
- [14] M. Quinn, *Parallel Computing. Theory and Practice*, McGraw Hill, 1994.
- [15] D. E. Culler and J. Pal Singh with A. Gupta, *Parallel Computer Architecture. A Hardware/Software Approach*, Morgan Kaufmann, 1999.
- [16] R. Hempel, “The MPI Standard for Message Passing”, *Lecture Notes in Computer Science*, Vol. 797, pp. 247-252, 1994.