

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 849 123**

21 Número de solicitud: 202030120

51 Int. Cl.:

G06N 3/04

(2013.01)

12

PATENTE DE INVENCION CON EXAMEN

B2

22 Fecha de presentación:

13.02.2020

43 Fecha de publicación de la solicitud:

13.08.2021

Fecha de modificación de las reivindicaciones:

10.11.2021

Fecha de concesión:

28.02.2023

45 Fecha de publicación de la concesión:

07.03.2023

73 Titular/es:

**UNIVERSITAT DE LES ILLES BALEARS (100.0%)
Campus Universitario, Ctra. de Valldemossa, Km.
7,5, Edifici Son Lledó
07071 Palma de Mallorca (Illes Balears) ES**

72 Inventor/es:

**FRANCO FRASSER, Christiam Carrilo y
ROSELLO SANZ, Josep Lluís**

74 Agente/Representante:

SALVÀ FERRER, Joan

54 Título: **Elemento de generación de señales estocásticas, neurona estocástica y red neuronal a partir de esta**

57 Resumen:

La presente invención consiste en un elemento de generación de señales estocásticas que comprende un primer conversor de binario a estocástico, el cual a su vez comprende una primera entrada para recibir una señal binaria y una segunda entrada para recibir una señal aleatoria, y el cual se encuentra configurado para convertir dicha señal binaria en una primera señal estocástica utilizando dicha señal aleatoria, estando el elemento de generación de señales estocásticas, caracterizado porque comprende una unidad de procesamiento con una primera entrada para recibir dicha primera señal estocástica y una segunda entrada para recibir una señal estocástica de referencia, esta última estando generada a partir de una señal de valor constante y utilizando dicha señal aleatoria, procesándolas según al menos una función aritmética, cuyo resultado es una señal de salida estocástica representativa de dicho procesamiento. También, consiste en una neurona computacional que implementa dicho elemento de generación de señales estocásticas, así como una red neuronal que implementa dicha neurona computacional.

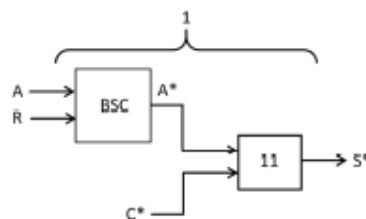


Fig. 1

Aviso: Se puede realizar consulta prevista por el art. 41 LP 24/2015.
Dentro de los seis meses siguientes a la publicación de la concesión en el Boletín Oficial de la Propiedad Industrial cualquier persona podrá oponerse a la concesión. La oposición deberá dirigirse a la OEPM en escrito motivado y previo pago de la tasa correspondiente (art. 43 LP 24/2015).

ES 2 849 123 B2

DESCRIPCIÓN

Elemento de generación de señales estocásticas, neurona estocástica y red neuronal a partir de esta

Campo de la invención

- 5 La presente invención estaría enmarcada en el campo de la tecnología electrónica y de la inteligencia computacional. Un primer aspecto de la invención consistente en un elemento de generación de señales estocásticas, basado en hardware digital y técnicas de computación probabilística, adecuado para sistemas de computación, como por ejemplo los sistemas de computación mediante redes convolucionales de neuronas estocásticas. Otro aspecto de la
- 10 invención consiste en una neurona estocástica que comprende dicho elemento de generación de señales estocásticas, así como una red neuronal (de tipo convolucional, por ejemplo, conocida como CNN por sus siglas en inglés), formada a partir de dicha neurona estocástica, con la cual se posibilita la obtención de altos grados de paralelismo y por tanto un reconocimiento de patrones a muy alta velocidad.

15 Antecedentes de la invención

En los últimos años el uso de redes neuronales extensas (Deep Neural Networks, DNN) han adquirido gran relevancia por su gran capacidad de extraer información útil a partir de grandes cantidades de datos. Su implementación hardware posibilita el aumento en velocidad de operación al realizarse la computación en paralelo comparándolo con aproximaciones

20 basadas en el uso de microprocesadores que utilizan arquitecturas de computación secuenciales del tipo Von-Neuman (soluciones Software). Las CNN son un tipo de DNN con una conexión del tipo Feed-Forward (sin realimentación), especialmente útiles para el reconocimiento de formas en imágenes, como se puede ver en las referencias siguientes:

- LeCun, Y., Bottou, L., Bengio, Y., Haffner, P. Gradient-based learning applied to document recognition (1998) Proceedings of the IEEE, 86 (11), pp. 2278-2323.
- Lawrence, S., Giles, C.L., Tsoi, A.C., Back, A.D. Face recognition: A convolutional neural-network approach (1997) IEEE Transactions on Neural Networks, 8 (1), pp. 98-113.

En concreto, una CNN es un tipo de red artificial donde las neuronas corresponden a campos receptivos de una manera muy similar a las neuronas en la corteza visual primaria de un cerebro biológico, donde su aplicación es realizada en matrices bidimensionales, por lo que son muy efectivas para tareas de visión artificial, como en la clasificación y segmentación de imágenes, entre otras aplicaciones. Consisten en múltiples capas de neuronas artificiales que actúan como filtros convolucionales de una o más dimensiones, donde después de cada capa, por lo general, se añade una función para realizar un mapeo causal no-lineal.

En ellas, a partir de la imagen a procesar (datos en bruto proporcionados por los objetos observados), se realizan una serie de cálculos mediante distintas neuronas dispuestas en capas. Cada neurona realiza un procesamiento relativamente simple a partir de sus entradas (que pueden proceder de la imagen a procesar o de otras neuronas) y la salida que proporciona consiste en una función no-lineal (llamada función de activación) de la suma ponderada de sus entradas como se indica en la ecuación siguiente:

$$y_i = f\left(\sum_{j=1}^N \omega_{ij}x_j - T_i\right)$$

En la fórmula anterior se muestra la dependencia de la salida de la neurona i -ésima (y_i) en relación a un total de N entradas definidas como x_j , así como de una selección de pesos para cada una de estas entradas y exclusivos de cada neurona (parámetros w_{ij}) y de un valor de sesgo propio de cada neurona (T_i). La función de activación f es una función no-lineal, como la función de Heaviside, la sigmoideal o la función rectificadora conocida como Unidad Lineal Rectificada (ReLU por sus siglas en inglés), la cual evalúa el máximo entre la combinación lineal de las entradas x_j de la fórmula anterior y un valor de referencia que puede ser cero.

En una red del tipo CNN, las neuronas se reúnen en capas de procesamiento, en donde cada neurona de la primera capa recibiría información directa de la imagen a procesar mientras que las capas interiores estarían alimentadas por las respuestas de las neuronas dispuestas en la capa precedente. Las funcionalidades de cada capa suelen ser de dos tipos: convolucional y de reducción. La capa convolucional consiste en la convolución de la imagen precedente mediante un kernel (lo que también se suele llamar respuesta al impulso unitario); matemáticamente, las capas convolucionales se pueden pensar como la realización de una proyección de la información de entrada sobre un determinado subespacio definido por el kernel, y en donde la reducción de información es relativamente pequeña. La segunda

funcionalidad de las capas consiste en un proceso drástico de reducción de la información en donde se seleccionan diferentes áreas de la imagen a procesar, y se selecciona un valor dominante que puede ser la señal de mayor valor (a dicho tipo de capas se le denominará max-pooling) o un valor medio de dichas señales (average pooling). El resultado final es una información reducida de la señal de entrada procedente de la capa convolucional anterior.

Con la concatenación de ambos tipos de capas neuronales (convolucionales y de pooling), se va cambiando la orientación de la información de la imagen de entrada y seleccionando determinadas características de la misma, para así ir reduciendo la información, pero aumentando el grado de abstracción. De esta forma, los bits de información de la entrada pueden representar valores de intensidad de la imagen a procesar mientras que a la salida obtendríamos una categoría que se quiere reconocer.

Por tanto, la aplicación principal de las CNN sería el procesamiento de imágenes para su reconocimiento e interpretación. De todas formas, debido a su transversalidad, el campo de las aplicaciones de las CNN es enorme y actualmente hay miles de trabajos científicos basándose en el uso de dicha arquitectura neuronal de procesamiento, abarcando procesamientos como el reconocimiento facial del trabajo de Lawrence et al., mencionado anteriormente, o el reconocimiento de texto como el divulgado en el documento de patente EP 1598770, con título: "Reconocimiento óptico de caracteres de baja resolución para documentos adquiridos con cámara".

Uno de los inconvenientes del procesamiento de las CNN es su alto coste computacional debido al gran número de operaciones que se extienden en la totalidad de las imágenes que se van generando a partir de la original (sucesivas convoluciones y procesos de pooling). Dicho coste computacional, se concreta en un tiempo de procesamiento por imagen, así como en el consumo energético asociado a dicho procesamiento, que pueden ser prohibitivos en caso de querer ser aplicados a grandes cantidades de datos (como es el caso del procesamiento de alta velocidad de vídeo). Para el caso de los sistemas tradicionales de procesamiento de datos (soluciones software), las arquitecturas de propósito generalista basadas en el uso de procesadores no son óptimas para ser aplicadas a la implementación de las CNN, y por tanto, tienen asociado tanto un consumo, como un tiempo de respuesta que son prohibitivos para según qué tipo de aplicaciones. Es por esta razón que se han ido desarrollando sistemas hardware que intentan implementar y paralelizan de forma óptima

dichos procesos. Como ejemplos de intentos de optimizar CNN mediante el uso de hardware encontramos la siguiente publicación:

- Zhang, C., Li, P., Sun, G., Guan, Y., Xiao, B., Cong, J. "Optimizing FPGA-based accelerator design for deep convolutional neural networks" (2015) FPGA 2015 - 2015 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays, pp. 161-170.

En este trabajo, Zhang y sus colaboradores muestran una metodología de optimización de CNN y la implementan en dispositivos de lógica reconfigurable (FPGAs). Su aproximación está basada en el uso de lógica digital clásica, en donde se aumenta el paralelismo en relación a las implementaciones basadas en el uso de microprocesadores generalistas pero que no explota al completo el paralelismo intrínseco de las CNN.

Con el objetivo de aumentar el paralelismo y adecuar al máximo el hardware de procesamiento a la arquitectura de las DNN en general, se han desarrollado estrategias en donde se usan lógicas digitales no convencionales como la computación estocástica. La computación estocástica es una lógica digital en donde las operaciones entre señales obedecen a reglas probabilísticas. Es el caso del trabajo de la siguiente publicación, aplicado a un tipo de redes profundas como son las Deep Belief Networks y que son un tipo de redes que se suelen utilizar para el reconocimiento de formas y que tiene su inspiración en la física estadística:

- Sanni, K., Garreau, G., Molin, J.L., Andreou, A.G. "FPGA implementation of a Deep Belief Network architecture for character recognition using stochastic computation", (2015) 2015 49th Annual Conference on Information Sciences and Systems, CISS 2015, art. no. 7086904.

Este trabajo utiliza una aproximación lineal a la función no-lineal sigmoideal, aunque no deja claro cómo implementar determinadas capas como el Max-pooling para el caso en el que las neuronas se tengan que disponer en forma de red del tipo CNN. Por otro lado, asume solamente el uso de señales descorrelacionadas para implementación de la red, lo que deja de lado la posibilidad de explotar correlaciones entre señales estocásticas y que ayudarían mucho a reducir el hardware.

En relación a la implementación de metodologías no-convencionales para la optimización del hardware pero aplicadas a las redes neuronales convolucionales, se encuentra la siguiente publicación:

- 5 - Alawad, M., Lin, M. "Stochastic-Based Deep Convolutional Networks with Reconfigurable Logic Fabric" (2016) IEEE Transactions on Multi-Scale Computing Systems, 2 (4), art. no. 7547913, pp. 242-256.

En este estudio, se utilizan ciertas propiedades de la teoría de probabilidades como es la relación entre la función densidad de probabilidad de la suma de dos variables aleatorias independientes y la densidad de probabilidad de dichas variables por separado (relacionadas mediante una convolución de ambas). La base de la aceleración del proceso de las CNN, se basa en la implementación de dicha propiedad probabilística en lugar de utilizar elementos neuronales individuales.

Otra metodología no-convencional, también basada en el uso de propiedades probabilísticas se encuentra en la siguiente publicación:

- 15 - Ren, A., Li, Z., Ding, C., Qiu, Q., Wang, Y., Li, J., Qian, X., Yuan, B. "SC-DCNN: Highly-scalable deep convolutional neural network using stochastic computing" (2017) International Conference on Architectural Support for Programming Languages and Operating Systems - ASPLOS, Part F127193, pp. 405-418.

En este trabajo se utiliza la lógica estocástica para implementar tanto el proceso de convolución como el de max-pooling. Por otro lado, utiliza máquinas de estado para implementar la función tangente-hiperbólica (función f de la mencionada ecuación), lo que puede complicar considerablemente el diseño (como se puede ver en su Figura 6). Tampoco explota el uso de señales correlacionadas para simplificar la implementación de la función de activación.

25 Por otro lado, también se encuentra una propuesta de red neuronal estocástica en la reciente publicación siguiente:

- 30 - Li, Z., Li, J., Ren, A., Cai, R., Ding, C., Qian, X., Draper, J., Yuan, B., Tang, J., Qiu, Q., Wang, Y. "HEIF: Highly Efficient Stochastic Computing-Based Inference Framework for Deep Neural Networks" (2019) IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 38 (8), art. no. 8403283, pp. 1543-1556.

En este trabajo se utiliza un bloque binario conocido como contador paralelo aproximado (APC por sus siglas en inglés), mostrado en su Fig.4c. Este se usa para realizar la suma ponderada de las entradas neuronales, pero se realiza un complicado circuito para la implementación de la función de activación tipo ReLU o del bloque de max-pooling (en Fig 6 de dicha referencia).

- 5 De hecho, la función de activación conseguida no es una ReLU clásica sino una "clipped ReLU", es decir, una ReLU saturada.

La lógica estocástica (o computación estocástica) es una alternativa a la lógica binaria tradicional en donde las cantidades se codifican mediante la frecuencia de conmutación de los bits, forzando que dicha conmutación tenga una naturaleza del tipo estocástico. La
10 computación estocástica (SC) es una metodología informática aproximada que representa señales que utilizan la frecuencia de conmutación de flujos de bits dependientes del tiempo. Cada señal SC está compuesta de pulsos que representan la probabilidad de encontrar un valor alto ('1' lógico) en una secuencia de bits. Por ejemplo, el número 0.75 podría representarse mediante un flujo de bits en el que la probabilidad de encontrar un '1' lógico a
15 lo largo del flujo de bits es el 75%, (1,1,0,1) para cuatro bits, o (0,1,1,0,1,1,1,1) para una trama de ocho bits. Esta codificación se conoce como representación unipolar y cada valor estará comprendido entre cero y uno. Para incluir valores negativos se requiere una codificación diferente, como es el caso de la codificación bipolar, donde el número de ceros se resta del número de unos y finalmente se divide por el número total de bits. Esta codificación es
20 equivalente a implementar el cambio de variables $p^* = 2p - 1$, donde 'p' es la representación unipolar del número. La codificación bipolar proporciona un rango $[-1,1]$ de valores posibles.

Una de las principales ventajas de usar la lógica estocástica es el bajo costo en recursos de hardware para implementar funciones complejas y que es dependiente también de la correlación entre señales. Por ejemplo, una puerta XNOR implementa una multiplicación en
25 codificación bipolar cuando ambas señales están temporalmente descorrelacionadas ($f = x \cdot y$), pero implementa la unidad menos el valor absoluto de la diferencia entre ambas señales en caso de estar correlacionadas ($f = 1 - |x - y|$).

En la literatura, se han propuesto diferentes diseños de neuronas estocásticas, aunque ninguno de ellos ha explotado la correlación de señales que puede simplificar las CNN de
30 manera considerable. En la presente invención, se utiliza la correlación entre las señales de salida de las neuronas para la implementación eficiente de redes neuronales convolucionales.

Descripción de la invención

La presente invención consiste principalmente en un elemento de generación de señales estocásticas, así como una neurona computacional, en particular una neurona estocástica, que comprende dicho elemento, y a una red neuronal computacional que comprende una pluralidad de tales neuronas. La neurona propuesta está orientada a implementar redes neuronales artificiales profundas, en especial, redes neuronales convolucionales.

El primer aspecto de la invención consiste en un elemento de generación de señales estocásticas, el cual comprende un primer conversor de binario a estocástico (o BSC por sus siglas en inglés), el cual suele consistir en un comparador binario complemento a dos. Este BSC, a su vez, comprende una primera entrada para recibir una señal binaria y una segunda entrada para recibir una señal aleatoria (entendiéndose por aleatoria que contiene algún tipo de aleatoriedad, ya sea aleatoria pura o pseudo-aleatoria), y se encuentra configurado para convertir a su salida dicha señal binaria en una primera señal estocástica a partir de dicha señal aleatoria.

El presente elemento de generación de señales estocásticas se caracteriza porque comprende una unidad de procesamiento, la cual comprende a su vez, una primera entrada para recibir dicha primera señal estocástica del BSC y una segunda entrada para recibir una segunda señal estocástica, utilizada como señal estocástica de referencia y generada a partir de una señal de valor constante (como por ejemplo el valor cero), habiendo utilizado la misma señal aleatoria utilizada por el BSC para generar esta segunda señal estocástica. Esta unidad de procesamiento está configurada para procesar dicha primera señal estocástica del BSC y dicha segunda señal estocástica (o señal estocástica de referencia) según al menos una función aritmética, para generar como resultado, por tanto, una señal de salida estocástica representativa de dicho procesamiento.

Con la condición de que la señal estocástica de referencia recibida por la unidad de procesamiento esté generada habiendo utilizado la misma señal aleatoria utilizada por el BSC, se consigue una correlación entre señales de diferentes elementos de generación de señales estocásticas.

Gracias a ello, resulta posible que la unidad de procesamiento aplique una sencilla función aritmética, como por ejemplo la aplicada mediante una puerta lógica de tipo OR o de tipo AND,

e implementar con ello una función de activación no saturada para obtener una señal estocástica de salida correlacionable con otros eventuales elementos de generación de señales estocásticas.

Por tanto, para un ejemplo de realización del elemento de generación de señales estocásticas del primer aspecto de la invención, la unidad de procesamiento es una puerta lógica de tipo OR, por lo que la función aritmética aplicada por esta consiste en una función *máxima* ($\max(a,b)$), que es la base para efectuar la Unidad Lineal Rectificada (o ReLU por sus siglas en inglés). Para otro ejemplo de realización la unidad de procesamiento es una puerta lógica de tipo AND, por lo que la función aritmética aplicada por esta consiste en una función de activación del tipo $\min(a, b)$.

Dicho elemento de generación de señales estocásticas, puede formar parte, por ejemplo, de una neurona estocástica para red neuronal computacional, el cual es otro aspecto de la presente invención, en particular un segundo aspecto para el que la neurona estocástica comprende un elemento de generación de señales estocásticas según el primer aspecto.

La neurona estocástica de la presente invención comprende preferiblemente un contador paralelo aproximado (APC por sus siglas en inglés) dispuesto para recibir una pluralidad de señales estocásticas de entrada y configurado para sumar dicha pluralidad de señales de entrada y convertirlas en una señal de salida en notación binaria codificada en complemento a dos a su salida. Esta señal de salida será la señal binaria de entrada del anteriormente descrito BSC del elemento de generación de señales estocásticas.

A su vez, para un ejemplo de realización, la neurona estocástica del segundo aspecto de la presente invención comprende una pluralidad de subunidades de procesamiento, cada una de ellas dispuesta para recibir una señal estocástica externa (como podría ser la señal de otras neuronas), así como una señal estocástica de peso de conexión, y configurada para procesarlas aplicándoles una función aritmética, para generar unas señales de salida. Dado que ambas señales recibidas por las subunidades de procesamiento no están correlacionadas, para un ejemplo de realización en que estas subunidades consisten en puertas lógicas XNOR, la función aplicada es la multiplicación de ambas señales estocásticas en codificación bipolar. En otra posible realización, estas consisten en una puerta lógica AND, y la función aplicada es la multiplicación de ambas señales estocásticas en codificación unipolar. Posteriormente, las señales de salida generadas por estas subunidades de

procesamiento se encuentran vinculadas operativamente con dicho APC, de forma que correspondan a dichas señales estocásticas de entrada del APC.

Otro aspecto de la presente invención, en particular un tercer aspecto, consiste en una red neuronal computacional implementada con una pluralidad de las neuronas estocásticas
5 definidas según el segundo aspecto de la invención, descritas anteriormente, donde unas se encuentran interconectadas operativamente con otras de ellas.

Preferiblemente, la presente red neuronal computacional comprende un segundo conversor de binario a estocástico o BSC configurado para generar dicha señal estocástica de referencia a partir de la entrada de una señal de valor constante y la entrada de dicha señal aleatoria, y
10 enviarla a las mencionadas unidades de procesamiento de las diferentes neuronas de forma simultánea. Así mismo, para un ejemplo de realización, la red neuronal del tercer aspecto de la presente invención comprende un generador de números aleatorios configurado para generar dicha señal aleatoria y enviarla a los diferentes conversores de binario a estocástico de forma simultánea, correlacionando por tanto todas las señales estocásticas generadas por
15 los APCs de las neuronas.

En la mayoría de las implementaciones de CNN, las operaciones comunes que se utilizan son: la multiplicación, la suma y la función máxima o ReLU. Esas operaciones pueden implementarse fácilmente en circuitería estocástica si la correlación se utiliza correctamente. Las principales ventajas aportadas al utilizar señales correlacionadas para la implementación
20 de redes neuronales convolucionales consisten en:

- a) un ahorro en recursos de hardware utilizados, al no necesitar implementar diferentes generadores de números aleatorios para cada neurona, sin perder precisión en el resultado(en las implementaciones con lógica estocástica, el mayor porcentaje de recursos se utilizan en la generación de los números aleatorios/ aleatorios),
- 25 b) una simplificación en la implementación de la función max-pooling y de la convolución mediante sólo la inclusión de una unidad de computación simple, como puede ser una puerta lógica (diferentes esfuerzos y diseños se han presentado en la literatura para poder implementar dicha función en redes neuronales, pero al utilizar señales no-correlacionadas, los diseños propuestos necesitan de gran espacio en hardware),

- c) la posibilidad de agregar más capas de profundidad en la red neuronal sin necesidad de utilizar varios generadores de números aleatorios, ya que la cantidad de generadores es constante independientemente del número de capas de la red (al contrario que en otras implementaciones, donde a medida que las capas de la red aumentan, la cantidad de generadores de números aleatorios también).

En una realización preferente de la red neuronal computacional, el generador de números aleatorios es del tipo Linear Feedback Shift Register (LFSR), dado su reducido coste en recursos de hardware y su fácil implementación.

Para un ejemplo de realización, la red neuronal computacional del tercer aspecto de la presente invención comprende una puerta OR grupal, configurada para recibir las señales de salida estocásticas de un grupo de neuronas estocásticas y proveer el valor máximo de ellas a su salida; o bien, implementar una función del tipo Min-pooling substituyendo la puerta OR grupal por una puerta AND grupal.

Para otro ejemplo de realización, la red neuronal computacional comprende una matriz de conversores de binario a estocástico, estando sus conversores configurados para convertir una señal inicial recibida en sus respectivas primeras entradas y convertirla a sus salidas en una señal inicial estocástica, utilizando dicha señal aleatoria recibida en sus respectivas segundas entradas.

De forma preferente, la red neuronal de la presente invención comprende un segundo generador de números aleatorios, así como una matriz de conversores de binario a estocástico. Los conversores de esta matriz están configurados para convertir una señal de peso de conexión recibida en una primera entrada y convertirla a su salida en dicha señal de peso estocástica, utilizando para ello la mencionada señal aleatoria del segundo generador. Así pues, se pueden utilizar señales descorrelacionadas para realizar el producto entre entradas y pesos, y utilizar señales correlacionadas para generar la función de transferencia de salida de las neuronas.

En una primera realización, la red neuronal comprende al menos una capa del tipo Max-pooling, es decir, se encuentra implementada con neuronas que comprenden elementos de generación de señales estocásticas cuya unidad de procesamiento es una puerta lógica de tipo OR. En una segunda realización, la red neuronal comprende una capa del tipo Min-pooling, es decir, se encuentra implementada con neuronas que comprenden elementos de

generación de señales estocásticas cuya unidad de procesamiento es una puerta lógica de tipo AND. Y en una realización aún más preferente, la red neuronal se encuentra implementada como una red convolucional que comprende más de una capa de tipo Max-pooling, así como de tipo Min-pooling.

- 5 En redes neuronales convolucionales, óptimas para el procesamiento de imágenes 2D, 3D o en general con una estructura N-dimensional, toda la red se dispone en distintas capas de neuronas. Después de extraer determinadas características mediante una capa neuronal de convolución, se suele aplicar una operación de sub-muestreo para reducir las dimensiones del proceso de datos en la siguiente capa. Como se ha comentado, una de las operaciones
- 10 más utilizadas es el bloque Max-pooling, donde el sub-muestreo se lleva a cabo extrayendo el valor máximo de la salida de cada neurona que está comprendida dentro de una ventana de la capa de neuronas convolucional. Las implementaciones del estado de la técnica usan un conjunto de bloques (acumulador, comparador y contadores) para implementar la función Max-pooling, lo que implica un gran uso de recursos. Sin embargo, con la presente invención,
- 15 se consigue aprovechar la correlación conseguida para todas las neuronas de la red (al usar la misma señal aleatoria en todos los conversores BSC de cada neurona), y así, implementar la función de agrupación máxima mediante una sola puerta lógica OR. Es decir, resulta posible implementar la función Max-pooling (muy utilizada en la implementación de redes neuronales profundas) a las señales de salida de las neuronas de una capa de la red mediante una puerta
- 20 OR, o bien implementar una función del tipo Min-pooling mediante una puerta AND.

Breve descripción de las figuras

Figura 1.- Representación esquemática del elemento de generación de señales estocásticas de la presente invención.

- Figura 2.- Representación esquemática de una realización preferente de una neurona estocástica que utiliza una puerta lógica OR y con ello implementa una función de activación del tipo ReLU.
- 25

Figura 3.- Representación esquemática de una realización preferente de una capa de tipo max-pooling de una red neuronal, que comprende dos neuronas estocásticas con sendas puertas lógicas OR.

Figura 4.- Representación esquemática de una realización preferente de red neuronal de una capa de tipo max-pooling con cuatro neuronas, cuya salida se dirige a una puerta lógica OR para generar la función de transferencia de salida de las neuronas, en este caso siendo la señal máxima de todas ellas.

- 5 Figura 5.- Representación esquemática de una realización preferente de red neuronal convolucional con varias capas, que utiliza señales descorrelacionadas para realizar el producto entre entradas y pesos, y señales correlacionadas para generar la función de transferencia de salida de las neuronas.

- Figura 6. - Tabla comparativa del rendimiento de la presente red neuronal respecto a otras
10 implementaciones de red neuronal.

DESCRIPCIÓN DE REALIZACIONES PREFERENTES

A continuación, se describen las realizaciones preferentes de las diferentes invenciones reivindicadas, haciendo referencia a las figuras de la 1 a la 6.

- En la Figura 1 se muestra una representación esquemática del elemento de generación de
15 señales estocásticas, objeto de un primer aspecto de la presente invención. Como se puede apreciar, el elemento de generación de señales estocásticas (1) comprende un primer conversor de binario a estocástico (BSC), el cual a su vez comprende una primera entrada para recibir una señal binaria (A) y una segunda entrada para recibir una señal aleatoria (R), y a su salida se encuentra convertida dicha señal binaria (A) en una señal estocástica (A*). A
20 continuación, comprende una unidad de procesamiento (11) que recibe dicha señal estocástica (A*) y una señal estocástica de referencia (C*). Esta última, está generada a partir de una señal de valor constante (C) y utilizando dicha señal aleatoria (R). La salida de esta unidad de procesamiento (11) es una señal de salida estocástica (S*) representativa de la función aritmética procesada.

- 25 En relación a un segundo aspecto de la presente invención, la Figura 2 muestra una representación esquemática de una realización preferente de una neurona estocástica (10), que incluye una realización preferente del elemento de generación de señales estocásticas (1), la cual comprende una puerta lógica OR como unidad de procesamiento para implementar una función de activación del tipo ReLU a su salida (S*). En esta realización de neurona
30 estocástica (10), a su vez, comprende una pluralidad de puertas lógicas XNOR, cada una de

ellas dispuesta para recibir una señal estocástica externa ($X1^* - Xn^*$) y una señal de peso estocástica ($w1^* - wn^*$). Considerando que estas dos señales no están correlacionadas, la función aplicada mediante las puertas XNOR consiste en la multiplicación de ambas señales estocásticas en codificación bipolar.

- 5 Posteriormente, las salidas de dichas puertas XNOR de la presente realización de neurona estocástica (10) se encuentran vinculadas operativamente con las entradas de un contador paralelo aproximado (APC), el cual se encuentra configurado para hacer el sumatorio de dichas señales de entrada ($Y1^* - Yn^*$) y convertirlas en notación binaria codificada en complemento a dos a su salida. Dicho de otra forma, el contador paralelo aproximado (APC)
- 10 evalúa cuantas señales a '1' hay presentes a las salidas de las puertas XNOR y proporciona a su salida una señal binaria del sumatorio. A su vez, dicho sumatorio se envía al conversor de binario a estocástico (BSC) del elemento de generación de señales estocásticas (1). Tal y como se ha comentado, esta realización de elemento de generación de señales estocásticas (1) comprende una puerta lógica OR para implementar una función de activación del tipo ReLU
- 15 a su salida (S^*) mediante una misma señal aleatoria (R) recibida en su conversor de binario a estocástico (BSC).

- La Figura 3 muestra esquemáticamente una realización de ejemplo de la conexión de dos neuronas de una red neuronal del conexionado de una red neuronal, lo cual es objeto de un tercer aspecto de la presente invención. Esta realización comprende dos neuronas
- 20 estocásticas (10) con sendas puertas lógicas OR, tales como la mostrada en la Figura 2 y descrita anteriormente. Así mismo, comprende un segundo conversor de binario a estocástico (BSC2), configurado para generar dicha señal estocástica de referencia (C^*) y enviarla a las puertas lógicas OR de ambas neuronas estocásticas (10). Como se puede apreciar, la señal estocástica de referencia (C^*) se genera en el segundo conversor de binario a estocástico
- 25 (BSC2) a partir de una señal de valor constante (C) y dicha señal aleatoria (R).

- Como un experto en la materia podrá deducir, la implementación para generar señales neuronales correlacionadas de la presente invención contrasta con las implementaciones del estado de la técnica, donde resulta más complicado generar una función ReLU, que además tiene el inconveniente de estar saturada, y por tanto, no ser la ReLU estándar utilizada en los
- 30 típicos procesos de Machine Learning.

Como efecto adicional de la presente invención, tal y como se puede deducir en la Figura 4, al estar todas las salidas de las neuronas de la red correlacionadas, resulta posible

implementar de forma eficiente la función Max-pooling (muy utilizada en la implementación de redes neuronales profundas) a las señales de salida estocásticas ($S_0^* - S_3^*$) de un grupo de neuronas estocásticas ($n_0 - n_3$) de una capa de una red mediante una puerta OR grupal (3) y proveer el valor máximo de ellas a su salida (S_{max}^*); o bien, implementar una función del tipo

5 Min-pooling substituyendo la puerta OR grupal (3) por una puerta AND grupal.

La Figura 5 es un diagrama de bloques donde se muestra de una manera más general una red neuronal, donde se utilizan dos generadores de números aleatorios (2, 2'). Como se aprecia, el primer generador de números aleatorios (2) se utiliza para la conversión de las señales de entrada (x), la señal de referencia (0) y las señales de salida de los contadores paralelos aproximados de las neuronas ($n_0 - n_n$, $n'_0 - n'_n$) de una red convolucional. El

10 segundo generador de números aleatorios (2') se utiliza únicamente para la conversión a estocástico de los pesos (w) de la red.

Como se puede deducir, la salida de cada capa de la red neuronal de la presente invención es compatible con las señales de entrada de las neuronas de la siguiente capa sin ningún

15 riesgo de imprecisión, ya que al ser generadas mediante un primer generador de números aleatorios (2), se pueden multiplicar con las señales estocásticas de pesos (w^*), las cuales no están correlacionadas con dichas señales de entrada, al ser generadas por matrices de conversores de binario a estocástico (BSC array') que utilizan los números aleatorios provenientes de un segundo generador de números aleatorios (2').

20 Como se puede observar, la presente implementación de neuronas estocásticas permite que toda la red neuronal se ejecute con solo dos generadores de números aleatorios (2, 2') para toda la red, algo que no se ha considerado todavía en el estado de la técnica, y que simplifica el coste en términos de puertas digitales que se han de usar para implementar toda la red.

En la Figura 6 se muestra una tabla comparativa del rendimiento de la presente y de otras

25 implementaciones de red neuronal profunda, la que se conoce en el área de Machine Learning como LeNet-5 (tal y como se muestra en la publicación *Y. LeCun, L. Bottou, Y. Bengio and P. Haffner: Gradient-Based Learning Applied to Document Recognition, Proceedings of the IEEE, 86(11):2278-2324, November 1998*). Las implementaciones hardware con las que se compara el modelo propuesto son los siguientes:

- 30 - FPGA16: S. I. Venieris and C. Bouganis, "fpgaconvnet: A framework for mapping convolutional neural networks on fpgas," in 2016 IEEE 24th Annual International

Symposium on Field-Programmable Custom Computing Machines (FCCM) , May 2016, pp. 40–47.

- FPGA17a: Z. Liu, Y. Dou, J. Jiang, J. Xu, S. Li, Y. Zhou, and Y. Xu, “Throughput-optimized fpga accelerator for deep convolutional neural networks,” TETS, vol. 10, pp. 17:1–17:23, 2017.
- FPGA17b: Z. Li, L. Wang, S. Guo, Y. Deng, Q. Dou, H. Zhou, and W. Lu, “Laius: An 8-bit fixed-point cnn hardware inference engine,” in 2017 IEEE International Symposium on Parallel and Distributed Processing with Applications and 2017 IEEE International Conference on Ubiquitous Computing and Communications (ISPA/IUCC), Dec 2017, pp. 143–150.
- FPGA18: S.-S. Park, K.-B. Park, and K. Chung, “Implementation of a cnn accelerator on an embedded soc platform using sdsoc,” 02 2018, pp. 161–165.

Para realizar la comparación se ha implementado la red LeNet-5 en una FPGA. Este diseño de CNN está orientado a procesar una base de datos muy estandarizado en Machine Learning sobre reconocimiento de números manuscritos (MNIST) y que está compuesto por 60.000 imágenes de entrenamiento y 10.000 imágenes de test. La arquitectura CNN del tipo LeNet-5 consta de dos capas convolucionales y tres capas completamente conectadas (Fully Connected). La tasa de aciertos en MNIST de dicha CNN es del orden de 98.5%, mientras que la implementación hardware que utiliza el modelo neuronal propuesto es de 97.6% (un tanto menor que en software al binarizar señales y al tratarse de una metodología de computación probabilística).

La implementación se ha llevado a cabo utilizando un FPGA Arria 10, 10AX115H2F34E1SG que funciona a una frecuencia de reloj de 150MHz con una precisión de ocho bits para las señales binarias.

Como se puede apreciar, el método propuesto supera a las demás arquitecturas en términos de rendimiento y eficiencia energética. Los resultados muestran que la implementación de la CNN estocástica propuesta logra un rendimiento de 27 veces más (medido en inferencias por segundo y por megahercio) comparado con FPGA17a y de 6.3 veces más eficiencia

energética (medido en inferencias por Julio) comparado con FPGA17b, lo que lo hace prometedor para aplicaciones de sistemas empotrados.

Gracias al nuevo diseño de neurona propuesto, que reduce el consumo de área de todo el sistema, la arquitectura propuesta ajusta una CNN completa en una sola FPGA y de forma totalmente paralela, al contrario de lo realizado en las otras tres implementaciones que solamente implementan una parte de la red, iterando de forma recurrente (implementación secuencial).

Este paralelismo agregado es la razón principal de la menor latencia del diseño propuesto, ya que no es necesaria la sucesiva iteración de la red.

En la tabla de la figura 6 también se destacan los recursos de hardware necesarios por tarea. Las implementaciones anteriores necesitan un área de gran tamaño mediante el uso de bloques de hardware específicos, como memoria RAM y los bloques de procesamiento de señal digital (DSP). En la presente propuesta, los bloques DSP se evitan ya que se utiliza una técnica de computación no convencional (Stochastic Computing) en lugar de la lógica binaria clásica. Al mismo tiempo, no se requieren bloques de memoria ya que el cálculo no se realiza de forma recursiva, lo que reduce la fuente principal de consumo de energía, que es la transferencia de datos y las operaciones de acceso a la memoria con la memoria fuera del chip.

Lo que caracteriza el primer aspecto de la presente propuesta es la combinación de un conversor de binario a estocástico (BSC) con una unidad de procesamiento (11) que utilizan señales estocásticas correlacionadas. Otro aspecto de la invención es su posterior combinación con una matriz de subunidades de procesamiento, como por ejemplo puertas XNOR, y un contador paralelo aproximado (APC) vinculado al conversor de binario a estocástico (BSC) para implementar una neurona estocástica (10). Y un tercer aspecto es su implementación en una red con un generador de números aleatorios (2) común para todas las neuronas. Considerando que el generador de números aleatorios (2) está generando una señal aleatoria (R) de un número tanto negativo como positivo (la probabilidad de que el bit de signo generado sea igual a '0' o '1' es del 50%), la señal estocástica de nivel cero consistirá en un bit que aleatoriamente está fluctuando entre 0 y 1, con un 50% de probabilidad para cada nivel. El punto clave de esta invención es el de combinar correctamente dicho valor estocástico de referencia con la señal procedente del contador paralelo aproximado (APC), la cual ha sido convertida a estocástico por el conversor de binario a estocástico (BSC) utilizando

la misma señal aleatoria (R). Por lo tanto, la señal estocástica de nivel cero y la señal estocástica (A^*) proporcionada por el conversor de binario a estocástico (BSC) estarán completamente correlacionadas, y su combinación, por ejemplo, en una puerta OR, podrá proporcionar la señal máxima.

REIVINDICACIONES

1. Elemento de generación de señales estocásticas (1) que comprende:

- un primer conversor de binario a estocástico (BSC), el cual a su vez comprende una primera entrada para recibir una señal binaria (A) y una segunda entrada para recibir una señal aleatoria (R), y el cual se encuentra configurado para convertir dicha señal binaria (A) en una primera señal estocástica (A*) utilizando dicha señal aleatoria (R),
- una unidad de procesamiento (11) que comprende una primera entrada para recibir dicha primera señal estocástica (A*) y una segunda entrada para recibir una segunda señal estocástica (C*), esta última estando generada a partir de una señal de valor constante (C), estando dicha unidad de procesamiento (11) configurada para procesar dicha primera señal estocástica (A*) y dicha señal estocástica de referencia (C*) según al menos una función aritmética, para generar, como resultado, una señal de salida estocástica (S*) representativa de dicho procesamiento,

estando el elemento de generación de señales estocásticas (1) **caracterizado** por que la segunda señal estocástica (C*) está generada a partir de la misma señal aleatoria (R) que la utilizada para convertir la primera señal estocástica (A*).

2. Elemento de generación de señales estocásticas (1) según la reivindicación 1, caracterizado por que la unidad de procesamiento (11) es una puerta lógica de tipo OR, por lo que la función aritmética aplicada por esta consiste en una función de activación del tipo Unidad lineal rectificadora ReLU.

3. Elemento de generación de señales estocásticas (1) según la reivindicación 1, caracterizado por que la unidad de procesamiento (11) es una puerta lógica de tipo AND, por lo que la función aritmética aplicada por esta consiste en una función de activación del tipo min (A*, C*).

4. Neurona estocástica (10) para red neuronal computacional, caracterizada por que comprende un elemento de generación de señales estocásticas (1) según cualquiera de las reivindicaciones anteriores.

5. Neurona estocástica (10) para red neuronal computacional según la reivindicación 4, caracterizada por que comprende un contador paralelo aproximado (APC) dispuesto para

recibir una pluralidad de señales estocásticas de entrada ($Y1^* - Yn^*$), el cual se encuentra configurado para hacer un sumatorio de dicha pluralidad de señales de entrada ($Y1^* - Yn^*$) y convertirlas en una señal de salida en notación binaria codificada en complemento a dos a su salida, siendo dicha señal de salida la señal binaria (A) de entrada de dicho conversor de binario a estocástico (BSC).

6. Neurona estocástica (10) para red neuronal computacional según la reivindicación 5, caracterizada por que comprende una pluralidad de subunidades de procesamiento, cada una de ellas dispuesta para recibir una señal estocástica externa ($X1^* - Xn^*$) y una señal de peso estocástica ($w1^* - wn^*$), y configurada para procesarlas aplicándoles una función aritmética, para generar unas señales de salida que constituyen las señales estocásticas de entrada ($Y1^* - Yn^*$) de dicho contador paralelo aproximado (APC).

7. Neurona estocástica (10) para red neuronal computacional según la reivindicación 6, caracterizada por que dichas subunidades de procesamiento consisten en puertas lógicas XNOR, cada una configurada para multiplicar de forma bipolar dicha señal estocástica externa ($X1^* - Xn^*$) con la correspondiente señal de peso estocástica ($w1^* - wn^*$).

8. Neurona estocástica (10) para red neuronal computacional según la reivindicación 6, caracterizada por que dichas subunidades de procesamiento consisten en puertas lógicas AND, cada una configurada para multiplicar de forma unipolar dicha señal estocástica externa ($X1^* - Xn^*$) con la correspondiente señal de peso estocástica ($w1^* - wn^*$).

9. Red neuronal computacional caracterizada por que comprende una pluralidad de neuronas estocásticas (10) según descritas en cualquiera de las reivindicaciones de la 4 a la 8, estando unas de dicha pluralidad de neuronas estocásticas (10) interconectadas operativamente con otras de ellas.

10. Red neuronal computacional según la reivindicación 9, caracterizada por que comprende un segundo conversor de binario a estocástico (BSC2), configurado para generar dicha señal estocástica de referencia (C^*) a partir de una entrada de señal de valor constante (C) y una entrada de dicha señal aleatoria (R), y enviarla a los primeros conversores de binario a estocástico (BSC) y a las diferentes unidades de procesamiento (11) de dicha pluralidad de neuronas estocásticas (10) de forma simultánea.

11. Red neuronal computacional según la reivindicación 10, caracterizada por que comprende un generador de números aleatorios (2) configurado para generar dicha señal aleatoria (R), y enviarla a los diferentes conversores de binario a estocástico (BSC, BSC2) de dicha pluralidad de neuronas estocásticas (10) de forma simultánea.
- 5 12. Red neuronal computacional según la reivindicación 11, caracterizada por que el generador de números aleatorios (2) es del tipo Linear Feedback Shift Register.
13. Red neuronal computacional según cualquiera de las reivindicaciones de la 9 a la 12, caracterizada por que comprende una puerta OR grupal (3), configurada para recibir las señales de salida estocásticas (S_0^* - S_3^*) de un grupo de neuronas estocásticas (n_0 - n_3)
10 y proveer el valor máximo de ellas a su salida (S_{max}^*).
14. Red neuronal computacional según cualquiera de las reivindicaciones de la 9 a la 13, caracterizada por que comprende una matriz de conversores de binario a estocástico (BSC array), estando sus conversores configurados para convertir una señal inicial (x) recibida en sus respectivas primeras entradas y convertirla a sus salidas en una señal
15 inicial estocástica (x^*), utilizando dicha señal aleatoria (R) recibida en sus respectivas segundas entradas.
15. Red neuronal computacional según cualquiera de las reivindicaciones de la 9 a la 14, caracterizada por que comprende un segundo generador de números aleatorios (2') y una matriz de conversores de binario a estocástico para pesos (BSC array'), estando sus
20 conversores configurados para convertir una señal de peso (w) recibida en una primera entrada y convertirla a su salida en dicha señal de peso estocástica (w^*), utilizando una segunda señal aleatoria recibida del segundo generador de números aleatorios (2').
16. Red neuronal computacional según cualquiera de las reivindicaciones de la 9 a la 15, que comprende una pluralidad de neuronas estocásticas (10) según descritas en la
25 reivindicación 2, caracterizada por que comprende una capa de tipo Max-pooling, con neuronas estocásticas (10) que comprenden elementos de generación de señales estocásticas (1) cuya unidad de procesamiento (11) es una puerta lógica de tipo OR.
17. Red neuronal computacional según cualquiera de las reivindicaciones de la 9 a la 15, que comprende una pluralidad de neuronas estocásticas (10) según descritas la reivindicación
30 3, caracterizada por que comprende una capa de tipo Min-pooling, con neuronas

artificiales (10) que comprenden elementos de generación de señales estocásticas (1) cuya unidad de procesamiento (11) es una puerta lógica de tipo AND.

- 5 18. Red neuronal computacional según las reivindicaciones 16 o 17, caracterizada por que se encuentra implementada como una red convolucional que comprende más de una capa de tipo Max-pooling, así como de tipo Min-pooling.

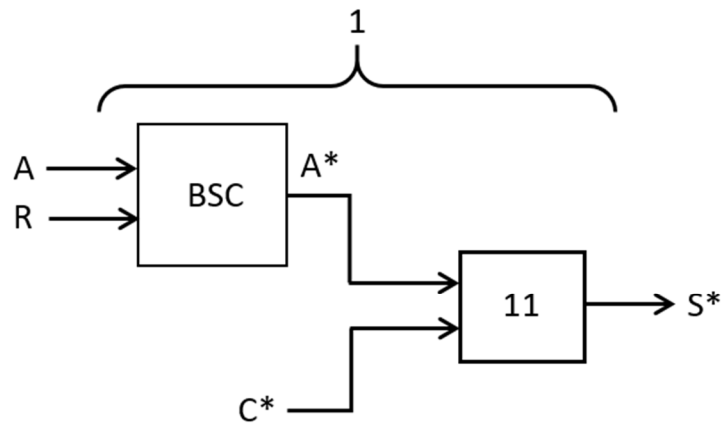


Fig. 1

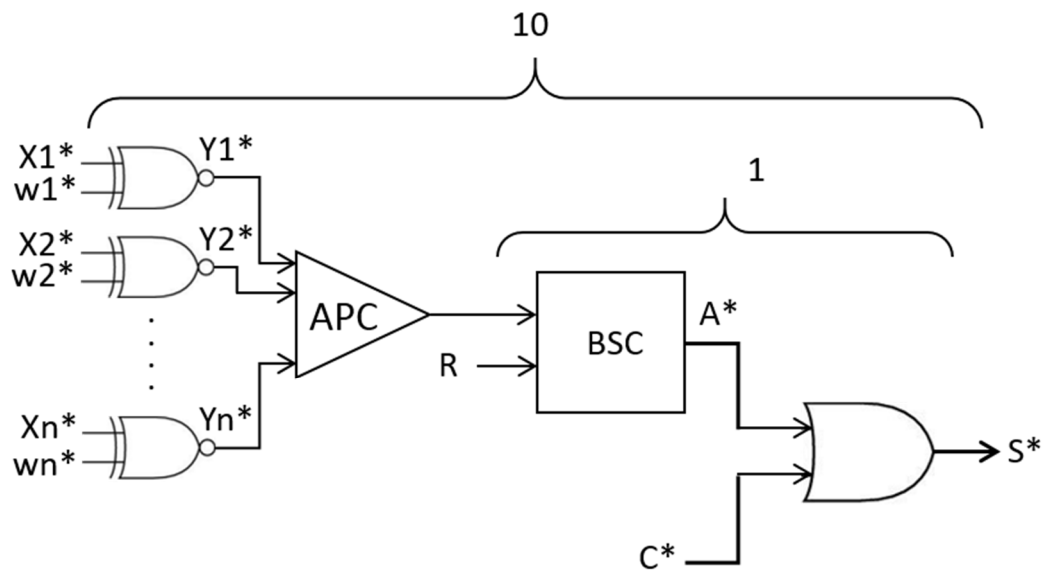


Fig. 2

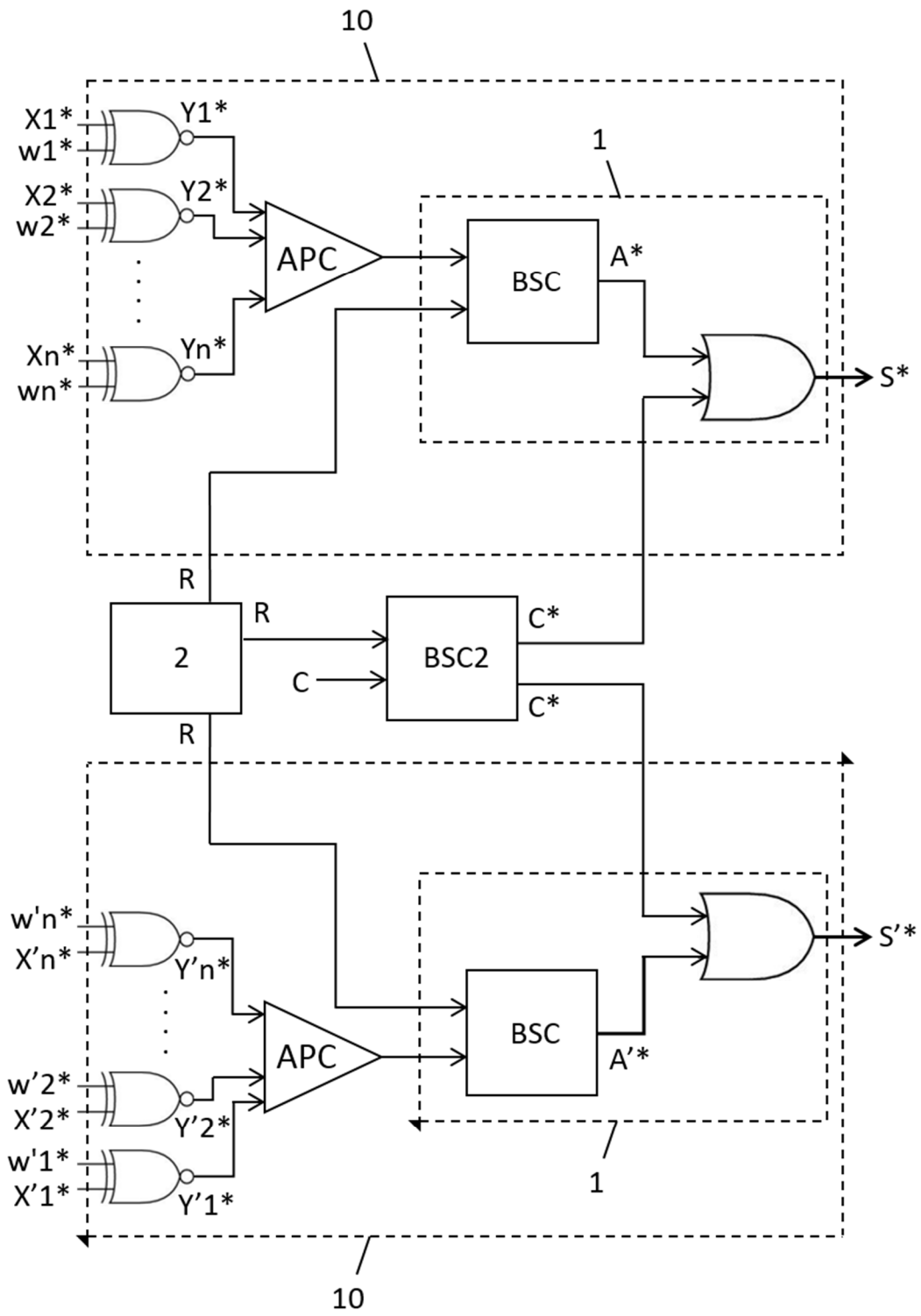


Fig. 3

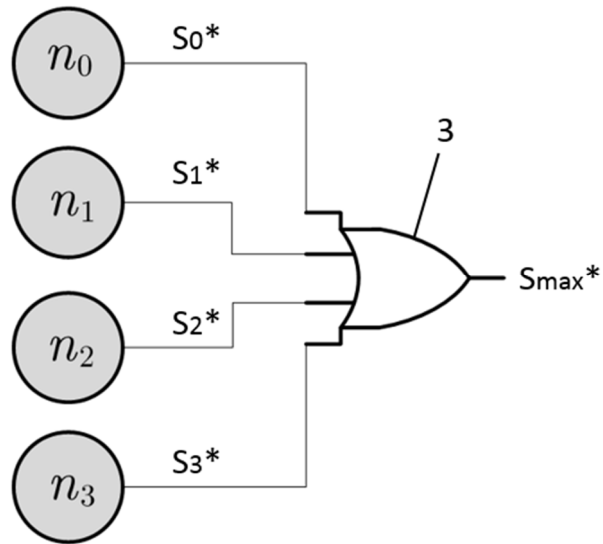


Fig. 4

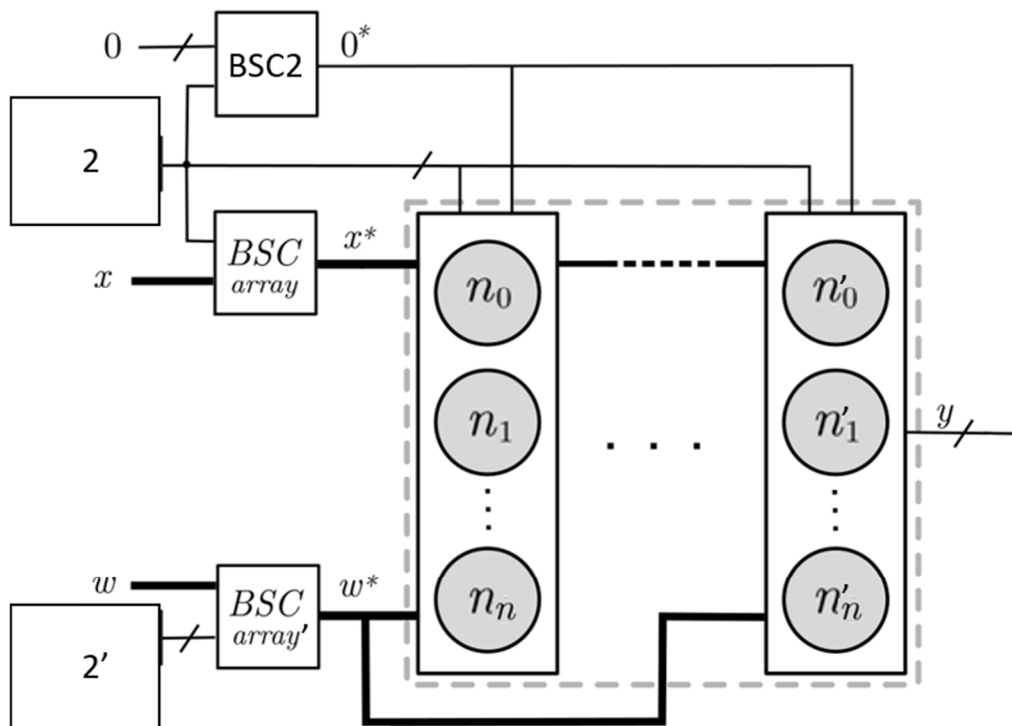


Fig. 5

COMAPRACIÓN DE CARACTERÍSTICAS TÉCNICAS CON OTROS TRABAJOS EN LA IMPLEMENTACIÓN DE LA LENET-5

Modelo	FPGA16	FPGA17a	FPGA17b	FPGA18	Propuesto
Arquitectura usada	Secuencial (binario)	Secuencial (binario)	Secuencial (binario)	Secuencial (binario)	Paralelo (estocástico)
Plataforma FPGA	Zynq XC7Z020	Virtex7 VX690T	Virtex7 485t	Zynq zc706	Arria10 GX1150
Frecuencia (MHz)	100	100	—	166	150
Latencia (us)	7916	94.2	960	1600	3.4
Kilo-Inferencias por segundo (KIPS)	0.13	10.6	1.04	0.625	294.1
Velocidad de proceso (KIPS/MHz)	0.001	0.1	—	0.003	1.96
Consumo (W)	—	25.2	0.47	10.98	21
Eficiencia energética (KI/J)	—	0.42	2.21	0.056	14
Lógica utilizada (LUT/ALM)	9682	233K	7204	39837	343.4K
Número de DSPs utilizadas	2.4	2907	574	59	0
Bloques de memoria usados	6.16	477	343.3	97	0

Fig. 6