

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 864 149**

51 Int. Cl.:

G06N 3/08

(2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **06.03.2015 PCT/US2015/019236**

87 Fecha y número de publicación internacional: **11.09.2015 WO15134900**

96 Fecha de presentación y número de la solicitud europea: **06.03.2015 E 15757786 (7)**

97 Fecha y número de publicación de la concesión europea: **03.03.2021 EP 3114540**

54 Título: **Red neuronal y método de entrenamiento de red neuronal**

30 Prioridad:

06.03.2014 US 201461949210 P
22.01.2015 US 201562106389 P

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
13.10.2021

73 Titular/es:

PROGRESS, INC. (100.0%)
2252 Keylon Dr.
West Bloomfield, Michigan 48324, US

72 Inventor/es:

PESCIANSCHI, DMITRI

74 Agente/Representante:

SÁEZ MAESO, Ana

ES 2 864 149 T3

Aviso: En el plazo de nueve meses a contar desde la fecha de publicación en el Boletín Europeo de Patentes, de la mención de concesión de la patente europea, cualquier persona podrá oponerse ante la Oficina Europea de Patentes a la patente concedida. La oposición deberá formularse por escrito y estar motivada; sólo se considerará como formulada una vez que se haya realizado el pago de la tasa de oposición (art. 99.1 del Convenio sobre Concesión de Patentes Europeas).

DESCRIPCIÓN

Red neuronal y método de entrenamiento de red neuronal

5 Campo técnico

La divulgación se refiere a una red neuronal artificial y un método de entrenamiento de la misma.

Antecedentes

10 En el aprendizaje automático, las redes neuronales artificiales son una familia de algoritmos de aprendizaje estadístico inspirados en las redes neuronales biológicas, también conocidas como el sistema nervioso central de los animales, en particular el cerebro. Las redes neuronales artificiales se utilizan principalmente para estimar o aproximar funciones generalmente desconocidas que pueden depender de una gran cantidad de entradas. Dichas redes neuronales se han utilizado para una amplia variedad de tareas que son difíciles de resolver utilizando la programación ordinaria basada en reglas, que incluyen la visión por ordenador y el reconocimiento de voz.

15 Las redes neuronales artificiales se presentan en general como sistemas de "neuronas" que pueden calcular valores a partir de entradas y, como resultado de su naturaleza adaptativa, son capaces de aprendizaje automático, así como de reconocimiento de patrones. Cada neurona se conecta frecuentemente con varias entradas a través de sinapsis que tienen pesos sinápticos.

20 Las redes neuronales no se programan como software típico, sino que se entrenan. Dicho entrenamiento se logra normalmente mediante el análisis de un número suficiente de ejemplos representativos y mediante la selección estadística o algorítmica de pesos sinápticos, de modo que un conjunto dado de imágenes de entrada corresponda a un conjunto dado de imágenes de salida. Una crítica común a las redes neuronales clásicas es que con frecuencia se requieren mucho tiempo y otros recursos para su entrenamiento.

25 Se describen varias redes neuronales artificiales en las siguientes patentes de Estados Unidos: 4,979,124; 5,479,575; 5,493,688; 5,566,273; 5,682,503; 5,870,729; 7,577,631; y 7,814,038 Fashandi H. et al: "Face Detection Using CMAC Neural Network", 18 May 2004 (2004-05-18), Artificial Intelligence and Soft Computing - ICAISC 2004, SPRINGER-VERLAG, Berlin/Heidelberg, páginas 724 - 729, ISBN: 978-3-540-22123-4 divulga una red neuronal implementada por ordenador que está configurada para reconocer patrones en imágenes.

35 Resumen

La invención se expone en el conjunto de reivindicaciones adjuntas.

40 Las características y ventajas de la presente divulgación serán fácilmente evidentes a partir de la siguiente descripción detallada de la(s) realización(es) y el(los) mejor(es) modo(s) para llevar a cabo la divulgación descrita cuando se toma en relación con los dibujos acompañantes y las reivindicaciones adjuntas.

Breve descripción de los dibujos

45 La FIGURA 1 es una ilustración de una red neuronal clásica artificial de la técnica anterior.
La FIGURA 2 es una ilustración de una "red neuronal progresiva" (red p) que tiene una pluralidad de sinapsis, un conjunto de distribuidores, y una pluralidad de pesos correctivos asociados con cada sinapsis.
La FIGURA 3A es una ilustración de una porción de la red p mostrada en la Figura 2, que tiene una pluralidad de sinapsis y un peso sináptico posicionado en dirección ascendente de cada distribuidor.
50 La FIGURA 3B es una ilustración de una porción de la red p mostrada en la Figura 2, que tiene una pluralidad de sinapsis y un conjunto de pesos sinápticos posicionado en dirección descendente de la respectiva pluralidad de pesos correctivos.
La FIGURA 3C es una ilustración de una porción de la red p mostrada en la Figura 2, que tiene una pluralidad de sinapsis y un peso sináptico posicionado en dirección ascendente de cada distribuidor y un conjunto de pesos sinápticos posicionado en dirección descendente de la respectiva pluralidad de pesos correctivos.
55 La FIGURA 4A es una ilustración de una porción de la red p mostrada en la Figura 2, que tiene un único distribuidor para todas las sinapsis de una entrada dada y un peso sináptico posicionado en dirección ascendente de cada distribuidor.
La FIGURA 4B es una ilustración de una porción de la red p mostrada en la Figura 2, que tiene un único distribuidor para todas las sinapsis de una entrada dada y un conjunto de pesos sinápticos posicionado en dirección descendente de la respectiva pluralidad de pesos correctivos.
60 La FIGURA 4C es una ilustración de una porción de la red p mostrada en la Figura 2, que tiene un único distribuidor para todas las sinapsis de una entrada dada, y que tiene un peso sináptico posicionado en dirección ascendente de cada distribuidor y un conjunto de pesos sinápticos posicionado en dirección descendente de la respectiva pluralidad de pesos correctivos.
65

La FIGURA 5 es una ilustración de la división del rango de valor de señal de entrada en intervalos individuales en la red p mostrada en la Figura 2.

La FIGURA 6A es una ilustración de una realización de una distribución para los valores del coeficiente de impacto de pesos correctivos en la red p mostrada en la Figura 2.

5 La FIGURA 6B es una ilustración de otra realización de la distribución para los valores del coeficiente de impacto de pesos correctivos en la red p mostrada en la Figura 2.

La FIGURA 6C es una ilustración de aún otra realización de la distribución para los valores del coeficiente de impacto de pesos correctivos en la red p mostrada en la Figura 2.

10 La FIGURA 7 es una ilustración de una imagen de entrada para la red p mostrada en la Figura 2, así como una tabla correspondiente que representa la imagen en la forma de códigos digitales y otra tabla correspondiente que representa la misma imagen como un conjunto de respectivos intervalos.

La FIGURA 8 es una ilustración de una realización de la red p mostrada en la Figura 2 entrenada para reconocimiento de dos distintas imágenes, en las que la red p se configura para reconocer una foto que incluye algunas características de cada imagen;

15 La FIGURA 9 es una ilustración de una realización no reivindicada de la red p mostrada en la Figura 2 con un ejemplo de distribución de pesos sinápticos alrededor de una neurona "central".

La FIGURA 10 es una ilustración de una realización de la red p mostrada en la Figura 2, que representa una distribución uniforme de la desviación de entrenamiento entre pesos correctivos.

20 La FIGURA 11 es una ilustración de una realización de la red p mostrada en la Figura 2, que emplea la modificación de los pesos correctivos durante el entrenamiento de red p.

La FIGURA 12 es una ilustración de una realización de la red p mostrada en la Figura 2, en la que el algoritmo básico genera un conjunto primario de sumas de neuronas de salida, y en la que el conjunto generado se utiliza para generar varias sumas "ganadoras" con ya sea valores retenidos o aumentados y se niega la contribución de las sumas restantes.

25 La FIGURA 13 es una ilustración de una realización de la red p mostrada en la Figura 2 que reconoce una imagen compleja con elementos de imágenes múltiples.

La FIGURA 14 es una ilustración de un modelo de programación orientada a objetos para la red p mostrada en la Figura 2 utilizando Lenguaje de Modelado Unificado (UML).

La FIGURA 15 es una ilustración de una secuencia de formación general de la red p mostrada en la Figura 2.

30 La FIGURA 16 es una ilustración de análisis representativo y preparación de datos para la formación de la red p mostrada en la Figura 2.

La FIGURA 17 es una ilustración de la creación de entrada representativa que permite la interacción de la red p mostrada en la Figura 2 con los datos de entrada durante el entrenamiento y la aplicación de red p.

35 La FIGURA 18 es una ilustración de la creación representativa de unidades de neurona para la red p mostrada en la Figura 2.

La FIGURA 19 es una ilustración de la creación representativa de cada sinapsis conectada con las unidades de neurona.

La FIGURA 20 es una ilustración del entrenamiento de la red p mostrada en la Figura 2.

La FIGURA 21 es una ilustración del entrenamiento de la unidad de neurona en la red p mostrada en la Figura 2.

40 La FIGURA 22 es una ilustración de la extensión de las sumas de neuronas durante el entrenamiento de la red p mostrada en la Figura 2.

La FIGURA 23 es un diagrama de flujo de un método utilizado para entrenar la red neuronal que se muestra en las Figuras 2-22.

45 Descripción detallada

Una red 10 neuronal artificial clásica, como se muestra en la Figura 1, incluye normalmente dispositivos 12 de entrada, sinapsis 14 con pesos 16 sinápticos, neuronas 18, que incluyen un sumador 20 y un dispositivo 22 de función de activación, salidas 24 de neuronas y calculadora 26 de corrección de peso. Cada neurona 18 está conectada a través de sinapsis 14 a dos o más dispositivos 12 de entrada. Los valores de los pesos 16 sinápticos se representan comúnmente utilizando resistencia eléctrica, conductividad, voltaje, carga eléctrica, propiedad magnética u otros parámetros.

55 El entrenamiento supervisado de la red 10 neuronal clásica se basa generalmente en una aplicación de un conjunto de pares 28 de entrenamiento. Cada par 28 de entrenamiento consiste comúnmente en una imagen 28-1 de entrada y una imagen 28-2 de salida deseada, también conocida como una señal de supervisión. El entrenamiento de la red 10 neuronal clásica se proporciona normalmente como sigue. Una imagen de entrada en forma de un conjunto de señales de entrada (I_1-I_m) entra en los dispositivos 12 de entrada y se transfiere a los pesos 16 sinápticos con pesos iniciales (W_i). El valor de la señal de entrada se modifica por los pesos, normalmente al multiplicar o dividir el valor de cada señal (I_1-I_m) por el peso respectivo. A partir de los pesos 16 sinápticos, las señales de entrada modificadas se transfieren a las neuronas 18 respectivas. Cada neurona 18 recibe un conjunto de señales de un grupo de sinapsis 14 relacionadas con la neurona 18 del sujeto. El sumador 20 incluido en la neurona 18 suma todas las señales de entrada modificadas por los pesos y recibidas por la neurona del sujeto. Los dispositivos 22 de función de activación reciben las respectivas sumas de neuronas resultantes y modifican las sumas de acuerdo con la función o funciones matemáticas, formando de esta manera las respectivas imágenes de salida como conjuntos de señales de salida de neuronas ($\Sigma F_1... \Sigma F_n$).

La imagen de salida de neurona obtenida definida por las señales de salida de neurona ($\Sigma F_1 \dots \Sigma F_n$) se compara mediante una calculadora 26 de corrección de peso con imágenes de salida deseadas predeterminadas ($O_1 \dots O_n$). En base a la diferencia determinada entre la imagen de salida de la neurona obtenida ΣF_n y la imagen de salida deseada O_n , se forman señales de corrección para cambiar los pesos 16 sinápticos utilizando un algoritmo preprogramado. Después de realizadas las correcciones a todos los pesos 16 sinápticos, el conjunto de señales de entrada ($I_1 \dots I_m$) se reintroduce en la red 10 neuronal y se realizan nuevas correcciones. El ciclo anterior se repite hasta que se determina que la diferencia entre la imagen de salida de la neurona obtenida ΣF_n y la imagen de salida deseada O_n es menor que algún error predeterminado. Un ciclo de entrenamiento en red con todas las imágenes individuales se identifica normalmente como una "iteración de entrenamiento". Generalmente, con cada iteración de entrenamiento, se reduce la magnitud del error. Sin embargo, dependiendo del número de entradas individuales ($I_1 \dots I_m$), así como del número de entradas y salidas, el entrenamiento de la red 10 neuronal clásica puede requerir un número significativo de iteraciones de entrenamiento, que, en algunos casos, puede llegar a cientos de miles.

Existe una variedad de redes neuronales clásicas, que incluyen la red Hopfield, la Máquina de Boltzmann Restrictada, la red de función de base radial y la red neuronal recurrente. Las tareas específicas de clasificación y agrupamiento requieren un tipo específico de red neuronal, los Mapas Autoorganizados que utilizan solo imágenes de entrada como información de entrenamiento de entrada de red, mientras que la imagen de salida deseada, correspondiente a una determinada imagen de entrada, se forma directamente durante el proceso de entrenamiento basado en una sola neurona ganadora que tiene una señal de salida con el valor máximo.

Como se señaló anteriormente, una de las principales preocupaciones con las redes neuronales clásicas existentes, tales como la red 10 neuronal, es que el entrenamiento exitoso de las mismas puede requerir una duración significativa de tiempo. Algunas preocupaciones adicionales con las redes clásicas pueden ser un gran consumo de recursos informáticos, lo que a su vez impulsaría la necesidad de ordenadores potentes. Preocupaciones adicionales son la incapacidad de aumentar el tamaño de la red sin un reentrenamiento completo de la red, una predisposición a fenómenos como la "parálisis de la red" y la "congelación en un mínimo local", que hacen imposible predecir si una red neuronal específica es capaz de ser entrenado con un conjunto dado de imágenes en una secuencia determinada. También puede haber limitaciones relacionadas con la secuenciación específica de imágenes que se introducen durante el entrenamiento, donde cambiar el orden de introducción de las imágenes de entrenamiento puede conducir a congelaciones de la red, así como a la imposibilidad de realizar entrenamiento adicional de una red ya entrenada.

Con referencia a los dibujos restantes, en los que números de referencia similares se refieren a componentes similares, la Figura 2 muestra una vista esquemática de una red neuronal progresiva, en adelante "red progresiva" o "red p" 100. La red 100 p incluye una pluralidad o un conjunto de entradas 102 de la red p. Cada entrada 102 se configura para recibir una señal 104 de entrada, en la que las señales de entrada se representan como $I_1, I_2 \dots I_m$ en la Figura 2. Cada señal de entrada $I_i, I_2 \dots I_m$ representa un valor de algunas características de una imagen 106 de entrada, por ejemplo, una magnitud, frecuencia, fase, ángulo de polarización de señal o asociación con diferentes partes de la imagen 106 de entrada. Cada señal 104 de entrada tiene un valor de entrada, en el que juntas la pluralidad de señales 104 de entrada describe generalmente la imagen 106 de entrada.

Cada valor de entrada puede estar dentro de un rango de valores que se encuentra entre $-\infty$ y $+\infty$ y se puede establecer en formas digitales y/o analógicas. El rango de los valores de entrada puede depender de un conjunto de imágenes de entrenamiento. En el caso más simple, los valores de entrada de rango podrían ser la diferencia entre los valores más pequeños y más grandes de las señales de entrada para todas las imágenes de entrenamiento. Por razones prácticas, el rango de los valores de entrada se puede limitar al eliminar los valores de entrada que se consideran demasiado altos. Por ejemplo, tal limitación del rango de los valores de entrada se puede lograr mediante métodos estadísticos conocidos para la reducción de la varianza, tal como el muestreo de importancia. Otro ejemplo de limitación del rango de los valores de entrada puede ser la designación de todas las señales que son inferiores a un nivel mínimo predeterminado a un valor mínimo específico y la designación de todas las señales que exceden un nivel máximo predeterminado a un valor máximo específico.

La red 100 p también incluye una pluralidad o un conjunto de sinapsis 118. Cada sinapsis 118 se conecta a una de la pluralidad de entradas 102, incluye una pluralidad de pesos 112 correctivos, y también pueden incluir un peso 108 sináptico, como se muestra en la Figura 2. Cada peso 112 correctivo se define por un valor 112 de peso respectivo. La red 100 p también incluye un conjunto de distribuidores 114. Cada distribuidor 114 se conecta de forma operativa a una de la pluralidad de entradas 102 para recibir la respectiva señal 104 de entrada. Adicionalmente, cada distribuidor 114 se configura para seleccionar uno más pesos correctivos de la pluralidad de pesos 112 correctivos en correlación con el valor de entrada.

La red 100 p adicionalmente incluye un conjunto de neuronas 116. Cada neurona 116 tiene al menos una salida 117 y se conecta con al menos una de la pluralidad de entradas 102 a través de una sinapsis 118. Cada neurona 116 se configura para agregar o sumar los valores de pesos correctivos de los pesos 112 correctivos seleccionados de cada sinapsis 118 conectada a la respectiva neurona 116 y de esta manera generar y emitir una suma 120 de neuronas, de otra forma designado como Σn . Se puede utilizar un distribuidor 114 separado para cada sinapsis 118 de una entrada 102 dada, como se muestra en las Figuras 3A, 3B y 3C, o se puede utilizar un solo distribuidor para todas estas

sinapsis, como se muestra en las Figuras 4A, 4B y 4C. Durante la formación o configuración de la red 100 p, a todos los pesos 112 correctivos se les asignan valores iniciales, que pueden cambiar durante el proceso de entrenamiento de red p. El valor inicial del peso 112 correctivo se puede asignar como en la red 10 neuronal clásica, por ejemplo, los pesos se pueden seleccionar aleatoriamente, calcular con la ayuda de una función matemática predeterminada, seleccionar de una plantilla predeterminada, etc.

La red 100 p también incluye una calculadora 122 de corrección de peso. La calculadora 122 de corrección de peso se configura para recibir una señal 124 de salida deseada, es decir, predeterminada, que tiene un valor de señal y que representa una porción de una imagen 126 de salida. La calculadora 122 de corrección de peso también se configura para determinar una desviación 128 de la suma 120 de neuronas a partir del valor de la señal 124 de salida deseada, también conocida como error de entrenamiento, y modificar los respectivos valores de pesos correctivos utilizando la desviación 128 determinada. A continuación, sumando los valores de pesos correctivos modificado para determinar la suma 120 de neuronas se minimiza la desviación de la suma de neuronas del sujeto del valor de la señal 124 de salida deseada y, como resultado, es eficaz para entrenar la red 100 p.

Por analogía con la red 10 clásica discutida con respecto a la Figura 1, la desviación 128 también se puede describir como el error de entrenamiento entre la suma 120 determinada de neuronas y el valor de la señal 124 de salida deseada. En comparación con la red 10 neuronal clásica discutida con respecto a la Figura 1, en la red 100 p los valores de entrada de la señal 104 de entrada solo cambian en el proceso de configuración general de la red, y no se cambian durante el entrenamiento de la red p. En lugar de cambiar el valor de entrada, el entrenamiento de la red 100 p se proporciona al cambiar los valores 112 de los pesos 112 correctivos. Adicionalmente, aunque cada neurona 116 incluye una función de suma, en la que la neurona suma los valores del peso correctivo, la neurona 116 no requiere, y de hecho, se caracteriza por la ausencia de una función de activación, tal como la que proporciona el dispositivo 22 de función de activación en la red 10 neuronal clásica.

En la red 10 neuronal clásica, la corrección de peso durante el entrenamiento se logra al cambiar los pesos 16 sinápticos, mientras que en la red 100 p, se proporciona la corrección de peso correspondiente al cambiar los valores 112 de los pesos correctivos, como se muestra en la Figura 2. El correctivo respectivo Los pesos 112 se pueden incluir en los bloques 110 de corrección de peso colocados en todas o algunas de las sinapsis 118. En las emulaciones informáticas de redes neuronales, cada peso sináptico y correctivo puede estar representado ya sea por un dispositivo digital, como una celda de memoria, y/o por un dispositivo analógico. En las emulaciones de software de redes neuronales, los valores de los pesos 112 correctivos se pueden proporcionar mediante un algoritmo programado apropiado, mientras que en las emulaciones de hardware, se podrían utilizar métodos conocidos para el control de la memoria.

En la red 100 p, la desviación 128 de la suma 120 de neuronas de la señal 124 de salida deseada se puede representar como una diferencia calculada matemáticamente entre ellos. Adicionalmente, la generación de los respectivos pesos 112 correctivos modificados puede incluir la repartición de la diferencia calculada para cada peso correctivo utilizado para generar la suma 120 de neuronas. En dicha realización no reivindicada, la generación de los respectivos pesos 112 correctivos modificados permitirá que la suma 120 de neuronas converja en el valor de señal de salida deseado dentro de un pequeño número de iteraciones, en algunos casos necesitando solo una iteración, para entrenar rápidamente la red 100 p. En un caso específico, la repartición de la diferencia matemática entre los pesos 112 correctivos utilizado para generar la suma 120 de neuronas puede incluir dividir la diferencia determinada por igual entre cada peso correctivo utilizado para generar la suma 120 de neuronas respectiva.

En una realización separada, la determinación de la desviación 128 de la suma 120 de neuronas del valor de señal de salida deseada puede incluir la división del valor de señal de salida deseada por la suma de neuronas para generar de esta manera un coeficiente de desviación. En dicho caso específico, la modificación de los pesos 112 correctivos modificados respectivos incluye la multiplicación de cada peso correctivo utilizado para generar la suma 120 de neuronas por el coeficiente de desviación. Cada distribuidor 114 se puede configurar adicionalmente para asignar una pluralidad de coeficientes de impacto 134 a la pluralidad de pesos 112 correctivos. En la presente realización, cada coeficiente de impacto 134 se puede asignar a una de la pluralidad de pesos 112 correctivos en alguna proporción predeterminada para generar la respectiva suma 120 de neuronas. Para correspondencia con cada respectivo peso 112 correctivo, cada coeficiente de impacto 134 se puede asignar a una nomenclatura " $C_{i,d,n}$ ", como se muestra en las Figuras.

Cada una de la pluralidad de coeficientes de impacto 134 que corresponde a la sinapsis 118 específica se define por una respectiva función 136 de distribución de impacto. La función 136 de distribución de impacto puede ser la misma, ya sea para todos los coeficientes de impacto 134 o solo para la pluralidad de coeficientes de impacto 134 que corresponden a una sinapsis 118 específica. Se puede recibir cada una de la pluralidad de valores de entrada en un rango 138 de valor dividido en intervalos o subdivisiones "d" de acuerdo con una función de distribución del intervalo 140, de tal manera que cada valor de entrada se recibe dentro de un respectivo intervalo "d" y cada peso correctivo corresponde a uno de dichos intervalos. Cada distribuidor 114 puede utilizar el respectivo valor de entrada recibido para seleccionar el respectivo intervalo "d", y para asignar la respectiva pluralidad de coeficientes de impacto 134 al peso 112 correctivo que corresponde al respectivo intervalo seleccionado "d" y a al menos un peso correctivo que corresponde a un intervalo adyacente al respectivo intervalo seleccionado, tal como $W_{i,d+1,n}$ o $W_{i,d-1,n}$. En otro ejemplo

no limitante, la proporción predeterminada de los coeficientes de impacto 134 se puede definir de acuerdo con una distribución estadística.

5 Generar la suma 120 de neuronas puede incluir asignar inicialmente los respectivos coeficientes de impacto 134 a cada peso 112 correctivo de acuerdo con el valor 102 de entrada y luego multiplicar los coeficientes de impacto del sujeto por valores de los respectivos pesos 112 correctivos empleados. Luego, sumar mediante cada neurona 116 los productos individuales del peso 112 correctivo y el coeficiente asignado de impacto 134 para todas las sinapsis 118 conectadas a la misma.

10 Se puede configurar la calculadora 122 de corrección de peso para aplicar los respectivos coeficientes de impacto 134 para generar los pesos 112 correctivos modificados respectivos. Específicamente, la calculadora 122 de corrección de peso puede aplicar una porción de la diferencia matemática calculada entre la suma 120 de neuronas y la señal 124 de salida deseada a cada peso 112 correctivo utilizado para generar la suma 120 de neuronas de acuerdo con la proporción establecida por los respectivos coeficientes de impacto 134. Adicionalmente, la diferencia matemática dividida entre los pesos 112 correctivos utilizados para generar la suma 120 de neuronas se puede dividir adicionalmente por el respectivo coeficiente de impacto 134. Posteriormente, el resultado de la división de la suma 120 de neuronas por el respectivo coeficiente de impacto 134 se puede agregar al peso 112 correctivo con el fin converger la suma 120 de neuronas en el valor de señal de salida deseada.

20 Normalmente, la formación de la red 100 p tendrá lugar antes de que comience el entrenamiento de la red p. Sin embargo, en una realización separada no reivindicada, si durante el entrenamiento la red 100 p recibe una señal 104 de entrada para la cual están ausentes los pesos correctivos iniciales, se pueden generar los pesos 112 correctivos apropiados. En tal caso, el distribuidor 114 específico determinará el intervalo "d" apropiado para la señal 104 de entrada particular, y se generará un grupo de pesos 112 correctivos con valores iniciales para la entrada 102 dada, el intervalo "d" dado, y todas las neuronas 116 respectivas. Adicionalmente, se puede asignar un coeficiente de impacto 134 correspondiente a cada peso 112 correctivo recién generado.

30 Cada peso 112 correctivo se puede definir por un conjunto de índices configurados para identificar una posición de cada respectivo peso correctivo sobre la red 100 p. El conjunto de índices puede incluir específicamente un índice de entrada "i" configurado para identificar el peso 112 correctivo que corresponde a la entrada 102 específica, un índice de intervalo "d" configurado para especificar el intervalo seleccionado discutido anteriormente para el respectivo peso correctivo, y un índice de neurona "n" configurado para especificar el peso 112 correctivo que corresponde a la neurona 116 específica con nomenclatura " $W_{i,d,n}$ ". Por tanto, a cada peso 112 correctivo correspondiente a una entrada 102 específica se le asigna el índice específico "i" en el subíndice para indicar la posición del sujeto. De manera similar, a cada peso correctivo "W" correspondiente a una neurona 116 específica y una sinapsis 118 respectiva se le asignan los índices específicos "n" y "d" en el subíndice para indicar la posición del sujeto del peso correctivo en la red 100 p. El conjunto de índices también puede incluir un índice de acceso "a" configurado para contar un número de veces que se accede al respectivo peso 112 correctivo mediante la señal 104 de entrada durante el entrenamiento de la red 100 p. En otras palabras, cada vez un intervalo específico "d" y el respectivo peso 112 correctivo se selecciona para entrenamiento de la pluralidad de pesos correctivos en correlación con el valor de entrada, el índice de acceso "a" se incrementa para contar la señal de entrada. El índice de acceso "a" se puede utilizar para especificar o definir además un estado actual de cada peso correctivo al adoptar una nomenclatura " $W_{i,d,n,a}$ ". Cada uno de los índices "i", "d", "n" y "a" pueden ser valores numéricos en el rango de 0 a $+\infty$.

45 En la Figura 5 se muestran varias posibilidades de dividir el rango de señales 104 de entrada en intervalos $d_0, d_1 \dots d_m$. La distribución de intervalo específica puede ser uniforme o lineal, lo que, por ejemplo, se puede lograr especificando todos los intervalos "d" con el mismo tamaño. Se puede considerar que todas las señales 104 de entrada que tienen su valor de señal de entrada respectivo menor que un nivel más bajo predeterminado tienen un valor cero, mientras que todas las señales de entrada que tienen su valor de señal de entrada respectivo mayor que un nivel más alto predeterminado se pueden asignar a dicho nivel más alto, como también como se muestra en la Figura 5. La distribución de intervalo específico también puede ser no uniforme o no lineal, tal como simétrica, asimétrica o ilimitada. La distribución no lineal de los intervalos "d" puede ser útil cuando el rango de las señales 104 de entrada se considera impracticablemente grande, y una cierta parte del rango podría incluir señales de entrada consideradas más críticas, como al principio, en el medio o al final del rango. La distribución de intervalo específico también se puede describir mediante una función aleatoria. Todos los ejemplos anteriores son de naturaleza no limitante, ya que también son posibles otras variantes de distribución de intervalos.

60 El número de intervalos "d" dentro del rango seleccionado de señales 104 de entrada se puede aumentar para optimizar la red 100 p. Dicha optimización de la red 100 p puede ser deseable, por ejemplo, con el aumento en la complejidad del entrenamiento de las imágenes 106 de entrada. Por ejemplo, puede ser necesario un mayor número de intervalos para imágenes multicolores en comparación con imágenes monocromáticas, y puede ser necesario un mayor número de intervalos para ornamentos complejos que para gráficos simples. Puede ser necesario un mayor número de intervalos para el reconocimiento preciso de imágenes con gradientes de color complejos en comparación con las imágenes descritas por contornos, así como para un mayor número total de imágenes de entrenamiento. También puede ser necesaria una reducción en el número de intervalos "d" en casos con una gran magnitud de ruido, una gran variación en las imágenes de entrenamiento y un consumo excesivo de recursos informáticos.

Dependiendo de la tarea o tipo de información manejada por la red 100 p, por ejemplo, se pueden asignar datos visuales o textuales, datos de sensores de diversa naturaleza, diferente número de intervalos y el tipo de distribución de los mismos. Para cada intervalo de valor de señal de entrada "d", se puede asignar un peso correctivo correspondiente de la sinapsis dada con el índice "d". Por tanto, un cierto intervalo "d" incluirá todos los pesos 112 correctivos con el índice "i" relevante para la entrada dada, el índice "d" relevante para el intervalo dado; y todos los valores para el índice "n" desde 0 hasta n. En el proceso de entrenamiento de la red 100 p, el distribuidor 114 define cada valor de la señal de entrada y, por tanto, relaciona la señal de entrada del sujeto 104 con el intervalo "d" correspondiente. Por ejemplo, si hay 10 intervalos iguales "d" dentro del rango de señales de entrada desde 0 hasta 100, la señal de entrada que tenga un valor entre 30 y 40 estará relacionada con el intervalo 3, es decir, "d" = 3.

Para todos los pesos 112 correctivos de cada sinapsis 118 conectada con la entrada 102 dada, el distribuidor 114 puede asignar valores del coeficiente de impacto 134 de acuerdo con el intervalo "d" relacionado con la señal de entrada particular. El distribuidor 114 también puede asignar valores del coeficiente de impacto 134 de acuerdo con una distribución predeterminada de valores del coeficiente de impacto 134 (mostrado en la Figura 6), tal como una curva de distribución sinusoidal, normal, logarítmica o una función de distribución aleatoria. En muchos casos, la suma o integral del coeficiente de impacto 134 o $C_{i,d,n}$ para una señal 102 de entrada específica relacionada con cada sinapsis 118 tendrá un valor de 1 (uno).

$$\sum_{Sinapsis} C_{i,d,n} = 1 \quad \text{o} \quad \int_{Sinapsis} C_{i,d,n} = 1 [1]$$

En el caso más simple, se puede asignar el peso 112 correctivo que corresponde más estrechamente al valor de señal de entrada un valor de 1 (uno) al coeficiente de impacto 134 ($C_{i,d,n}$), mientras que los pesos correctivos para otros intervalos pueden recibir un valor de 0 (cero).

La red 100 p se centra en la reducción de la duración del tiempo y el uso de otros recursos durante el entrenamiento de la red p, en comparación con la red 10 neuronal clásica. Aunque algunos de los elementos divulgados en este documento como parte de la red 100 p están designados por ciertos nombres o identificadores conocidos por aquellos familiarizados con las redes neuronales clásicas, los nombres específicos se utilizan para simplificar y se pueden emplear de manera diferente a sus contrapartes en las redes neuronales clásicas. Por ejemplo, los pesos 16 sinápticos que controlan las magnitudes de las señales de entrada (I_1-I_m) se establecen durante el proceso de configuración general de la red 10 neuronal clásica y se cambian durante el entrenamiento de la red clásica. Por otro lado, el entrenamiento de la red 100 p se logra al cambiar los pesos 112 correctivos, mientras que los pesos 108 sinápticos no cambian durante el entrenamiento. Adicionalmente, como se discutió anteriormente, cada una de las neuronas 116 incluye un componente que suma o sumador, pero no incluye un dispositivo 22 de función de activación que es típico de la red 10 neuronal clásica.

En general, se entrena la red 100 p al entrenar cada unidad 119 de neurona que incluye una neurona 116 respectiva y todas las sinapsis 118 de conexión, que incluyen la neurona particular y todas las respectivas sinapsis 118 y pesos 112 de corrección conectados con la neurona del sujeto. De acuerdo con lo anterior, el entrenamiento de la red 100 p incluye cambiar los pesos 112 correctivos que contribuyen a la neurona 116 respectiva. Los cambios en los pesos 112 correctivos tienen lugar en base a un algoritmo de entrenamiento grupal incluido en un método 200 que se divulga en detalle a continuación. En el algoritmo divulgado, el error de entrenamiento, es decir, la desviación 128, se determina para cada neurona, en base a los valores de corrección que se determinan y asignan a cada uno de los pesos 112 utilizados para determinar la suma obtenida por cada neurona 116 respectiva. La introducción de dichos valores de corrección durante el entrenamiento está destinada a reducir las desviaciones 128 para la neurona 116 del sujeto a cero. Durante el entrenamiento con imágenes adicionales, pueden volver a aparecer nuevos errores relacionados con las imágenes utilizadas anteriormente. Para eliminar dichos errores adicionales, después de completar una iteración de entrenamiento, se pueden calcular los errores para todas las imágenes de entrenamiento de toda la red 100 p, y si dichos errores son mayores que los valores predeterminados, se pueden realizar una o más iteraciones de entrenamiento adicionales hasta que los errores sean menores que un valor objetivo o predeterminado.

La Figura 23 representa el método 200 de entrenamiento de la red 100 p, como se describió anteriormente con respecto a las Figuras 2-22. El método 200 comienza en la trama 202 donde el método incluye recibir, a través de la entrada 102, la señal 104 de entrada que tiene el valor de entrada. Después de la trama 202, el método avanza a la trama 204. En la trama 204, el método incluye comunicar la señal 104 de entrada al distribuidor 114 conectado operativamente a la entrada 102. Ya sea en la trama 202 o la trama 204, el método 200 puede incluir definir cada peso 112 correctivo por el conjunto de índices. Como se describió anteriormente con respecto a la estructura de la red 100 p, el conjunto de índices puede incluir el índice de entrada "i" configurado para identificar el peso 112 correctivo que corresponde a la entrada 102. El conjunto de índices también puede incluir el índice de intervalo "d" configurado para especificar el intervalo seleccionado para el respectivo peso 112 correctivo, y el índice de neurona "n" configurado para especificar el peso 112 correctivo que corresponde a la neurona 116 específica como " $W_{i,d,n}$ ". El conjunto de índices pueden incluir adicionalmente el índice de acceso "a" configurado para contar el número veces que el

respectivo peso 112 correctivo es accesado por la señal 104 de entrada durante el entrenamiento de la red 100 p. De acuerdo lo anterior, el presente estado de cada peso correctivo puede adoptar la nomenclatura " $W_{i,d,n,a}$ ".

Después de la trama 204, el método pasa a la trama 206, en la que el método incluye seleccionar, a través del distribuidor 114, en correlación con el valor de entrada, uno más pesos 112 correctivos de la pluralidad de pesos correctivos ubicada sobre la sinapsis 118 conectada a la entrada 102 del sujeto. Como se describió anteriormente, cada peso 112 correctivo se define por su valor de peso respectivo. En la trama 206 el método puede incluir adicionalmente asignar, a través del distribuidor 114, la pluralidad de coeficientes de impacto 134 a la pluralidad de pesos 112 correctivos. En la trama 206 el método también puede incluir asignar cada coeficiente de impacto 134 a una de la pluralidad de pesos 112 correctivos en una proporción predeterminada para generar la suma 120 de neuronas. También, en la trama 206 el método puede incluir agregar, a través de la neurona 116, un producto del peso 112 correctivo y el coeficiente asignado de impacto 134 para todas las sinapsis 118 conectadas a esta. Adicionalmente, en la trama 206 el método puede incluir aplicar, a través de la calculadora 122 de corrección de peso, una porción de la diferencia determinada a cada peso 112 correctivo utilizado para generar la suma 120 de neuronas de acuerdo con la proporción establecida por el respectivo coeficiente de impacto.

Como se describió anteriormente con respecto a la estructura de la red 100 p, se puede definir la pluralidad de coeficientes de impacto 134 por una función 136 de distribución de impacto. En dicho caso, el método adicionalmente puede incluir recibir el valor de entrada en el rango 138 de valor dividido en intervalos "d" de acuerdo con la función de distribución del intervalo 140, de tal manera que el valor de entrada se recibe dentro de un respectivo intervalo, y cada peso 112 correctivo corresponde a uno de los intervalos. También, el método puede incluir utilizar, a través del distribuidor 114, el valor de entrada recibido para seleccionar el respectivo intervalo "d" y asignar la pluralidad de coeficientes de impacto 134 al peso 112 correctivo que corresponde al respectivo intervalo seleccionado "d" y a al menos un peso correctivo que corresponde a un intervalo adyacente al respectivo intervalo seleccionado "d". Como se describió anteriormente con respecto a la estructura de la red 100 p, se pueden identificar los pesos 112 correctivos que corresponden a un intervalo adyacente al respectivo intervalo seleccionado "d", por ejemplo, como $W_{i,d+1,n}$ o $W_{i,d-1,n}$.

Luego de la trama 206, el método avanza a la trama 208. En la trama 208, el método incluye agregar los valores de peso de los pesos 112 correctivos seleccionados por la neurona 116 específica conectada con la entrada 102 a través de la sinapsis 118 para generar la suma 120 de neuronas. Como se describió anteriormente con respecto a la estructura de la red 100 p, cada neurona 116 incluye al menos una salida 117. Después de la trama 208, el método procede a la trama 210, en la que el método incluye recibir, a través de la calculadora 122 de corrección de peso, la señal 124 de salida deseada que tiene el valor de señal. Luego de la trama 210, el método avanza a la trama 212 en la que el método incluye determinar, a través de la calculadora 122 de corrección de peso, la desviación 128 de la suma 120 de neuronas a partir del valor de la señal 124 de salida deseada.

Como se divulgó anteriormente en la descripción de la red 100 p, la determinación de la desviación 128 de la suma 120 de neuronas del valor de señal de salida deseada puede incluir determinar la diferencia matemática entre ellos. Adicionalmente, la modificación de los pesos 112 correctivos respectivos puede incluir la repartición de la diferencia matemática a cada peso correctivo utilizado para generar la suma 120 de neuronas. Alternativamente, la repartición de la diferencia matemática puede incluir dividir la diferencia determinada por igual entre cada peso 112 correctivo utilizado para generar la suma 120 de neuronas. En una realización aún separada, la determinación de la desviación 128 también puede incluir dividir el valor de la señal 124 de salida deseada por la suma 120 de neuronas para generar de esta manera el coeficiente de desviación. Adicionalmente, en dicho caso, la modificación de los respectivos pesos 112 correctivos puede incluir multiplicar cada peso 112 correctivo utilizado para generar la suma 120 de neuronas por el coeficiente de desviación generado.

Después de la trama 212, el método procede a la trama 214. En la trama 214 el método incluye modificar, a través de la calculadora 122 de corrección de peso, los valores de pesos correctivos respectivos utilizando la desviación 128 determinada. Los valores de pesos correctivos modificados posteriormente se pueden agregar o sumar y luego utilizar para determinar una nueva suma 120 de neuronas. Los valores de pesos correctivos modificados sumados luego pueden servir para minimizar la desviación de la suma 120 de neuronas a partir del valor de la señal 124 de salida deseada y de esta manera entrenar la red 100 p. Después de la trama 214, el método 200 puede incluir volver a la trama 202 para realizar iteraciones de entrenamiento adicionales hasta que la desviación de la suma 120 de neuronas del valor de la señal 124 de salida deseada se minimice suficientemente. En otras palabras, se pueden realizar iteraciones de entrenamiento adicionales para hacer converger la suma 120 de neuronas en la señal 124 de salida deseada dentro de la desviación predeterminada o valor de error, de modo que la red 100 p se puede considerar entrenada y lista para funcionar con nuevas imágenes.

Generalmente, las imágenes 106 de entrada necesitan ser preparadas para el entrenamiento de la red 100 p. La preparación de la red 100 p para el entrenamiento generalmente comienza con la formación de un conjunto de imágenes de entrenamiento, que incluye las imágenes 106 de entrada y, en la mayoría de los casos, las imágenes 126 de salida deseadas corresponden a las imágenes de entrada del sujeto. Las imágenes 106 de entrada (mostradas en la Figura 2) definidas por las señales de entrada I_1, I_2, \dots, I_m para el entrenamiento de la red 100 p se seleccionan de acuerdo con las tareas que la red p asigna para manejar, por ejemplo, el reconocimiento de imágenes humanas u

otros objetos, reconocimiento de determinadas actividades, agrupamiento o clasificación de datos, análisis de datos estadísticos, reconocimiento de patrones, previsión o control de determinados procesos. De acuerdo con lo anterior, las imágenes 106 de entrada se pueden presentar en cualquier formato adecuado para introducción en un ordenador, por ejemplo, utilizando formatos jpeg, gif o pptx, en forma de tablas, cuadros, diagramas y gráficos, varios formatos de documentos o un conjunto de símbolos.

La preparación para el entrenamiento de la red 100 p también puede incluir la conversión de las imágenes 106 de entrada seleccionadas para su unificación que sea conveniente para el procesamiento de las imágenes del sujeto por la red 100 p, por ejemplo, transformando todas las imágenes en un formato que tiene el mismo número de señales o, en el caso de imágenes, el mismo número de píxeles. Las imágenes en color se podrían presentar, por ejemplo, como una combinación de tres colores básicos. La conversión de imágenes también podría incluir la modificación de características, por ejemplo, cambiar una imagen en el espacio, cambiar las características visuales de la imagen, como resolución, brillo, contraste, colores, punto de vista, perspectiva, distancia focal y punto focal, así como agregar símbolos, números o notas.

Después de la selección del número de intervalos, una imagen de entrada específica se puede convertir en una imagen de entrada en formato de intervalo, es decir, los valores de señal reales se pueden registrar como números de intervalos a los que pertenecen las señales respectivas del sujeto. Este procedimiento se puede realizar en cada iteración de entrenamiento para la imagen dada. Sin embargo, la imagen también se puede formar una vez como un conjunto de números de intervalo. Por ejemplo, en la Figura 7 la imagen inicial se presenta como una imagen, mientras que en la tabla "Imagen en formato digital" la misma imagen se presenta en forma de códigos digitales, y en la tabla "Imagen en formato de intervalo" entonces la imagen es presentado como un conjunto de números de intervalo, donde se asigna un intervalo separado para cada 10 valores de códigos digitales.

La estructura descrita de la red 100 p y el algoritmo o método de entrenamiento 200 como se describe permite el entrenamiento continuo o iterativo de la red p, por lo que no es necesario formar un conjunto completo de imágenes 106 de entrada de entrenamiento 106 al comienzo del proceso de entrenamiento. Es posible formar un conjunto inicial relativamente pequeño de imágenes de entrenamiento, y dicho conjunto inicial podría ampliarse según sea necesario. Las imágenes 106 de entrada se pueden dividir en distintas categorías, por ejemplo, un conjunto de imágenes de una persona, un conjunto de fotografías de gatos o un conjunto de fotografías de coches, de modo que cada categoría corresponda a una única imagen de salida, tal como el nombre de la persona o una etiqueta específica. Las imágenes 126 de salida deseadas representan un campo o tabla de digital, donde cada punto corresponde a un valor numérico específico de $-\infty$ a $+\infty$, o valores analógicos. Cada punto de la imagen 126 de salida deseada puede corresponder a la salida de una de las neuronas de la red 100 p. Las imágenes 126 de salida deseadas se pueden codificar con códigos digitales o analógicos de imágenes, tablas, texto, fórmulas, conjuntos de símbolos, tales como códigos de barras o sonidos.

En el caso más simple, cada imagen 106 de entrada puede corresponder a una imagen de salida, que codifica la imagen de entrada del sujeto. A uno de los puntos de dicha imagen de salida se le puede asignar un valor máximo posible, por ejemplo, 100%, mientras que a todos los demás puntos se le puede asignar un valor mínimo posible, por ejemplo, cero. En tal caso, tras el entrenamiento, se habilitará el reconocimiento probabilístico de varias imágenes en forma de porcentaje de similitud con las imágenes de entrenamiento. La Figura 8 muestra un ejemplo de cómo la red 100 p entrenada para el reconocimiento de dos imágenes, un cuadrado y un círculo, puede reconocer una imagen que contiene algunas características de cada figura que se expresan en porcentajes, con la suma no necesariamente igual al 100 %. Dicho proceso de reconocimiento de patrones al definir el porcentaje de similitud entre diferentes imágenes utilizadas para el entrenamiento se puede utilizar para clasificar imágenes específicas.

Para mejorar la precisión y excluir errores, la codificación se puede lograr utilizando un conjunto de varias salidas neuronales en lugar de una salida (ver más abajo). En el caso más simple, las imágenes de salida se pueden preparar antes del entrenamiento. Sin embargo, también es posible tener las imágenes de salida formadas por la red 100 p durante el entrenamiento.

En la red 100 p, también existe la posibilidad de invertir las imágenes de entrada y salida. En otras palabras, las imágenes 106 de entrada pueden tener la forma de un campo o tabla de valores digitales o analógicos, donde cada punto corresponde a una entrada de la red p, mientras que las imágenes de salida se pueden presentar en cualquier formato adecuado para introducción en el ordenador, por ejemplo, utilizando formatos jpeg, gif, pptx, en forma de tablas, cuadros, diagramas y gráficos, varios formatos de documentos o un conjunto de símbolos. La red 100 p resultante puede ser muy adecuada para sistemas de archivo, así como para una búsqueda asociativa de imágenes, expresiones musicales, ecuaciones o conjuntos de datos.

Después de la preparación de las imágenes 106 de entrada, normalmente se debe formar la red 100 p y/o se deben establecer los parámetros de una red p existente para manejar las tareas dadas. La formación de la red 100 p puede incluir las siguientes designaciones:

- dimensiones de la red 100 p, según se define por el número de entradas y salidas;
- pesos 108 sinápticos para todas las entradas;

- número de pesos 112 correctivos;
- distribución de coeficientes de impacto de peso correctivo ($C_{i,d,n}$) para diferentes valores de las señales 104 de entrada; y
- precisión de entrenamiento deseada

El número de entradas se determina en base a los tamaños de las imágenes 106 de entrada. Por ejemplo, se puede utilizar una serie de píxeles para las imágenes, mientras que el número seleccionado de salidas puede depender del tamaño de las imágenes 126 de salida deseadas. En algunos casos, el número seleccionado de resultados puede depender del número de categorías de imágenes de entrenamiento.

Los valores de los pesos 108 sinápticos individuales pueden estar en el intervalo de $-\infty$ a $+\infty$. Los valores de pesos 108 sinápticos que son menores que 0 (cero) pueden denotar la amplificación de la señal, que se puede utilizar para mejorar el impacto de señales de entradas específicas, o de imágenes específicas, por ejemplo, para un reconocimiento más efectivo de rostros humanos en fotos que contienen una gran cantidad de individuos u objetos diferentes. Por otro lado, los valores de pesos 108 sinápticos que son mayores que 0 (cero) se pueden utilizar para denotar la atenuación de la señal, que se puede utilizar para reducir el número de cálculos requeridos y aumentar la velocidad operativa de la red 100 p. Generalmente, cuanto mayor es el valor del peso sináptico, más atenuada es la señal transmitida a la neurona correspondiente. Si todos los pesos 108 sinápticos correspondientes a todas las entradas son iguales y todas las neuronas están igualmente conectadas con todas las entradas, la red neuronal se volverá universal y será más eficaz para tareas comunes, como cuando se sabe muy poco sobre la naturaleza de las imágenes en avance. Sin embargo, dicha estructura generalmente aumentará el número de cálculos requeridos durante el entrenamiento y operación.

La Figura 9 muestra una realización no reivindicada de la red 100 p en la que la relación entre una entrada y las neuronas respectivas se reduce de acuerdo con la distribución normal estadística. La distribución desigual de los pesos 108 sinápticos puede dar como resultado que la señal de entrada completa se comunique a una neurona diana o "central" para la entrada dada, asignando de esta manera un valor de cero al peso sináptico del sujeto. Adicionalmente, la distribución desigual de los pesos sinápticos puede provocar que otras neuronas reciban valores de señal de entrada reducidos, por ejemplo, utilizando una distribución normal, logarítmica normal, sinusoidal u otra. Los valores de los pesos 108 sinápticos para las neuronas 116 que reciben valores de señal de entrada reducidos pueden aumentar junto con el aumento de su distancia desde la neurona "central". En dicho caso, se puede reducir el número de cálculos y se puede acelerar el funcionamiento de la red p. Dichas redes, que son una combinación de redes neuronales conocidas completamente conectadas y no completamente conectadas, pueden ser sumamente efectivas para el análisis de imágenes con patrones internos fuertes, por ejemplo, rostros humanos o tramas consecutivas de una película.

La Figura 9 muestra una realización no reivindicada de la red 100 p que es eficaz para el reconocimiento de patrones locales. Para mejorar la identificación de patrones comunes, el 10-20% de las conexiones fuertes, donde los valores de los pesos 108 sinápticos son pequeños o nulos, se pueden distribuir a lo largo de toda la red 100 p, en una forma determinista, tal como en la forma de una cuadrícula, o un enfoque aleatorio. La formación real de la red 100 p destinada a manejar una tarea en particular se realiza utilizando un programa, por ejemplo, escrito en un lenguaje de programación orientado a objetos, que genera elementos principales de la red p, tal como sinapsis, pesos sinápticos, distribuidores, pesos correctivos, neuronas, etc., como objetos de software. Dicho programa puede asignar relaciones entre los objetos anotados y los algoritmos que especifican sus acciones. En particular, se pueden formar pesos sinápticos y correctivos al comienzo de la formación de la red 100 p, junto con el establecimiento de sus valores iniciales. La red 100 p se puede formar completamente antes del inicio de su entrenamiento y modificar o agregar en una trama posterior, según sea necesario, por ejemplo, cuando se agota la capacidad de información de la red, o en caso de un error fatal. También es posible completar la red 100 p mientras continúa el entrenamiento.

Si la red 100 p se forma de antemano, el número de pesos correctivos seleccionados en una sinapsis particular puede ser igual al número de intervalos dentro del rango de señales de entrada. Adicionalmente, se pueden generar pesos correctivos después de la formación de la red 100 p, como señales en respuesta a la aparición de intervalos individuales. De manera similar a la red 10 neuronal clásica, la selección de parámetros y configuraciones de la red 100 p se proporciona con una serie de experimentos dirigidos. Dichos experimentos pueden incluir (1) la formación de la red p con los mismos pesos 108 sinápticos en todas las entradas, y (2) la evaluación de los valores de la señal de entrada para las imágenes seleccionadas y la selección inicial del número de intervalos. Por ejemplo, para el reconocimiento de imágenes binarias (de un color), puede ser suficiente tener solo 2 intervalos; para el reconocimiento cualitativo de imágenes de 8 bits, se pueden utilizar hasta 256 intervalos; la aproximación de dependencias estadísticas complejas puede requerir decenas o incluso cientos de intervalos; para grandes bases de datos, el número de intervalos podría ser de miles.

En el proceso de entrenamiento de la red 100 p, los valores de las señales de entrada se pueden redondear a medida que se distribuyen entre los intervalos específicos. Por lo tanto, es posible que no se requiera una precisión de las señales de entrada mayor que el ancho del rango dividido por el número de intervalos. Por ejemplo, si el rango de valores de entrada se establece en 100 unidades y el número de intervalos es 10, no se requerirá una precisión mejor que ± 5 . Dichos experimentos también pueden incluir (3) la selección de una distribución uniforme de intervalos en todo

el rango de valores de las señales de entrada y la distribución más simple para los coeficientes de impacto del peso correctivo $C_{i,d,n}$ se puede establecer igual a 1 para el peso correctivo correspondiente al intervalo para la señal de entrada particular, mientras que el impacto del peso correctivo para todos los pesos correctivos restantes se puede establecer en 0 (cero). Dichos experimentos pueden incluir adicionalmente (4) el entrenamiento de red 100 p con una, más o todas las imágenes de entrenamiento preparadas con precisión predeterminada.

El tiempo de entrenamiento de la red 100 p para una precisión predeterminada se puede establecer mediante experimentación. Si la precisión y el tiempo de entrenamiento de la red 100 p son satisfactorios, los ajustes seleccionados podrían mantenerse o cambiarse, mientras se continúa la búsqueda de una variante más eficaz. Si no se logra la precisión requerida, con fines de optimización, se puede evaluar la influencia de una modificación específica, que se puede realizar de una en el tiempo o en grupos. Dicha evaluación de modificaciones puede incluir cambiar, ya sea aumentar o reducir, el número de intervalos; cambiar el tipo de distribución de los coeficientes de impacto del peso correctivo ($C_{i,d,n}$), probar variantes con una distribución de intervalos no uniforme, tal como utilizar una distribución normal, de potencia, logarítmica o logarítmica normal; y cambiar los valores de los pesos 108 sinápticos, por ejemplo, su transición a una distribución no uniforme.

Si el tiempo de entrenamiento requerido para un resultado exacto se considera excesivo, el entrenamiento con un mayor número de intervalos se puede evaluar por su efecto sobre el tiempo de entrenamiento. Si, como resultado, se redujo el tiempo de entrenamiento, el aumento en el número de intervalos se puede repetir hasta que se obtenga el tiempo de entrenamiento deseado sin perder la precisión requerida. Si el tiempo de entrenamiento aumenta al aumentar el número de intervalos en lugar de reducirse, se puede realizar un entrenamiento adicional con un número reducido de intervalos. Si el número reducido de intervalos da como resultado una reducción del tiempo de entrenamiento, el número de intervalos se podría reducir aún más hasta obtener el tiempo de entrenamiento deseado.

La formación de los ajustes de red 100 p se puede realizar mediante entrenamiento con un tiempo de entrenamiento predeterminado y determinación experimental de la precisión del entrenamiento. Los parámetros se podrían mejorar mediante cambios experimentales similares a aquellos descritos anteriormente. La práctica real con varias redes p ha mostrado que el procedimiento de selección de la configuración es generalmente sencillo y no requiere mucho tiempo.

El entrenamiento real de la red 100 p como parte del método 200, que se muestra en la Figura 23, comienza con la carga de las señales de imagen de entrada $I_1, I_2...I_m$ a los dispositivos 102 de entrada de red, desde donde se transmiten a la sinapsis 118, pasan a través del peso 108 sináptico y entran en el distribuidor (o un grupo de distribuidores) 114. Basado en el valor de la señal de entrada, el distribuidor 114 establece el número del intervalo "d" al que corresponde la señal 104 de entrada dada, y asigna coeficientes de impacto de peso correctivo $C_{i,d,n}$ para todos los pesos 112 correctivos de los bloques 110 de corrección de peso de todas las sinapsis 118 conectadas con la entrada 102 respectiva. Por ejemplo, si el intervalo "d" se puede establecer en 3 para la primera entrada, para todos los pesos $W_{1,3,n}$ $C_{1,3,n} = 1$ se establece en 1, mientras que para todos los demás pesos con $i \neq 1$ y $d \neq 3$, $C_{i,d,n}$ se puede establecer en 0 (cero).

Para cada neurona 116, identificada como "n" en la siguiente relación, las sumas de salida de neuronas $\Sigma 1, \Sigma 2... \Sigma n$ se forman al multiplicar cada peso 112 correctivo, identificado como $W_{i,d,n}$ en la siguiente relación, por un coeficiente correspondiente de impacto de peso correctivo $C_{i,d,n}$ para todas las sinapsis 118 que contribuyen a la neurona particular y al agregar todos los valores obtenidos:

$$\Sigma^n = \sum_{i,d,n} W_{i,d,n} \times C_{i,d,n} [2]$$

La multiplicación de $W_{i,d,n} \times C_{i,d,n}$ se puede realizar mediante varios dispositivos, por ejemplo mediante distribuidores 114, dispositivos con pesos almacenados o directamente por neuronas 116. Las sumas se transfieren a través de la salida 117 de neurona a la calculadora 122 de corrección de peso. Las señales de salida deseadas $O_1, O_2...O_n$ que describen la imagen 126 de salida deseada también se cargan a la calculadora 122.

Como se discutió anteriormente, la calculadora 122 de corrección de peso es un dispositivo de cálculo para calcular el valor modificado para pesos correctivos mediante la comparación de las sumas de salida de neuronas $\Sigma 1, \Sigma 2... \Sigma n$ con las señales de salida deseadas $O_1, O_2...O_n$. La Figura 11 muestra un conjunto de pesos correctivos $W_{i,d,1}$, que contribuyen a la suma de salida de neuronas $\Sigma 1$, que se multiplican por el coeficiente correspondiente de impacto de peso correctivo $C_{i,d,1}$, y estos productos se agregan posteriormente mediante la suma de salida de neuronas $\Sigma 1$:

$$\Sigma 1 = W_{1,0,1} \times C_{1,0,1} + W_{1,1,1} \times C_{1,1,1} + W_{1,2,1} \times C_{1,2,1} + ... [3]$$

A medida que comienza el entrenamiento, es decir, durante la primera iteración, los pesos correctivos $W_{i,d,1}$ no corresponden a la imagen 106 de entrada utilizada para el entrenamiento, por lo tanto, las sumas de salida de neuronas $\Sigma 1$ no son iguales a la imagen 126 de salida deseada correspondiente. En base a los pesos correctivos iniciales $W_{i,d,1}$, el sistema de corrección de peso calcula el valor de corrección $\Delta 1$, que se utiliza para cambiar todos los pesos correctivos que contribuyen a la suma de salida de neuronas $\Sigma 1$ ($W_{i,d,1}$). La red 100 p permite varias opciones o variantes para su formación y utilización de señales correctivas colectivas para todos los pesos correctivos $W_{i,d,n}$ que contribuyen a una neurona 116 específica.

A continuación se muestran dos variantes de ejemplo y no limitantes para la formación y utilización de las señales correctivas colectivas. Variante 1: formación y utilización de señales correctivas basadas en la diferencia entre las señales de salida deseadas y las sumas de salida obtenidas de la siguiente manera:

- cálculo del valor de corrección igual Δ_n para todos los pesos correctivos que contribuyen a la neurona "n" de acuerdo con la ecuación:

$$\Delta_n = (O_n - \sum_n) / S [4],$$

Donde:

O_n : señal de salida deseable correspondiente a la suma de salida de la neurona \sum_n ;
S - número de sinapsis conectadas a la neurona "n".

- modificación de todos los pesos correctivos $W_{i,d,n}$ que contribuyen a la neurona "n" de acuerdo con la ecuación:

$$W_{i,d,n} \text{ modificado} = W_{i,d,n} + \Delta_n / C_{i,d,n} [5],$$

Variante 2 - formación y utilización de señales correctivas basadas en la relación de las señales de salida deseadas frente a las sumas de salida obtenidas como sigue:

- cálculo del valor de corrección igual Δ_n para todos los pesos correctivos que contribuyen a la neurona "n" de acuerdo con la ecuación:

$$\Delta_n = O_n / \sum_n [6],$$

- modificación de todos los pesos correctivos $W_{i,d,n}$ que contribuyen a la neurona "n" de acuerdo con la ecuación:

$$W_{i,d,n} \text{ modificado} = W_{i,d,n} \times \Delta_n [7],$$

La modificación de los pesos correctivos $W_{i,d,n}$ por cualquier variante disponible está destinada a reducir el error de entrenamiento para cada neurona 116 al hacer converger su suma de salida \sum_n en el valor de la señal de salida deseada. De esta manera, el error de entrenamiento para una imagen dada se puede reducir hasta que se iguale o se acerque a cero.

En la Figura 11 se muestra un ejemplo de modificación de los pesos correctivos $W_{i,d,n}$ durante el entrenamiento. Los valores de los pesos correctivos $W_{i,d,n}$ se establecen antes de que comience el entrenamiento en forma de distribución de peso aleatoria con los valores de peso que se establece en $0 \pm 10\%$ del rango de peso de corrección y alcanza la distribución de peso final después del entrenamiento. El cálculo descrito de señales colectivas se realiza para todas las neuronas 116 en la red 100 p. El procedimiento de entrenamiento descrito para una imagen de entrenamiento se puede repetir para todas las demás imágenes de entrenamiento. Dicho procedimiento puede dar lugar a la aparición de errores de entrenamiento para algunas de las imágenes previamente entrenadas, ya que algunos pesos correctivos $W_{i,d,n}$ pueden participar en varias imágenes. De acuerdo con lo anterior, el entrenamiento con otra imagen puede interrumpir parcialmente la distribución de los pesos correctivos $W_{i,d,n}$ formados para las imágenes anteriores. Sin embargo, debido al hecho de que cada sinapsis 118 incluye un conjunto de pesos correctivos $W_{i,d,n}$, que entrena con nuevas imágenes mientras posiblemente aumenta el error de entrenamiento, no borra las imágenes, para las cuales la red 100 p fue entrenada previamente. Más aún, cuantas más sinapsis 118 contribuyan a cada neurona 116 y mayor será el número de pesos correctivos $W_{i,d,n}$ en cada sinapsis, menos entrenamiento para una imagen específica afecta al entrenamiento para otras imágenes.

Cada iteración de entrenamiento generalmente termina con la convergencia sustancial del error de entrenamiento total y/o errores de entrenamiento locales para todas las imágenes de entrenamiento. Los errores se pueden evaluar utilizando métodos estadísticos conocidos, tal como, por ejemplo, el Error Cuadrático Medio (MSE), el Error Absoluto Medio (MAE) o el Error Medio Estándar (SEM). Si el error total o algunos de los errores locales son demasiado altos, se puede realizar una iteración de entrenamiento adicional hasta que el error se reduzca a menos de un valor de error predeterminado. El proceso de reconocimiento de imágenes descrito anteriormente con la definición del porcentaje de similitud entre diferentes imágenes utilizadas para el entrenamiento (mostrado en la Figura 8) es en sí mismo un proceso de clasificación de imágenes a lo largo de categorías previamente definidas.

Para agrupar, es decir, dividir imágenes en clases naturales o grupos que no se especificaron previamente, el algoritmo de entrenamiento básico del método 200 se puede modificar con el enfoque de Mapas Autoorganizados (SOM) modificado. La imagen 126 de salida deseada que corresponde a una imagen de entrada dada se puede formar directamente en el proceso de entrenamiento de la red 100 p basado en un conjunto de neuronas ganadoras con un valor máximo de las sumas 120 de neuronas de salida. La Figura 22 muestra cómo el uso del algoritmo básico del método 200 puede generar un conjunto primario de sumas de neuronas de salida, donde el conjunto se convierte

adicionalmente de modo que varias sumas mayores retienen su valor, o aumentan, mientras que todas las demás sumas se consideran iguales a cero. Este conjunto transformado de sumas de neuronas de salida se puede aceptar como la imagen 126 de salida deseada.

Formado como se describió anteriormente, el conjunto de imágenes 126 de salida deseadas incluye agrupaciones o grupos. Como tal, el conjunto de imágenes 126 de salida deseadas permite el agrupamiento de imágenes linealmente inseparables, que es distinta de la red 10 clásica. La Figura 13 muestra cómo el enfoque descrito puede ayudar a agrupar una imagen hipotética compleja "gato-coche", donde diferentes características de las imágenes se asignan a diferentes grupos: gatos y coches. Se puede utilizar un conjunto de imágenes 126 de salida deseadas creadas como se describe, por ejemplo, para crear diferentes clasificaciones, análisis estadístico, selección de imágenes basada en criterios formados como resultado del agrupamiento. También, las imágenes 126 de salida deseadas generadas por la red 100 p se pueden utilizar como imágenes de entrada para otra red p adicional, que también se puede formar a lo largo de las líneas descritas para la red 100 p del sujeto. Por lo tanto, las imágenes 126 de salida deseadas, formadas se pueden utilizar para una capa posterior de una red p de múltiples capas.

El entrenamiento de la red 10 neuronal clásica se proporciona generalmente a través de un método de entrenamiento supervisado que se basa en pares preparados de forma preliminar de una imagen de entrada y una imagen de salida deseada. El mismo método general también se utiliza para entrenar la red 100 p, sin embargo, la mayor velocidad de entrenamiento de la red 100 p también permite entrenar con un entrenador externo. La función del formador externo se puede realizar, por ejemplo, por un individuo o por un programa informático. Actuando como un entrenador externo, el individuo puede participar en la realización de una tarea física u operar en un entorno de juego. La red 100 p recibe señales de entrada en forma de datos con respecto a una situación particular y cambia a la misma. Las señales que reflejan las acciones del entrenador se pueden introducir como imágenes 126 de salida deseadas y permitir que la red 100 p sea entrenada de acuerdo con el algoritmo básico. De esta forma, la red 100 p puede generar modelos de varios procesos en tiempo real.

En un ejemplo no cubierto por la invención reivindicada, se puede entrenar la red 100 p para conducir un vehículo al recibir información sobre las condiciones de la carretera y las acciones del conductor. A través del modelado de una gran variedad de situaciones críticas, la misma red 100 p puede ser entrenada por muchos conductores diferentes y acumular más habilidades de conducción de las que generalmente es posible para un solo conductor. En otro ejemplo no cubierto por la invención reivindicada, la red 100 p es capaz de evaluar una condición de la carretera específica en 0.1 segundos o más rápido y acumular una "experiencia de conducción" sustancial que puede mejorar la seguridad del tráfico en una variedad de situaciones. La red 100 p también se puede entrenar para cooperar con un ordenador con una máquina de juego de ajedrez. La capacidad de la red 100 p para cambiar fácilmente del modo de entrenamiento al modo de reconocimiento y viceversa permite la realización de un modo de "aprender de los errores", cuando la red 100 p es entrenada por un entrenador externo. En dicho caso, la red 100 p parcialmente entrenada puede generar sus propias acciones para controlar un proceso tecnológico en un ejemplo que no está cubierto por la invención reivindicada. El entrenador podría controlar las acciones de la red 100 p y corregir esas acciones cuando sea necesario. Por tanto, podría proporcionarse entrenamiento adicional de la red 100 p.

La capacidad de información de la red 100 p es muy grande, pero no ilimitada. Con las dimensiones establecidas, tal como el número de entradas, salidas e intervalos, de la red 100 p, y con un aumento en el número de imágenes con las que se entrena la red p, después de un cierto número de imágenes, también se puede aumentar el número y la magnitud de errores de entrenamiento. Cuando se detecta tal aumento en la generación de errores, el número y/o la magnitud de los errores se puede reducir al aumentar el tamaño de la red 100 p, ya que la red p permite aumentar el número de neuronas 116 y/o el número de intervalos de señal "d" a través de la red p o en sus componentes entre iteraciones de entrenamiento. Se puede proporcionar la expansión de la red 100 p al agregar nuevas neuronas 116, agregar nuevas entradas 102 y sinapsis 118, cambiar la distribución de los coeficientes de impacto del peso correctivo $C_{i,d,n}$, y dividir los intervalos "d" existentes.

En la mayoría de los casos, la red 100 p se entrenará para asegurar su capacidad para reconocer imágenes, patrones y correlaciones inherentes a la imagen, o a un conjunto de imágenes. El proceso de reconocimiento en el caso más simple repite las primeras etapas del proceso de entrenamiento de acuerdo con el algoritmo básico divulgado como parte del método 200. En particular:

- el reconocimiento directo comienza con el formateo de la imagen de acuerdo con las mismas reglas que se utilizan para formatear imágenes para entrenamiento;
- la imagen se envía a las entradas de la red 100 p entrenada, los distribuidores asignan los pesos correctivos $W_{i,d,n}$ que corresponden a los valores de las señales de entrada que se establecieron durante el entrenamiento, y las neuronas generan las respectivas sumas de neuronas, como se muestra en la Figura 8;
- si las sumas de salida resultantes que representan la imagen 126 de salida cumplen totalmente con una de las imágenes con las que se está entrenando la red 100 p, hay un reconocimiento exacto del objeto; y
- si la imagen 126 de salida cumple parcialmente con varias imágenes con las que se está entrenando la red 100 p, el resultado muestra la tasa de coincidencia con diferentes imágenes como un porcentaje. La Figura 13 demuestra que durante el reconocimiento de la imagen compleja que se elabora en base a una combinación de imágenes de un gato

y un vehículo, la imagen 126 de salida representa la combinación de imágenes dada e indica el porcentaje de la contribución de cada imagen inicial en la combinación.

Por ejemplo, si se utilizaron varias fotos de una persona específica para el entrenamiento, la imagen reconocida puede corresponder en un 90 % a la primera foto, un 60 % a la segunda foto y un 35 % a la tercera foto. Puede ser que la imagen reconocida se corresponda con cierta probabilidad con las imágenes de otras personas o incluso de animales, lo que significa que existe cierta semejanza entre las fotos. Sin embargo, es probable que la probabilidad de tal semejanza sea menor. Con base en tales probabilidades, la confiabilidad del reconocimiento se puede determinar, por ejemplo, con base en el teorema de Bayes.

Con la red 100 p también es posible implementar el reconocimiento de múltiples etapas que combina las ventajas de los métodos de reconocimiento de redes neuronales y algorítmicas. Dicho reconocimiento de múltiples etapas puede incluir:

- reconocimiento inicial de una imagen por una red previamente entrenada mediante el uso de no todas, sino solo del 1 % al 10 % de las entradas, que en este documento se denominan “entradas básicas”. Dicha porción de las entradas se puede distribuir dentro de la red 100 p de manera uniforme, aleatoria o mediante cualquier otra función de distribución. Por ejemplo, el reconocimiento de una persona en la fotografía que incluye una pluralidad de otros objetos;
- seleccionar los objetos o partes de objetos más informativos para un reconocimiento más detallado. Dicha selección se puede proporcionar de acuerdo con estructuras de objetos específicos que están preestablecidos en la memoria, como en el método algorítmico, o de acuerdo con un gradiente de colores, brillo y/o profundidad de la imagen. Por ejemplo, en el reconocimiento de retratos se pueden seleccionar las siguientes zonas de reconocimiento: ojos, comisuras de la boca, forma de la nariz, así como ciertas características específicas, tales como tatuajes, números de matrícula de vehículos o números de casa también se pueden seleccionar y reconocer utilizando un enfoque similar;
- y
- también es posible el reconocimiento detallado de las imágenes seleccionadas, si es necesario.

La formación de una emulación por ordenador de la red 100 p y su entrenamiento se puede proporcionar basándose en la descripción anterior al utilizar cualquier lenguaje de programación. Por ejemplo, se puede utilizar una programación orientada a objetos, en la que los pesos 108 sinápticos, pesos 112 correctivos, distribuidores 114 y neuronas 116 representan objetos de programación o clases de objetos, se establecen relaciones entre clases de objetos a través de enlaces o mensajes, y algoritmos de la interacción se establece entre objetos y entre clases de objetos.

La formación y entrenamiento de la emulación de software de red 100 p puede incluir lo siguiente:

1. Preparación para la formación y entrenamiento de la red 100 p, en particular:

- conversión de conjuntos de imágenes de entrada de entrenamiento en forma digital de acuerdo con una tarea determinada;
- análisis de las imágenes digitales resultantes, que incluye la selección de parámetros de las señales de entrada que se utilizarán para el entrenamiento, por ejemplo, frecuencias, magnitudes, fases o coordenadas; y
- establecer un rango para las señales de entrenamiento, un número de intervalos dentro del rango del sujeto y una distribución de coeficientes de impacto de peso correctivo $C_{i,d,n}$.

2. Formación de la emulación de software de red p, que incluye:

- formación de un conjunto de entradas a la red 100 p. Por ejemplo, el número de entradas puede ser igual al número de señales en la imagen de entrada de entrenamiento;
- formación de un conjunto de neuronas, donde cada neurona representa un dispositivo agregado;
- formación de un conjunto de sinapsis con pesos sinápticos, donde cada sinapsis está conectada a una entrada de red p y una neurona;
- formación de bloques de corrección de peso en cada sinapsis, donde los bloques de corrección de peso incluyen distribuidores y pesos correctivos, y donde cada peso correctivo tiene las siguientes características:
 - Índice de entrada de peso correctivo (i);
 - Índice de neuronas de peso correctivo (n);
 - Índice de intervalo de peso correctivo (d); y
 - Valor inicial del peso correctivo ($W_{i,d,n}$).

- designar una correlación entre intervalos y pesos correctivos.

3. Entrenamiento de cada neurona con una imagen de entrada, que incluye:

- designar coeficientes de impacto de peso correctivo $C_{i,d,n}$, que incluyen:

- determinar un intervalo correspondiente a la señal de entrada de la imagen de entrada de entrenamiento recibida por cada entrada; y
- designar magnitudes de los coeficientes de impacto del peso correctivo $C_{i,d,n}$ para todos los pesos correctivos para todas las sinapsis.

- 5
- calcular la suma de salida de neuronas (Σ_n) para cada neurona "n" al agregar el valor de peso correctivo $W_{i,d,n}$ de todos los pesos sinápticos que contribuyen a la neurona multiplicado por los coeficientes correspondientes de impacto de peso correctivo $C_{i,d,n}$:

$$\Sigma_n = \sum_{i,d,n} W_{i,d,n} \times C_{i,d,n}$$

- 10
- calcular desviación o error de entrenamiento (T_n) mediante la resta de la suma de salida de la neurona Σ_n de la señal de salida deseada correspondiente O_n :

$$T_n = O_n - \Sigma_n$$

- 15
- calcular el valor de corrección igual (Δ_n) para todos los pesos correctivos que contribuyen a la neurona "n" al dividir el error de entrenamiento por el número de sinapsis "S" conectadas a la neurona "n":

$$\Delta_n = T_n / S$$

- 20
- modificar todos los pesos correctivos $W_{i,d,n}$ que contribuyen a la neurona respectiva al agregar a cada peso correctivo el valor de corrección Δ_n dividido por los coeficientes correspondientes de impacto del peso correctivo $C_{i,d,n}$:

$$W_{i,d,n} \text{ modificado} = W_{i,n,d} + \Delta_n / C_{i,d,n}$$

Otro método para calcular el valor de corrección igual (Δ_n) y modificar los pesos correctivos $W_{i,d,n}$ para todos los pesos correctivos que contribuyen a la neurona "n" puede incluir lo siguiente:

- 25
- dividir la señal de la imagen de salida deseada O_n por una suma de salida de neurona Σ_n :

$$\Delta_n = O_n / \Sigma_n$$

- 30
- modificar los pesos correctivos $W_{i,n,d}$ que contribuyen a la neurona al multiplicar los pesos correctivos por el valor de corrección Δ_n :

$$W_{i,d,n} \text{ modificado} = W_{i,n,d} \times \Delta_n$$

4. Entrenamiento de la red 100 p utilizando todas las imágenes de entrenamiento, que incluyen:

- 35
- repetir el proceso descrito anteriormente para todas las imágenes de entrenamiento seleccionadas que se incluyen en una iteración de entrenamiento; y
 - determinar un error o errores de la iteración de entrenamiento específica, comparar esos errores con un nivel de error aceptable predeterminado y repetir las iteraciones de entrenamiento hasta que los errores de entrenamiento sean menores que el nivel de error aceptable predeterminado.

40

Un ejemplo real de emulación de software de la red 100 p utilizando programación orientada a objetos se describe a continuación y se muestra en las Figuras 14-21.

La formación de una clase de objeto NeuronUnit puede incluir la formación de:

- 45
- un conjunto de objetos de la clase Synapse;
 - neurona 116 que presenta una variable, en la que la suma se realiza durante el entrenamiento; y
 - calculadora 122 que presenta una variable, en la que se almacena el valor de la suma 120 de neuronas deseada y se realiza el cálculo de los valores de corrección Δ_n durante el proceso de entrenamiento.

50

La clase NeuronUnit proporciona un entrenamiento de red 100 p que puede incluir:

- entrenamiento de sumas 120 de neuronas;
 - configuración de sumas deseadas;
 - cálculo del valor de corrección Δ_n ; y
- 55
- suma del valor de corrección calculado Δ_n para los pesos correctivos $W_{i,n,d}$.

La formación de la clase de objeto Synapse puede incluir:

- conjunto de pesos correctivos $W_{i,n,d}$; y

- puntero que indica la entrada conectada a la sinapsis 118.

La clase Synapse puede realizar las siguientes funciones:

- inicialización de los pesos correctivos $W_{i,n,d}$;
- multiplicación de los pesos $W_{i,n,d}$ por los coeficientes $C_{i,d,n}$; y
- corrección de pesos $W_{i,n,d}$.

La formación de la clase de objeto InputSignal puede incluir:

- un conjunto de índices en las sinapsis 118 conectadas a una entrada 102 dada;
- variable que incluye el valor de la señal 104 de entrada;
- valores de posible señal de entrada mínima y máxima;
- número de intervalos "d"; y
- duración del intervalo.

La clase InputSignal puede proporcionar las siguientes funciones:

- formación de la estructura de la red 100 p, que incluye:
 - Agregar y eliminar enlaces entre una entrada 102 y las sinapsis 118; y
 - Establecer el número de intervalos "d" para las sinapsis 118 de una entrada 102 particular.
- configuración de los parámetros de las señales 104 de entrada mínima y máxima;
- contribución a la operación de la red 100 p:
 - establecer una señal 104 de entrada; y
 - establecer coeficientes de impacto del peso correctivo $C_{i,d,n}$.

La formación de la clase de objeto PNet incluye un conjunto de clases de objeto:

- NeuronUnit; y
- InputSignal.

La clase PNet proporciona las siguientes funciones:

- configuración del número de objetos de la clase InputSignal;
- configuración del número de objetos de la clase NeuronUnit; y
- solicitud de grupo de funciones de los objetos NeuronUnit y InputSignal.

Durante el proceso de entrenamiento se pueden formar los ciclos, donde:

- se forma la suma de salida de neuronas que es igual a cero antes de que comience el ciclo;
- se revisan todas las sinapsis que contribuyen a la NeuronUnit dada. Para cada sinapsis 118:
 - Basado en la señal 102 de entrada, el distribuidor forma un conjunto de coeficientes de impacto de peso correctivo $C_{i,d,n}$;
 - se revisan todos los pesos $W_{i,n,d}$ de dicha sinapsis 118, y para cada peso:
 - El valor del peso $W_{i,n,d}$ se multiplica por el correspondiente coeficiente de impacto del peso correctivo $C_{i,d,n}$;
 - El resultado de la multiplicación se agrega a la suma de salida de la neurona en entrenamiento;
 - se calcula el valor de corrección Δ_n ;
 - se divide el valor de corrección Δ_n por el coeficiente de impacto de peso correctivo $C_{i,d,n}$, es decir, $\Delta_n/C_{i,d,n}$; y
 - se revisan todas las sinapsis 118 que contribuyen a la NeuronUnit dada. Para cada sinapsis 118, se revisan todos los pesos $W_{i,n,d}$ de la sinapsis del sujeto, y para cada peso su valor se modifica al valor de corrección correspondiente Δ_n .

La posibilidad anteriormente señalada de entrenamiento adicional de la red 100 p permite una combinación de entrenamiento con el reconocimiento de la imagen que permite acelerar el proceso de entrenamiento y mejorar su precisión. Al entrenar la red 100 p en un conjunto de imágenes que cambian secuencialmente, como el entrenamiento en tramas consecutivas de la película que son ligeramente diferentes entre sí, el entrenamiento adicional puede incluir:

- entrenamiento con la primera imagen;

- reconocimiento de la siguiente imagen e identificación de un porcentaje de similitud entre la nueva imagen y la imagen con la que se entrenó inicialmente la red. No se requiere entrenamiento adicional si el error de reconocimiento es menor que su valor predeterminado; y
- si el error de reconocimiento excede el valor predeterminado, se proporciona entrenamiento adicional.

El entrenamiento de la red 100 p mediante el algoritmo de entrenamiento básico anterior es efectivo para resolver problemas de reconocimiento de imágenes, pero no excluye la pérdida o corrupción de datos debido a imágenes superpuestas. Por lo tanto, el uso de la red 100 p con fines de memoria, aunque es posible, puede no ser del todo confiable. El entrenamiento de la red 100 p proporciona protección contra la pérdida o corrupción de información. Se puede introducir una restricción adicional en el algoritmo de entrenamiento de red básico que requiere que cada peso correctivo $W_{i,n,d}$ se pueda entrenar solo una vez. Después del primer ciclo de entrenamiento, el valor del peso $W_{i,n,d}$ permanece fijo o constante. Esto se puede lograr al ingresar un índice de acceso adicional "a" para cada peso correctivo, que es el índice descrito anteriormente que representa el número de accesos al peso correctivo sujeto $W_{i,n,d}$ durante el proceso de entrenamiento.

Como se describió anteriormente, cada peso correctivo puede tomar la nomenclatura de $W_{i,n,d,a}$, en donde "a" es el número de accesos al peso del sujeto durante el proceso de entrenamiento. En el caso más simple, para los pesos no modificados, es decir, no fijos, $a = 0$, mientras que para los pesos que han sido modificados o fijados por el algoritmo básico descrito, $a = 1$. Más aún, mientras que se aplica el algoritmo básico, los pesos correctivos $W_{i,n,d,a}$ con el valor fijo $a = 1$ se pueden excluir de los pesos para los cuales se están realizando las correcciones. En dicho caso, las ecuaciones [5], [6] y [7] se pueden transformar de la siguiente manera:

Valor	Algoritmo básico	Algoritmo de entrenamiento con pesos fijos
Valor de corrección igual - Variante 1	$\Delta_n = (O_n - \Sigma_n)/S$ [4],	$\Delta_n = (O_n - \Sigma_n)/S_0$ [8], donde S_0 - suma $C_{i,d,n,a}$ de todos los pesos correctivos $W_{i,n,d,a}$ que contribuyen a la neurona del sujeto y que tienen el índice $a = 0$
Peso correctivo modificado - Variante 1	$W_{i,n,d}$ modificado = $W_{i,n,d} + \Delta_n/C_{i,d,n}$ [5],	$W_{i,n,d,0}$ modificado = $W_{i,n,d,0} + \Delta_n/C_{i,d,n,0}$ [9], donde $W_{i,n,d,0}$ son pesos que contribuyen a la neurona del sujeto y que tienen el índice $a = 0$, y $G_{i,d,n,0}$ son coeficientes del impacto de peso correctivo para los pesos correctivos que contribuyen a someter la neurona y que tienen el índice $a = 0$
Peso correctivo modificado - Variante 2	$W_{i,n,d}$ modificado = $W_{i,n,d} \times \Delta_n$ [7]	$W_{i,n,d,0}$ modificado = $W_{i,n,d,0} \times \Delta_n$ [10]

La restricción anterior se puede aplicar parcialmente a la corrección de los pesos correctivos previamente entrenados $W_{i,n,d,a}$, pero solo a los pesos que forman las imágenes más importantes. Por ejemplo, dentro de la capacitación sobre un conjunto de retratos de una sola persona, una imagen específica se puede declarar principal y asignársele prioridad. Después de entrenar en dicha imagen de prioridad, todos los pesos correctivos $W_{i,n,d,a}$ que se cambian en el proceso de entrenamiento se pueden fijar, es decir, donde el índice $a = 1$, designando de esta manera el peso como $W_{i,n,d,1}$, y otras imágenes de la misma persona pueden permanecer modificables. Dicha prioridad puede incluir otras imágenes, por ejemplo, aquellas que se utilizan como claves de cifrado y/o contienen datos numéricos críticos.

Los cambios en los pesos correctivos $W_{i,n,d,a}$ también pueden no estar completamente prohibidos, pero limitados al crecimiento del índice "a". Es decir, cada uso posterior del peso $W_{i,n,d,a}$ se puede utilizar para reducir su capacidad de cambio. Cuanto más a menudo se utiliza un peso correctivo particular $W_{i,n,d,a}$, menos cambia el peso con cada acceso y, por lo tanto, durante el entrenamiento de las imágenes posteriores, las imágenes almacenadas anteriores se modifican menos y experimentan una menor corrupción. Por ejemplo, si $a = 0$, es posible cualquier cambio en el peso $W_{i,n,d,a}$; cuando $a = 1$, la posibilidad de cambio del peso puede reducirse al $\pm 50\%$ del valor del peso; con $a = 2$ la posibilidad de cambio se puede reducir al $\pm 25\%$ del valor del peso.

Después de alcanzar el número predeterminado de accesos, como lo indica el índice "a", por ejemplo, cuando $a = 5$, se puede prohibir el cambio adicional del peso $W_{i,n,d,a}$. Dicho enfoque puede proporcionar una combinación de alta inteligencia y seguridad de la información dentro de una sola red 100 p. Utilizando el mecanismo de cálculo de errores de red, se pueden establecer niveles de errores permitidos de manera que se pueda guardar información con pérdidas dentro de un rango de precisión predeterminado, en donde el rango de precisión se puede asignar de acuerdo con una tarea en particular. En otras palabras, para la red 100 p que opera con imágenes visuales, el error se puede establecer en un nivel que no puede ser capturado a simple vista, lo que proporcionaría un "factor de" aumento significativo en la capacidad de almacenamiento. Lo anterior puede permitir la creación de un almacenamiento altamente eficaz de información visual, por ejemplo, películas.

La capacidad de limpiar selectivamente la memoria del ordenador puede ser valiosa para el funcionamiento continuo de alto nivel de la red 100 p. Dicha limpieza selectiva de la memoria se puede realizar al eliminar ciertas imágenes sin pérdida o corrupción del resto de la información almacenada. Dicha limpieza se puede realizar de la siguiente manera:

- identificación de todos los pesos correctivos $W_{i,n,d,a}$ que participan en la formación de la imagen, por ejemplo, al introducir la imagen en la red o compilar la lista de pesos correctivos utilizados para cada imagen;
- reducción del índice "a" para los respectivos pesos correctivos $W_{i,n,d,a}$; y
- reemplazo de los pesos correctivos $W_{i,n,d,a}$ ya sea por cero o por un valor aleatorio cercano al medio del rango de valores posibles para el peso del sujeto cuando el índice "a" se reduce a cero.

Se puede seleccionar experimentalmente un orden y sucesión apropiados de reducción del índice "a" para identificar patrones fuertes ocultos en la secuencia de imágenes. Por ejemplo, por cada 100 imágenes introducidas en la red 100 p durante el entrenamiento, puede haber una reducción del índice "a" en una cuenta de uno, hasta que "a" alcance el valor cero. En dicho caso, el valor de "a" puede crecer de acuerdo con lo anterior con la introducción de nuevas imágenes. La competencia entre el crecimiento y la reducción de "a" puede conducir a una situación en la que los cambios aleatorios se eliminan gradualmente de la memoria, mientras que los pesos correctivos $W_{i,n,d,a}$ que se han utilizado y confirmado muchas veces se pueden guardar. Cuando la red 100 p se entrena en un gran número de imágenes con atributos similares, por ejemplo, del mismo sujeto o entorno similar, los pesos correctivos $W_{i,n,d,a}$ que se utilizan con frecuencia, confirman constantemente su valor y la información en estas áreas se vuelve muy estable. Adicionalmente, el ruido aleatorio desaparecerá gradualmente. En otras palabras, la red 100 p con una disminución gradual del índice "a" puede servir como un filtro de ruido eficaz.

El entrenamiento de la red 100 p descrito sin pérdida de información permite crear una memoria de red p con alta capacidad y confiabilidad. Dicha memoria se puede utilizar como una memoria de ordenador de alta velocidad de gran capacidad que proporciona mayor velocidad incluso que el sistema de "memoria de efectivo", pero no aumentará el coste y la complejidad del ordenador como es típico con el sistema de "memoria de efectivo". Según los datos publicados, en general, mientras se graba una película con redes neuronales, la memoria se puede comprimir decenas o cientos de veces sin una pérdida significativa de la calidad de grabación. En otras palabras, una red neuronal puede operar como un programa de archivo muy eficaz. La combinación de esta capacidad de las redes neuronales con la capacidad de entrenamiento de alta velocidad de la red 100 p puede permitir la creación de un sistema de transmisión de datos de alta velocidad, una memoria con alta capacidad de almacenamiento y un programa de descifrado de archivos multimedia de alta velocidad, es decir, un código.

Debido al hecho de que los datos de la red 100 p se almacenan como un conjunto de pesos correctivos $W_{i,n,d,a}$, que es un tipo de grabación de código, es poco probable la decodificación o acceso no autorizado a la red p a través de métodos existentes y sin el uso de una red y clave idénticas. Por tanto, la red 100 p puede ofrecer un grado considerable de protección de datos. También, a diferencia de la memoria de ordenador convencional, el daño a los elementos de almacenamiento individuales de la red 100 p presenta un efecto perjudicial insignificante, ya que otros elementos compensan significativamente las funciones perdidas. En el proceso de reconocimiento de imágenes, los patrones inherentes de la imagen que se utiliza prácticamente no se distorsionan como resultado del daño a uno o más elementos. Lo anterior puede mejorar drásticamente la confiabilidad de los ordenadores y permitir el uso de ciertos bloques de memoria, que bajo condiciones normales se considerarían defectuosos. Además, este tipo de memoria es menos vulnerable a los ataques de piratas informáticos debido a la ausencia de direcciones permanentes para bytes críticos en la red 100 p, lo que lo hace impermeable al ataque de dicho sistema por una variedad de virus informáticos.

El proceso de reconocimiento de imágenes anteriormente señalado con determinación del porcentaje de similitud entre diferentes imágenes utilizadas en el entrenamiento también puede emplearse como un proceso de clasificación de imágenes de acuerdo con las categorías definidas anteriormente, como se señaló anteriormente. Para el agrupamiento, que es una división de las imágenes en clases o grupos naturales no predefinidos, se puede modificar el proceso de entrenamiento básico. Una realización no reivindicada puede incluir:

- preparación de un conjunto de imágenes de entrada para entrenamiento, sin incluir imágenes de salida preparadas;
- formación y entrenamiento de la red con la formación de las sumas de salida de neuronas como se hace de acuerdo con el algoritmo básico;
- selección en la imagen de salida resultante de la salida con la suma de salida máxima, es decir, la salida ganadora, o un grupo de salidas ganadoras, que se pueden organizar de manera similar a la red Kohonen;
- creación de una imagen de salida deseada, en la que la salida ganadora o el grupo de salidas ganadoras reciben valores máximos. Al mismo tiempo:
 - El número de salidas ganadoras seleccionadas se puede predeterminar, por ejemplo, en un rango de 1 a 10, o las salidas ganadoras se pueden seleccionar de acuerdo con la regla "no menos del N% de la suma máxima de neuronas", donde "N" puede estar, por ejemplo, dentro del 90 – 100 %; y
 - Todas las demás salidas se pueden establecer en cero.
- entrenamiento de acuerdo con el algoritmo básico con el uso de la imagen de salida deseada creada, Fig. 13; y
- repetición de todos los procedimientos para otras imágenes con formación para cada imagen de diferentes ganadores o grupos de ganadores.

El conjunto de imágenes de salida deseadas formado de la manera anterior se puede utilizar para describir agrupaciones o grupos en los que la pluralidad de imágenes de entrada se puede separar de forma natural. Este

conjunto de imágenes de salida deseadas se puede utilizar para producir diferentes clasificaciones, tales como para la selección de imágenes de acuerdo con los criterios establecidos y en análisis estadístico. Lo anterior también se puede utilizar para la inversión antes mencionada de imágenes de entrada y salida. En otras palabras, las imágenes de salida deseadas se pueden utilizar como imágenes de entrada para otra red, es decir, adicional, y la salida de la red adicional pueden ser imágenes presentadas en cualquier forma adecuada para entrada de ordenador.

En la red 100 p, después de un único ciclo de entrenamiento con el algoritmo descrito anteriormente, se pueden generar imágenes de salida deseadas con una pequeña variación de suma de salida, lo que puede ralentizar el proceso de entrenamiento y también puede reducir su precisión. Para mejorar el entrenamiento de la red 100 p, la variación inicial de puntos puede aumentarse o extenderse artificialmente, de modo que la variación de la magnitud de los puntos cubra todo el rango de posibles valores de salida, por ejemplo -50 a +50, como se muestra en la Fig. 21. Dicha extensión de la variación inicial de puntos puede ser lineal o no lineal.

Puede desarrollarse una situación en la que el valor máximo de una determinada salida sea un valor atípico o un error, por ejemplo, una manifestación de ruido. Esto se puede manifestar por la aparición de un valor máximo rodeado por una multitud de pequeñas señales. Cuando se seleccionan las salidas ganadoras, los valores de las señales pequeñas se pueden ignorar mediante la selección de las señales más grandes rodeadas de otras señales grandes como ganadoras. Para este propósito, se pueden utilizar técnicas estadísticas conocidas de reducción de la varianza, tal como el muestreo por importancia. Dicho enfoque puede permitir eliminar el ruido manteniendo patrones básicos valiosos. La creación de grupos ganadores permite agrupar imágenes linealmente inseparables, es decir, imágenes que se relacionan con más de un grupo, como se muestra en la Figura 13. Lo anterior puede proporcionar una mejora significativa en la precisión y disminuir el número de errores de agrupamiento.

En el proceso de entrenamiento de la red 100 p, los errores típicos que se someten a corrección son:

Error típico de la red neuronal	Método de corrección de la red 100 p
Errores en la selección de imágenes de entrenamiento. Por ejemplo, el conjunto de imágenes humanas incluye una imagen de un gato	Borrado de la imagen de salida deseada correspondiente o restricción de su demostración
Errores de red que no fueron corregidos durante el entrenamiento. Por ejemplo, una determinada imagen se reconoce incorrectamente porque la red no puede dividir algunas características del objeto (el efecto de la inseparabilidad lineal).	Entrenamiento adicional de la red 100 p después de que se detecta el error; introducción de una imagen de salida deseada adicional
Disminución de la precisión debido a que se alcanzó el límite de la capacidad de información de la red	Expansión de la red 100 p

La corrección de errores también es posible con la ayuda del algoritmo descrito anteriormente en el entrenamiento con un entrenador externo.

La descripción detallada y los dibujos o figuras apoyan y describen la divulgación.

REIVINDICACIONES

1. Una red (100) neuronal implementada por ordenador configurada para reconocer patrones en imágenes, la red (100) neuronal implementada por ordenador comprende:

una pluralidad de entradas (102) de la red neuronal, cada entrada configurada para recibir una señal (104, I_1-I_m , 106) de entrada que tiene un valor de entrada, en el que la señal de entrada representa una característica de una imagen de entrada;

una pluralidad de sinapsis (118), en la que cada sinapsis se conecta a una de la pluralidad de entradas e incluye una pluralidad de pesos (112) correctivos, en los que cada peso correctivo se define por un valor de peso;

un conjunto de distribuidores (114), en el que cada distribuidor se conecta de forma operativa a una de la pluralidad de entradas para recibir la respectiva señal de entrada y se configura para seleccionar uno más pesos correctivos de la pluralidad de pesos (112) correctivos en correlación con el valor de entrada;

un conjunto de neuronas (116), en el que cada neurona tiene al menos una salida (117) y se conecta con al menos una de la pluralidad de entradas a través de una de la pluralidad de sinapsis (118), y en la que cada neurona (116) se configura para agregar los valores de peso de los pesos (112) correctivos seleccionados de cada sinapsis (118) conectada a la respectiva neurona (116) y generar de esta manera una suma (120, $\Delta 1-\Delta n$) de neuronas; y

una calculadora (122) de corrección de peso configurada para recibir una señal (124, O_1-O_n) de salida deseada que tiene un valor que representa un patrón reconocido en la imagen de entrada, determinar una desviación ($\Delta 1-\Delta n$) de la suma de neuronas (120, $\Sigma 1-\Sigma n$) del valor (124, O_1-O_n) señal de salida deseada, y modificar valores de peso correctivo respectivos utilizando la desviación determinada para minimizar la desviación de la suma de neuronas del valor de señal de salida deseada cuando los valores de peso correctivo modificados se agregan para determinar la suma de neuronas durante el entrenamiento de la red neuronal, y de esta manera entrenar la red neuronal para reconocer dichos patrones en imágenes; en los que

la determinación de la desviación de la suma de neuronas de la señal de salida deseada incluye la división del valor de señal de salida deseada por la suma de neuronas para generar de esta manera un coeficiente de desviación; y la modificación de los valores de peso correctivo respectivos incluye la multiplicación de cada peso correctivo utilizado para generar la suma de neuronas por el coeficiente de desviación.

2. La red (100) neuronal implementada por ordenador de la reivindicación 1, en la que:

cada distribuidor (114) se configura adicionalmente para asignar una pluralidad de coeficientes de impacto a la respectiva pluralidad de pesos correctivos, de tal manera que cada coeficiente de impacto se asigna a una de la pluralidad de pesos correctivos en una proporción predeterminada para generar la respectiva suma de neuronas;

cada neurona se configura para agregar un producto del peso correctivo y el coeficiente asignado de impacto para toda la sinapsis conectada a la misma; y

la calculadora de corrección de peso se configura para aplicar una porción de la diferencia determinada a cada peso correctivo utilizado para generar la suma de neuronas de acuerdo con la proporción establecida por el respectivo coeficiente de impacto.

3. La red (100) neuronal implementada por ordenador de la reivindicación 2, en la que:

cada respectiva pluralidad de coeficientes de impacto (134) se define por una función de distribución de impacto;

la pluralidad de los valores de entrada se recibe en un rango de valor dividido en intervalos (d_0-d_3) de acuerdo con una función de distribución de intervalo, de tal manera que cada valor de entrada se recibe dentro de un respectivo intervalo, y cada peso (112) correctivo corresponde a uno de los intervalos (d_0-d_3); y

cada distribuidor (114) se configura para utilizar el respectivo valor de entrada recibido para seleccionar el respectivo intervalo, y para asignar la respectiva pluralidad de coeficientes de impacto (134) al peso (112) correctivo que corresponde al respectivo intervalo seleccionado (d_0-d_3) y al menos uno peso correctivo que corresponde a un intervalo adyacente al respectivo intervalo seleccionado.

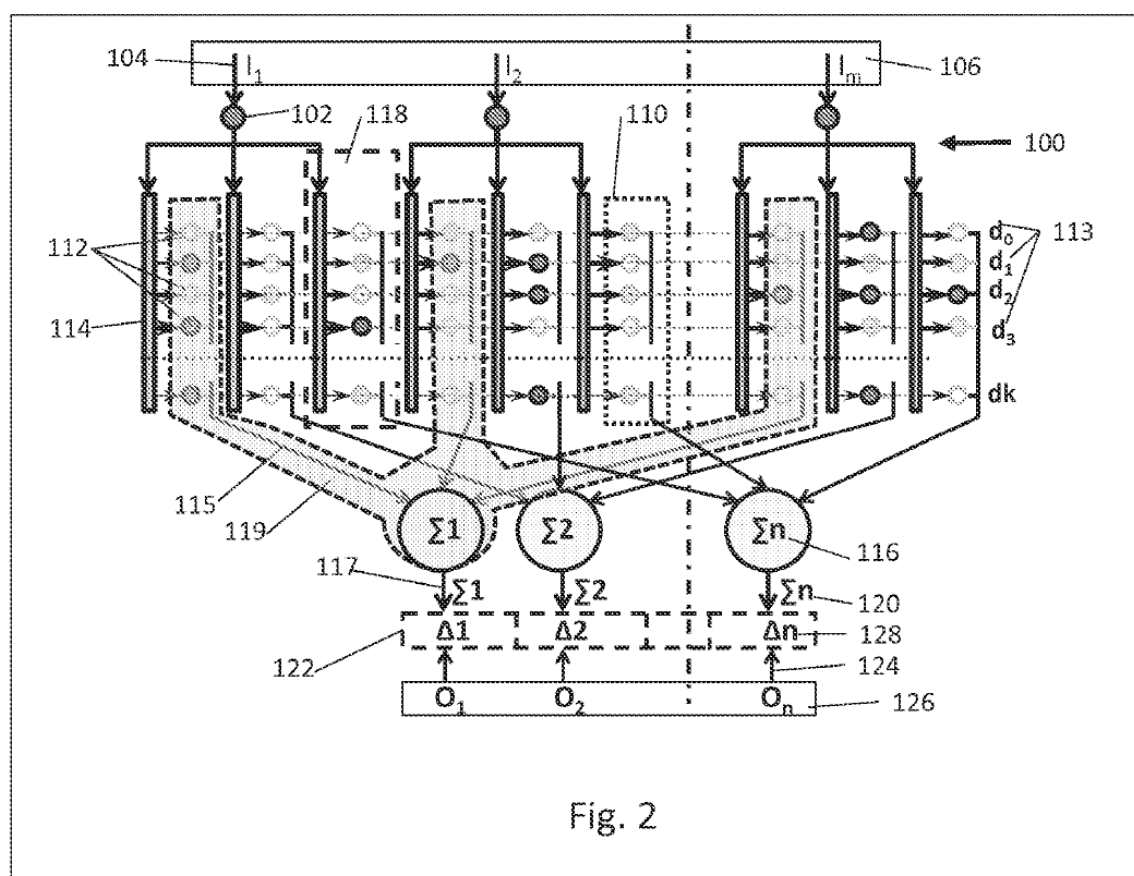
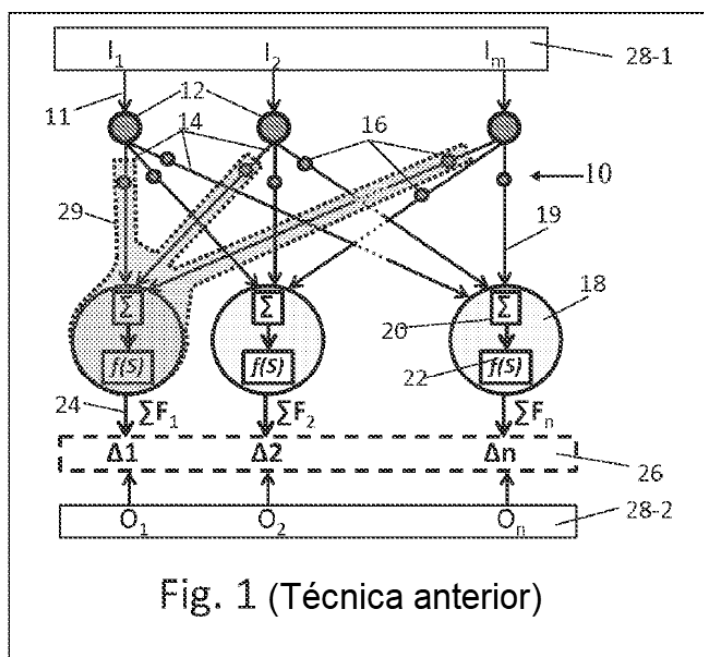
4. La red (100) neuronal implementada por ordenador de la reivindicación 3, en la que cada peso (112) correctivo se define adicionalmente por un conjunto de índices que incluye:

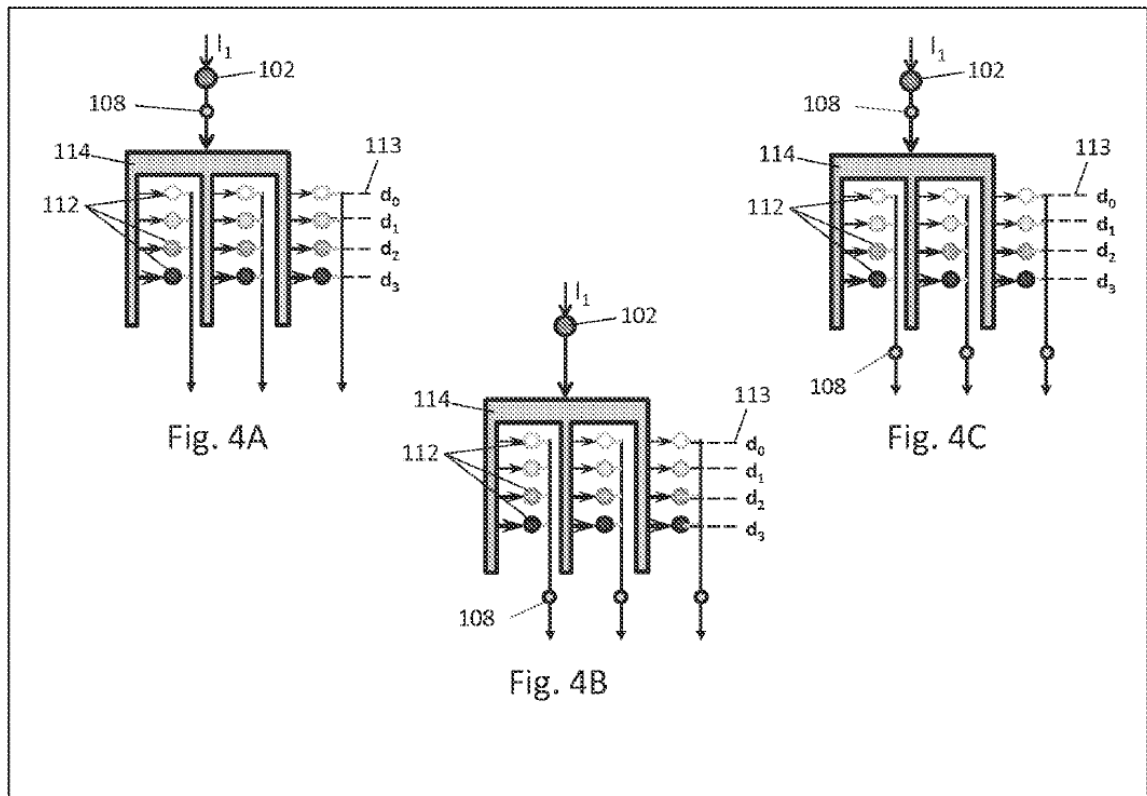
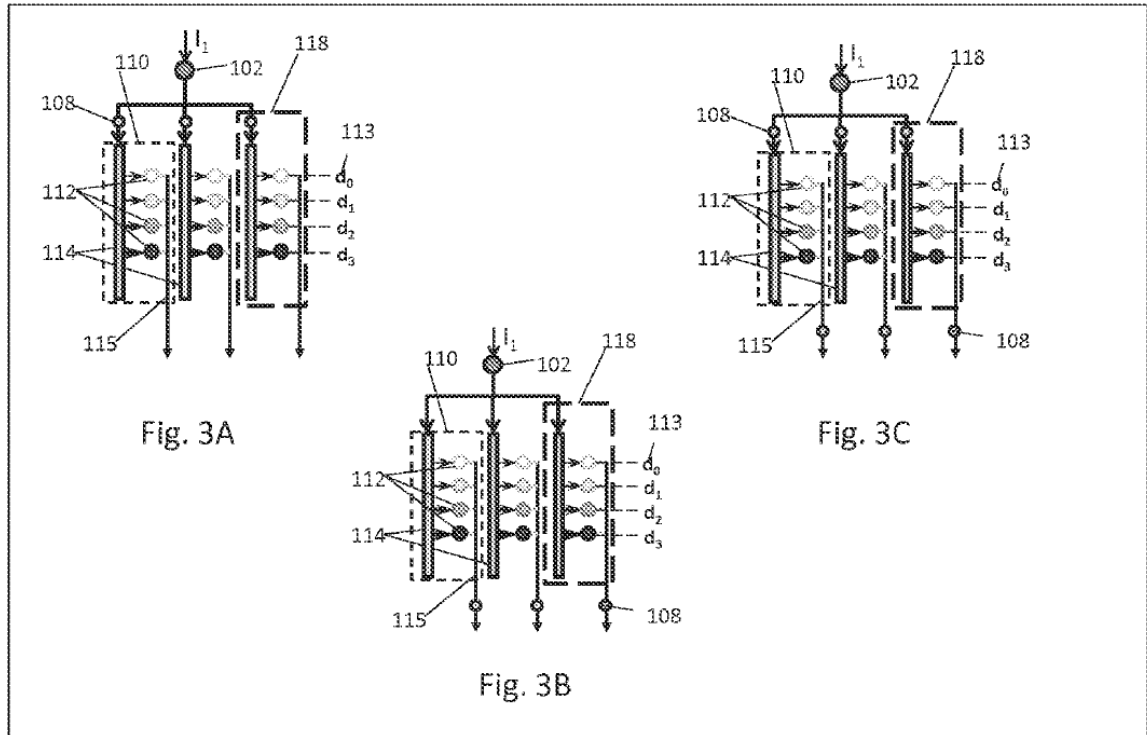
un índice de entrada ("i") configurado para identificar el peso correctivo que corresponde a la entrada;

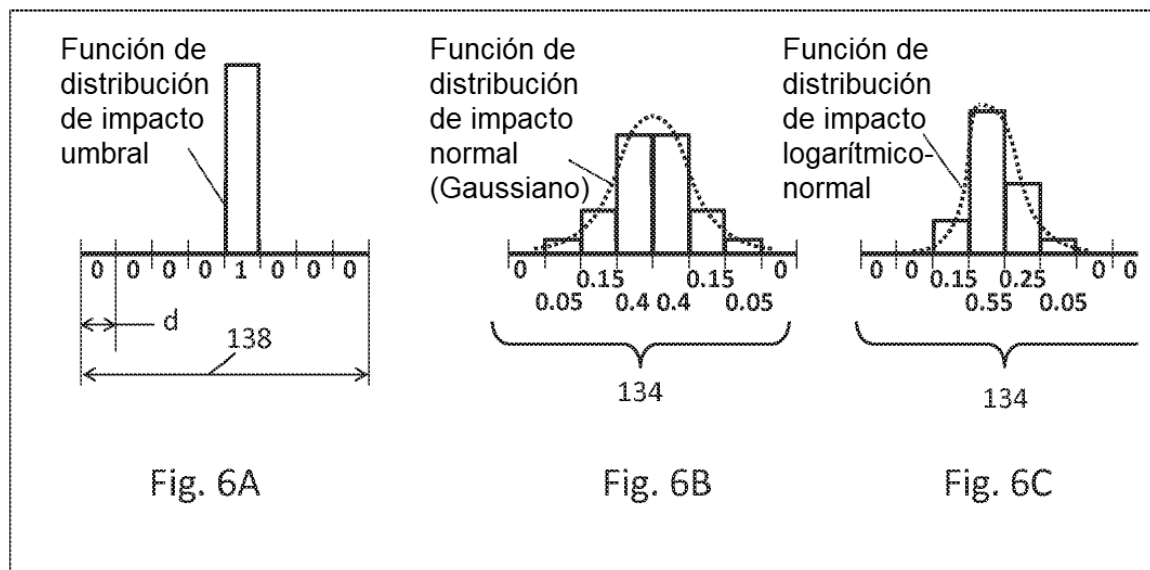
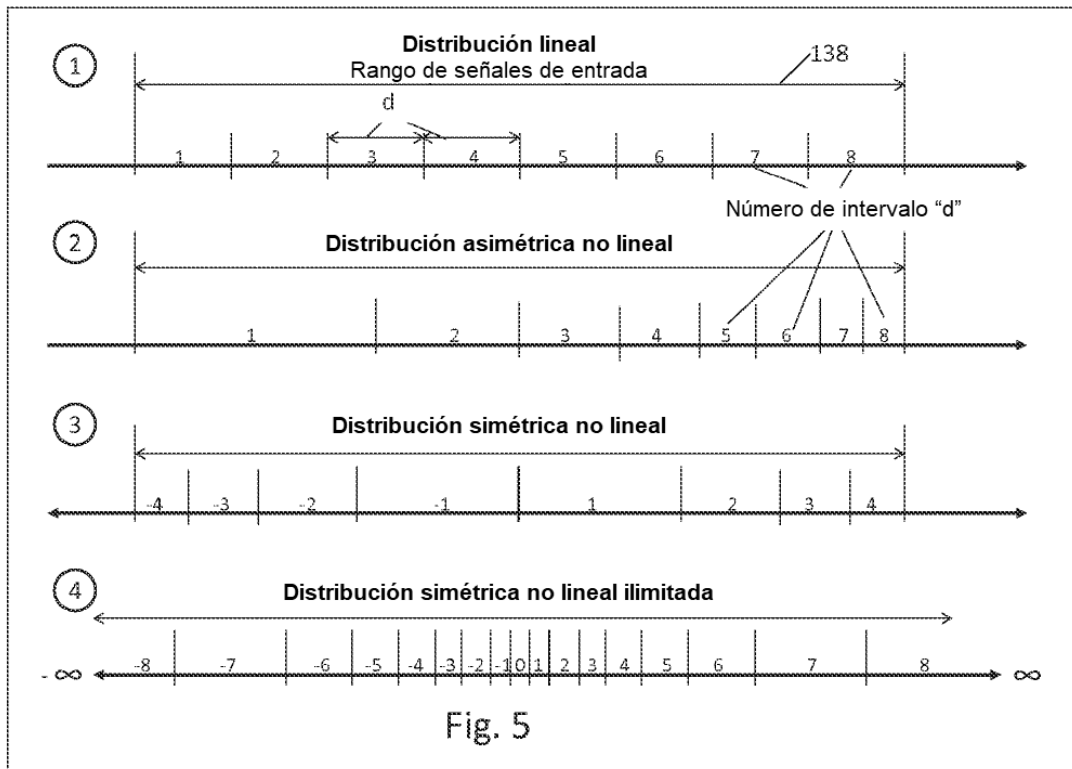
un índice de intervalo ("d") configurado para especificar el intervalo seleccionado para el respectivo peso correctivo; y

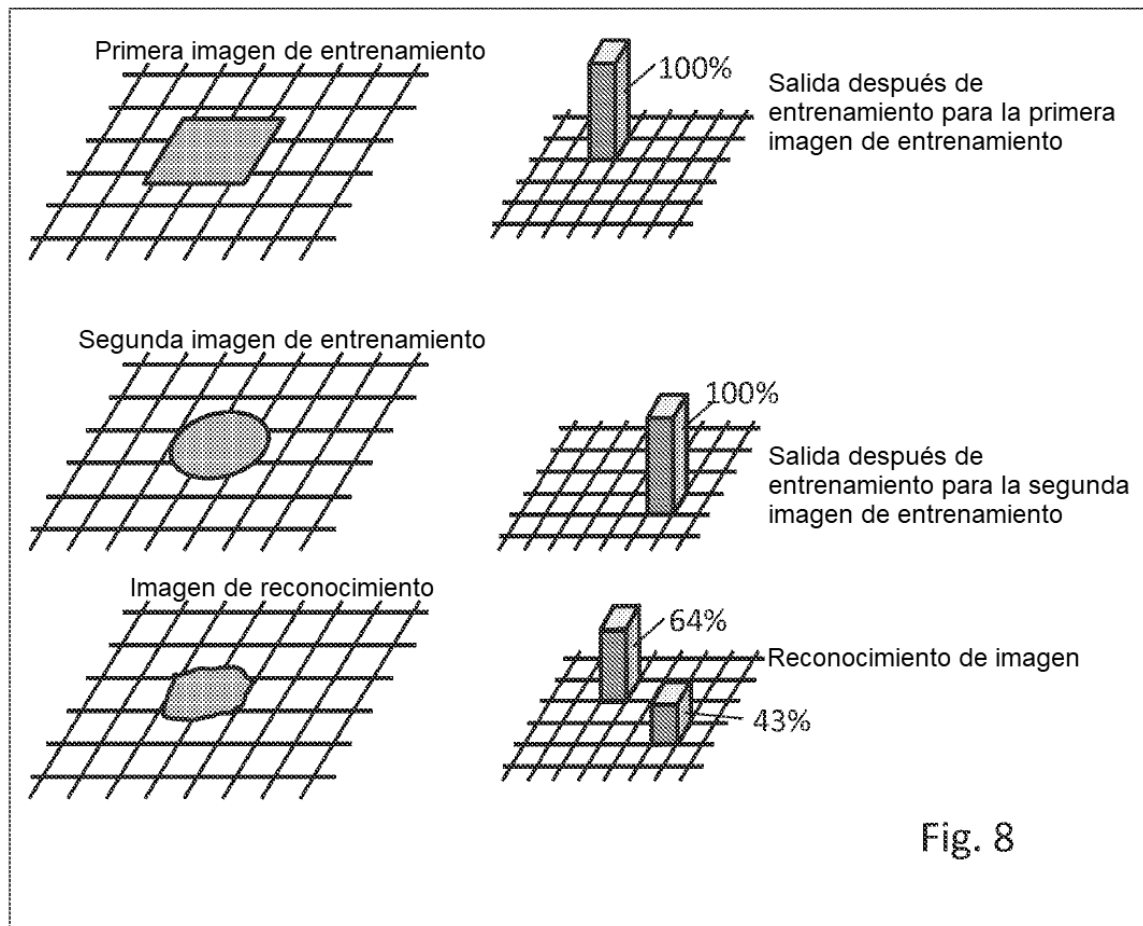
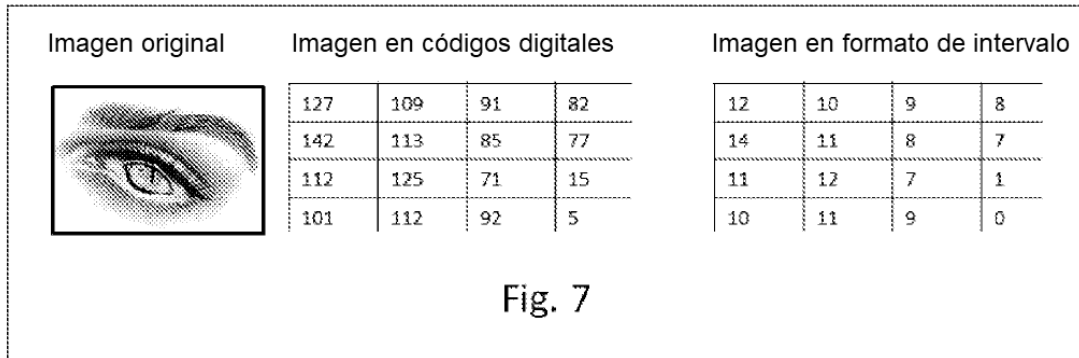
un índice de neurona ("n") configurado para especificar el peso correctivo que corresponde a la neurona (116).

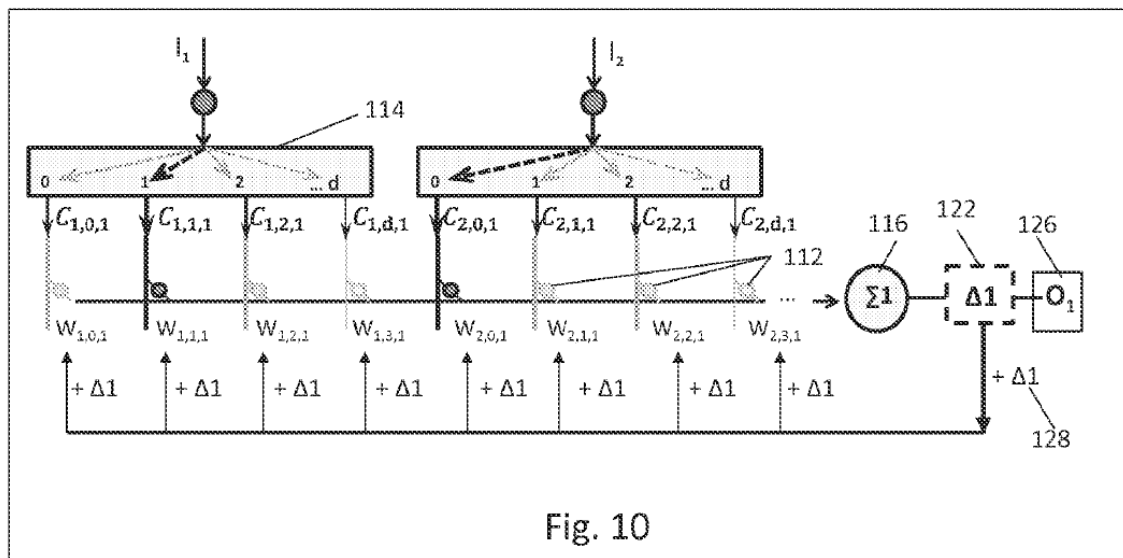
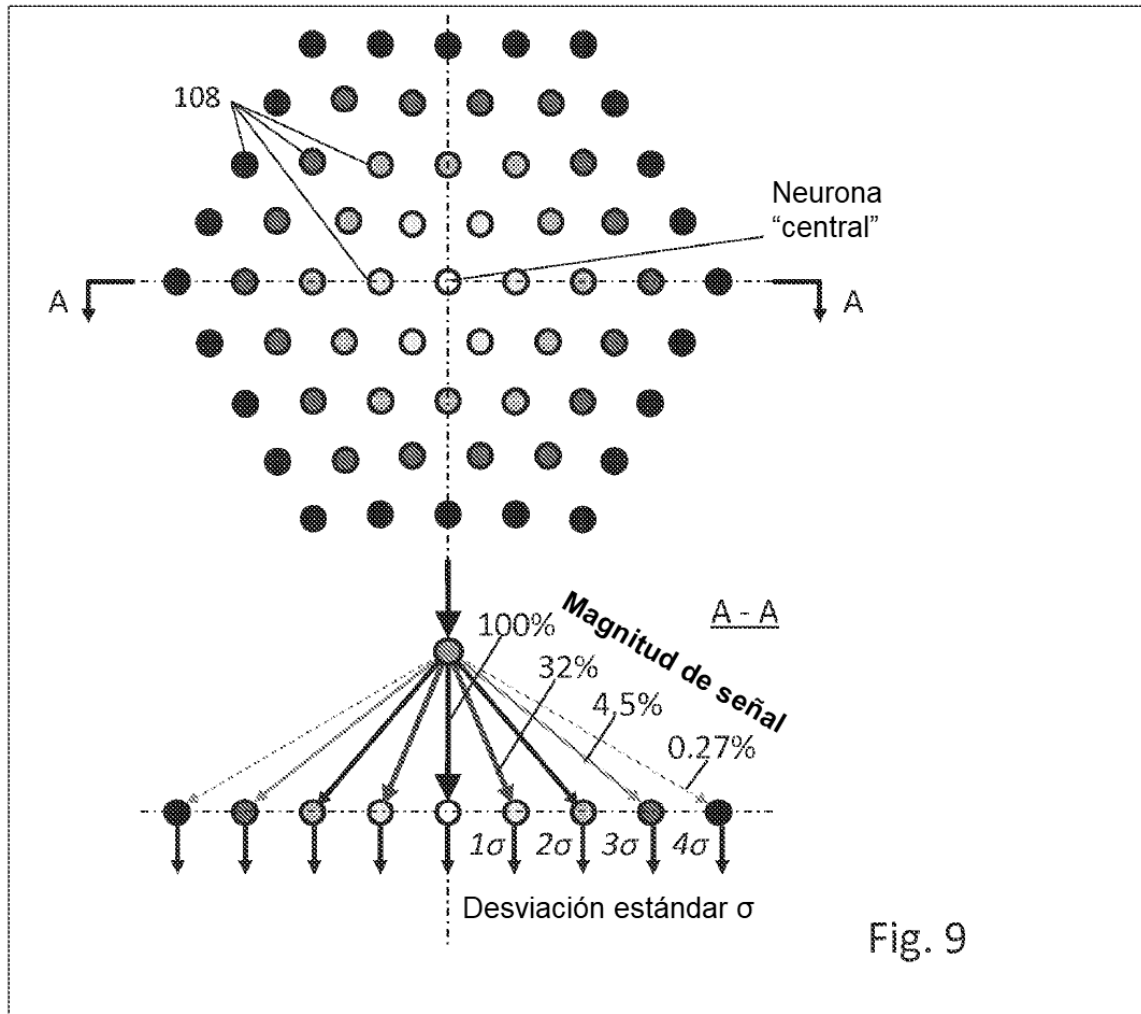
5. La red (100) neuronal implementada por ordenador de la reivindicación 4, en la que cada peso (112) correctivo se define adicionalmente por un índice de acceso ("a") configurado para contar el número de veces que la señal de entrada accede al respectivo peso correctivo durante el entrenamiento de la red neuronal.

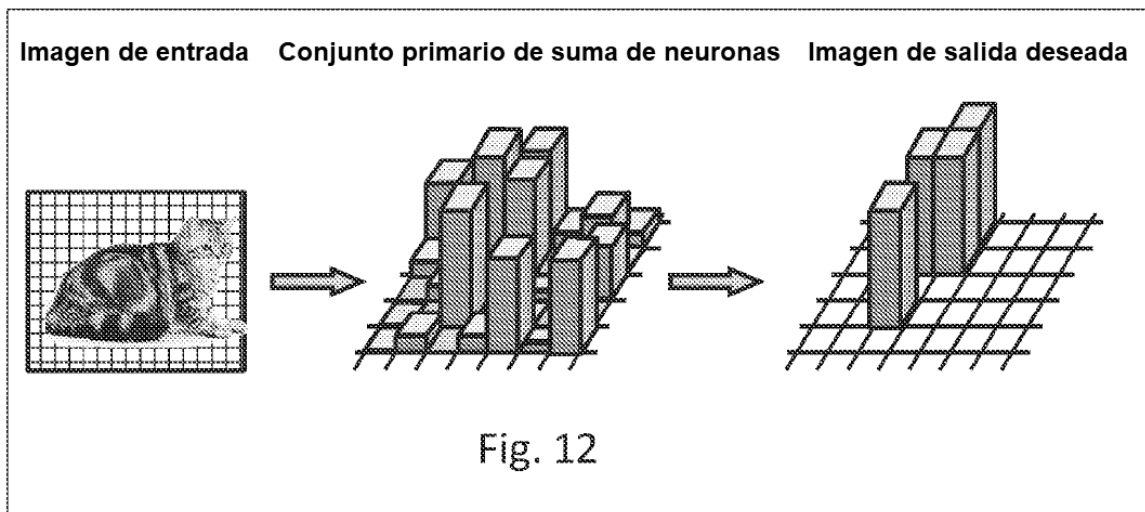
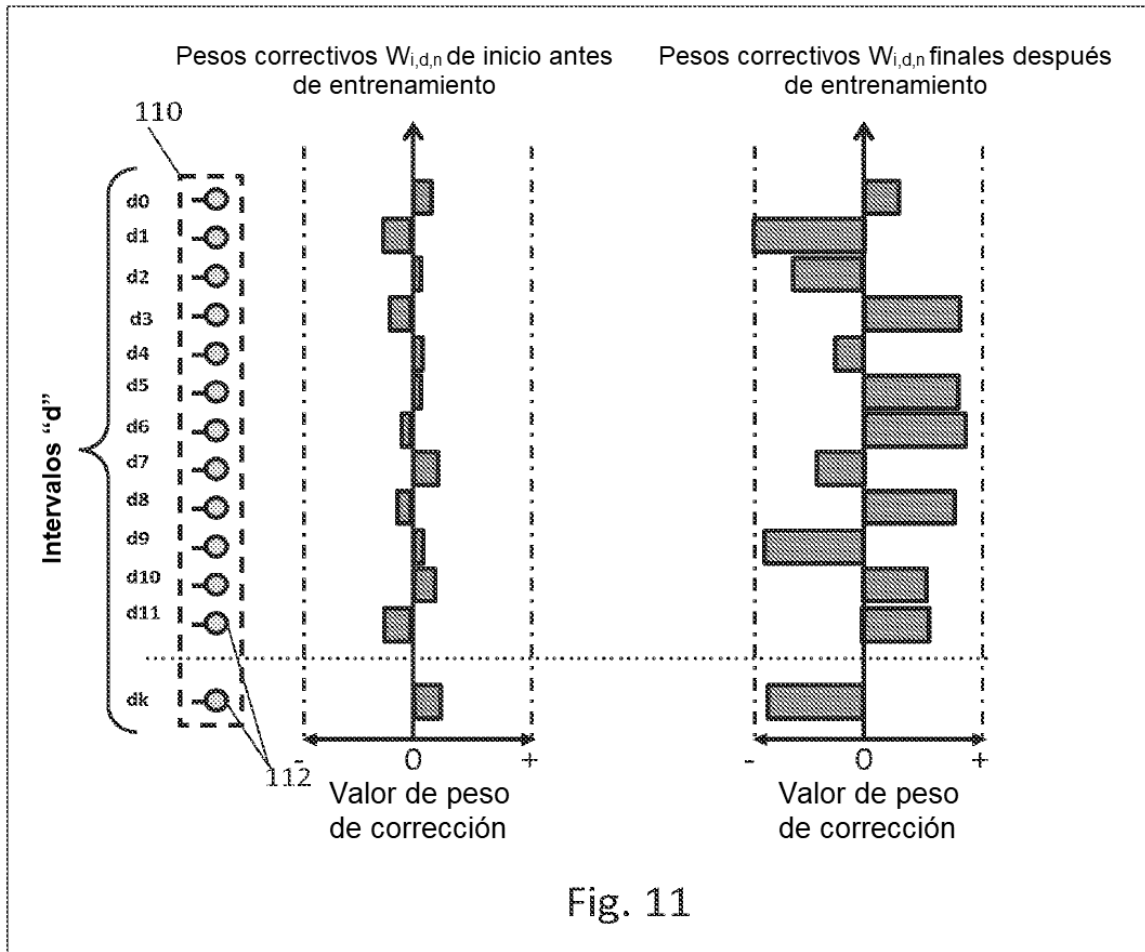


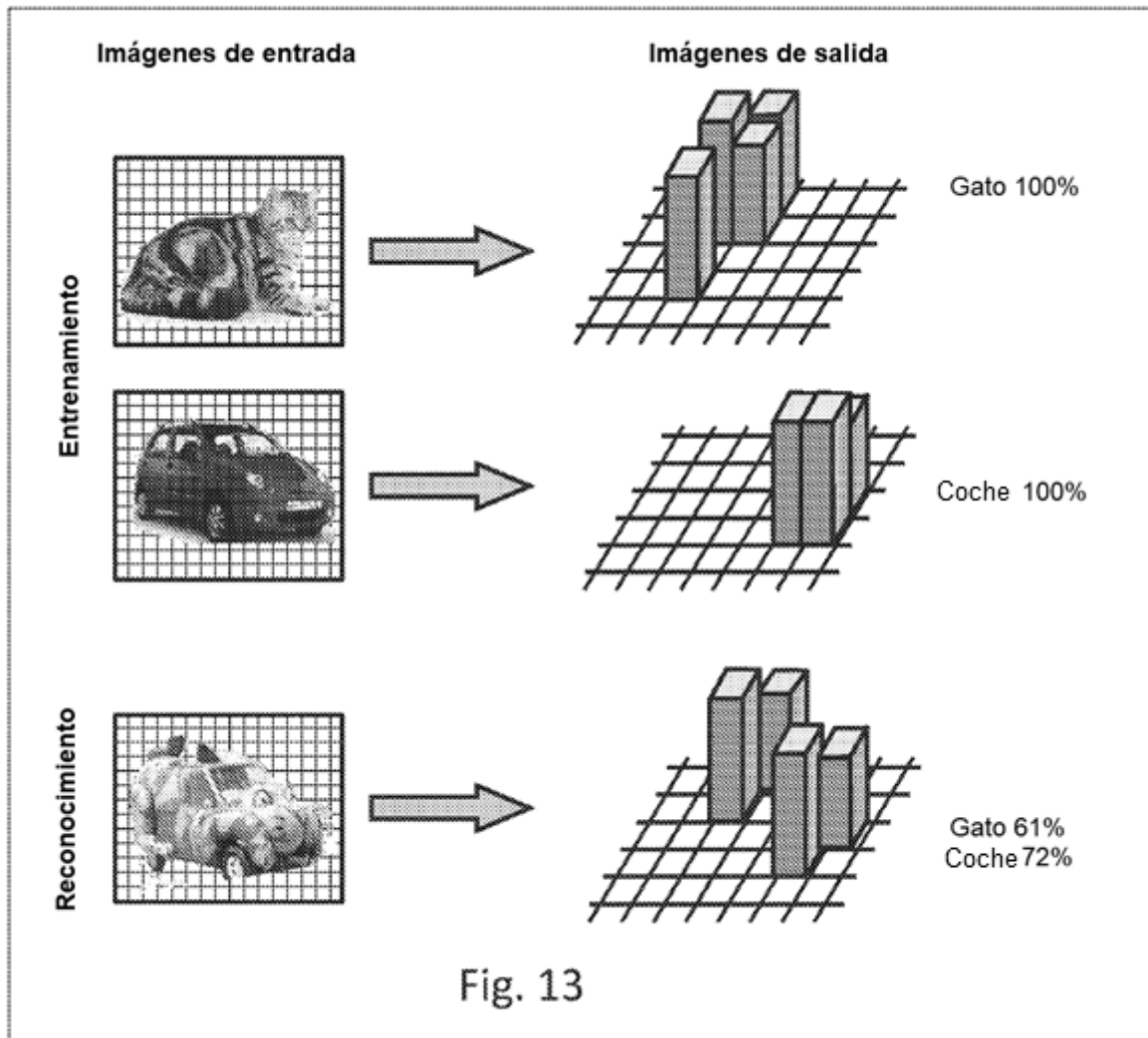


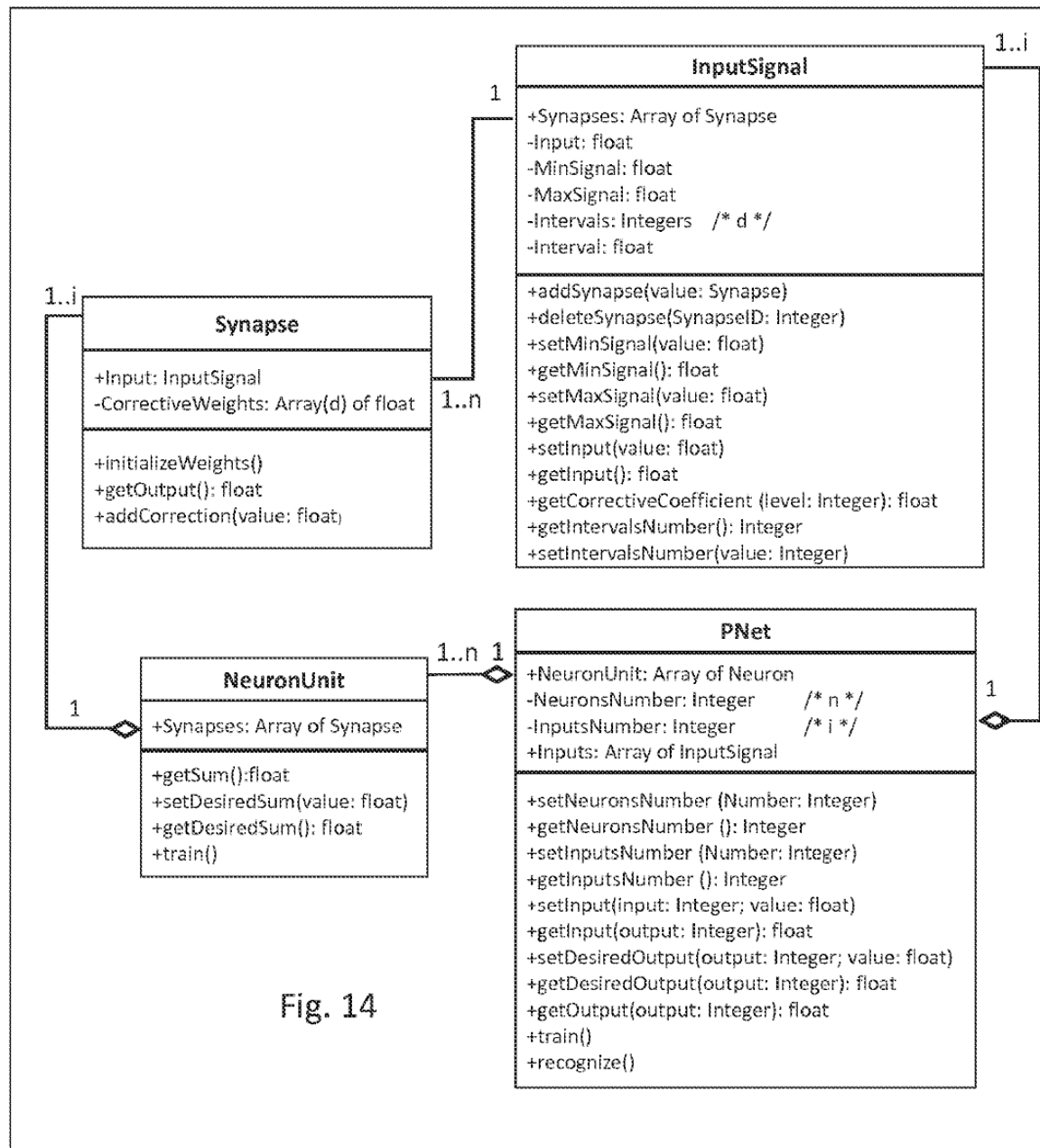


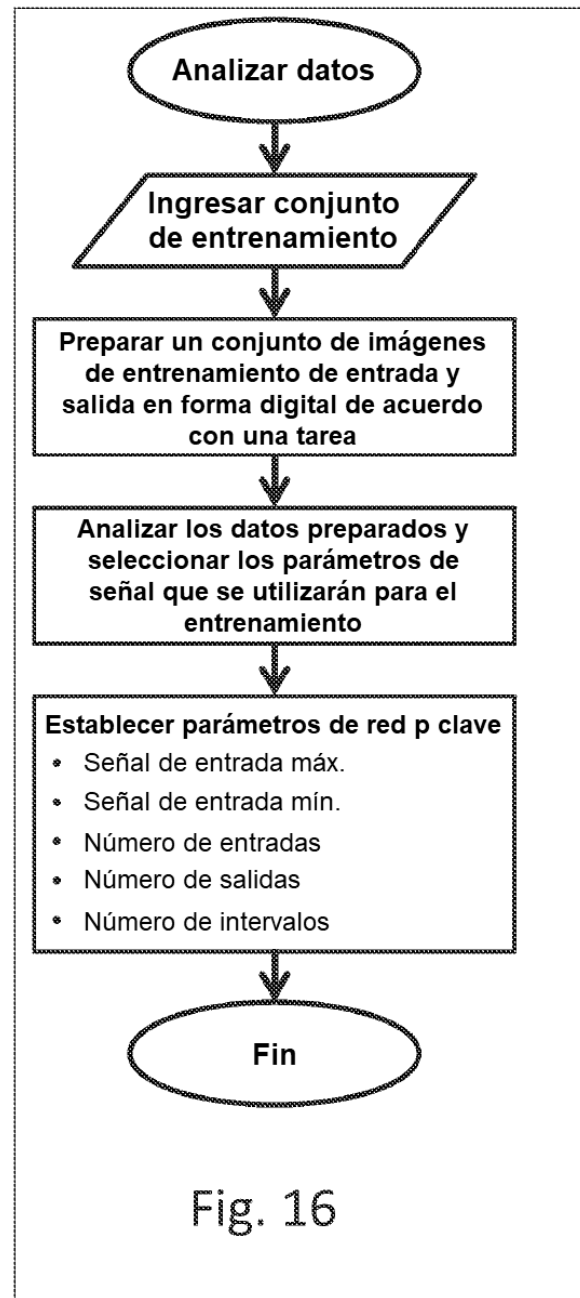
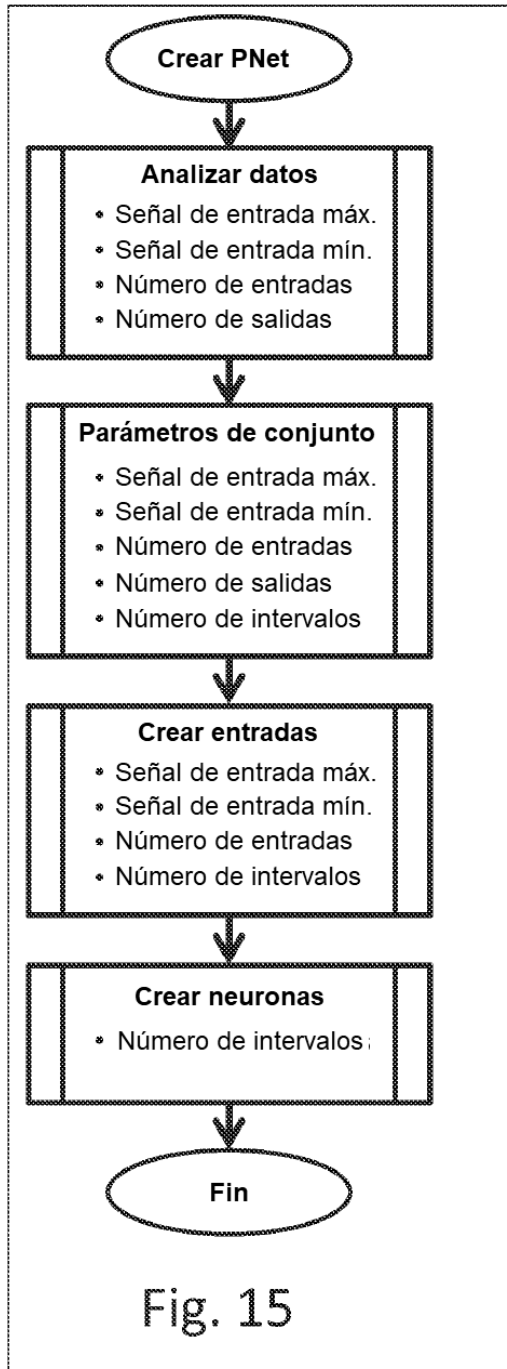


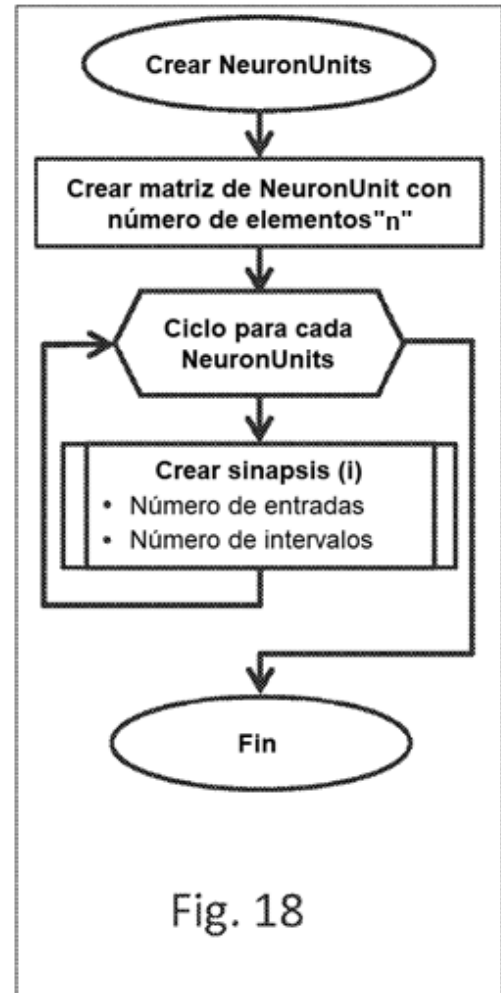
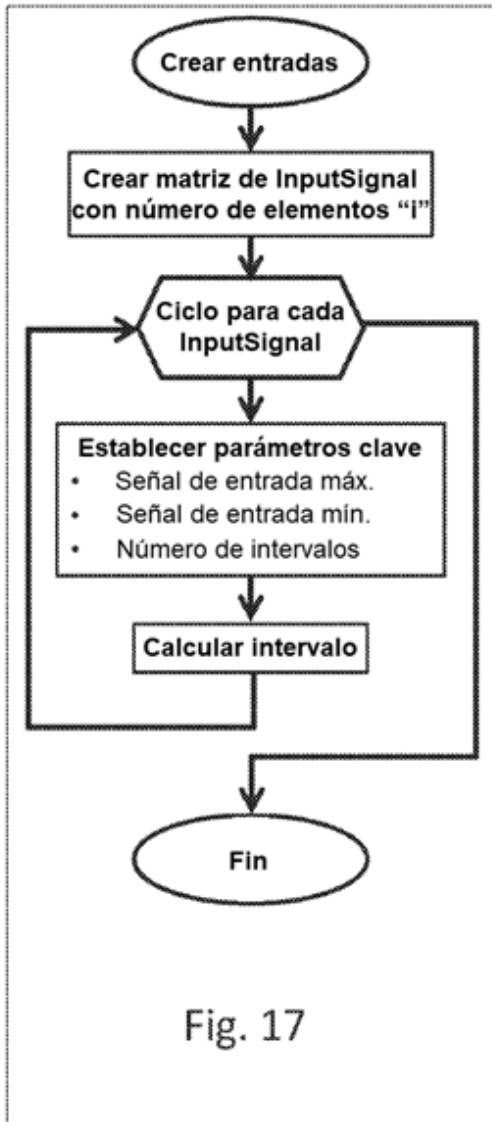


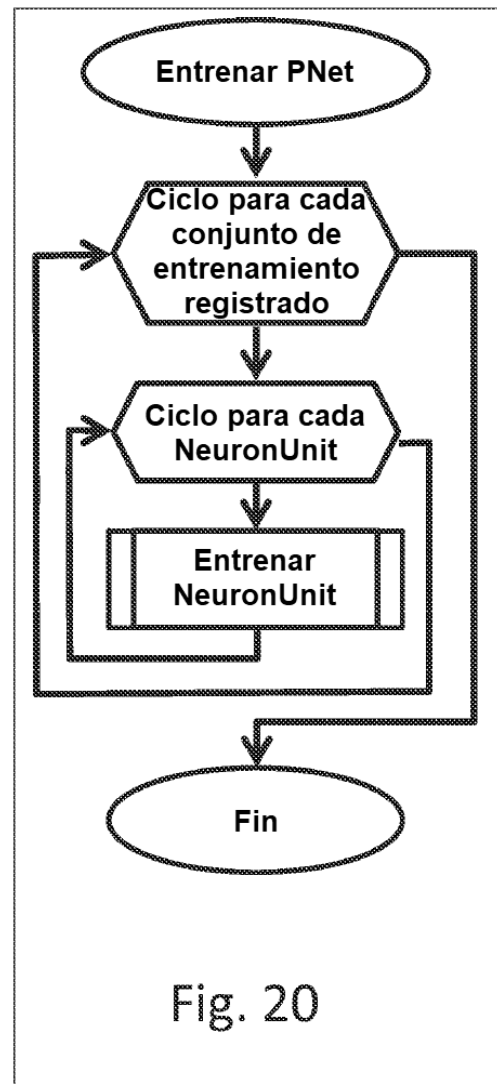
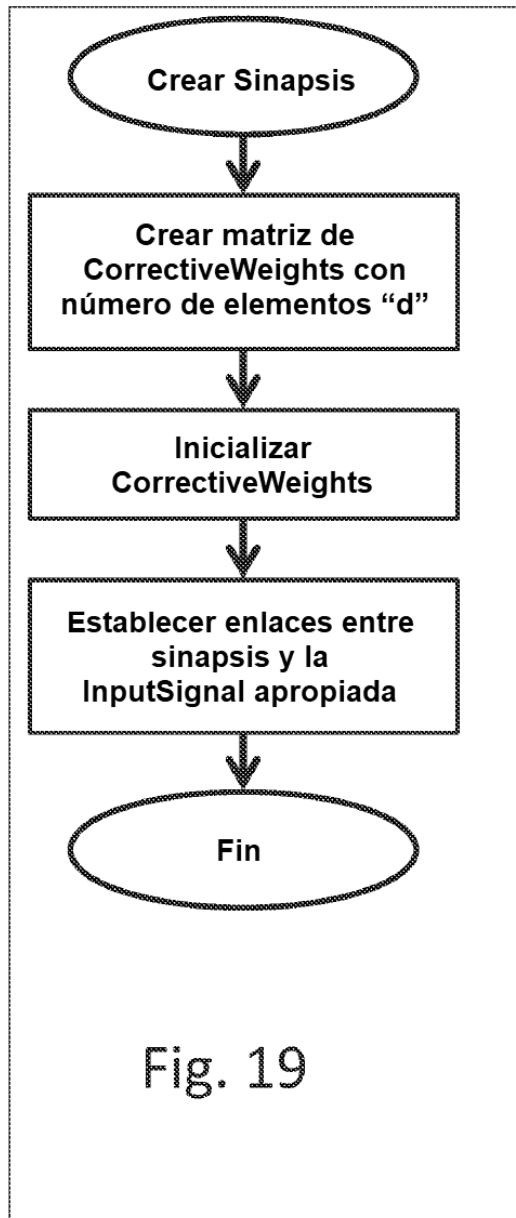












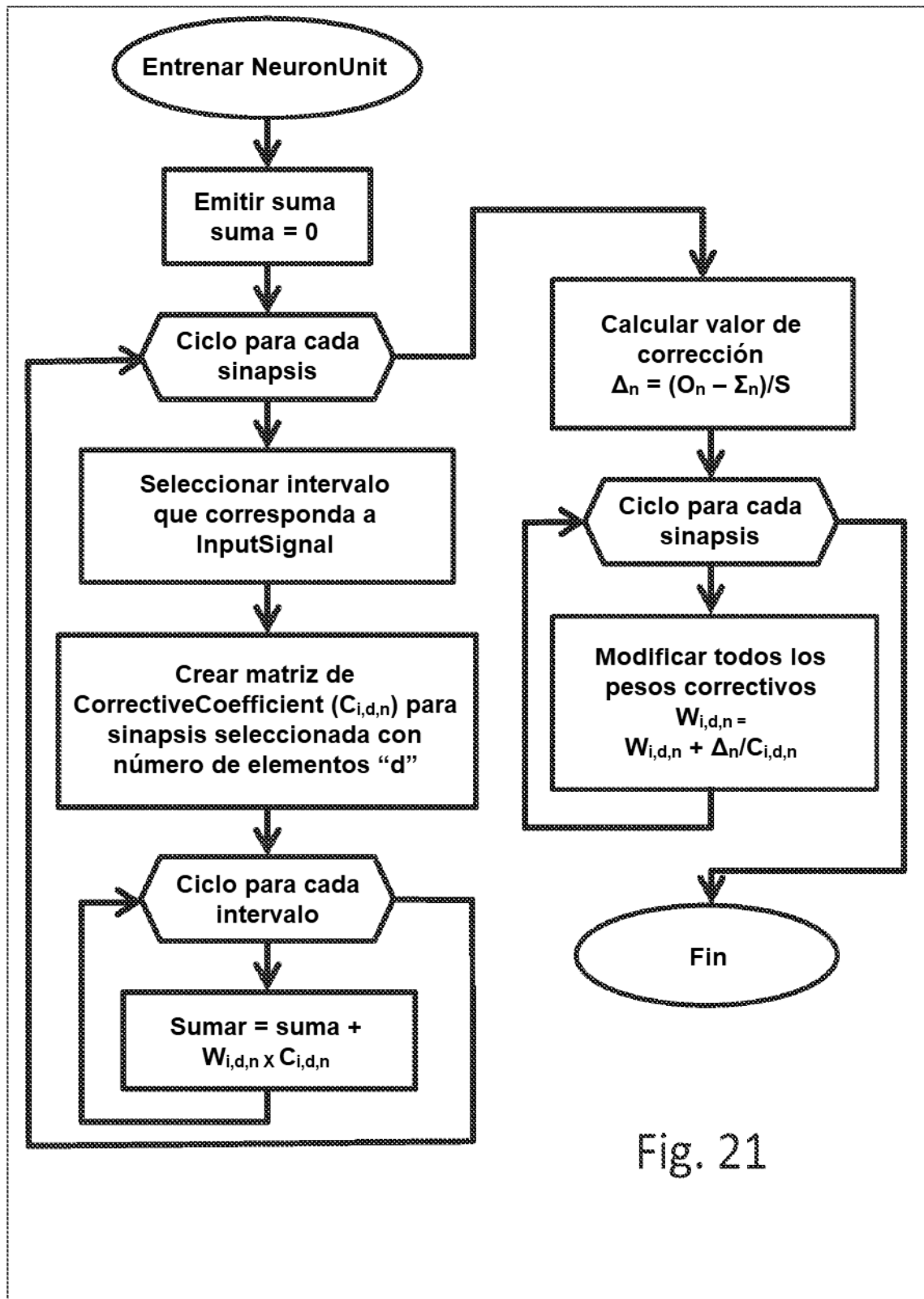


Fig. 21

