

Introducción

David Marr (1982). *La visión*, San Francisco: W. H. Freeman, pp. 19-38, 54-61

David Marr fue un científico brillante que tuvo una influencia importante sobre varias áreas de la neurobiología teórica. Se añade a su leyenda era su muerte extemporánea en el cúlmen de su productividad. Comenzó su carrera con varios trabajos impresionantes que modelan el cerebelo y corteza cerebral (Marr, 1969, 1970). Su trabajo más conocido es la *Visión*, que ha sido muy influyente en el campo de la visión computacional. Hemos incluido aquí únicamente secciones breves, pero el libro contiene muchas ideas importantes. Es difícil extractar a causa de la complejidad de algunos de los argumentos, y, ya que es fácilmente disponible, la única manera de hacer justicia a las ideas está en leer una copia.

Marr sugirió que la manera para acercarse al análisis del mecanismo del procesamiento de la información era darse cuenta de que hay tres niveles separados de función a ser entendidos. El primero está "la teoría computacional," que es la meta del cómputo; el segundo es el de "la representación y el algoritmo" usado para alcanzar la meta; y el tercero es el de "la implementación de soporte físico" usado para realizar el algoritmo que realiza la meta. Un ejemplo podría ser el tres en raya (tic-tac-toe), que Marr de menciona brevemente. La meta consiste en ganar el juego consiguiendo tres de sus marcadores en una línea, o, si esto no es posible, no permitir que el adversario lo logre. La representación podría incluir el tablero de juego y los marcadores, pero podría incluir también algunos rasgos importantes del tablero: los cuadrados vacíos en líneas o ubicaciones críticas de dos marcadores con un cuadrado vacío entre ellos. Hay un número de algoritmos conocido para el tres en raya, y describir formalmente la representación interna del tablero y el cómputo requerido sería el segundo nivel. Una vez se escribió el algoritmo en forma de LISP o programa de Pascal, podría ejecutarse sobre cualquier máquina conveniente que fuera lo suficientemente rápida para el propósito deseado, desde un Cray a, como Marr menciona, un dispositivo de propósito especial hecho por Tinkertoys. Mientras la máquina pudiera ejecutar el algoritmo, el tipo de soporte físico usado sería materia de

conveniencia, por ejemplo, llegaría a ser pertinente el tipo de costo y eficiencia.

El análisis de fenómenos perceptuales en estos niveles ha llegado a ser un artículo de fe en algunos laboratorios, y los niveles se interpretan quizás más rígidamente de lo que Marr podría haber querido. También, aunque Marr sugirió que los niveles estaban "libremente acoplados", muchos modelistas connexionistas argumentarían que la conexión ha de ser ajustada, especialmente, entre el nivel algorítmico y el nivel de soporte físico. Hay fuertes interacciones internivel: algunos algoritmos deseables no pueden ejecutarse eficientemente o incluso de manera total en algunas máquinas. Elegidos los algoritmos y sus metas igualadas, el soporte físico puede ser principalmente conducido por sistemas Biológicos relacionados con la eficiencia y la velocidad; es generalmente mejor tener razón la mayoría de las veces y ser rápido que tener razón siempre y ser lento. De colonización para segunda mejor podría ser mejor para un sistema real, una limitación que afectaría las metas, los algoritmos, y el soporte físico. (Se ha indicado que en biología, "lo mejor es enemigo de lo bueno".)

Aunque los ingenieros, que pueden comenzar de nuevo, pueden tener las metas funcionales a la vista desde el principio, la evolución tiene que trabajar a partir del soporte físico existente. Un ejemplo bien conocido de esto es el vuelo del pájaro, donde es una materia de discusión caliente con respecto a cual era la función de las alas antes de que fueran suficientemente grandes para usarse para volar. ¿Eran radiadores que se usaban para descargar el excesivo calor, o quizás membranas para coger insectos que también llegaron a ser útiles en la locomoción? La evolución trabaja primero del soporte físico al cómputo; los diseñadores de dispositivos o algoritmos trabajan frecuentemente de la otra manera.

Una vez que el soporte físico desempeña una función útil, sin embargo, la evolución puede trabajar también la otra vía, para hacer un cómputo existente más eficiente. Marr menciona, en una sección llamada "El propósito de visión," una serie de ejemplos biológicos de la manera que impresionantes sistemas biológicos hacen lo que hace y lo hacen muy eficientemente: la mosca son pequeños autómatas

móviles con algunas representaciones especializadas de aspectos relevantes del mundo visual.

La noción de representación llegará a ser uno de los puntos clave en redes neuronales en los próximos años. Marr era consciente de la importancia de esto. Para él, "Una representación es un sistema formal que hace explícitas ciertas entidades o tipos de información, junto con una especificación de como el sistema hace esto" (p. 20). En la visión humana, esto implica una serie de análisis consecutivamente más complejos de la imagen en niveles más altos y más altos de abstracción a partir de la imagen (ver Tabla I, t, p. 37). Para reconocer un objeto, por ejemplo, comenzamos a partir de una serie de valores de intensidad en puntos sobre una imagen retinal y concluimos con una representación de "un objeto-centrado", donde nuestra descripción mental es desacoplada (decoupled) desde la imagen exacta que le dio origen. Mucha de la utilidad de la idea que el mundo puede representarse como objetos separados viene de esto: que los objetos pueden por lo menos parcialmente entenderse y ser descritos independientemente del punto de vista de un observador particular, o incluso del ejemplo particular del objeto vistos. Este nivel de representación es similar en alguna manera a los conceptos en la ciencia cognoscitiva. Ya que es tan difícil hacer prácticamente esta cantidad de abstracción, debe hacerse en un número de pasos, alguno de los cuales se describe en forma detallada en el libro.

El último extracto del libro (la sección 2,2) describe brevemente uno de los pasos primeros en el largo proceso de alcanzar la representación final centrada del objeto: para encontrar donde están los bordes en una imagen. Es sorprendente que algo que nosotros hacemos tan fácilmente sea tan difícil de hacer artificialmente. Marr sugiere que la mejor manera de buscar cambios de intensidad que pueden corresponder a los bordes es, primero, para mirar varias escalas diferentes de frecuencia espacial, ya que el mismo suceso que ocurre en varios canales podría corresponder a un borde cierto, y no a ruido aleatorio. El segundo, es muy probable que un borde se corresponda con un cambio súbito en la intensidad; i. e., la derivada de intensidad será grande a través del borde. Un cambio tal puede escogerse computando la segunda derivado de las intensidades de imagen y anotando los lugares donde la segunda derivada llega a ser cero, esto es, un cero cruzado. Marr describe un número de criterios

técnicos que deben encontrarse para un buen filtro capaz de detectar tales lugares, y argumenta que el operador Laplaciano que opera sobre la distribución Gaussiana es la elección más práctica. Es fácil cambiar la escala de tal un operador, y responde a la segunda derivada adecuadamente. También da buenos resultados cuando se usa sobre una imagen, y varios ejemplos de su uso se dan en el extracto corto.

Las partes más detalladas e impresionantes de Visión se encuentran en sus discusiones de visión de bajo nivel . Es desafortunado que Marr no tuvo tiempo suficiente para extender su enfoque a niveles más altos de análisis de la imagen en el detalle comparable.

Referencias

D. Marr (1969), "A theory of cerebellar cortex," *Journal of Physiology* 202:437-470.

D. Marr (1970), "A theory for cerebral neocortex," *Proceedings of the Royal Society of London, Series B* 176:161-234

D. Marr (1982). *Vision*, San Francisco: W. H. Freeman pp. 18-38, 54-61

1.2. Comprensión de Sistemas complejos de Procesamiento de información.

Casi nunca un sistema complejo de cualquier tipo puede entenderse como una extrapolación simple de las propiedades de sus componentes elementales. Considerar, por ejemplo, algunos gases en una botella. Una descripción de los efectos termodinámicos de temperatura, presión, densidad, y las relaciones entre estos factores no es formulada usando un gran conjunto de ecuaciones, una para cada una de las partículas implicadas. Tales efectos se describen en su nivel propio, el de una enorme colección de partículas; el esfuerzo está en mostrar que en un principio las descripciones microscópicas y macroscópicas son consistentes una con la otra. Si uno espera lograr una comprensión plena de un sistema tan complicado como un sistema nervioso, un embrión en desarrollo, un conjunto de vías metabólicas, una botella de gas, o incluso un gran programa de ordenador entonces hay que estar dispuesto a contemplar tipos diferentes de explicación en niveles diferentes de descripción que se vinculan, por lo menos en el principio, en una totalidad coherente, aún

cuando si la vinculación de los niveles en el detalle completo es no práctica. Para el caso específico de un sistema que resuelve un problema de procesamiento de información, hay además el doble cordón de proceso y representación, y ambas ideas necesitan alguna discusión.

Representación y Descripción

Una representación es un sistema formal para hacer explícitas ciertas entidades o tipos de información, junto con una especificación de como el sistema hace esto. Llamaré el resultado de usar una representación para describir una entidad determinada una descripción de la entidad en esa representación (Marr y Nishihara, 1978).

Por ejemplo, los sistemas Árabe, Romano y numerales binarios son todos sistemas formales para representar números. La representación Árabe consiste en una cadena de símbolos obtenida a partir del conjunto (0, 1, 2, 3, 4, 5, 6, 7, 8, 9), y la regla para construir la descripción de un entero particular n es descomponer n en una suma de múltiplos de potencias de 10 y unir estos múltiplos en una cadena con las potencias más grandes hacia la izquierda y las más pequeñas hacia la derecha. Así, treinta y siete es igual $3 \times 10^1 + 7 \times 10^0$, que da lugar a 37, la descripción del número del sistema numeral Árabe. Lo que esta descripción hace explícito es la descomposición del número en potencias de 10. La descripción del sistema numeral binario del número treinta y siete es 100101, y esta descripción hace explícita la descomposición del número en potencias de 2. En el sistema numeral Romano, treinta y siete se representa como XXXVII.

Esta definición de representación es bastante general. Por ejemplo, una representación para la forma sería un esquema formal para describir algunos aspectos de forma, junto con las reglas que especifican como se aplica el esquema a cualquier forma particular. Una partitura musical provee una manera de representar una sinfonía; el alfabeto permite la construcción de una representación escrita de palabras y así sucesivamente. La frase "el esquema formal" es crítico a la definición, pero el lector no debería asustarse por ello. La razón es simplemente que nosotros repartimos con máquinas de procesamiento de la información y la manera en que tales máquinas

trabajan es usando símbolos en lugar de cosas para representar cosas, en nuestra terminología. Decir que algo es un esquema formal significa sólo que es un conjunto de símbolos con reglas para ponerlos juntos ni más ni menos.

Una representación, por lo tanto, no es una idea extraña ya que habitualmente usamos representaciones. Sin embargo, la noción de que uno puede captar algún aspecto de realidad haciendo su descripción usando un símbolo y que hacerlo puede ser útil parece mí una idea poderosa y fascinante. Pero incluso los ejemplos simples que hemos discutido introducen algunos puntos más bien generales e importantes que surgen cuando uno pretende usar una representación particular. Por ejemplo, si uno escoge la representación numeral Árabe, es fácil descubrir si un número es un potencia de 10 pero difícil descubrir si es una potencia de 2. Si uno escoge la representación binaria, la situación se invierte. Así, existe un compromiso; cualquier representación particular hace explícita cierta información en la información consumida encajada en el medio ambiente que puede ser bastante difícil de recuperar.

Este punto es importante, porque como está representada la información puede afectar mucho a la facilidad de hacer cosas diferentes con ella. Esto es evidente incluso en nuestros ejemplos numéricos: Es fácil sumar, restar, e incluso multiplicar si se usan representaciones binarias o Árabes, pero no es tan fácil hacer estas cosas, especialmente multiplicación, con numerales Romanos. Esta es una razón clave de por qué la cultura Romana fracasó en desarrollar las matemáticas en la manera en que lo hicieron las culturas Árabes más recientes.

Un problema análogo encaran hoy los ingenieros de ordenadores. La tecnología electrónica favorece mucho más al sistema numeral binario que al sistema convencional de base 10, aún los humanos introducen sus datos y requieren los resultados en base 10. La decisión de diseño que encara el ingeniero, por lo tanto, es, Debería pagar el costo de conversión en base 2, efectuando la aritmética en una representación binaria, y entonces volver a convertir en números decimales en el output; o debería sacrificar la eficiencia del sistema de circuitos para efectuar operaciones directamente en una representación decimal? En conjunto, las computadoras de los

negocios y las calculadoras de bolsillo siguen el segundo enfoque, y los ordenadores de propósito general siguen el primero. Pero si bien uno no está restringido a usar simplemente un sistema de representación para un tipo determinado de información, la elección de cual usar es importante y no puede tomarse a la ligera. Determina qué información se hace explícita y por tanto cual sigue adelante en el trasfondo, y tiene un efecto de largo alcance sobre la facilidad y dificultad con que las operaciones pueden consecutivamente efectuarse sobre esa información.

Proceso

El término proceso es muy amplio. Por ejemplo, la adición es un proceso, y también lo es realizar una transformada de Fourier. Pero también lo es hacer una taza de té, o ir a comprarlo. Para los propósitos de este libro, quiero restringir nuestra atención a los significados asociados con máquinas que efectúan tareas de procesamiento de información. Así examinemos en profundidad las nociones subyacentes a un simple mecanismo, un caja registradora en el mostrador de control de un supermercado.

Hay varios niveles que uno necesita para comprender tal mecanismo, y es quizás muy útil pensar desde el punto de vista de tres de ellos. El más abstracto es el nivel de qué hace el dispositivo y por qué. Lo que hace es aritmética, así nuestra tarea primera es dominar la teoría de la adición. La adición es una comparación (emparejamiento), comúnmente denotada por $+$, de pares de números en números únicos; por ejemplo, $+$ empareja el par $(3, 4)$ a 7 , y Yo escribiré esto en la forma $(3 + 4) = 7$. Sin embargo, la adición tiene un número de propiedades abstractas. Es conmutativa: tanto $(3 + 4)$ como $(4 + 3)$ tienen por valor 7 ; y asociativa: la suma de $3 + (4 + 5)$ es igual que la suma de $(3 + 4) + 5$. Además hay el elemento distinguido único, cero, cuya suma no tiene efecto: $(4 + 0) = 4$. También, para cada número hay un único "inverso", escrito como (-4) en el caso de 4 , que cuando se agrega al número da cero: $[4 + (-4)] = 0$.

Notar que estas propiedades son parte de la teoría fundamental de adición. Ellas son ciertos sin que importe que los números estén escritos en representación binaria, árabe, o romana, ni como se

ejecute la adición. Así parte de este primer nivel es algo que podría caracterizarse como qué está siendo computado.

La otra la mitad de este nivel de explicación tiene que ver con la pregunta de por qué la caja registradora ejecuta la adición y no, por ejemplo, la multiplicación al combinar los precios de los artículos comprados para llegar en una cuenta final. La razón es que las reglas que sentimos intuitivamente que son apropiadas para combinar los precios individuales definen de hecho la operación matemática de la adición. Estas pueden formularse como restricciones de la manera siguiente:

1. Si Ud. No compra nada, no le debería costar nada; y comprar nada y algo debería costar lo mismo que comprar simplemente algo. (Reglas de cero.)

2. El orden en que mercancías se ponen de delante del cajero no debe afectar el total. (Commutatividad).

3. Organizar las mercancías en dos pilas y pagar cada pila separadamente no debe afectar a la cantidad total pagada. (Asociatividad; la operación básica para combinar precios).

4. Si Ud. compra un artículo y lo devuelve para que se lo reembolsen, su gasto total debería ser cero. (Inversa).

Es un teorema matemático que estas condiciones definen la operación de adición, por lo que es, por lo tanto, el cómputo apropiado a utilizar.

Este argumento global es el que llamo la teoría computacional de la caja registradora. Sus rasgos importantes son (1) que contiene los argumentos separados sobre qué se computa y por qué y (2) que la operación resultante se define singularmente por las limitaciones que satisfacer. En la teoría de los procesos visuales, la tarea subyacente consiste en derivar fiablemente las propiedades del mundo a partir de sus imágenes; la empresa de aislar las restricciones que sean a la vez suficientemente poderosas para permitir un proceso a ser definido y generalmente cierto del mundo es un tema central de nuestro examen.

Para que un proceso funcione realmente, sin embargo, uno tiene que realizarlo en alguna manera y por lo tanto seleccionar una representación para las entidades que manipula el proceso. El segundo de nivel del análisis de un proceso, por lo tanto, implica seleccionar dos cosas: (1) una representación para el input y para el output del proceso y (2) un algoritmo por el que la transformación puede realmente realizarse. Para la adición, por supuesto, las representaciones de input y output pueden ambas ser las mismas porque ambos consisten en números. Sin embargo esto no es cierto en general. En el caso de una transformada de Fourier, por ejemplo, la representación de input puede ser un dominio temporal, y el output, un dominio de frecuencia. Si el primero de nuestros niveles especifica qué y por qué, el segundo nivel especifica cómo. Para la adición, podríamos elegir numerales Árabes para las representaciones, y para el algoritmo nosotros podríamos seguir las reglas usuales de añadir primero los dígitos menos importantes y "llevar" si la suma excede de 9. Las cajas registradoras, sean mecánicas o electrónicas, comúnmente usan este tipo de representación y algoritmo.

Hay tres puntos importantes aquí. Primero, hay comúnmente una amplia elección de representación. Segundo, a menudo la elección del algoritmo depende más bien críticamente de la representación particular que es empleada. Y tercero, incluso para una representación determinada fija, hay frecuentemente varios algoritmos posibles para efectuar el mismo proceso. Cual de ellos se elige dependerá comúnmente de algunas características particularmente deseables o indeseables que los algoritmos pueden tener; por ejemplo, un algoritmo puede ser mucho más eficiente que otro, u otro puede ser ligeramente menos eficiente pero más robusto (esto es, menos sensibles a ligeras inexactitudes en los datos de que debe ejecutarse). O nuevamente, un algoritmo puede ser paralelo y otro serial. La elección, entonces, puede depender del tipo de soporte físico o la maquinaria en que el algoritmo ha de ser incorporado físicamente.

Esto nos lleva al tercero nivel, el del dispositivo en que el proceso ha de ser realizado físicamente. El punto importante aquí es que, una vez más, el mismo algoritmo puede implementarse en tecnologías bastante diferentes. El niño quien metódicamente agrega dos números de la derecha a la izquierda, "llevando" un dígito cuando es necesario, puede estar usando el mismo algoritmo que es implementado por los

cables y los transistores de la caja registradora en el supermercado de la esquina, pero la realización física del algoritmo es bastante diferente en estos dos casos. Otro ejemplo: Mucha gente ha escrito programas de ordenador para jugar al tres en raya, y hay un algoritmo más o menos estándar que no puede perder. Este algoritmo ha sido implementado de hecho por W. D. Hillis y B. Silverman en una tecnología bastante diferente, en una computadora hecha Tinkertoys, un conjunto de construcción de madera de niños. El ingenio completo monstruosamente desgarbado, que realmente funciona, está actualmente en un museo en la Universidad de Missouri en St. Louis.

Algunos estilos de algoritmo favorecerán a algunos substratos físicos mejor que a otros. Por ejemplo, en computadoras digitales convencionales, el número de conexiones es comparable al número de puertas, mientras en un cerebro, el número de conexiones es mucho más grande ($\times 10^4$) que el número de células del nervio. La razón subyacente es que los cables son más bien baratos en la arquitectura biológica, por lo que pueden crecer individualmente y en tres dimensiones. En la tecnología convencional, el cable a colocar es más o menos restringido a dos dimensiones, lo que restringe bastante severamente el ámbito de uso de algoritmos y técnicas paralelas; las mismas operaciones son frecuentemente mejor efectuadas en serie.

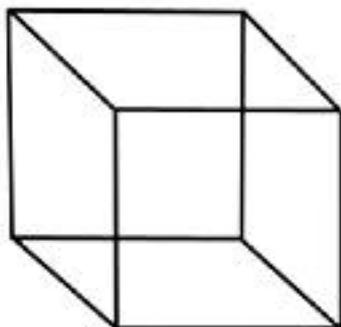
Teoría Computacional	Representación y algoritmo	Realización del soporte físico
¿Cual es la meta del cómputo, por qué es asignada, y cual es la lógica de la estrategia por la que puede ser efectuada?	¿Como puede implementarse esta teoría computacional? En particular, ¿cual es la representación del input y output, y cual es el algoritmo para la transformación?	¿Como puede realizarse físicamente el algoritmo y la representación?

Figura 1-4. Los tres niveles en que debe ser comprendida cualquier máquina que efectúe un proceso de información.

Los tres niveles

Podemos resumir nuestra discusión de manera semejante a la mostrada en la Figura 1-4, que ilustra los niveles diferentes en que un mecanismo de procesamiento de información debe comprenderse antes de uno pueda decir que lo ha entendido completamente. En un de extremo, el nivel superior, es la teoría computacional abstracta del dispositivo, en la que se caracteriza el desempeño del dispositivo como

emparejamiento de un de tipo de información con otra, se definen con precisión las propiedades abstractas de este emparejamiento, y se demuestra su adecuación y la suficiencia para la tarea a mano. En el centro está la elección de la representación para el input y el output y el algoritmo a utilizar para transferir de una información a la otra. Y en el otro extremo están los detalles de como el algoritmo y la representación se ejecutan físicamente, por así decirlo, la arquitectura detallada de la computadora. Estos tres niveles están acoplados, pero sólo flojamente. La elección de un algoritmo es influenciada por ejemplo, por lo qué se ha de hacer y por el soporte físico en que debe ejecutarse. Pero hay una posibilidad amplia de elección disponible en cada nivel, y la explicación de cada nivel implica aspectos que son más bien independientes de la de los otros dos.



La figura 1-5. La llamada ilusión de Necker, por el naturalista Suizo L. A. Necker que la desarrolló en 1832. La esencia del tema es que la representación bidimensional (a) ha destruido la profundidad de un cubo y que un cierto aspecto de visión humana consiste en recuperar esta tercera dimensión perdida. La profundidad del cubo puede desde luego percibirse, pero hay dos interpretaciones posibles, (b) y (c). La percepción de una persona fluctúa de una a otra.

Cada uno de los tres niveles de descripción tendrá su lugar en la comprensión eventual del procesamiento de la información perceptual, y por supuesto ellos están relacionados lógicamente y causalmente. Pero un punto importante a notar es que ya que los tres niveles están más bien relacionados sólo débilmente, algunos fenómenos sólo pueden explicarse en uno o dos de ellos. Esto significa, por ejemplo, que una explicación correcta de alguna observación psicofísica debe formularse en el nivel apropiado. En intentos de relacionar problemas psicofísicos con la fisiología, ocurre demasiado frecuentemente la confusión sobre el nivel en el que deben tratarse los

problemas. Por ejemplo, algunos se relacionan principalmente con los mecanismos físicos de visión tal como las postimágenes (por ejemplo, la que uno ve después de mirar fijamente una bombilla) o el hecho de que cualquier color puede ser equiparado por una mezcla conveniente de los tres primarios (una consecuencia principalmente de que nosotros los humanos tenemos tres tipos de conos). Por otra parte, la ambigüedad del cubo de Necker (la Figura 1-5) parece exigir un tipo diferente de explicación. Con seguridad, parte de la explicación de su inversión perceptual debe tener que ver con una red neuronal biestable (esto es, con dos estados estables distintos) en alguna parte dentro del cerebro. Pero pocos se sentirían satisfechos por una explicación que falle en mencionar la existencia de dos diferentes, pero perfectamente plausibles, interpretaciones tridimensionales de esta imagen bidimensional.

Para algunos fenómenos, el tipo de explicación requerido es bastante obvio. La neuroanatomía, por ejemplo, está claramente vinculada principalmente al nivel tercer, la realización física del cómputo. Lo mismo vale para los mecanismos sinápticos, potenciales de acción, interacciones inhibitorias, y así sucesivamente. La neurofisiología, también, se relaciona mayormente con este nivel, pero puede igualmente ayudarnos a comprender el tipo de representaciones que han de usarse, particularmente si se acepta algo de las líneas del punto de vista de Barlow que cité anteriormente. Pero uno tiene que llevar una precaución extrema al hacer inferencias a partir de los hallazgos neurofisiológicos sobre los algoritmos y las representaciones que están siendo usados, particularmente hasta uno tiene una idea clara sobre qué información necesita representarse y qué procesos necesitan implementarse.

La psicofísica, por otra parte, se relaciona más directamente con el nivel de algoritmo y representación. Algoritmos diferentes tienden a fracasar de manera radicalmente diferente según se les fuerza a los límites de su desempeño o se les priva de información crítica. Como veremos, primariamente la evidencia psicofísica nos demostró a Poggio y a mí que nuestro primer algoritmo de emparejamiento estéreo (Marr y Poggio, 1976) no era el único usado por el cerebro, y la mejor evidencia de que nuestro segundo algoritmo (Marr y Poggio, 1979) es el único que se usa de modo preliminar también viene de la psicofísica. Por supuesto, la teoría computacional subyacente

permaneció la misma en ambos casos, sólo los algoritmos eran diferentes.

La psicofísica puede también ayuda a determinar la naturaleza de una representación. El trabajo de Roger Shepard (1975), Eleanor Rosch (1978), o Elizabeth Warrington (1975) ofrecen algunos indicios interesantes en esta dirección. Más específicamente, Stevens (1979) argumentó a partir de experimentos psicofísicos que la orientación de superficie está representada por las coordenadas de sesgo e inclinación, más bien que (por ejemplo) la más tradicional del espacio (p, q) de gradiente (ver Capítulo 3). También dedujo a partir de la uniformidad del tamaño de errores hecho por sujetos que juzgan orientación de superficie sobre una gama amplia de orientaciones que las cantidades representacionales usadas para el sesgo y la inclinación son los ángulos puros y no, por ejemplo, sus cosenos, senos, o tangentes.

Más generalmente, si se tiene en la mente con claridad la idea de que fenómenos diferentes necesitan ser explicados a niveles diferentes, ello ayuda frecuentemente en la evaluación de la validez de los tipos diferentes de objeciones que se plantean de vez en cuando. Por ejemplo, una objeción favorita es que el cerebro es muy diferente de una computadora porque el uno trabaja en paralelo y la otra en serie. La respuesta a esto, por supuesto, es que la distinción entre serial y paralelo es una distinción a nivel de algoritmo; en general no es fundamental que cualquier cosa programada en paralelo pueda ser reescrita en serie (aunque no necesariamente viceversa). La distinción, por lo tanto, no ofrece bases para argumentar que el cerebro opera de manera tan diferente de una computadora que una computadora no puede programarse para desempeñar las mismas tareas.

La importancia de la Teoría Computacional

Aunque algoritmos y mecanismos son empíricamente más accesibles, es el nivel superior, el nivel de teoría computacional, el que es críticamente importante desde un punto de vista de procesamiento de la información. La razón es que la naturaleza de los cálculos que subyacen a la percepción depende más de los problemas computacionales que tienen que ser resueltos que del soporte físico particular en que sus soluciones se implementan. A expresar el tema

de otra manera, es probable que un algoritmo se entienda más fácilmente comprendiendo la naturaleza del problema que se está resolviendo que examinando el mecanismo (y el soporte físico) en que se incorpora.

En manera similar, tratar de comprender la percepción estudiando sólo las neuronas es como tratar de comprender el vuelo del pájaro estudiando sólo las plumas: Simplemente no puede hacerse. A fin de comprender el vuelo del pájaro, tenemos que comprender la aerodinámica; únicamente entonces tiene sentido tratar la estructura de las plumas y las diferentes formas de alas de pájaros. Más a claramente, como veremos, no podemos comprender por qué las células retinales del ganglio y las neuronas geniculadas laterales tienen los campos receptivos simplemente estudiando su anatomía y fisiología. Podemos comprender como estas células y las neuronas se comportan como lo hacen estudiando su instalación alámbrica e interacciones. Pero a fin de comprender por qué los campos receptivos son como son, por qué son circularmente simétricos y por qué sus regiones excitadoras e inhibitorias tienen las formas características y distribuciones, tenemos que saber un poco de la teoría de operadores diferenciales, canales de bandas de paso, y matemáticas del principio de incertidumbre (ver Capítulo 2).

Quizás no sea sorprendente que las disciplinas empíricas muy especializadas de las neurociencias fracasaron en apreciar totalmente la ausencia de la teoría computacional; pero es sorprendente que este nivel de enfoque no jugara un papel más enérgico en el desarrollo inicial de inteligencia artificial. Sería demasiado pedir, un programa heurístico para efectuar alguna tarea que fuera una teoría de esa tarea, y la distinción entre lo qué un programa hizo y como hizo no fuera tomada seriamente. Como resultado, (1) un estilo de explicación evolucionado que invocara el uso de mecanismos especiales para resolver problemas particulares. (2) estructuras particulares de datos, tales como las listas de los pares de valor de atributo llamadas listas de propiedad en el lenguaje de programación LISP, tienen lugar para sumarse a las teorías de la representación de conocimiento. Y (3) con frecuencia no había otra manera de determinar si un programa podría tratar un caso particular que corriendo el programa.

El fracaso en reconocer esta distinción teórica entre qué y cómo también impidió mucho la comunicación entre los campos de

lingüística e inteligencia artificial. La teoría de la gramática transformacional de Chomsky (1965) es una verdadera teoría computacional en el sentido definido anteriormente. Se preocupa únicamente en especificar cual debe ser la descomposición sintáctica de una oración Inglesa, y no en todos los casos que deberían lograrse con esa descomposición. Chomsky mismo fue muy claro sobre esto es alrededor de su distinción entre competencia y desempeño (*competence and performance*), aunque su idea de desempeño incluyó otros factores, como parar en mitad de la expresión, pero el hecho que su teoría fuera definida por transformaciones, que se parece a los cálculos, parece haber confundido a mucha gente. Winograd (1972), por ejemplo, se sintió capaz de criticar la teoría de Chomsky sobre la base de que no puede invertirse y que no puede hacerse para correr sobre una computadora; yo había oído reflejos del mismo argumento hecho por colegas de Chomsky en la lingüística en los que vuelven su atención a como la estructura gramatical puede computarse realmente a partir de una oración inglesa verdadera.

La explicación es simplemente que encontrar algoritmos por los que teoría de Chomsky pueda implementarse es un empeño completamente diferente de la formulación de la teoría misma. En nuestros términos, es un estudio en un nivel diferente, y ambas tareas tienen que hacerse. Este punto era apreciado por Marcus (1980), quien trató precisamente de como la teoría de Chomsky puede ser realizada y los tipos de restricciones sobre el poder del procesador gramatical humano que pueden dar origen a las restricciones estructurales en la sintaxis que Chomsky encontró. Igual aparece que la teoría de gramatical de "rastros" emergente (Chomsky y Lasnik, 1977) puede ofrecer una manera de sintetizar los dos enfoques mostrando que, por ejemplo, algunas de las restricciones más bien ad hoc que forman parte de la teoría computacional pueden ser consecuencias de las debilidades en el poder computacional que está disponible para implementar la descodificación sintáctica.