(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **3 014 274**

(51) Int. Cl.:

G10L 25/51 (2013.01)
G06N 3/045 (2013.01)
G05B 23/02 (2006.01)
G06N 3/088 (2013.01)
G06N 3/048 (2013.01)
G06N 5/01 (2013.01)
G10L 25/30 (2013.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(96) Fecha de presentación y número de la solicitud europea: **04.07.2019 E 23156617 (5)**

(97) Fecha y número de publicación de la concesión europea: **19.02.2025 EP 4216216**

(54) Título: **Aparato de aprendizaje de distribución de probabilidad y aparato de aprendizaje de autocodificación**

(30) Prioridad:

10.08.2018 JP 2018151412
07.11.2018 JP 2018209416

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
21.04.2025

(73) Titular/es:

NIPPON TELEGRAPH AND TELEPHONE CORPORATION (100.00%)
5-1, Otemachi 1-chome, Chiyoda-ku
Tokyo 100-8116, JP

(72) Inventor/es:

YAMAGUCHI, MASATAKA;
KOIZUMI, YUMA y
HARADA, NOBORU

(74) Agente/Representante:

ELZABURU, S.L.P

DESCRIPCIÓN

Aparato de aprendizaje de distribución de probabilidad y aparato de aprendizaje de autocodificación

Campo técnico

La presente invención se refiere a una técnica de detección de anomalías y a una técnica de transformación de dominios.

Antecedentes de la técnica

Por ejemplo, si el funcionamiento de un equipo comercial, como una gran máquina de fabricación y una máquina de moldeo instaladas en una fábrica, se detiene debido a un problema, el servicio se ve obstruido en gran medida. Por lo tanto, es necesario monitorizar, diariamente, la condición de funcionamiento y, en caso de que se produzca una anomalía, hacer frente inmediatamente a la anomalía. Como método para abordar este problema, existe un método en el que un proveedor de servicios de gestión de equipos comerciales envía, regularmente, un trabajador de mantenimiento al sitio, y el trabajador de mantenimiento confirma la abrasión, o similar, de piezas. Sin embargo, debido a que este método cuesta mucho dinero (por ejemplo, gastos de personal y gastos de viaje) y requiere una gran cantidad de mano de obra, es difícil realizar este método en todos los equipos y fábricas comerciales. Por lo tanto, como técnica alternativa, existe un método en el que se monitoriza, diariamente, la condición de funcionamiento en función del sonido de funcionamiento recogido por un micrófono provisto dentro del equipo. Específicamente, se analiza el sonido del funcionamiento, y en caso de que se detecte un sonido (sonido anómalo) que se considera anómalo, se emite una alerta. Una técnica para determinar si el equipo a monitorizar está en un estado normal o en un estado anormal utilizando el sonido de esta manera se conoce como técnica de detección de sonidos anómalos.

En la técnica de detección de sonidos anómalos, cuesta dinero establecer un tipo de sonido anómalo y un método de detección para cada tipo o para cada pieza del equipo. Por tanto, es necesario diseñar, automáticamente, una regla para detectar sonidos anómalos. Como una de las soluciones a esto, es bien conocida la detección de sonidos anómalos en función de métodos estadísticos (bibliografía No de patente 1). Esta detección de sonidos anómalos en función de los métodos estadísticos se divide, aproximadamente, en detección supervisada de sonidos anómalos y detección no supervisada de sonidos anómalos. En la detección supervisada de sonidos anómalos, se recoge una gran cantidad de sonido normal y de sonido anómalo como datos de aprendizaje, y se aprende un discriminador de modo que se maximice una tasa de discriminación (una tasa de sonido normal que puede discriminarse del sonido anómalo). Mientras tanto, en la detección no supervisada de sonidos anómalos, se recoge una gran cantidad sólo de sonido normal como datos de aprendizaje, y se aprende una distribución de probabilidad (en lo sucesivo, denominada modelo normal) con respecto a la aparición de un sonido normal. Luego, el sonido se determina como normal en un caso en el que se determine, utilizando el modelo normal, que el sonido recién recogido (es decir, el sonido para el cual se va a realizar la detección de anomalías) es similar a (muy probablemente sea) un sonido normal, mientras que el sonido se determina como anómalo en un caso en el que se determine que el sonido recién recogido no es similar a (poco probable que sea) un sonido normal.

Debido a que es difícil recopilar una gran cantidad de datos de aprendizaje de sonidos anómalos en un campo de aplicación industrial, a menudo se emplea la detección no supervisada de sonidos anómalos. Además, en el campo de aplicación industrial, existe un caso en el que se desea establecer, respectivamente, un gran número de piezas del equipo del mismo tipo como objetivos de detección de anomalías. Por ejemplo, un caso de este tipo puede incluir un caso donde se desee monitorizar sonido anómalo de una enorme cantidad de servidores existentes en un centro de datos. En este caso, mientras que los sistemas de detección de anomalías se aplican, respectivamente, a los respectivos servidores, se supone que la distribución del sonido emitido desde los respectivos servidores varía ligeramente debido a las ubicaciones donde están instalados los servidores o debido a errores de montaje. Por lo tanto, como método para aplicar sistemas de detección de anomalías a una cantidad tan grande de piezas del equipo del mismo tipo, puede existir el siguiente método.

(1) Un modelo normal común a algunas piezas del equipo se aprende utilizando el sonido normal recogido de estas piezas del equipo. Luego, se realiza detección de anomalías en todas las piezas del equipo utilizando este modelo común.

(2) Diferentes modelos normales para cada pieza del equipo se aprenden utilizando el sonido normal recogido para cada pieza del equipo. Luego, se realiza detección de anomalías de cada pieza del equipo utilizando estos modelos individuales.

Con el método de (1), debido a que el aprendizaje no se realiza para cada pieza de equipo, incluso si aumenta el número de piezas del equipo que se desean monitorizar, no es necesario recopilar datos de aprendizaje ni realizar aprendizaje del modelo, de modo que es posible reducir los costes relacionados con la recopilación de datos y el aprendizaje. Sin embargo, debido a que es imposible capturar una ligera diferencia con respecto a la aparición de sonido normal para cada pieza del equipo, existe una posibilidad de que la detección de anomalías no se pueda realizar con alta precisión. Además, con el método de (2), si bien se espera que se genere un modelo normal con alta precisión porque el aprendizaje se realiza utilizando sólo sonido normal obtenido de las respectivas piezas del equipo, debido a que es necesario recopilar datos de aprendizaje para cada pieza del equipo para realizar el aprendizaje del modelo, existe el problema de que cuesta dinero recopilar datos y realizar el aprendizaje de acuerdo con el aumento

en el número de piezas del equipo que se desean monitorizar.

La transformación del dominio se describirá a continuación. La transformación del dominio es una técnica de transformación de datos de un determinado dominio en datos de un dominio diferente al del dominio. Los datos objetivo son aquí, por ejemplo, una imagen o un sonido. Por ejemplo, como se describe en la bibliografía 1 de Referencia no de patente, la transformación del dominio es una técnica para transformar una "imagen de una fotografía de un paisaje" en una "imagen de un paisaje" o transformar una "imagen de una fotografía de un caballo" en una "imagen de una fotografía de una cebra". (bibliografía 1 de Referencia no de patente: Jun-Yan Zhu, Taesung Park, Phillip Isola, Alexei A. Efros et al., "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks", arXiv:1703.10593v5, https://arxiv .org/abs/1703.10593v5)

Para realizar la transformación del dominio, sólo es necesario crear un transformador de datos que transforme un dominio D en un dominio D'. Para crear dicho transformador de datos, puede existir un método en el que, por ejemplo, se recopilen como datos de aprendizaje un gran número de pares de "fotografías de paisajes" y "paisajes" que se obtienen retratando con precisión las fotografías de paisajes, y la transformación de las "fotografías de paisajes" en "paisajes" se aprende utilizando una red neuronal. En lo sucesivo, un marco para hacer que un transformador de datos realice aprendizaje utilizando pares de datos de dos dominios de esta manera se denominará transformación del dominio con datos de pares. La transformación del dominio con datos de pares tiene la ventaja de que es posible configurar un transformador de datos con relativa facilidad mediante aprendizaje utilizando pares de datos de dos dominios que se ingresan y una respuesta correcta a la entrada como datos de aprendizaje. Sin embargo, es necesario recopilar un gran número de datos de aprendizaje. En el ejemplo de las "fotografías de paisajes" y los "paisajes" descrito anteriormente, primero es necesario recopilar "fotografías de paisajes", y luego, crear "paisajes" que se obtienen retratando con precisión las "fotografías de paisajes" (por ejemplo, solicitando a un artista que cree los paisajes). Además, en el ejemplo de una "fotografía de un caballo" y una "fotografía de una cebra", debido a que es difícil tomar fotografías con la misma composición, es realmente imposible recopilar datos de aprendizaje.

Por lo tanto, en los últimos años, para resolver un problema relacionado con la recopilación de datos de aprendizaje, se ha propuesto un marco que es capaz de aprender un transformador de datos sin utilizar datos de pares. Este marco se denominará transformación del dominio sin datos de pares. En la transformación del dominio sin datos de pares, se aprende un transformador de datos que transforma datos de un dominio D en datos de un dominio D' utilizando los datos del dominio D y los datos del dominio D'. Aquí, los datos del dominio D y los datos del dominio D' que se van a utilizar para el aprendizaje no tienen que ser un par. Por lo tanto, es posible aprender un transformador de datos incluso para la transformación entre dominios, para la que es difícil recopilar datos de pares, como una "fotografía de un caballo" y una "fotografía de una cebra".

Como ejemplo de transformación del dominio sin datos de pares, se ha propuesto, por ejemplo, un método llamado StarGAN descrito en la bibliografía no de patente 2.

También se hace referencia a la bibliografía de patente 1, que divulga un dispositivo de discriminación para una señal acústica según el preámbulo de la reivindicación 1 adjunta.

BIBLIOGRAFÍA DE LA TÉCNICA ANTERIOR

BIBLIOGRAFÍA DE PATENTE

Bibliografía de patente 1: Solicitud de patente japonesa abierta a inspección pública No. H07-219581

BIBLIOGRAFÍA NO DE PATENTE

Bibliografía no de patente 1, Tsuyoshi Ide, Masashi Sugiyama et al., "Anomaly Detection and Change Detection", Kodansha, págs. 6-7, 2015.

Bibliografía no de patente 2: Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, Jaegul Choo, et al., "StarGAN: Unified Generative Adversarial Networks for Multi-Domain Image-to-Image Translation", Conferencia del IEEE sobre Visión por Ordenador y Reconocimiento de Patrones (CVPR) 2018, págs.8789-8797, 2018.

Compendio de la invención

Problemas a resolver por la invención

En primer lugar, se describirá un primer problema.

Como se describió anteriormente, con los métodos de (1) y (2), existe un problema de compensación entre el coste relacionado con la recopilación y el aprendizaje de los datos y la precisión de la detección de anomalías. Por lo tanto, como tercer método, puede existir el siguiente método.

(3) Un modelo normal común a algunas piezas del equipo se aprende utilizando el sonido normal recogido de estas piezas del equipo. Luego, se aprenden, de forma adaptativa, diferentes modelos normales para cada pieza del equipo a partir de este modelo común utilizando el sonido normal recogido para cada pieza del equipo. Luego, la detección de anomalías de cada pieza del equipo se realiza utilizando estos modelos normales individuales adaptados.

Si bien, con el método de (3), existe la posibilidad de que se pueda resolver el problema de compensación descrito anteriormente, para realizar realmente dicho método, sólo es necesario permitir que se recoja una cantidad relativamente pequeña de sonido normal para cada pieza del equipo para la cual se va a realizar la detección de anomalías, y permitir un aprendizaje eficiente de modelos normales adaptados a las respectivas piezas del equipo a partir del modelo normal común utilizando el sonido normal. Sin embargo, hasta el momento no se ha desarrollado un método de este tipo.

Este es el primer problema.

A continuación, se describirá un segundo problema.

Con StarGAN descrito en la bibliografía No de patente 2, mientras que la transformación del dominio sin datos de pares se realiza utilizando un método llamado Redes Generativas Adversarias (GAN) descrito en la bibliografía 2 de Referencia no de patente, existe el problema de que el aprendizaje es inestable.

(Bibliografía 2 de referencia no de patente: Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, Yoshua Bengio et al., "Generative Adversarial Nets", Avances en los Sistemas Neuronales 27 de Procesamiento de Información (NIPS 2014), 2018.)

Este es el segundo problema.

Por lo tanto, la presente invención está dirigida a proporcionar una técnica de detección de anomalías que logra una alta precisión al mismo tiempo que reduce el coste requerido para el aprendizaje del modelo normal, y una técnica de transformación del dominio que permite un aprendizaje estable sin datos de pares.

Medios para resolver los problemas

La presente invención proporciona un aparato de aprendizaje de distribución de probabilidad con las características de la reivindicación 1 adjunta. En la reivindicación 2 se describe una realización preferida.

Un ejemplo que no está abarcado por las reivindicaciones pero que es útil para entender la presente invención es un aparato de detección de anomalías que comprende una unidad de estimación del grado de anomalía configurado para estimar un grado de anomalía de equipo objetivo de detección de anomalías del sonido emitido desde el equipo objetivo de detección de anomalías (de ahora en adelante, en el presente documento referido como sonido objetivo de detección de anomalías) basado en la asociación entre una primera distribución de probabilidad que indica la distribución del sonido normal emitido por una o más piezas de equipamiento diferente del equipo objetivo de detección de anomalías y el sonido normal emitido desde el equipo objetivo de detección de anomalías (de ahora en adelante, en el presente documento referido como sonido normal para aprendizaje adaptativo).

Un ejemplo que no está abarcado por las reivindicaciones pero que es útil para entender la presente invención es un aparato de aprendizaje de distribuciones de probabilidad que comprende una unidad de aprendizaje configurada para aprender una primera distribución de probabilidad que indica la distribución del sonido normal emitido desde uno o más equipos diferentes del equipo objetivo de detección de anomalías (en lo sucesivo, denominado sonido normal para el aprendizaje) a partir del sonido normal emitido desde la una o más piezas del equipo diferentes del equipo objetivo de detección de anomalías, en donde una variable x de la primera distribución $q_1(x;\theta)$ de probabilidad es una variable que indica datos de entrada generados a partir del sonido normal emitido desde la una o más piezas del equipo diferentes del equipo objetivo de detección de anomalías, la variable x se expresa como $x = f_K(f_{K-1}(...(f_1(z_0))...))$ utilizando transformaciones f_i (i = 1, ..., K, K es un número entero de 1 o mayor, y existen transformaciones inversas f_i^{-1} para las transformaciones f_i) y una variable latente z_0, $q_0(z_0)$ se establece como una distribución de probabilidad de la variable latente z_0, una densidad $q_1(x;\theta)$ de probabilidad de los datos x de entrada se calcula utilizando una densidad $q_0(z_0)$ de probabilidad de la variable latente $z_0 = f_1^{-1}(f_2^{-1}(...(f_K^{-1}(x))...))$ correspondiente a los datos x de entrada, y a, al menos, una transformación inversa de las transformaciones f_i(i = 1, ..., K) es una normalización adaptativa por lotes.

Un ejemplo que no está abarcado por las reivindicaciones pero que es útil para entender la presente invención es un aparato de detección de anomalías que comprende una unidad de estimación del grado de anomalía configurada para estimar un grado de anomalía que indica un grado de anomalía del del sonido emitido desde el equipo objetivo de detección de anomalías(de ahora en adelante en el presente documento referido como sonido objetivo de detección de anomalías) basado en la asociación entre un primer autocodificador que restaura el sonido normal emitido desde una o más piezas de equipamiento diferente del equipo objetivo de detección de anomalías y el sonido normal emitido desde el equipo objetivo de detección de anomalías (de ahora en adelante en el presente documento referido como sonido para aprendizaje adaptativo).

Un ejemplo que no está abarcado por las reivindicaciones pero que es útil para entender la presente invención es un aparato de aprendizaje con codificador automático que comprende una unidad de aprendizaje configurada para aprender un primer codificador automático que restaura el sonido normal emitido desde uno o más equipo diferentes del equipo objetivo de detección de anomalías (en lo sucesivo, denominado sonido normal para el aprendizaje) a partir del sonido normal emitido desde la una o más piezas del equipo diferentes del equipo objetivo de detección de anomalías, en donde el primer codificador automático se configura como una red neuronal que incluye una capa AdaBN, que ejecuta el cálculo de la normalización adaptativa por lotes.

Un ejemplo que no está abarcado por las reivindicaciones pero que es útil para entender la presente invención es un aparato de detección de anomalías que comprende una unidad de estimación del grado de anomalía configurada para estimar un grado de anomalía que indica un grado de anomalía del equipo objetivo de detección de anomalías a partir del sonido emitido desde el equipo objetivo de detección de anomalías (en lo sucesivo, denominado sonido objetivo de detección de anomalías) en función de la asociación entre un conjunto de sonidos normales emitido desde uno o más equipos diferentes del equipo objetivo de detección de anomalías y del sonido normal emitido desde el equipo objetivo de detección de anomalías (en lo sucesivo, denominado sonido normal para el aprendizaje adaptativo).

Un ejemplo que no está abarcado por las reivindicaciones pero que es útil para entender la presente invención es un aparato de transformación de datos que comprende una unidad de cálculo de la variable latente configurada para calcular una variable latente a partir de los datos de entrada correspondientes a los datos de dominio de un primer dominio, y una unidad de cálculo de datos de salida configurada para calcular los datos de salida correspondientes a los datos de dominio de un segundo dominio a partir de la variable latente, en donde la unidad de cálculo de la variable latente realiza cálculos utilizando una función predeterminada (en lo sucesivo, denominada primera función) con una función inversa, la unidad de cálculo de datos de salida realiza cálculos utilizando una función predeterminada (en lo sucesivo, denominada segunda función) con una función inversa, y la primera función y la segunda función se derivan de una función predeterminada que transforma la variable latente z_0 en una variable x.

Efectos de la invención

Según la presente invención, es posible realizar detección de anomalías con alta precisión y al mismo tiempo reducir el coste requerido para el aprendizaje del modelo normal. Además, según la presente invención, es posible realizar una transformación del dominio que permita un aprendizaje estable sin datos de pares.

Breve descripción de los dibujos

La Fig. 1 es un diagrama de bloques que ilustra un ejemplo de una configuración de un aparato 100 de aprendizaje de distribuciones de probabilidad;

La Fig. 2 es un diagrama de flujo que ilustra un ejemplo de funcionamiento del aparato 100 de aprendizaje de distribuciones de probabilidad;

La Fig. 3 es un diagrama de bloques que ilustra un ejemplo de una configuración de un aparato 200 de aprendizaje adaptativo de distribuciones de probabilidad;

La Fig. 4 es un diagrama de flujo que ilustra un ejemplo de funcionamiento del aparato 200 de aprendizaje adaptativo de distribuciones de probabilidad;

La Fig. 5 es un diagrama de bloques que ilustra un ejemplo de una configuración de un aparato 300 de detección de anomalías;

La Fig. 6 es un diagrama de flujo que ilustra un ejemplo de funcionamiento del aparato 300 de detección de anomalías;

La Fig. 7 es un diagrama de bloques que ilustra un ejemplo de una configuración de una unidad 320 de estimación del grado de anomalía;

La Fig. 8 es un diagrama de flujo que ilustra un ejemplo de funcionamiento de la unidad 320 de estimación del grado de anomalía;

La Fig. 9 es un diagrama de bloques que ilustra un ejemplo de una configuración de un aparato 400 de aprendizaje con codificador automático;

La Fig. 10 es un diagrama de flujo que ilustra un ejemplo de funcionamiento del aparato 400 de aprendizaje con codificador automático;

La Fig. 11 es un diagrama de bloques que ilustra un ejemplo de una configuración de un aparato 500 de aprendizaje adaptativo con codificador automático;

La Fig. 12 es un diagrama de flujo que ilustra un ejemplo de funcionamiento del aparato 500 de aprendizaje adaptativo con codificador automático;

La Fig. 13 es un diagrama de bloques que ilustra un ejemplo de una configuración de un aparato 600 de detección de anomalías;

La Fig. 14 es un diagrama de flujo que ilustra un ejemplo de funcionamiento del aparato 600 de detección de anomalías;

La Fig. 15 es un diagrama de bloques que ilustra un ejemplo de una configuración de una unidad 620 de estimación del grado de anomalía;

La Fig. 16 es un diagrama de flujo que ilustra un ejemplo de funcionamiento de la unidad 620 de estimación del grado de anomalía;

La Fig. 17 es un diagrama de bloques que ilustra un ejemplo de una configuración de un aparato 1100 de aprendizaje de distribuciones de probabilidad;

La Fig. 18 es un diagrama de flujo que ilustra un ejemplo de funcionamiento del aparato 1100 de aprendizaje de distribuciones de probabilidad;

La Fig. 19 es un diagrama de bloques que ilustra un ejemplo de una configuración de un aparato 1200 de transformación de datos;

La Fig. 20 es un diagrama de flujo que ilustra un ejemplo de funcionamiento del aparato 1200 de transformación de datos;

La Fig. 21 es una vista que ilustra un aspecto de procesamiento mediante una unidad 1220 de cálculo de la variable latente y una unidad 230 de cálculo de datos de salida;

La Fig. 22 es un diagrama de bloques que ilustra un ejemplo de una configuración de un aparato 1300 de transformación de datos;

La Fig. 23 es un diagrama de flujo que ilustra un ejemplo de funcionamiento del aparato 1300 de transformación de datos; y

La Fig. 24 es una vista que ilustra un ejemplo de una configuración funcional de un ordenador que realiza cada aparato en las realizaciones de la presente invención.

Descripción detallada de las realizaciones

Las realizaciones de la presente invención se describirán en detalle a continuación. Tenga en cuenta que se asignarán los mismos números de referencia a componentes que tengan la misma función, y se omitirán las descripciones redundantes.

<Notación>

_(barra baja) indica un subíndice. Por ejemplo, x^{y_z} indica que y_z es un superíndice de x, y x_{y_z} indica que y_z es un subíndice de x.

En primer lugar, se describirán los antecedentes técnicos de la primera realización a la tercera realización de la presente invención, y cada realización.

<Antecedentes técnicos>

Las realizaciones de la presente invención proporcionan un marco de detección de anomalías que puede aplicarse a una pluralidad de piezas del equipo del mismo tipo. Específicamente, se proporciona el marco que utiliza el método de (3) descrito anteriormente. En función de una hipótesis de que una diferencia en la distribución del sonido normal emitido desde las respectivas piezas del equipo se expresa con una cantidad de estadísticas de cantidades de características con respecto al sonido, generando un promedio que es una cantidad primaria de estadísticas de las cantidades de características y una varianza, que es una cantidad secundaria de estadísticas para hacer coincidir diferentes piezas del equipo, es posible derivar modelos normales de diferentes piezas del equipo a partir de un modelo.

En primer lugar, se describirá la técnica relacionada que se utilizará en las realizaciones de la presente invención.

<<Detección no supervisada de sonidos anómalos>>

La detección de sonidos anómalos es una tarea de determinar si el estado de un equipo a monitorizar, que emite un sonido (señal de observación) para el cual se va a realizar la detección de anomalías, es normal o anormal. Aquí, como datos x de entrada se generará, a partir de la señal de observación, por ejemplo, un vector que incluya un espectro $\ln|X_{t,f}|$ de amplitud logarítmica de la señal de observación como elemento puede utilizarse como en la siguiente expresión.

[Fórmula 1]

$$x := (\ln|X_{t-Q,1}|,\ \ln|X_{t-Q,2}|,\ \ldots,\ \ln|X_{t-Q,F}|,\ \ln|X_{t-Q+1,1}|,\ \ln|X_{t-Q+1,2}|,\ \ldots,\ \ln|X_{t+Q,F}|)^T \quad \ldots(1)$$

Aquí, t = {1, ..., T}, f = {1, ..., F} indican, respectivamente, un índice de tiempo y un índice de una frecuencia. Además, Q indica el número de tramas en el pasado y en el futuro, lo que se tiene en cuenta en la entrada.

Los datos x de entrada no se limitan a los ejemplos descritos anteriormente, y pueden utilizarse otras cantidades de características como datos de entrada que se generarán a partir de la señal de observación.

La detección de sonidos anómalos en función de la estimación de densidad se describirá a continuación. Se supone que los datos de entrada que se generarán a partir del sonido normal (en lo sucesivo, denominados, simplemente, datos de entrada del sonido normal) se generan de acuerdo con una distribución p(x) de probabilidad. Primero, se diseña una distribución q(x;θ) de probabilidad con un parámetro θ. Entonces, un parámetro θ* con el que q(x;θ) se vuelve la más cercano a p(x) se obtiene utilizando un conjunto $\{x_i\}_{i=1}^N$ de datos de entrada de N piezas de sonido normal generadas a partir de la distribución p(x) de probabilidad, y q(x;θ*) se establece como distribución aproximada de p(x). Entonces, en el caso de que se introduzcan datos de entrada de un sonido para el que pase va a realizar la detección de anomalías, se obtiene un grado A(x;θ*) de anomalía con respecto a los datos de entrada utilizando, por ejemplo, la siguiente expresión.

[Fórmula 2]

$$A(x;\theta^*) = -\ln q(x;\theta^*) \quad \ldots(2)$$

Por último, se obtiene un resultado R de determinación utilizando, por ejemplo, la siguiente expresión.

[Fórmula 3]

$$R = H(A(x;\theta^*) - \phi) \quad \ldots(3)$$

Aquí, un umbral ϕ es una constante predeterminada, y H(•) es una función escalonada que devuelve 1 si un argumento no es negativo, y que devuelve 0 si el argumento es negativo. En un caso donde R = 1, se determina que el equipo que ha emitido el sonido para el cual se va a realizar la detección de anomalía es anormal, mientras que, en un caso donde R = 0, se determina que el equipo es normal. En otras palabras, si el grado A(x;θ*) de anomalía es mayor que un umbral ϕ establecido de antemano, se determina que el equipo es anormal.

Como se describió anteriormente, en la detección de sonidos anómalos en función de la estimación de densidad, es necesario (1) diseñar la distribución q(x;θ) de probabilidad, y (2) decidir el parámetro θ. En (1) el diseño de la distribución q(x;θ) de probabilidad, puede utilizarse el Flujo de Normalización. Además, en (2) la decisión del parámetro θ, por ejemplo, aprendizaje utilizando un método de descenso de gradiente en el que una función de pérdida (función objetivo) es $L(\theta) = -\Sigma \log q(x_i;\theta)$ (es decir, una suma de probabilidades logarítmicas negativas con respecto a un conjunto $\{x_i\}_{i=1}^N$ de los datos de entrada de un sonido normal).

<<Flujo de Normalización>>

El Flujo de Normalización es un método para obtener una distribución que se aproxime a una distribución p(x) de probabilidad respecto a la generación de datos.

El Flujo de Normalización se describirá a continuación. Se supone que $\{f_i(z)\}_{i=1}^K$ son K transformaciones que tienen transformaciones inversas (donde $f_i(z):R^D \rightarrow R^D$, R es un conjunto de números reales y D es un número entero de 1 o mayor). Además, se supone que $f_i^{-1}(z)$ (i = 1, ..., K) es una transformación inversa de $f_i(z)$.

En el Flujo de Normalización se considera que las variables latentes $\{z_{0,i}\}_{i=1}^N$ correspondientes, respectivamente, al conjunto $\{x_i\}_{i=1}^N$ de N piezas de datos de entrada existen, y los datos x_i de entrada se obtienen transformando la variable latente $z_{0,i}$ correspondiente utilizando la siguiente expresión que utiliza K transformaciones $\{f_i(z)\}_{i=1}^K$ y una variable latente z_0 de x.

[Fórmula 4]

$$x = f_K(f_{K-1}(\ldots(f_1(z_0))\ldots)) \quad \ldots(4)$$

En otras palabras, la siguiente expresión es válida para i = 1, ..., K.

[Fórmula 5]

$$x_i = f_K(f_{K-1}(...(f_1(z_{0,i}))...)) \qquad ...(4)'$$

Tenga en cuenta que $z_1 = f_1(z_0)$, $z_2 = f_2(z_1)$, ..., $x = f_K(z_{K-1})$.

Además, se supone que la variable latente $\{z_{0,i}\}_{i=1}^N$ se genera a partir de una distribución $q_0(z_0)$ de probabilidad como, por ejemplo, una distribución Gaussiana isotrópica, con la que es fácil realizar un muestreo Monte Carlo. En este momento, la distribución $q(x;\theta)$ de probabilidad (donde x es una variable que indica datos de entrada) con la que cumple un conjunto de datos $\{x_i\}_{i=1}^N$ de entrada puede expresarse de la siguiente forma.

[Fórmula 6]

$$q(x;\theta) = q_0(z_0)\left|\det(\partial f_K(z_{K-1};\theta_K)/\partial z_{K-1})\right| \left|\det(\partial f_{K-1}(z_{K-2};\theta_{K-1})/\partial z_{K-2})\right| ...$$

$$\left|\det(\partial f_1(z_0;\theta_1)/\partial z_0)\right| \qquad ...(5)$$

Aquí, $z_0 = f_1^{-1}(f_2^{-1}(...(f_K^{-1}(x))...))$. Además, $\{\theta_i\}_{i=1}^K$ es un parámetro correspondiente a una transformación $\{f_i(z)\}_{i=1}^K$, y $\theta = [\theta_1^T, \theta_2^T, ..., \theta_K^T]^T$.

Tenga en cuenta que la distribución $q_0(z_0)$ de probabilidad no se limita a una distribución con la que sea fácil realizar un muestreo de Monte Carlo, y sólo se requiere que sea una distribución con la que sea fácil realizar una estimación estricta de la densidad de probabilidad. Ejemplos de distribuciones con las que es fácil realizar una estimación estricta de la densidad de probabilidad pueden incluir una distribución p(x) de probabilidad que satisfaga las siguientes condiciones.

(Condición 1) Una función no negativa $g(x)$ (≥ 0) en R^D existe para la distribución p(x) de probabilidad, y $p(x) = g(x)/\int g(x)dx$ para x arbitrario $\in R^D$.

(Condición 2) Es fácil calcular $\int g(x)dx$ para una función g(x).

Los ejemplos de una función que satisface la Condición 2 pueden incluir una distribución Gaussiana. Mientras tanto, los ejemplos de una función que no satisface la Condición 2 pueden incluir $g(x) = \exp(\sin(x) - x^2)$.

En el Flujo de Normalización, el parámetro θ de la distribución $q(x;\theta)$ de probabilidad se aprende utilizando un conjunto de los datos $\{x_i\}_{i=1}^N$ de entrada . Luego, la distribución p(x) de probabilidad original respecto a la generación de datos se aproxima mediante la distribución $q(x;\theta^*)$ de probabilidad, que utiliza el parámetro θ^* obtenido a través del aprendizaje.

En el Flujo de Normalización, es posible utilizar varios tipos de transformación como la transformación $\{f_i(z)\}_{i=1}^K$. Por ejemplo, es posible utilizar Normalización por Lotes, una ReLU (Unidad Lineal Rectificada) con Fugas, o similares, descritos en la bibliografía 3 de Referencia no de patente. Además, también es posible utilizar la siguiente transformación lineal descrita en la bibliografía 4 de Referencia no de patente.

[Fórmula 7]

$$f(z) = LUz \qquad ...(6)$$

Aquí, L, U $\in R^{D\times D}$ son, respectivamente, una matriz triangular inferior y una matriz triangular superior. Esta transformación está caracterizada por que, debido a que un valor absoluto $|\det(\partial f(z;\theta)/\partial z)|$ de un determinante Jacobiano puede calcularse utilizando un valor absoluto (es decir, $|\Pi_{i=1}^D L_{ii}U_{ii}|$) de un producto de elementos diagonales de L y U, es posible calcular, fácilmente, una densidad $q(x;\theta)$ de probabilidad de los datos x de entrada (es posible reducir el coste del cálculo de la densidad $q(x;\theta)$ de probabilidad de los datos x de entrada) (véase la expresión (5)).

(Bibliografía 3 de Referencia no de patente: S. Ioffe, C. Szegedy, et al., "Batch normalization: accelerating deep network training by reducing internal covariate shift", ICML 2015, 2015.)

(Bibliografía 4 de Referencia no de patente: J. Oliva, et al., "Transformation Autoregressive Networks", ICML 2018, 2018.)

Normalización por Lotes BN: $x \rightarrow y$ ($x, y \in R^D$) se describirá, brevemente, a continuación. En la Normalización por Lotes BN, después de realizar el ajuste para que un promedio de elementos de las respectivas dimensiones del conjunto $\{x_i\}_{i=1}^N$ de los datos de entrada se convierta en 0 y una varianza se convierta en 1, se realizan transformación de escala y transformación de desplazamiento. Específicamente, $y_i = BN(x_i)$ se calcula utilizando la siguiente expresión.

[Fórmula 8]

$$m = \frac{1}{N}\sum_{i=1}^{N} x_i \quad \cdots(7a)$$

$$s^2 = \frac{1}{N}\sum_{i=1}^{N} (x_i - m)^2 \quad \cdots(7b)$$

$$\hat{x}_i = \frac{x_i - m}{\sqrt{s^2 + \varepsilon}} \quad \cdots(7c)$$

$$y_i = \gamma \hat{x}_i + \beta \quad \cdots(7d)$$

Aquí, γ y β son, respectivamente, un parámetro de transformación de escala y un parámetro de transformación de desplazamiento, y ambos son parámetros que deben aprenderse.

Además, ε es un número real no negativo, y, sólo es necesario establecer un número real positivo como ε en un caso donde se desee evitar la división por cero, y establecer cero como ε en un caso donde no sea necesario evitar la división por cero.

Tenga en cuenta que, para especificar, claramente, el parámetro γ de transformación de escala y el parámetro β de transformación de desplazamiento, también existe un caso donde BN(•) se expresa como $BN_{\gamma\beta}(\bullet)$.

Además, no todas las K transformaciones tienen que ser transformaciones del mismo tipo. Por lo tanto, por ejemplo, también es posible combinar algunos tipos de transformaciones de modo que una transformación $f_1(z)$ es una normalización por lotes y una transformación $f_2(z)$ es una transformación lineal.

<<AdaBN (Normalización Adaptativa por Lotes)>>

La adaptación del dominio es una técnica para ajustar un modelo aprendido de modo que, en un caso donde la distribución de los datos de aprendizaje que se utilizará para el aprendizaje del modelo sea diferente de la distribución de los datos de prueba, que es un objetivo del procesamiento que utiliza el modelo aprendido, la precisión del procesamiento que utiliza el modelo aprendido no se degrada debido a una diferencia entre la distribución. Aquí, un conjunto de datos de aprendizaje y un conjunto de datos de prueba son dominios, y a veces se los denomina, respectivamente, dominio para aprendizaje y dominio para prueba.

Si bien existen varios métodos para la adaptación del dominio que pueden combinarse con una red neuronal profunda (DNN), aquí, se describirá la normalización adaptativa por lotes (véase la bibliografía 5 de Referencia no de patente). La normalización adaptativa por lotes es un método en el que se realiza el cálculo de un promedio y de una varianza y el ajuste de un promedio y de una varianza en la normalización por lotes (véanse las expresiones (7a) a (7d)) para cada dominio. En otras palabras, el cálculo utilizando las expresiones (7a) a (7c) se realiza para cada dato en el mismo dominio. En la prueba real, se calcula una cantidad de estadísticas (promedio y varianza) para un conjunto $\{x_i\}_{i=1}^{N}$ de los datos de entrada del dominio para prueba, y se genera un resultado y_i del procesamiento utilizando la expresión (7c) y la expresión (7d) que utilizan la cantidad de estadísticas. Tenga en cuenta que, en un caso donde la transformación sea una normalización adaptativa por lotes, existe un caso donde la transformación se expresa como AdaBN: $x \rightarrow y$ ($x, y \in R^D$).

(Bibliografía de referencia no de patente 5: Y. Li, et al., "Revisiting Batch Normalization for Practical Domain Adaptation", ICLR 2017, 2016.)

La detección de anomalías en las realizaciones de la presente invención se describirá a continuación. En primer lugar, se describirá el problema de establecimiento para la detección de anomalías en las realizaciones de la presente invención. A continuación, se describirá una configuración específica de detección de anomalías en las realizaciones de la presente invención utilizando la técnica relacionada anteriormente descrita.

<<Establecimiento del problema>>

Un problema a resolver es "aprender una segunda distribución de probabilidad que es un modelo normal que puede utilizarse para el equipo objetivo de detección de anomalías, utilizando una primera distribución de probabilidad que es un modelo normal común aprendido utilizando una gran cantidad de sonido normal obtenido de una pluralidad de piezas del equipo y una pequeña cantidad de sonido normal obtenido del equipo objetivo de detección de anomalías,

y permitir la detección de anomalías a partir del sonido emitido desde el equipo objetivo de detección de anomalías utilizando esta segunda distribución de probabilidad". Por tanto, se tratarán los siguientes datos.

(1) Datos de aprendizaje: se supone que los datos de aprendizaje son sonidos normales emitidos desde uno o más equipos diferentes del equipo objetivo de detección de anomalías y que es posible preparar una gran cantidad de datos de aprendizaje. Debido a que los datos de aprendizaje se utilizan para el aprendizaje, el sonido se denominará sonido normal para el aprendizaje. Además, un conjunto de datos de aprendizaje se denominará dominio para aprendizaje. Tenga en cuenta que el tipo de equipo desde el cual se va a recoger el sonido normal preferiblemente es el mismo que el tipo de equipo objetivo de detección de anomalías.

(2) Datos de aprendizaje adaptativo: se supone que los datos de aprendizaje adaptativo son sonidos normales emitidos desde el equipo objetivo de detección de anomalías y que sólo puede prepararse una pequeña cantidad de datos de aprendizaje adaptativo. Debido a que los datos del aprendizaje adaptativo se utilizan para el aprendizaje adaptativo, el sonido se denominará sonido normal para el aprendizaje adaptativo. Tenga en cuenta que un conjunto de los datos de aprendizaje adaptativo es un dominio para prueba que se describirá más adelante.

(3) Datos de prueba: los datos de prueba son sonidos emitidos desde el equipo objetivo de detección de anomalías, y se determina si el equipo es normal o anormal a partir de este sonido. Por lo tanto, este sonido se denominará sonido objetivo de detección de anomalías. Además, un conjunto de los datos de prueba se denominará dominio para prueba.

En lo sucesivo, una fase en la que se realiza el aprendizaje utilizando sonido normal para el aprendizaje se denominará fase de aprendizaje, una fase en la que se realiza el aprendizaje adaptativo utilizando sonido normal para el aprendizaje adaptativo se denominará fase de aprendizaje adaptativo, y una fase en la que se detecta una anomalía a partir del sonido objetivo de detección de anomalías se denominará fase de prueba (fase de detección de anomalías).

<<Puntos>>

En las realizaciones de la presente invención, para permitir que la segunda distribución de probabilidad se aprenda, de forma adaptativa, a partir de una pequeña cantidad de datos de aprendizaje adaptativo con un pequeño coste de cálculo, se introduce una normalización adaptativa por lotes para el Flujo de Normalización. Específicamente, al menos, una transformación inversa $f_i^{-1}(z)$ de la K transformaciones $\{f_i(z)\}_{i=1}^{k}$ que se utilizarán en el Flujo de Normalización es una normalización adaptativa por lotes. Tenga en cuenta que, en el cálculo de la normalización adaptativa por lotes, pueden omitirse la transformación de escala y la transformación de desplazamiento, es decir, el cálculo en la expresión (7d). En otras palabras, también es posible expresar que la transformación inversa $f_i^{-1}(z)$ es una normalización adaptativa por lotes en la que $\gamma = 1$ y $\beta = 0$.

<<Configuración específica>>

Se describirá a continuación una configuración específica.

(1) Fase de aprendizaje

Primero, una red neuronal en la que los datos x de entrada generados a partir del sonido normal (es decir, datos de aprendizaje) emitido desde uno o más equipos diferentes del equipo objetivo de detección de anomalías, se transforman en una variable latente $z_0(\sim q(z_0))$ que se considera generada de acuerdo con la distribución $q_0(z_0)$ de probabilidad que se describirá. Aquí, se describirá un caso donde se utilizan cinco transformaciones $\{f_i(z)\}_{i=1}^{5}$. En otras palabras, z_0 puede obtenerse como $z_0 = f_1^{-1}(f_2^{-1}(f_3^{-1}(f_4^{-1}(f_5^{-1}(x)))))$.

Las cinco transformaciones $\{f_i(z)\}_{i=1}^{5}$ descritas anteriormente se definirán utilizando la siguiente expresión. Tenga en cuenta que, por conveniencia, en lugar de ser indicada una transformación f_i, se indica una transformación inversa f_i^{-1} de la transformación f_i (donde $z_4 = f_5^{-1}(x)$, $z_3 = f_4^{-1}(z_4)$, $z_2 = f_3^{-1}(z_3)$, $z_1 = f_2^{-1}(z_2)$, $z_0 = f_1^{-1}(z_1)$).

[Fórmula 9]

$$f_5^{-1}(x) = L_5 D_5 U_5 x \qquad ...(8a)$$

$$f_4^{-1}(z_4) = \mathrm{AdaBN}_{\gamma 4, \beta 4}(z_4) \qquad ...(8b)$$

$$f_3^{-1}(z_3) = \mathrm{LeakyReLU}(z_3) = \max(z_3, \alpha_3 z_3) \qquad ...(8c)$$

$$f_2^{-1}(z_2) = L_2 D_2 U_2 z_2 \qquad ...(8d)$$

$$f_1^{-1}(z_1) = \mathrm{AdaBN}_{\gamma 1, \beta 1}(z_1) \qquad ...(8e)$$

Aquí, L_2, $L_5 \in R^{D \times D}$ es una matriz triangular inferior cuyo elemento diagonal es 1, y todos los elementos $L_{2,ij}$, $L_{5,ij}$ ($i \geq j$) distintos de una parte triangular superior son parámetros objetivo de aprendizaje (es decir, un parámetro θ_2 o un parámetro θ_5). D_2, $D_5 \in R^{D \times D}$ es una matriz diagonal y los elementos diagonales $D_{2,ij}$, $D_{5,ij}$ ($i = j$) son parámetros objetivo de aprendizaje (es decir, un parámetro θ_2 o un parámetro θ_5). U_2, $U_5 \in R^{D \times D}$ es una matriz triangular superior cuyo elemento diagonal es 1, y todos los elementos $U_{2,ij}$, $U_{5,ij}$ ($i \leq j$) distintos de una parte triangular inferior son parámetros objetivo de aprendizaje (es decir, un parámetro θ_2 o un parámetro θ_5). Además, α_3 (≥ 0) es un parámetro de LeakyReLU, y puede establecerse como un hiper parámetro, o puede establecerse como un parámetro objetivo de aprendizaje (es decir, un parámetro θ_3) (en un caso donde α_3 se establece como objetivo de aprendizaje, ReLU se denomina ReLU Paramétrica (bibliografía 6 de Referencia no de patente)). Además, $AdaBN_{\gamma\_4\beta\_4}(\bullet)$ y $AdaBN_{\gamma\_1\beta\_1}(\bullet)$ son la normalización adaptativa por lotes descrita anteriormente, y γ_1, β_1, γ_4, β_4 son parámetros objetivo de aprendizaje (es decir, un parámetro θ_1 o un parámetro θ_4).

(Bibliografía 6 de Referencia no de patente: K. He, et al., "Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification", ICCV 2015, páginas 1026-1034, 2015.)

Además, los valores absolutos de los determinantes Jacobianos de las transformaciones $\{f_i(z)\}_{i=1}^{5}$ se calculan, respectivamente, utilizando la siguiente expresión (donde $x = f_5(z_4)$, $z_4 = f_4(z_3)$, $z_3 = f_3(z_2)$, $z_2 = f_2(z_1)$, $z_1 = f_1(z_0)$).

[Fórmula 10]

$$\left|\det\left(\partial f_5(z_4)/\partial z_4\right)\right| = 1\Big/\left|\prod_{i=1}^{D} D_{5,ii}\right| \quad \cdots(9a)$$

$$\left|\det\left(\partial f_4(z_3)/\partial z_3\right)\right| = \sqrt{s_4'^2+\varepsilon}\Big/\gamma_4 \quad \cdots(9b)$$

$$\left|\det\left(\partial f_3(z_2)/\partial z_2\right)\right| = 1\Big/\alpha_3^{\delta} \quad \cdots(9c)$$

$$\left|\det\left(\partial f_2(z_1)/\partial z_1\right)\right| = 1\Big/\left|\prod_{i=1}^{D} D_{2,ii}\right| \quad \cdots(9d)$$

$$\left|\det\left(\partial f_1(z_0)/\partial z_0\right)\right| = \sqrt{s_1'^2+\varepsilon}\Big/\gamma_1 \quad \cdots(9e)$$

Aquí, s_4' es una desviación estándar de z_4 (correspondiente a los datos x de entrada generados a partir de los datos de aprendizaje), δ es el número de elementos por debajo de cero entre z_3 (correspondiente a los datos x de entrada generados a partir de los datos de aprendizaje), y s_1' es una desviación estándar de z_1 (correspondiente a los datos x de entrada generados a partir de los datos de aprendizaje). Tenga en cuenta que los valores absolutos $|\det(\partial f_4(z_3)/\partial z_3)|$, $|\det(\partial f_1(z_0)/\partial z_0)|$ de los determinantes Jacobianos para las transformaciones f_4 y f_1 se expresan utilizando valores absolutos de los determinantes Jacobianos tras la deducción en lugar de tras el aprendizaje (es decir, tras el procesamiento utilizando el modelo aprendido).

Además, como se describió anteriormente, se supone que la distribución $q_0(z_0)$ de probabilidad es una distribución de probabilidad con la que es fácil realizar una estimación estricta de la densidad de probabilidad. Por ejemplo, si una distribución Gaussiana N(0, I), en la que un promedio es 0 y una varianza es una matriz identidad I, se establece como la distribución $q_0(z_0)$ de probabilidad, la distribución $q_0(z_0)$ de probabilidad puede expresarse utilizando la siguiente expresión.

[Fórmula 11]

$$q_0(z_0) = -(2\pi)^{-D/2}\exp\left(-\|z_0\|_2^2\right) \quad \cdots(10)$$

Por lo tanto, puede entenderse que, al establecer la distribución de probabilidad de los datos x de entrada generados a partir de los datos de aprendizaje como $q_1(x;\theta)$ y utilizando la expresión (5), es posible calcular una densidad $q_1(X_i;\theta)$ de probabilidad de los datos x_i de entrada a partir de la densidad $q_0(z_{0,i})$ de probabilidad de la variable latente $z_{0,i}$.

Posteriormente, se describirá un método de aprendizaje del parámetro θ. De manera similar al aprendizaje convencional de una red neuronal, es posible realizar el aprendizaje utilizando, por ejemplo, un método de descenso de gradiente, Momentum SGD (Descenso de Gradiente Estocástico), ADAM (Estimación Adaptativa de Momentos) o una combinación de los mismos, utilizando una función $L(\theta)$ de pérdida. En un caso donde se utilice un Flujo de Normalización, a menudo se utiliza como función $L(\theta)$ de pérdida un promedio de probabilidades logarítmicas negativas definidas mediante la siguiente expresión.

[Fórmula 12]

$$L(\theta) = -1/N \sum_{i=1}^{N} \log q_1(x_i;\theta) \quad \cdots(11)$$

Tenga en cuenta que es posible utilizar un método de aprendizaje en mini lotes que se realiza en unidades de un conjunto de datos de aprendizaje denominado mini lote en el aprendizaje descrito anteriormente. Aquí, un mini lote se refiere a una pluralidad de datos de aprendizaje seleccionados, aleatoriamente, de todos los datos de aprendizaje. Se calcula un valor de la función L(θ) de pérdida para cada mini lote.

(2) Fase de aprendizaje adaptativo

A continuación, se describirá un método de aprendizaje adaptativo de la distribución $q_2(x;\theta)$ de probabilidad de los datos x de entrada generados a partir de sonido normal (es decir, datos de aprendizaje adaptativo) emitidos desde el equipo objetivo de detección de anomalías. Por ejemplo, sólo es necesario ejecutar el aprendizaje en el siguiente procedimiento utilizando $z_4 = f_5^{-1}(x)$, $z_3 = f_4^{-1}(z_4)$, $z_2 = f_3^{-1}(z_3)$, $z_1 = f_2^{-1}(z_2)$, $z_0 = f_1^{-1}(z_1)$. Primero, $\{z'_{4,i}\}_{i=1}^{M}(z'_{4,i} = f_5^{-1}(x'_i))$ se calcula a partir de un conjunto de los datos $\{x'_i\}_{i=1}^{M}$ de entrada. Entonces, se obtienen un promedio y una varianza de $\{z'_{4,i}\}_{i=1}^{M}$. Por último, la media y la varianza obtenidas se sustituyen por m, s^2 en la expresión (7c). De manera similar, $\{z'_{1,i}\}_{i=1}^{M}(z'_{1,i} = f_2^{-1}(f_3^{-1}(f_4^{-1}(f_5^{-1}(x'_i)))))$ se calcula a partir del conjunto de los datos $\{x'_i\}_{i=1}^{M}$ de entrada. Entonces, se obtienen un promedio y una varianza de $\{z'_{1,i}\}_{i=1}^{M}$. Por último, la media y la varianza obtenidas se sustituyen por m, s^2 en la expresión (7c).

(3) Fase de prueba

Como método para la detección de anomalías, por ejemplo, es posible utilizar el método descrito en <<Detección no supervisada de sonidos anómalos>>.

<<Efectos>>

Al introducir normalización adaptativa por lotes al Flujo de Normalización, pueden obtenerse los siguientes efectos.

(1) Es posible ajustar una diferencia entre la distribución de los datos de aprendizaje y la distribución de los datos de prueba, de modo que sea posible reducir la degradación en la precisión de detección de anomalías en el dominio para prueba.

En un caso donde se utilice normalización adaptativa por lotes en una capa final, es posible corregir una brecha en la traducción paralela y una escala en una dirección axial de distribución de datos entre dominios. Además, en un caso donde una diferencia en la distribución de datos entre diferentes dominios se exprese con traducción paralela y escalado en una dirección axial, en principio, al introducir normalización adaptativa por lotes en una primera capa, es posible ejecutar la detección de anomalías con alta precisión en el dominio para prueba incluso si el parámetro θ, que se ha aprendido en el dominio para aprendizaje, se aplica al dominio para prueba sin cambios.

(2) Es posible aprender, de forma adaptativa, la segunda distribución de probabilidad con un bajo coste de cálculo.

El procesamiento que es necesario para el aprendizaje adaptativo de la segunda distribución de probabilidad es, como se describió anteriormente, básicamente un mero cálculo de una cantidad de estadísticas en la normalización adaptativa por lotes para los datos de aprendizaje adaptativo. Por lo tanto, el aprendizaje adaptativo puede ejecutarse con un coste de cálculo menor que en un caso donde se repite el aprendizaje normal, de modo que es posible ejecutar el aprendizaje adaptativo en línea en algunos casos.

Además, como en el ejemplo descrito anteriormente, en un caso donde se introduce una transformación lineal en el Flujo de Normalización, mediante una matriz W correspondiente a la transformación lineal que está sujeta a descomposición LU o a descomposición LDU, pueden obtenerse los siguientes efectos.

(3) Se reduce el coste del cálculo de la densidad de probabilidad, por lo que se reduce el coste del aprendizaje.

En el Flujo de Normalización, es necesario calcular cada determinante Jacobiano de la transformación lineal f. Por lo tanto, en un caso donde la matriz W se mantenga en una forma sin estar sujeta a descomposición LU o descomposición LDU, se requiere un coste del cálculo de $O(k^3)$ para calcular el determinante |W|, donde k es un orden de W. Sin embargo, en un caso donde la matriz W se mantenga en una forma que está sujeta a descomposición LU o a descomposición LDU de manera que W = LU o W = LDU, debido a que es posible obtener IWI utilizando |W| = |LU| = |L| × |U| (es decir, un producto de todos los elementos diagonales de L y U) o |W| = |LDU| = |L| × |D| × |U| (es decir, un producto de todos los elementos diagonales de L, D y U), es posible realizar el cálculo a una velocidad extremadamente alta.

[Primera realización]

Se considerará un caso donde, en una situación en la que existan dos o más equipos del mismo tipo, se detecte una anomalía de uno de ellos (que se denominará equipo objetivo de detección de anomalías). Para lograr esto, en primer lugar, una distribución de probabilidad (en lo sucesivo, denominada primera distribución de probabilidad) que indica una distribución de sonido normal (en lo sucesivo, denominado sonido normal para el aprendizaje) emitido desde uno o más equipos diferentes de este equipo objetivo de detección de anomalías se aprende a partir del sonido normal emitido desde uno o más equipos diferentes del equipo objetivo de detección de anomalías (fase de aprendizaje). Luego, una distribución de probabilidad (en lo sucesivo, denominada segunda distribución de probabilidad) que indica una distribución de sonido normal (en lo sucesivo, denominado sonido normal para el aprendizaje adaptativo) emitido desde el equipo objetivo de detección de anomalías se aprende, de forma adaptativa, a partir de la primera distribución de probabilidad, utilizando el sonido normal emitido desde el equipo objetivo de detección de anomalías (fase de aprendizaje adaptativo). Luego, se determina si el equipo es anormal o no a partir del sonido (en lo sucesivo, denominado sonido objetivo de detección de anomalías) emitido desde el equipo objetivo de detección de anomalías (fase de prueba (fase de detección de anomalías)).

El aparato 100 de aprendizaje de distribuciones de probabilidad aprende la primera distribución de probabilidad a partir del sonido normal para el aprendizaje. El aparato 200 de aprendizaje adaptativo de distribuciones de probabilidad aprende, de forma adaptativa, la segunda distribución de probabilidad a partir de la primera distribución de probabilidad utilizando el sonido normal para el aprendizaje adaptativo. El aparato 300 de detección de anomalías determina si el equipo es anormal o no a partir del sonido objetivo de detección de anomalías.

El aparato 100 de aprendizaje de distribuciones de probabilidad se describirá a continuación con referencia a la Fig. 1 y a la Fig. 2. La Fig. 1 es un diagrama de bloques que ilustra una configuración del aparato 100 de aprendizaje de distribuciones de probabilidad. La Fig. 2 es un diagrama de flujo que ilustra el funcionamiento del aparato 100 de aprendizaje de distribuciones de probabilidad. Como se ilustra en la Fig. 1, el aparato 100 de aprendizaje de distribuciones de probabilidad incluye una unidad 110 generadora de datos de entrada, una unidad 120 de estimación de la variable latente, una unidad 130 de cálculo de la función de pérdida, una unidad 140 de actualización del parámetro, una unidad 150 de determinación de la condición de convergencia y una unidad 190 de grabación. La unidad 190 de grabación es un componente que graba la información necesaria para el procesamiento del aparato 100 de aprendizaje de distribuciones de probabilidad según corresponda. La unidad 190 de grabación graba, por ejemplo, un parámetro θ de la primera distribución $q_1(x;\theta)$ de probabilidad que debe aprenderse. Como valor inicial del parámetro θ, se graba, por ejemplo, un valor generado utilizando un número aleatorio.

El aparato 100 de aprendizaje de distribuciones de probabilidad está conectado a una unidad 910 de grabación de sonido normal de aprendizaje. En la unidad 910 de grabación de sonido normal de aprendizaje, el sonido normal para el aprendizaje preparado de antemano se graba como datos de aprendizaje. Como se describió anteriormente, es preferible preparar el sonido normal para el aprendizaje tanto como sea posible.

El funcionamiento del aparato 100 de aprendizaje de distribuciones de probabilidad se describirá de acuerdo con la Fig. 2. La unidad 110 generadora de datos de entrada genera los datos x_i ($i = 1,..., N$) de entrada a partir del sonido normal para el aprendizaje s_i ($i = 1,..., N$) que se ingresa (S110). Por ejemplo, como en la expresión (1), sólo es necesario generar un vector que incluya un espectro de amplitud logarítmica del sonido normal para el aprendizaje s_i como elemento, y establecerlo como el dato x_i de entrada. Tenga en cuenta que también es posible utilizar una cantidad de característica acústica distinta a la descrita anteriormente como el dato x_i de entrada generado a partir del sonido normal para el aprendizaje s_i.

La unidad 120 de estimación de la variable latente estima una variable latente $z_{0,i}$ ($i = 1, ..., N$) correspondiente a los datos x_i de entrada de los datos x_i ($i = 1, ..., N$) de entrada generados en S110 utilizando el parámetro θ de la primera distribución $q_1(x;\theta)$ de probabilidad (S120). Tenga en cuenta que el parámetro θ utilizado aquí es un valor que se está aprendiendo.

Aquí, una variable x de la primera distribución $q_1(x;\theta)$ de probabilidad que se va a aprender es una variable que indica datos de entrada generados a partir del sonido normal emitido desde uno o más equipos diferentes del equipo objetivo de detección de anomalías, y la variable x se expresa como $x = f_K(f_{K-1}(...(f_1(z_0))...))$ utilizando transformaciones f_i ($i = 1, ..., K$, K es un número entero de 1 o mayor, y existen transformaciones inversas f_i^{-1} para las transformaciones f_i) y la variable latente z_0.

Por tanto, la variable latente $z_{0,i}$ correspondiente a los datos x_i de entrada se proporciona utilizando la siguiente expresión.

[Fórmula 13]

$$z_{0,i} = f_1^{-1}(f_2^{-1}(\cdots(f_K^{-1}(x_i))\cdots)) \quad \cdots(12)$$

Además, se supone que la variable latente $z_{0,i}$ (i = 1, ..., N) se genera de acuerdo con la distribución $q_0(z_0)$ de probabilidad de la variable latente z_0. Sin embargo, la distribución $q_0(z_0)$ de probabilidad tiene características que hacen que sea fácil realizar una estimación estricta de la densidad de probabilidad.

Por lo tanto, la distribución $q_1(x;\theta)$ de probabilidad puede expresarse utilizando la siguiente expresión que utiliza la distribución $q_0(z_0)$ de probabilidad (véase la expresión (5)).

[Fórmula 14]

$$q_1(x;\theta) = q_0(z_0)\left|\det(\partial f_K(z_{K-1};\theta_K)/\partial z_{K-1})\right|$$
$$\left|\det(\partial f_{K-1}(z_{K-2};\theta_{K-1})/\partial z_{K-2})\right|\cdots\left|\det(\partial f_1(z_0;\theta_1)/\partial z_0)\right|\cdots(5)'$$

Aquí, θ_i es un parámetro correspondiente a la transformación f_i, y $\theta = [\theta_1^T, \theta_2^T, ..., \theta_K^T]^T$.

La unidad 120 de estimación de la variable latente puede realizarse utilizando una red neuronal NN que calcula la variable latente z_0 a partir de los datos x de entrada. Tenga en cuenta que la expresión de cálculo es la siguiente.

[Fórmula 15]

$$z_0 = f_K^{-1}(f_2^{-1}(...(f_K^{-1}(x))...)) \qquad ...(12)'$$

En otras palabras, la red neuronal NN recibe los datos x de entrada como entrada, procede con el cálculo, secuencialmente, a partir de transformaciones inversas f_K^{-1}, f_{K-1}^{-1}, ..., f_2^{-1}, f_1^{-1}y finalmente, emite la variable latente z_0.

Al menos, una transformación inversa $f_{i\_0}^{-1}$ de las transformaciones f_i (i = 1, ..., K) descritas anteriormente (donde i_0 es un número entero que satisface $1 \leq i_0 \leq K$) se realiza normalización adaptativa por lotes. Mediante esta técnica, es posible aprender, de forma adaptativa, la segunda distribución $q_2(x;\theta)$ de probabilidad utilizando una cantidad relativamente pequeña de sonido normal para el aprendizaje adaptativo (véase <<Puntos>>).

Además, al menos, una transformación inversa $f_{i\_1}^{-1}$ de las transformaciones f_i (i = 1, ..., K) (donde i_1 es un número entero que satisface $1 \leq i_1 \leq K$) puede hacerse una transformación lineal, y una matriz correspondiente a la transformación lineal puede expresarse como un producto de una matriz triangular inferior y una matriz triangular superior, o como un producto de una matriz triangular inferior, una matriz diagonal y una matriz triangular superior. Mediante esta técnica, es posible ejecutar el cálculo (cálculo en la unidad 130 de cálculo de la función de pérdida que se describirá más adelante) de la densidad de probabilidad que se requiere tras el aprendizaje de la primera distribución $q_1(x;\theta)$ de probabilidad a bajo coste (véase (3) en <<Efectos>>).

Como ejemplo específico de las transformaciones f_i (i = 1, ..., K), es posible utilizar cinco transformaciones cuyas transformaciones inversas pueden expresarse utilizando la expresión (8a) a la expresión (8e) suponiendo que, por ejemplo, K = 5.

La unidad 130 de cálculo de la función de pérdida calcula un valor de la función $L(\theta)$ de pérdida que se utilizará para la optimización del parámetro θ de la primera distribución $q_1(x;\theta)$ de probabilidad de la variable latente $z_{0,i}$ (i = 1, ..., N) estimado en S120 (S130). La función $L(\theta)$ de pérdida puede, por ejemplo, definirse como un promedio de probabilidades logarítmicas negativas como en la expresión (11). En este momento, mientras que es necesario calcular la densidad $q_1(X_i;\theta)$ de probabilidad de los datos x_i (i = 1, ..., N) de entrada, la densidad $q_1(X_i;\theta)$ de probabilidad de los datos x_i de entrada puede calcularse utilizando la densidad $q_0(z_{0,i})$ de probabilidad de la variable latente $z_{0,i}$ correspondiente a los datos x_i de entrada. Por ejemplo, en un caso donde la distribución $q_0(z_0)$ de probabilidad es la distribución Gaussiana N(0, I), la densidad $q_0(z_{0,i})$ de probabilidad de la variable latente $z_{0,i}$ puede calcularse utilizando la siguiente expresión.

[Fórmula 16]

$$q_0(z_{0,i}) = -(2\pi)^{-D/2}\exp(-\|z_{0,i}\|_2^2) \quad \cdots(10)'$$

Por lo que es posible calcular la densidad $q_1(x_i;\theta)$ de probabilidad de los datos x_i de entrada a partir de la densidad $q_0(z_{0,i})$ de probabilidad calculada descrita anteriormente de la variable latente $z_{0,i}$ utilizando la expresión (5)'.

La unidad 140 de actualización del parámetro actualiza el parámetro θ de la primera distribución $q_1(x;\theta)$ de probabilidad para optimizar (minimizar) el valor de la función $L(\theta)$ de pérdida calculado en S130 (S140). Por ejemplo, se utiliza, preferentemente, un método de descenso de gradiente en la actualización del parámetro θ.

La unidad 150 de determinación de la condición de convergencia determina las condiciones de convergencia establecidas de antemano como condiciones de terminación de la actualización del parámetro, emite la primera distribución $q_1(x;\theta)$ de probabilidad utilizando el parámetro θ actualizado en S140 en un caso donde se cumplan las condiciones de convergencia, y repite el procesamiento desde S110 a S140 en un caso donde no se cumplan las condiciones de convergencia (S150). Como condiciones de convergencia, por ejemplo, puede emplearse una condición relativa a si el número de veces de ejecución del procesamiento desde S110 a S140 alcanza un número predeterminado de veces. A la salida, pueden emitirse el parámetro θ (este parámetro también se denominará parámetro aprendido) actualizado en S140 y las transformaciones inversas $f_K^{-1}(x;\theta_K)$, $f_{K-1}^{-1}(z_{K-1};\theta_{K-1})$, ..., $f_2^{-1}(z_2;\theta_2)$, $f_1^{-1}(z_1;\theta_1)$.

Obsérvese que un componente que incluya la unidad 110 generadora de datos de entrada, la unidad 120 de estimación de la variable latente, la unidad 130 de cálculo de la función de pérdida, la unidad 140 de actualización del parámetro y la unidad 150 de determinación de la condición de convergencia se denominará unidad 105 de aprendizaje. En otras palabras, la unidad 105 de aprendizaje es un componente que aprende la primera distribución de probabilidad a partir de sonido normal para el aprendizaje.

Según la realización de la presente invención, es posible aprender la primera distribución q1(x;θ) de probabilidad, lo que permite un fácil ajuste de una diferencia entre la distribución de los datos de aprendizaje y la distribución de los datos de prueba.

<Modificación>

Mientras que la descripción se ha proporcionado suponiendo que el aparato 100 de aprendizaje de distribuciones de probabilidad aprende una primera distribución $q_1(x;\theta)$ de probabilidad a partir de sonido normal para el aprendizaje, que es un sonido normal emitido desde uno o más equipos diferentes del equipo objetivo de detección de anomalías, el aparato 100 de aprendizaje de distribuciones de probabilidad puede aprender la primera distribución $q_1^{(1)}(x;\theta)$, ..., $q_1^{(W)}(x;\theta)$ de probabilidad que indica la distribución del sonido normal emitido desde los respectivos W equipos, mientras que el número de uno o más equipos diferentes del equipo objetivo de detección de anomalías se establece como W (W es un número entero de 1 o mayor). En este caso, mediante un promedio y una varianza en la normalización adaptativa por lotes que se calcula para cada dato de entrada generado a partir del sonido normal para el aprendizaje emitido desde el mismo equipo entre los datos x_i (i = 1,..., N) de entrada (correspondiente a un mini lote), la unidad 120 de estimación de la variable latente ejecuta el procesamiento de estimación de la variable latente. En otras palabras, en el cálculo en la normalización adaptativa por lotes, se utilizan W conjuntos de promedios y varianzas en lugar de un conjunto de un promedio y de una varianza. Sin embargo, el parámetro θ aprendido es común entre W conjuntos de la primera distribución $q_1^{(1)}(x;\theta)$, ..., $q_1^{(W)}(x;\theta)$ de probabilidad, y hay un conjunto del parámetro aprendido.

[Segunda realización]

La segunda realización descrita a continuación se refiere a un ejemplo que no está abarcado por las reivindicaciones pero es útil para entender la presente invención.

El aparato 200 de aprendizaje adaptativo de distribuciones de probabilidad se describirá a continuación con referencia a la Fig. 3 y a la Fig. 4. La Fig. 3 es un diagrama de bloques que ilustra una configuración del aparato 200 de aprendizaje adaptativo de distribuciones de probabilidad. La Fig. 4 es un diagrama de flujo que ilustra el funcionamiento del aparato 200 de aprendizaje adaptativo de distribuciones de probabilidad. Como se ilustra en la Fig. 3, el aparato 200 de aprendizaje adaptativo de distribuciones de probabilidad incluye una unidad 110 generadora de datos de entrada, una unidad 240 de actualización del parámetro, una unidad 250 de salida, y la unidad 190 de grabación. La unidad 190 de grabación es un componente que graba la información necesaria para el procesamiento del aparato 200 de aprendizaje adaptativo de distribuciones de probabilidad según corresponda. La unidad 190 de grabación, por ejemplo, graba el parámetro θ (es decir, el parámetro aprendido) de la primera distribución $q_1(x;\theta)$ de probabilidad aprendida utilizando el aparato 100 de aprendizaje de distribuciones de probabilidad. Este parámetro aprendido se convierte en un valor inicial del parámetro θ de la segunda distribución $q_2(x;\theta)$ de probabilidad. Tenga en cuenta que la variable x de la segunda distribución $q_2(x;\theta)$ de probabilidad es una variable que indica datos de entrada generados a partir del sonido normal emitido desde el equipo objetivo de detección de anomalías.

El aparato 200 de aprendizaje adaptativo de distribuciones de probabilidad está conectado a una unidad 920 de grabación de sonido normal de aprendizaje adaptativo. En la unidad 920 de grabación de sonido normal de aprendizaje adaptativo, el sonido normal para el aprendizaje adaptativo preparado de antemano se graba como datos de aprendizaje adaptativo. Como se describió anteriormente, sólo es necesario preparar una cantidad relativamente pequeña del sonido normal para el aprendizaje adaptativo en comparación con una cantidad de sonido normal para el aprendizaje.

El funcionamiento del aparato 200 de aprendizaje adaptativo de distribuciones de probabilidad se describirá de acuerdo con la Fig. 4. La unidad 110 generadora de datos de entrada genera los datos x'_i (i = 1, ..., M) de entrada a partir del sonido normal para el aprendizaje adaptativo s'_i (i = 1,..., M) que se introduce (S110). Aquí, mientras que el número de piezas de sonido normal para el aprendizaje adaptativo M es básicamente un número entero que no exceda el número de piezas de sonido normal para el aprendizaje N, el número de piezas de sonido normal para el aprendizaje adaptativo M puede ser un número entero que exceda el número de piezas de sonido normal para el aprendizaje N.

La unidad 240 de actualización del parámetro actualiza el parámetro θ de la segunda distribución $q_2(x;\theta)$ de probabilidad utilizando los datos x'_i (i = 1, ..., M) de entrada generados en S110 (S240). Específicamente, para una transformación f_{i_0} cuya transformación inversa es una normalización adaptativa por lotes, sólo es necesario actualizar un promedio y una varianza que se utilizarán para el cálculo con un promedio y una varianza de los datos $z'_{i\_0,i}(=f_{i\_0\text{-}1}^{-1}(f_{i\_0\text{-}2}^{-1}(...(f_K^{-1}(x'_1))...)))$ de entrada que se ingresarán a una transformación inversa $f_{i\_0}^{-1}$ calculada a partir de los datos x'_i (i = 1,..., M) de entrada. En otras palabras, un promedio y una varianza de los datos $z'_{i\_0,1}$ (i = 1, ..., M) de entrada se sustituyen por m y s^2 en la expresión (7c).

La unidad 250 de salida emite la segunda distribución $q_2(x;\theta)$ de probabilidad utilizando el parámetro θ actualizado en S240 (S250). Además, a la salida, pueden emitirse el parámetro θ (este parámetro también se denominará parámetro aprendido) actualizado en S240, y las transformaciones inversas $f_K^{-1}(x;\theta_K)$, $f_{K-1}^{-1}(z_{K-1};\theta_{K-1})$, ..., $f_2^{-1}(z_2;\theta_2)$, $f_1^{-1}(z_1;\theta_1)$.

Tenga en cuenta que un componente que incluya la unidad 110 generadora de datos de entrada, la unidad 240 de actualización del parámetro y la unidad 250 de salida se denominará unidad 205 de aprendizaje adaptativo. En otras palabras, la unidad 205 de aprendizaje adaptativo es un componente que aprende, de forma adaptativa, la segunda distribución de probabilidad a partir de la primera distribución de probabilidad utilizando el sonido normal para el aprendizaje adaptativo.

Según la realización de la presente invención, es posible aprender la segunda distribución $q_2(x;\theta)$ de probabilidad ajustando una diferencia entre la distribución de los datos de aprendizaje y la distribución de los datos de prueba.

<Modificación>

Además, en un caso donde el aparato 100 de aprendizaje de distribuciones de probabilidad aprenda las W primeras distribuciones $q_1^{(1)}(x;\theta)$, ..., $q_1^{(W)}(x;\theta)$ de probabilidad, sólo hay un conjunto del parámetro θ aprendido. El aparato 200 de aprendizaje adaptativo de distribuciones de probabilidad realiza el aprendizaje adaptativo utilizando este conjunto del parámetro.

[Tercera realización]

La tercera realización descrita a continuación se refiere a un ejemplo que no está abarcado por las reivindicaciones, pero es útil para entender la presente invención. El aparato 300 de detección de anomalías se describirá a continuación con referencia a la Fig. 5 y a la Fig. 6. La Fig. 5 es un diagrama de bloques que ilustra una configuración del aparato 300 de detección de anomalías. La Fig. 6 es un diagrama de flujo que ilustra el funcionamiento del aparato 300 de detección de anomalías. Como se ilustra en la Fig. 5, el aparato 300 de detección de anomalías incluye una unidad 110 generadora de datos de entrada, una unidad 320 de estimación del grado de anomalía, una unidad 330 de determinación de anomalías y una unidad 390 de grabación. La unidad 390 de grabación es un componente que graba la información necesaria para el procesamiento del aparato 300 de detección de anomalías según corresponda. La unidad 390 de grabación, por ejemplo, graba el parámetro θ (es decir, el parámetro aprendido) de la segunda distribución $q_2(x;\theta)$ de probabilidad que se ha aprendido utilizando el aparato 200 de aprendizaje adaptativo de distribuciones de probabilidad.

En otras palabras, el aparato 300 de detección de anomalías ejecuta la detección de anomalías utilizando la segunda distribución $q_2(x;\theta)$ de probabilidad que utiliza este parámetro aprendido, como la segunda distribución de probabilidad aprendida.

El funcionamiento del aparato 300 de detección de anomalías se describirá de acuerdo con la Fig. 6. La unidad 110 generadora de datos de entrada genera los datos x de entrada a partir del sonido s objetivo de detección de anomalías que se introduce (S110).

La unidad 320 de estimación del grado de anomalía estima un grado de anomalía que indica un grado de anomalía del equipo a partir de los datos x de entrada generados en S110 en función de la segunda distribución $q_2(x;\theta)$ de probabilidad aprendida (S320). La unidad 320 de estimación del grado de anomalía se describirá a continuación con referencia a la Fig. 7 y a la Fig. 8. La Fig. 7 es un diagrama de bloques que ilustra una configuración de la unidad 320 de estimación del grado de anomalía. La Fig. 8 es un diagrama de flujo que ilustra el funcionamiento de la unidad 320 de estimación del grado de anomalía. Como se ilustra en la Fig. 7, la unidad 320 de estimación del grado de anomalía incluye una unidad 321 de cálculo de la variable latente y una unidad 322 de cálculo del grado de anomalía.

El funcionamiento de la unidad 320 de estimación del grado de anomalía se describirá de acuerdo con la Fig. 8. La unidad 321 de cálculo de la variable latente calcula una variable latente z_0 correspondiente a los datos x de entrada a partir de los datos x de entrada generados en S110 (S321). Específicamente, la variable latente z_0 puede calcularse utilizando una red neuronal en la que el parámetro aprendido de la segunda distribución de probabilidad se establece como parámetro de la red neuronal NN.

La unidad 322 de cálculo del grado de anomalía calcula un grado $A(x;\theta)$ de anomalía con respecto a los datos x de entrada a partir de la variable latente z_0 calculada en S321 (S322). El grado de anomalía puede, por ejemplo, calcularse utilizando la siguiente expresión.

[Fórmula 17]

$$A(x;\theta) = -\ln q_2(x;\theta) \qquad ...(2)'$$

En un caso donde la distribución $q_0(z_0)$ de probabilidad es una distribución con la que es fácil realizar una estimación estricta de la densidad de probabilidad, la densidad $q_2(x;\theta)$ de probabilidad de los datos x de entrada puede calcularse utilizando la siguiente expresión.

[Fórmula 18]

$$q_2(x;\theta) = q_0(z_0)\left|\det(\partial f_K(z_{K-1};\theta_K)/\partial z_{K-1})\right| \left|\det(\partial f_{K-1}(z_{K-2};\theta_{K-1})/\partial z_{K-2})\right| ...$$

$$\left|\det(\partial f_1(z_0;\theta_1)/\partial z_0)\right| \qquad ...(5)''$$

Al calcular la expresión (5)", por ejemplo, en un caso donde la distribución $q_0(z_0)$ de probabilidad es la distribución Gaussiana N(0, I) y las transformaciones inversas son cinco transformaciones que pueden expresarse con la expresión (8a) a la expresión (8e), sólo es necesario utilizar la expresión (10) y la expresión (9a) a la expresión (9e).

La unidad 330 de determinación del grado de anomalía genera un resultado de determinación que indica si el equipo es anormal o no a partir del grado A(x;θ) de anomalía estimado en S320 (S330). Por ejemplo, sólo es necesario generar un resultado de determinación que indique anormal en un caso donde R = 1, y generar un resultado de determinación que indique normal en un caso donde R = 0, utilizando la expresión (3).

En otras palabras, puede decirse que el aparato 300 de detección de anomalías incluye la unidad 320 de estimación del grado de anomalía que estima un grado de anomalía que indica un grado de anomalía del equipo objetivo de detección de anomalías a partir del sonido emitido desde el equipo objetivo de detección de anomalías (sonido objetivo de detección de anomalías) en función de la asociación entre la primera distribución de probabilidad que indica la distribución del sonido normal emitido desde uno o más equipos diferentes del equipo objetivo de detección de anomalías y el sonido normal (sonido normal para el aprendizaje adaptativo) emitido desde el equipo objetivo de detección de anomalías. Un ejemplo de la asociación es la segunda distribución de probabilidad que indica la distribución del sonido normal emitido desde el equipo objetivo de detección de anomalías, que se obtiene actualizando la primera distribución de probabilidad utilizando el sonido normal para el aprendizaje adaptativo.

Según la realización de la presente invención, al realizar la detección de anomalías utilizando la segunda distribución de probabilidad obtenida ajustando una diferencia entre la distribución de los datos de aprendizaje y la distribución de los datos de prueba, es posible realizar la detección de anomalías con alta precisión. En otras palabras, es posible reducir la degradación de la precisión de detección de anomalías.

Tenga en cuenta que, si bien se ha descrito desde la primera realización hasta la tercera realización un método para calcular el grado A(x;θ) de anomalía utilizando un Flujo de Normalización, el grado de anomalía también puede obtenerse utilizando otros modelos estadísticos. Por ejemplo, también es posible utilizar el codificador automático (AE) descrito en la bibliografía 7 de Referencia no de patente. El codificador automático es un conjunto de un codificador y un decodificador.

(Bibliografía 7 de Referencia no de patente: Y. Koizumi, S. Saito, H. Uematsu, Y. Kawachi, y N. Harada, et al., "Unsupervised Detection of Anomalous Sound based on Deep Learning and the Neyman-Pearson Lemma", Transacciones del IEEE/ACM sobre Procesamiento de Audio, Voz, y Lenguaje, Vol.27-1, págs.212-224, 2019.)

Se describirán a continuación los antecedentes técnicos de una cuarta realización a una sexta realización de la presente invención y de cada realización.

<Antecedentes técnicos>

En un caso donde se utilice un codificador automático, el grado de anomalía puede calcularse utilizando la siguiente expresión.

[Fórmula 19]

$$A(x;\theta) = \left\|x - D(E((x,\theta_E),\theta_D)\right\|^2 \qquad ...(13)$$

Aquí, $\|\bullet\|$ indica una norma L_2, E y D indican, respectivamente, un codificador y un decodificador, θ_E y θ_D indican, respectivamente, un parámetro del codificador E y un parámetro del decodificador D. En otras palabras, $\theta = \{\theta_E, \theta_D\}$.

Tanto el codificador E como el decodificador D pueden configurarse como una red neuronal. En este caso, por ejemplo, θ se aprende de modo que un error de reconstrucción (restauración) de los datos de aprendizaje del sonido normal sea mínimo.

[Fórmula 20]

$$\theta \leftarrow \arg\min_{\theta} \frac{1}{N} \sum_{i=1}^{N} A(x_i;\theta) \quad \cdots(14)$$

Aquí, x_i son los datos de aprendizaje del i-ésimo sonido normal, y N es el número de muestras de los datos de aprendizaje del sonido normal.

Para realizar aprendizaje adaptativo utilizando un codificador automático, sólo es necesario configurar ambos o uno del codificador E y del decodificador D como una red neuronal utilizando normalización adaptativa por lotes (AdaBN). En otras palabras, se utiliza normalización adaptativa por lotes a mitad del cálculo de ambos o de uno del codificador E y del decodificador D. Por ejemplo, en lugar de configurar el codificador E como una red neuronal de tres capas de $E(x;\theta_E) = W_2[\sigma(W_1x+b_1)] + b_2$, sólo es necesario configurar el codificador E como una red neuronal que calcula la siguiente expresión en la que se inserta una capa AdaBN.

[Fórmula 21]

$$E(x,\theta_E) = W_2\{\mathrm{AdaBN}[\sigma(W_1x + b_1)]\} + b_2 \quad \ldots(15)$$

Aquí, W_1 y W_2 indican matrices de ponderación, b_1 y B_2 indican vectores de sesgo, σ indica una función de activación. La capa AdaBN es una capa que ejecuta el cálculo de AdaBN (normalización adaptativa por lotes) como en, por ejemplo, la expresión (8b) y la expresión (8e).

[Cuarta realización]

La cuarta realización descrita a continuación se refiere a un ejemplo que no está abarcado por las reivindicaciones pero que es útil para entender la presente invención.

Se considerará un caso donde, en una situación en la que existan dos o más equipos del mismo tipo, se detecte una anomalía de uno de ellos (que se denominará equipo objetivo de detección de anomalías). Para lograr esto, en primer lugar, un codificador automático (en lo sucesivo, denominado primer codificador automático) que restaura el sonido normal (en lo sucesivo, denominado sonido normal para el aprendizaje) emitido desde uno o más equipos diferentes de este equipo objetivo de detección de anomalías se aprende a partir del sonido normal emitido desde uno o más equipos diferentes del equipo objetivo de detección de anomalías (fase de aprendizaje). Luego, un codificador automático (en lo sucesivo, denominado segundo codificador automático) que restaura el sonido normal (en lo sucesivo, denominado sonido normal para el aprendizaje adaptativo) emitido desde el equipo objetivo de detección de anomalías se aprende, de forma adaptativa, a partir del primer codificador automático utilizando el sonido normal emitido desde el equipo objetivo de detección de anomalías (fase de aprendizaje adaptativo). Luego, se determina si el equipo es anormal o no a partir del sonido (en lo sucesivo, denominado sonido objetivo de detección de anomalías) emitido desde el equipo objetivo de detección de anomalías (fase de prueba (fase de detección de anomalías)).

El aparato 400 de aprendizaje con codificador automático aprende el primer codificador automático a partir del sonido normal para el aprendizaje. El aparato 500 de aprendizaje adaptativo con codificador automático aprende, de forma adaptativa, el segundo codificador automático a partir del primer codificador automático utilizando el sonido normal para el aprendizaje adaptativo. El aparato 600 de detección de anomalías determina si el equipo es anormal o no a partir del sonido objetivo de detección de anomalías.

El aparato 400 de aprendizaje con codificador automático se describirá a continuación con referencia a la Fig. 9 y a la Fig. 10. La Fig. 9 es un diagrama de bloques que ilustra una configuración del aparato 400 de aprendizaje con codificador automático. La Fig. 10 es un diagrama de flujo que ilustra el funcionamiento del aparato 400 de aprendizaje con codificador automático. Como se ilustra en la Fig. 9, el aparato 400 de aprendizaje con codificador automático incluye la unidad 110 generadora de datos de entrada, una unidad 420 de estimación de los datos de entrada restaurados, una unidad 430 de cálculo de la función de pérdida, una unidad 440 de actualización del parámetro, una unidad 450 de determinación de la condición de convergencia, y una unidad 490 de grabación. La unidad 490 de grabación es un componente que graba la información necesaria para el procesamiento del aparato 400 de aprendizaje con codificador automático según corresponda. La unidad 490 de grabación, por ejemplo, graba un parámetro θ del primer codificador automático, que debe aprenderse. Como valor inicial del parámetro θ, se graba, por ejemplo, un valor generado utilizando un número aleatorio.

El aparato 400 de aprendizaje con codificador automático está conectado a una unidad 910 de grabación de sonido normal de aprendizaje. En la unidad 910 de grabación de sonido normal de aprendizaje, el sonido normal para el aprendizaje preparado de antemano se graba como datos de aprendizaje. Como se describió anteriormente, es preferible preparar el sonido normal para el aprendizaje tanto como sea posible.

El funcionamiento del aparato 400 de aprendizaje con codificador automático se describirá de acuerdo con la Fig. 10. La unidad 110 generadora de datos de entrada genera los datos x_i (i = 1, ..., N) de entrada a partir del sonido normal para el aprendizaje s_i (i = 1, ..., N) que se introduce (S110).

La unidad 420 de estimación de los datos de entrada restaurados estima los datos y_i (i = 1, ..., N) de entrada restaurados correspondientes a los datos x_i de entrada a partir de los datos x_i (i = 1, ..., N) de entrada generados en S110 utilizando el parámetro θ del primer codificador automático (S420). Tenga en cuenta que el parámetro θ utilizado aquí es un valor que se está aprendiendo.

La unidad 420 de estimación de los datos de entrada restaurados puede realizarse utilizando una red neuronal que calcule los datos y_i de entrada restaurados a partir de los datos x_i de entrada. Tenga en cuenta que la expresión de cálculo es la siguiente.

[Fórmula 22]

$$y_i = D(E(x_i,\theta_E),\theta_D) \qquad ...(16)$$

Esta red neuronal es el primer codificador automático (que se denominará red neuronal NN).

Aquí, $\theta = \{\theta_E, \theta_D\}$ (donde θ_E y θ_D indican, respectivamente, un parámetro del codificador automático E y un parámetro del decodificador D). Además, al menos, una de una red neuronal que configura el codificador E y de una red neuronal que configura el decodificador D incluyen una capa AdaBN. La capa AdaBN se refiere a una capa que ejecuta el cálculo de AdaBN (normalización adaptativa por lotes). En otras palabras, la red neuronal NN incluye una capa AdaBN.

La unidad 430 de cálculo de la función de pérdida calcula un valor de la función L(θ) de pérdida que se utilizará para la optimización del parámetro θ del primer codificador automático a partir de los datos y_i (i = 1, ..., N) de entrada restaurados estimados en S420 (S430). La función L(θ) de pérdida puede establecerse como, por ejemplo, un promedio de un grado de anomalía definido utilizando la siguiente expresión.

[Fórmula 23]

$$L(\theta) = \frac{1}{N}\sum_{i=1}^{N} A(x_i;\theta) \quad \cdots(17)$$

La unidad 440 de actualización del parámetro actualiza el parámetro θ del primer codificador automático para optimizar (minimizar) el valor de la función L(θ) de pérdida calculada en S430 (S440). Por ejemplo, es preferible utilizar un método de descenso de gradiente en la actualización del parámetro θ.

La unidad 450 de determinación de la condición de convergencia determina las condiciones de convergencia establecidas de antemano como condiciones de terminación para la actualización del parámetro y, en un caso donde se cumplan las condiciones de convergencia, emite el parámetro θ actualizado en S440, mientras que, en un caso donde no se cumplan las condiciones de convergencia, repite el procesamiento desde S110 a S440 (S450). Como condiciones de convergencia, por ejemplo, es posible emplear una condición relativa a si el número de veces de ejecución del procesamiento desde S110 a S440 alcanza un número predeterminado de veces.

Tenga en cuenta que un componente que incluye la unidad 110 generadora de datos de entrada, la unidad 420 de estimación de los datos de entrada restaurados, la unidad 430 de cálculo de la función de pérdida, la unidad 440 de actualización del parámetro, y la unidad 450 de determinación de la condición de convergencia se denominará unidad 405 de aprendizaje. En otras palabras, la unidad 405 de aprendizaje es un componente que aprende (el parámetro θ de) el primer codificador automático a partir del sonido normal para el aprendizaje.

Según la realización de la presente invención, es posible aprender el primer codificador automático que permite un fácil ajuste de una diferencia entre la distribución de los datos de aprendizaje y la distribución de los datos de prueba.

[Quinta realización]

La quinta realización descrita a continuación se refiere a un ejemplo que no está abarcado por las reivindicaciones pero es útil para entender la presente invención.

El aparato 500 de aprendizaje adaptativo con codificador automático se describirá a continuación con referencia a la Fig. 11 y a la Fig. 12. La Fig. 11 es un diagrama de bloques que ilustra una configuración del aparato 500 de aprendizaje adaptativo con codificador automático. La Fig. 12 es un diagrama de flujo que ilustra el funcionamiento del aparato 500 de aprendizaje adaptativo con codificador automático. Como se ilustra en la Fig. 11, el aparato 500 de aprendizaje adaptativo con codificador automático incluye la unidad 110 generadora de datos de entrada, una unidad 540 de actualización del parámetro, una unidad 550 de salida, y una unidad 490 de grabación. La unidad 490 de grabación es un componente que graba la información necesaria para el procesamiento del aparato 500 de aprendizaje adaptativo con codificador automático según corresponda. La unidad 490 de grabación, por ejemplo, graba el

parámetro θ (es decir, el parámetro aprendido) del primer codificador automático aprendido utilizando el aparato 400 de aprendizaje con codificador automático. Este parámetro aprendido se convierte en un valor inicial del parámetro θ del segundo codificador automático.

El aparato 500 de aprendizaje adaptativo con codificador automático está conectado a una unidad 920 de grabación de sonido normal de aprendizaje adaptativo. En la unidad 920 de grabación de sonido normal de aprendizaje adaptativo, el sonido normal para el aprendizaje adaptativo preparado de antemano se graba como datos de aprendizaje adaptativo. Como se describió anteriormente, sólo es necesario preparar una cantidad relativamente pequeña de sonido normal para el aprendizaje adaptativo en comparación con una cantidad de sonido normal para el aprendizaje.

El funcionamiento del aparato 500 de aprendizaje adaptativo con codificador automático se describirá de acuerdo con la Fig. 12. La unidad 110 generadora de datos de entrada genera los datos x'_i (i = 1, ..., M) de entrada a partir del sonido normal para el aprendizaje adaptativo s'i (i = 1, ..., M) que se introduce (S110). Aquí, mientras que el número de piezas de sonido normal para el aprendizaje adaptativo M es, básicamente, un número entero que no exceda el número de piezas de sonido normal para el aprendizaje N, el número de piezas de sonido normal para el aprendizaje adaptativo M puede ser un número entero que exceda el número de piezas de sonido normal para el aprendizaje N.

La unidad 540 de actualización del parámetro actualiza el parámetro θ del segundo codificador automático utilizando los datos x'_i (i = 1, ..., M) de entrada generados en S110 (S540). Específicamente, para la capa AdaBN, que es una capa que calcula la normalización adaptativa por lotes y que está incluida en el primer codificador automático (red neuronal NN), sólo es necesario actualizar un promedio y una varianza utilizados en el cálculo con un promedio y una varianza de los datos y'_i de entrada restaurados calculados a partir de los datos x'_i (i = 1,..., M) de entrada.

La unidad 550 de salida emite el parámetro θ actualizado en S540 (S550).

Tenga en cuenta que un componente que incluya la unidad 110 generadora de datos de entrada, la unidad 540 de actualización del parámetro, y la unidad 550 de salida se denominará unidad 505 de aprendizaje adaptativo. En otras palabras, la unidad 505 de aprendizaje adaptativo es un componente que aprende, de forma adaptativa, el segundo codificador automático a partir del primer codificador automático utilizando el sonido normal para el aprendizaje adaptativo.

Según la realización de la presente invención, es posible aprender el segundo codificador automático ajustando una diferencia entre la distribución de los datos de aprendizaje y la distribución de los datos de prueba.

[Sexta realización]

La sexta realización descrita a continuación se refiere a un ejemplo que no está abarcado por las reivindicaciones, pero es útil para entender la presente invención.

El aparato 600 de detección de anomalías se describirá a continuación con referencia a la Fig. 13 y a la Fig. 14. La Fig. 13 es un diagrama de bloques que ilustra una configuración del aparato 600 de detección de anomalías. La Fig. 14 es un diagrama de flujo que ilustra el funcionamiento del aparato 600 de detección de anomalías. Como se ilustra en la Fig. 13, el aparato 600 de detección de anomalías incluye la unidad 110 generadora de datos de entrada, una unidad 620 de estimación del grado de anomalía, una unidad 630 de determinación de anomalías y una unidad 690 de grabación. La unidad 690 de grabación es un componente que graba la información necesaria para el procesamiento del aparato 600 de detección de anomalías según corresponda. La unidad 690 de grabación, por ejemplo, graba el parámetro θ (es decir, el parámetro aprendido) del segundo codificador automático aprendido utilizando el aparato 500 de aprendizaje adaptativo con codificador automático.

En otras palabras, el aparato 600 de detección de anomalías ejecuta la detección de anomalías utilizando el segundo codificador automático que utiliza este parámetro aprendido como el segundo codificador automático aprendido.

El funcionamiento del aparato 600 de detección de anomalías se describirá de acuerdo con la Fig. 14. La unidad 110 generadora de datos de entrada genera los datos x de entrada a partir del sonido s objetivo de detección de anomalías que se introduce (S110).

La unidad 620 de estimación del grado de anomalía estima un grado de anomalía que indica un grado de anomalía del equipo a partir de los datos x de entrada generados en S110 en función del segundo codificador automático aprendido (S620). La unidad 620 de estimación del grado de anomalía se describirá a continuación con referencia a la Fig. 15 y a la Fig. 16. La Fig. 15 es un diagrama de bloques que ilustra una configuración de la unidad 620 de estimación del grado de anomalía. La Fig. 16 es un diagrama de flujo que ilustra el funcionamiento de la unidad 620 de estimación del grado de anomalía. Como se ilustra en la Fig. 15, la unidad 620 de estimación del grado de anomalía incluye una unidad 621 de cálculo de los datos de entrada restaurados y una unidad 622 de cálculo del grado de anomalía.

El funcionamiento de la unidad 620 de estimación del grado de anomalía se describirá de acuerdo con la Fig. 16. La unidad 621 de cálculo de los datos de entrada restaurados calcula los datos y de entrada restaurados correspondientes a los datos x de entrada a partir de los datos x de entrada generados en S110 (S621). Específicamente, es posible calcular los datos y de entrada restaurados utilizando una red neuronal en la que el parámetro aprendido del segundo

codificador automático se establece como un parámetro de la red neuronal NN.

La unidad 622 de cálculo del grado de anomalía calcula el grado A(x;θ) de anomalía con respecto a los datos x de entrada a partir de los datos y de entrada restaurados, calculados en S621 (S622). El grado de anomalía puede calcularse utilizando, por ejemplo, la expresión (13).

La unidad 630 de determinación del grado de anomalía genera un resultado de determinación que indica si el equipo es anormal o no a partir del grado A(x;θ) de anomalía estimado en S620 (S630). Por ejemplo, sólo es necesario generar un resultado de determinación que indique anormal en un caso donde R = 1, y generar un resultado de determinación que indique normal en un caso donde R = 0, utilizando la expresión (3).

En otras palabras, puede decirse que el aparato 600 de detección de anomalías incluye la unidad 620 de estimación del grado de anomalía que estima un grado de anomalía que indica un grado de anomalía del equipo objetivo de detección de anomalías a partir del sonido emitido desde el equipo objetivo de detección de anomalías (sonido objetivo de detección de anomalías) en función de la asociación entre un primer codificador automático que restaura el sonido normal emitido desde uno o más equipos diferentes del equipo objetivo de detección de anomalías y el sonido normal emitido desde el equipo objetivo de detección de anomalías (sonido normal para el aprendizaje adaptativo). Un ejemplo de la asociación es un segundo codificador automático que restaura el sonido normal emitido desde el equipo objetivo de detección de anomalías, que se obtiene actualizando el primer codificador automático utilizando el sonido normal para el aprendizaje adaptativo.

Según la realización de la presente invención, al realizar la detección de anomalías utilizando el segundo codificador automático en el que se ajusta una diferencia entre la distribución de los datos de aprendizaje y la distribución de los datos de prueba, es posible realizar la detección de anomalías con alta precisión. En otras palabras, es posible reducir la degradación de la precisión de detección de anomalías.

Se describirán a continuación los antecedentes técnicos de una séptima realización a una novena realización de la presente invención y de cada realización.

<Antecedentes técnicos>

En las realizaciones de la presente invención, el aprendizaje de un transformador de datos de transformación del dominio sin datos de pares se realiza utilizando un Flujo de Normalización. El Flujo de Normalización está caracterizado por que el aprendizaje es más fácil que GAN, lo que permite un aprendizaje más estable que una técnica de transformación del dominio basada en GAN (StarGAN) sin datos de pares en la técnica relacionada.

Se describirán a continuación las técnicas relacionadas utilizadas en la realización de la presente invención.

<<Flujo de Normalización>>

El Flujo de Normalización es un método para obtener una distribución que se aproxima a la distribución p(x) de probabilidad con respecto a la generación de datos.

Se supone que $\{f_i(z)\}_{i=1}^{k}$ son K transformaciones que tienen transformaciones inversas (donde $f_i(z)$: $R^D \rightarrow R^D$, R es un conjunto de números reales, D es un número entero de 1 o mayor, y K es un número entero de 1 o mayor). Además, se supone que $f_i^{-1}(z)$ (i = 1, ..., K) es una transformación inversa de $f_i(z)$.

En el Flujo de Normalización se considera que existen las variables latentes $\{z_{0,i}\}_{i=1}^{N}$ correspondientes, respectivamente, al conjunto $\{x_i\}_{i=1}^{N}$ de N piezas de datos de entrada, y los datos x_i de entrada se obtienen transformando la variable latente $z_{0,i}$ correspondiente utilizando la expresión (21) que utiliza K transformaciones $\{f_i(z)\}_{i=1}^{k}$ y la variable latente z_0 de x.

[Fórmula 24]

$$x = f_K(f_{K-1}(...(f_1(z_0))...)) \qquad ...(21)$$

En otras palabras, la siguiente expresión es válida para i = 1, ..., K.

[Fórmula 25]

$$x_i = f_K(f_{K-1}(...(f_1(z_{0,i}))...)) \qquad ...(21)'$$

Tenga en cuenta que $z_1 = f_1(z_0)$, $z_2 = f_2(z_1)$, ..., $x = f_K(z_{K-1})$.

Además, se supone que la variable latente $\{z_{0,i}\}_{i=1}^{N}$ se genera a partir de la distribución $q_0(z_0)$ de probabilidad como, por ejemplo, una distribución Gaussiana isotrópica, con la que es fácil realizar un muestreo Monte Carlo. En este momento, la distribución q(x;θ) de probabilidad (donde x es una variable que indica datos de entrada) con la que cumple un conjunto de datos $\{x_i\}_{i=1}^{N}$ de entrada puede expresarse de la siguiente forma.

[Fórmula 26]

$$q(x;\theta) = q_0(z_0)\left|\det(\partial f_K(z_{K-1};\theta_K)/\partial z_{K-1})\right| \left|\det(\partial f_{K-1}(z_{K-2};\theta_{K-1})/\partial z_{K-2})\right| \ldots$$

$$\left|\det(\partial f_1(z_0;\theta_1)/\partial z_0)\right| \quad \ldots(22)$$

Aquí, $z_0 = f_1^{-1}(f_2^{-1}(\ldots(f_K^{-1}(x))\ldots))$. Además, $\{\theta_i\}_{i=1}^K$ es un parámetro correspondiente a una transformación $\{f_i(z)\}_{i=1}^K$, y $\theta = [\theta_1^T, \theta_2^T, \ldots, \theta_K^T]^T$.

Tenga en cuenta que la distribución $q_0(z_0)$ de probabilidad no se limita a una distribución con la que sea fácil realizar un muestreo de Monte Carlo, y sólo se requiere que sea una distribución con la que sea fácil realizar una estimación estricta de la densidad de probabilidad. Ejemplos de distribuciones con las que es fácil realizar una estimación estricta de la densidad de probabilidad pueden incluir una distribución p(x) de probabilidad que satisfaga las siguientes condiciones.

(Condición 1) Una función no negativa g(x) (≥0) en R^D existe para la distribución p(x) de probabilidad, y $p(x) = g(x)/\int g(x)dx$ para x arbitrario $\in R^D$.

(Condición 2) Es fácil calcular $\int g(x)dx$ para una función g(x).

Los ejemplos de la función que satisface la Condición 2 pueden incluir una distribución Gaussiana. Mientras tanto, los ejemplos de la función que no satisface la Condición 2 pueden incluir $g(x) = \exp(\sin(x) - x^2)$.

En el Flujo de Normalización, un parámetro θ de la distribución q(x;θ) de probabilidad se aprende utilizando un conjunto de los datos $\{x_i\}_{i=1}^N$ de entrada . Luego, la distribución p(x) de probabilidad original respecto a la generación de datos se aproxima mediante la distribución q(x;θ) de probabilidad, que utiliza el parámetro (denominado parámetro aprendido) θ obtenido a través del aprendizaje.

En el Flujo de Normalización, es posible utilizar varios tipos de transformación como la transformación $\{f_i(z)\}_{i=1}^K$. Por ejemplo, es posible utilizar Normalización por Lotes, una ReLU (Unidad Lineal Rectificada) con Fugas, o similares, descritos en la bibliografía 3 de Referencia no de patente. Además, también es posible utilizar la siguiente transformación lineal descrita en la bibliografía 4 de Referencia no de patente.

[Fórmula 27]

$$f(z) = LUz \quad \ldots(23)$$

Aquí, L, U $\in R^{D\times D}$ son, respectivamente, una matriz triangular inferior y una matriz triangular superior. Esta transformación está caracterizada por que, debido a que un valor absoluto $|\det(\partial f(z;\theta)/\partial z)|$ de un determinante Jacobiano puede calcularse utilizando un valor absoluto (es decir, $|\Pi_{i=1}^D L_{ii}U_{ii}|$) de un producto de elementos diagonales de L y U, es posible calcular, fácilmente, una densidad q(x;θ) de probabilidad de los datos x de entrada (en otras palabras, es posible reducir el coste del cálculo de la densidad q(x;θ) de probabilidad de los datos x de entrada) (véase la expresión (22)).

Normalización por Lotes BN: $x \rightarrow y$ ($x, y \in R^D$) se describirá, brevemente, a continuación. En la Normalización por Lotes BN, después de realizar el ajuste para que un promedio de elementos de las respectivas dimensiones del conjunto $\{x_i\}_{i=1}^N$ de los datos de entrada se convierta en 0 y una varianza se convierta en 1, se realizan transformación de escala y transformación de desplazamiento. Específicamente, $y_i = BN(x_i)$ se calcula utilizando la siguiente expresión.

[Fórmula 28]

$$m = \frac{1}{N}\sum_{i=1}^{N} x_i \quad \cdots(24a)$$

$$s^2 = \frac{1}{N}\sum_{i=1}^{N}(x_i - m)^2 \quad \cdots(24b)$$

$$\hat{x}_i = \frac{x_i - m}{\sqrt{s^2 + \varepsilon}} \quad \cdots(24c)$$

$$y_i = \gamma\hat{x}_i + \beta \quad \cdots(24d)$$

Aquí, γ y β son, respectivamente, un parámetro de transformación de escala, y un parámetro de transformación de desplazamiento, y ambos son parámetros que deben aprenderse. Además, ε es un número real no negativo, y, sólo es necesario establecer un número real positivo como ε en un caso donde se desee evitar la división por cero, y establecer cero como ε en un caso donde no sea necesario evitar la división por cero.

Tenga en cuenta que, para especificar, claramente, el parámetro γ de transformación de escala y el parámetro β de transformación de desplazamiento, también existe un caso donde BN(•) se expresa como $BN_{\gamma\beta}(\bullet)$.

Además, no todas las K transformaciones utilizadas en el Flujo de Normalización tienen que ser transformaciones del mismo tipo. Por lo tanto, por ejemplo, también es posible combinar algunos tipos de transformaciones de modo que una transformación $f_1(z)$ es una normalización por lotes, y una transformación $f_2(z)$ es una transformación lineal.

<<AdaBN (Normalización Adaptativa por Lotes)>>

La adaptación del dominio es una técnica para ajustar un modelo aprendido de modo que, en un caso donde la distribución de los datos de aprendizaje que se utilizará para el aprendizaje del modelo sea diferente de la distribución de los datos de prueba, que es un objetivo del procesamiento que utiliza el modelo aprendido, la precisión del procesamiento que utiliza el modelo aprendido no se degrada debido a una diferencia entre la distribución. Aquí, un conjunto de datos de aprendizaje y un conjunto de datos de prueba son dominios, y a veces se los denomina, respectivamente, dominio para aprendizaje y dominio para prueba.

Si bien existen varios métodos para la adaptación del dominio que pueden combinarse con una red neuronal profunda (DNN), aquí, se describirá la normalización adaptativa por lotes (véase la bibliografía 5 de Referencia no de patente). La normalización adaptativa por lotes es un método en el que se realiza el cálculo de un promedio y de una varianza y el ajuste de un promedio y de una varianza en la normalización por lotes (véanse las expresiones (24a) a (24d)) para cada dominio. En otras palabras, el cálculo utilizando las expresiones (24a) a (24c) se realiza para cada dato en el mismo dominio. En la prueba real, se calcula una cantidad de estadísticas (promedio y varianza) para un conjunto $\{x_i\}_{i=1}^N$ de los datos de entrada del dominio para prueba, y se genera un resultado y_i del procesamiento utilizando la expresión (24c) y la expresión (24d) que utilizan la cantidad de estadísticas. Tenga en cuenta que, en un caso donde la transformación sea una normalización adaptativa por lotes, existe un caso donde la transformación se expresa como AdaBN: $x \rightarrow y$ ($x, y \in R^D$).

<<AdaFlow>>

AdaFlow es un método en el que se introduce normalización adaptativa por lotes en el Flujo de Normalización. Específicamente, al menos, una transformación inversa $f_{i\_0}^{-1}(z)$ de las K transformaciones $\{f_i(z)\}_{i=1}^K$ que se utilizarán en el Flujo de Normalización es una normalización adaptativa por lotes. Tenga en cuenta que, en el cálculo de la normalización adaptativa por lotes, pueden omitirse la transformación de escala y la transformación de desplazamiento, es decir, el cálculo en la expresión (24d). En otras palabras, también es posible expresar que la transformación inversa $f_{i\_0}^{-1}(z)$ es una normalización adaptativa por lotes en la que $\gamma = 1$ y $\beta = 0$.

Si el aprendizaje se realiza utilizando AdaFlow, es posible generar una distribución de probabilidad de una pluralidad de dominios a partir de un modelo aprendido. Además, es posible realizar transformación de datos entre una pluralidad de dominios.

AdaFlow se describirá, específicamente, a continuación. AdaFlow es una red neuronal que transforma los datos x de entrada generados a partir de los datos de aprendizaje en variables latentes $z_0 (\sim q(z_0))$ que se consideran generadas de acuerdo con la distribución $q_0(z_0)$ de probabilidad. Aquí, se describirá un caso donde se utilizan cinco transformaciones $\{f_i(z)\}_{i=1}^5$. En otras palabras, z_0 puede obtenerse como $z_0 = f_1^{-1}(f_2^{-1}(f_3^{-1}(f_4^{-1}(f_5^{-1}(x)))))$.

Las cinco transformaciones $\{f_i(z)\}_{i=1}^5$ descritas anteriormente se definirán utilizando la siguiente expresión. Tenga en cuenta que, por conveniencia, en lugar de ser indicada una transformación f_i, se indica una transformación inversa f_i^{-1} de la transformación f_i (donde $z_4 = f_5^{-1}(x)$, $z_3 = f_4^{-1}(z_4)$, $z_2 = f_3^{-1}(z_3)$, $z_1 = f_2^{-1}(z_2)$, $z_0 = f_1^{-1}(z_1)$).

[Fórmula 29]

$$f_5^{-1}(x) = L_5 D_5 U_5 x \qquad ...(25a)$$

$$f_4^{-1}(z_4) = \mathrm{AdaBN}_{\gamma_4\beta_4}(z_4) \qquad ...(25b)$$

$$f_3^{-1}(z_3) = \mathrm{LeakyReLU}(z_3) = \max(z_3, \alpha_3 z_3) \qquad ...(25c)$$

$$f_2^{-1}(z_2) = L_2 D_2 U_2 z_2 \qquad ...(25d)$$

$$f_1^{-1}(z_1) = \mathrm{AdaBN}_{\gamma_1\beta_1}(z_1) \qquad ...(25e)$$

Aquí, L_2, $L_5 \in R^{D \times D}$ es una matriz triangular inferior cuyo elemento diagonal es 1, y todos los elementos $L_{2,ij}$, $L_{5,ij}$ ($i \geq j$) distintos de una parte triangular superior son parámetros objetivo de aprendizaje (es decir, un parámetro θ_2 o un parámetro θ_5). D_2, $D_5 \in R^{D \times D}$ es una matriz diagonal y los elementos diagonales $D_{2,ij}$, $D_{5,ij}$ ($i = j$) son parámetros objetivo de aprendizaje (es decir, un parámetro θ_2 o un parámetro θ_5). U_2, $U_5 \in R^{D \times D}$ es una matriz triangular superior cuyo elemento diagonal es 1, y todos los elementos $U_{2,ij}$, $U_{5,ij}$ ($i \leq j$) distintos de una parte triangular inferior son parámetros objetivo de aprendizaje (es decir, un parámetro θ_2 o un parámetro θ_5). Además, α_3 (≥ 0) es un parámetro de LeakyReLU, y puede establecerse como un hiper parámetro, o puede establecerse como un parámetro objetivo de aprendizaje (es decir, un parámetro θ_3) (en un caso donde α_3 se establece como objetivo de aprendizaje, ReLU se denomina ReLU Paramétrica (bibliografía 6 de Referencia no de patente)). Además, $AdaBN_{\gamma\_4\beta\_4}(\bullet)$ y $AdaBN_{\gamma\_1\beta\_1}(\bullet)$ son la normalización adaptativa por lotes descrita anteriormente, y γ_1, β_1, γ_4, β_4 son parámetros objetivo de aprendizaje (es decir, un parámetro θ_1 o un parámetro θ_4).

Además, los valores absolutos de los determinantes Jacobianos de las transformaciones $\{f_i(z)\}_{i=1}^{5}$ se calculan, respectivamente, utilizando la siguiente expresión (donde $x = f_5(z_4)$, $z_4 = f_4(z_3)$, $z_3 = f_3(z_2)$, $z_2 = f_2(z_1)$, $z_1 = f_1(z_0)$).

[Fórmula 30]

$$\left|\det(\partial f_5(z_4)/\partial z_4)\right| = 1/\left|\prod_{i=1}^{D} D_{5,ii}\right| \quad \cdots(26a)$$

$$\left|\det(\partial f_4(z_3)/\partial z_3)\right| = \sqrt{s_4'^2 + \varepsilon}/\gamma_4 \quad \cdots(26b)$$

$$\left|\det(\partial f_3(z_2)/\partial z_2)\right| = 1/\alpha_3^{\delta} \quad \cdots(26c)$$

$$\left|\det(\partial f_2(z_1)/\partial z_1)\right| = 1/\left|\prod_{i=1}^{D} D_{2,ii}\right| \quad \cdots(26d)$$

$$\left|\det(\partial f_1(z_0)/\partial z_0)\right| = \sqrt{s_1'^2 + \varepsilon}/\gamma_1 \quad \cdots(26e)$$

Aquí, s_4' es una desviación estándar de z_4 (correspondiente a los datos x de entrada generados a partir de los datos de aprendizaje), δ es el número de elementos por debajo de cero entre z_3 (correspondiente a los datos x de entrada generados a partir de los datos de aprendizaje), y s_1' es una desviación estándar de z_1 (correspondiente a los datos x de entrada generados a partir de los datos de aprendizaje). Tenga en cuenta que los valores absolutos $|\det(\partial f_4(z_3)/\partial z_3)|$, $|\det(\partial f_1(z_0)/\partial z_0)|$ de los determinantes Jacobianos para las transformaciones f_4 y f_1 se expresan utilizando valores absolutos de los determinantes Jacobianos tras la deducción en lugar de tras el aprendizaje (es decir, tras el procesamiento utilizando el modelo aprendido).

Además, como se describió anteriormente, se supone que la distribución $q_0(z_0)$ de probabilidad es una distribución de probabilidad con la que es fácil realizar una estimación estricta de la densidad de probabilidad. Por ejemplo, si una distribución Gaussiana N(0, I), en la que un promedio es 0 y una varianza es una matriz identidad I, se establece como la distribución $q_0(z_0)$ de probabilidad, la distribución $q_0(z_0)$ de probabilidad puede expresarse utilizando la siguiente expresión.

[Fórmula 31]

$$q_0(z_0) = -(2\pi)^{-D/2} \exp(-\| z_0 \|_2^2) \quad \cdots(27)$$

Por lo tanto, puede entenderse que, al establecer la distribución de probabilidad de los datos x de entrada generados a partir de los datos de aprendizaje como $q_1(x;\theta)$ y utilizando la expresión (5), es posible calcular una densidad $q_1(X_i;\theta)$ de probabilidad de los datos x_i de entrada a partir de la densidad $q_0(z_{0,i})$ de probabilidad de la variable latente $z_{0,i}$.

Posteriormente, se describirá un método de aprendizaje del parámetro θ. De manera similar al aprendizaje convencional de una red neuronal, es posible realizar el aprendizaje utilizando, por ejemplo, un método de descenso de gradiente, Momentum SGD (Descenso de Gradiente Estocástico), ADAM (Estimación Adaptativa de Momentos) o una combinación de los mismos, utilizando una función $L(\theta)$ de pérdida. En un caso donde se utilice un Flujo de Normalización, a menudo se utiliza como función $L(\theta)$ de pérdida un promedio de probabilidades logarítmicas negativas definidas mediante la siguiente expresión.

[Fórmula 32]

$$L(\theta) = -1/N \sum_{i=1}^{N} \log q(x_i;\theta) \quad \cdots(28)$$

Tenga en cuenta que es posible utilizar un método de aprendizaje en mini lotes que se realiza en unidades de un conjunto de datos de aprendizaje denominado mini lote en el aprendizaje descrito anteriormente. Aquí, un mini lote se refiere a una pluralidad de datos de aprendizaje seleccionados, aleatoriamente, de todos los datos de aprendizaje. Se calcula un valor de la función L(θ) de pérdida para cada mini lote.

[Séptima realización]

La séptima realización descrita a continuación se refiere a un ejemplo que no está abarcado por las reivindicaciones, pero es útil para entender la presente invención.

El aparato 1100 de aprendizaje de distribuciones de probabilidad aprende un parámetro θ del modelo de una red neuronal que transforma los datos x de entrada generados a partir de los datos de P tipos de dominios D_j (j = 1, ..., P) (en lo sucesivo, denominados datos de dominio) en una variable latente z_0(~ $q(z_0)$) que se considera generada de acuerdo con la distribución $q_0(z_0)$ de probabilidad. Es posible obtener la distribución q(x;θ) de probabilidad de los datos x de entrada a partir de la distribución $q_0(z_0)$ de probabilidad utilizando este parámetro θ del modelo (véase la expresión (22)). Aquí se supone que el dominio D_j incluye N_j datos de dominio. Por lo tanto, si el número de datos de dominio incluidos en una unión de P tipos de dominios es N, $N = \Sigma_j N_j$.

El aparato 1100 de aprendizaje de distribuciones de probabilidad se describirá a continuación con referencia a la Fig. 17 y a la Fig. 18. La Fig. 17 es un diagrama de bloques que ilustra una configuración del aparato 1100 de aprendizaje de distribuciones de probabilidad. La Fig. 18 es un diagrama de flujo que ilustra el funcionamiento del aparato 1100 de aprendizaje de distribuciones de probabilidad. Como se ilustra en la Fig. 17, el aparato 1100 de aprendizaje de distribuciones de probabilidad incluye una unidad 1110 generadora de datos de entrada, una unidad 1120 de estimación de la variable latente, una unidad 1130 de cálculo de la función de pérdida, una unidad 1140 de actualización del parámetro, una unidad 1150 de determinación de la condición de convergencia, y una unidad 1190 de grabación. La unidad 1190 de grabación es un componente que graba la información necesaria para el procesamiento del aparato 1100 de aprendizaje de distribuciones de probabilidad según corresponda. La unidad 1190 de grabación graba, por ejemplo, un parámetro θ de la distribución q(x;θ) de probabilidad de los datos x de entrada. Como valor inicial del parámetro θ, se graba, por ejemplo, un valor generado utilizando un número aleatorio.

El aparato 1100 de aprendizaje de distribuciones de probabilidad está conectado a una unidad 1910 de grabación de datos del dominio de aprendizaje. En la unidad 1910 de grabación de datos del dominio de aprendizaje, una tupla (s_i, j) de datos s_i (i = 1,..., N) de dominio preparada de antemano y un identificador j de dominio para identificar un dominio en el que los datos s_i de dominio se incluyen y se graban como datos de aprendizaje. Tenga en cuenta que (s_i, j) también se denominará datos de dominio para el aprendizaje.

El funcionamiento del aparato 1100 de aprendizaje de distribuciones de probabilidad se describirá de acuerdo con la Fig. 18. En S1110, la unidad 1110 generadora de datos de entrada recibe datos de dominio para el aprendizaje (s_i, j) (i = 1,..., N, s_i son datos de dominio, y j es un identificador de dominio de un dominio en el que s_i está incluido) que se ingresa como entrada, genera los datos x_i (i = 1, ..., N) de entrada a partir de los datos s_i de dominio y genera una tupla de los datos x_i de entrada y del identificador j de dominio. Cualquier método con el que puedan restaurarse datos de dominio a partir de los datos de entrada puede utilizarse como método para generar datos de entrada a partir de datos de dominio. Por ejemplo, en un caso donde los datos de dominio sean una imagen o un sonido, es preferible utilizar cantidades de características predeterminadas con las que pueda restaurarse una imagen o un sonido que sea una fuente de extracción, como los datos x_i de entrada generados a partir de los datos s_i de dominio.

En S1120, la unidad 1120 de estimación de la variable latente recibe la tupla de los datos x_i (i = 1, ..., N) de entrada y el identificador j de dominio del dominio en el que están incluidos los datos s_i de dominio, generados en S1110, como entrada, estima una variable latente $z_{0,i}$ (i = 1, ..., N) correspondiente a los datos x_i de entrada a partir de los datos x_i (i = 1, ..., N) de entrada utilizando el parámetro θ de la distribución q(x;θ) de probabilidad y emite la variable latente $z_{0,i}$ (i = 1, ..., N). Tenga en cuenta que el parámetro θ utilizado aquí es un valor que se está aprendiendo.

Aquí, se supone que una variable x de la distribución q(x;θ) de probabilidad es una variable que indica datos de entrada generados a partir de los datos de dominio de P tipos de dominios D_j, y la variable x se expresa como $x = f_K(f_K^{-1}(...(f_1(z_0))...))$ utilizando transformaciones f_i (i = 1, ..., K, K es un número entero de 1 o mayor, y existen transformaciones inversas f_i^{-1} para las transformaciones f_i) y la variable latente z_0.

Por tanto, la variable latente $z_{0,i}$ correspondiente a los datos x_i de entrada se proporciona utilizando la siguiente expresión.

[Fórmula 33]

$$z_{0,i} = f_1^{-1}(f_2^{-1}(...(f_K^{-1}(x_i))...)) \qquad ...(29)$$

Además, se supone que la variable latente $z_{0,i}$ (i = 1, ..., N) se genera de acuerdo con la distribución $q_0(z_0)$ de probabilidad de la variable latente z_0. Sin embargo, la distribución $q_0(z_0)$ de probabilidad tiene características que hacen que sea fácil realizar una estimación estricta de la densidad de probabilidad.

Por lo tanto, la distribución $q(x;\theta)$ de probabilidad puede expresarse utilizando la siguiente expresión que utiliza la distribución $q_0(z_0)$ de probabilidad.

[Fórmula 34]

$$q(x;\theta) = q_0(z_0)\left|\det(\partial f_K(z_{K-1};\theta_K)/\partial z_{K-1})\right| \left|\det(\partial f_{K-1}(z_{K-2};\theta_{K-1})/\partial z_{K-2})\right| ...$$

$$\left|\det(\partial f_1(z_0;\theta_1)/\partial z_0)\right| \qquad (22)$$

Aquí, θ_i es un parámetro correspondiente a la transformación f_i, y $\theta = [\theta_1^T, \theta_2^T, ..., \theta_K^T]^T$.

La unidad 1120 de estimación de la variable latente puede realizarse utilizando una red neuronal NN que calcula la variable latente z_0 a partir de los datos x de entrada. Tenga en cuenta que una expresión de cálculo de la variable latente z_0 es como sigue.

[Fórmula 35]

$$z_0 = f_1^{-1}(f_2^{-1}(...(f_K^{-1}(x))...)) \qquad ...(29)'$$

En otras palabras, la red neuronal NN recibe los datos x de entrada como entrada, procede con el cálculo secuencialmente a partir de las transformaciones inversas f_K^{-1}, f_{K-1}^{-1}, ..., f_2^{-1}, f_1^{-1}, y finalmente, emite la variable latente z_0.

Al menos, una transformación inversa $f_{i\_0}^{-1}$ de las transformaciones f_i (i = 1, ..., K) descritas anteriormente (donde i_0 es un número entero que satisface $1 \leq i_0 \leq K$) se realiza normalización adaptativa por lotes.

Además, una transformación inversa $f_{i\_1}^{-1}$ de unas transformaciones f_{i_1} (donde i_1 es un número entero que satisface $1 \leq i_1 \leq K$) incluida en las transformaciones f_i (i = 1, ..., K) puede ser una transformación lineal, y una matriz correspondiente a la transformación lineal puede expresarse como un producto de una matriz triangular inferior y una matriz triangular superior, o un producto de una matriz triangular inferior, una matriz diagonal y una matriz triangular superior.

Como ejemplo específico de las transformaciones f_i (i = 1, ..., K), es posible utilizar cinco transformaciones cuyas transformaciones inversas pueden expresarse utilizando la expresión (25a) a la expresión (25e) suponiendo que, por ejemplo, K = 5.

Además, debido a que la unidad 1120 de estimación de la variable latente calcula un promedio y una varianza de los datos $z_{i\_0,i}(= f_{i\_0\text{-}1}^{-1}(f_{i\_0\text{-}2}^{-1}(...(f_K^{-1}(x_i))...)))$ de entrada que se introducirán en la transformación inversa $f_{i\_0}^{-1}$, que se calcula a partir de los datos x_i (i = 1, ..., N) de entrada, para cada dominio, es decir, un promedio y una varianza utilizados para el cálculo de la transformación inversa $f_{i\_0}^{-1}$ que es una normalización adaptativa por lotes, $(z_{i\_0,i,j})$ (donde j es un identificador de dominio de los datos s_i de dominio que son una fuente de generación de los datos x_i de entrada) se graban en la unidad 1190 de grabación. En lo sucesivo, un promedio y una varianza del dominio D_j de los datos de entrada que se introducirán en la transformación inversa $f_{i\_0}^{-1}$ se expresan, respectivamente, como $m_{i\_0,j}$ y $s_{i\_0,j}^2$. Tenga en cuenta que, como se describirá más adelante, el promedio $m_{i\_0,j}$ y la varianza $s_{i\_0,j}^2$ se calculan mediante la unidad 1150 de determinación de la condición de convergencia.

En S1130, la unidad 1130 de cálculo de la función de pérdida recibe la variable latente $z_{0,1}$ (i = 1, ..., N) estimada en S1120 como entrada, calcula un valor de la función $L(\theta)$ de pérdida que se utilizará para la optimización del parámetro θ de la distribución $q(x;\theta)$ de probabilidad a partir del latente variable $z_{0,1}$ (i = 1, ..., N) y emite el valor de la función $L(\theta)$ de pérdida. La función $L(\theta)$ de pérdida puede, por ejemplo, definirse como un promedio de probabilidades logarítmicas negativas como en la expresión (28). En este momento, si bien es necesario calcular la densidad $q(x_i;\theta)$ de probabilidad de los datos x_i (i = 1, ..., N) de entrada, la densidad $q(x_i;\theta)$ de probabilidad de los datos x_i de entrada puede calcularse utilizando la densidad $q_0(z_{0,i})$ de probabilidad de la variable latente $z_{0,i}$ correspondiente a los datos x_i de entrada. Por ejemplo, en un caso donde la distribución $q_0(z_0)$ de probabilidad es una distribución Gaussiana N(0, I), la densidad $q_0(z_{0,i})$ de probabilidad de la variable latente $z_{0,i}$ puede calcularse utilizando la siguiente expresión.

[Fórmula 36]

$$q_0(z_{0,i}) = -(2\pi)^{-D/2} \exp(-\| z_{0,i} \|_2^2) \quad \cdots(27)'$$

Por lo que es posible calcular la densidad $q(x_i;\theta)$ de probabilidad de los datos x_i de entrada a partir de la densidad $q_0(z_{0,i})$ de probabilidad descrita anteriormente de la variable latente $z_{0,i}$ calculada utilizando la expresión (22).

En S1140, la unidad 1140 de actualización del parámetro recibe el valor de la función $L(\theta)$ de pérdida calculado en S1130 como entrada, y actualiza el parámetro θ de la distribución $q(x;\theta)$ de probabilidad para optimizar (minimizar) el valor de la función $L(\theta)$ de pérdida, y emite el parámetro θ. Por ejemplo, se utiliza, preferentemente, un método de descenso de gradiente en la actualización del parámetro θ.

En S1150, la unidad 1150 de determinación de la condición de convergencia determina las condiciones de convergencia establecidas de antemano como condiciones de terminación de la actualización del parámetro y, en un caso donde se cumplan las condiciones de convergencia, emite la distribución $q(x;\theta)$ de probabilidad utilizando el parámetro θ (este parámetro se denominará parámetro aprendido) actualizado en S1140. En este momento, la unidad 1150 de determinación de la condición de convergencia calcula el promedio $m_{i\_0,j}$ y la varianza $S_{i\_0,j}^2$ (j = 1, ..., P) del dominio D_{i_0} de los datos de entrada que se introducirán en la transformación inversa $f_{i\_0}^{-1}$ utilizando ($z_{i\_0,i,}$ j) (i = 1, ..., N) grabado en S1120, y emite el promedio $m_{i\_0,j}$ y la varianza $S_{i\_0,j}^2$ (j = 1, ..., P). Mientras tanto, en un caso donde no se cumplan las condiciones de convergencia, la unidad 1150 de determinación de la condición de convergencia repite el procesamiento desde S1110 a S1140. Como condiciones de convergencia, por ejemplo, es posible emplear una condición relativa a si el número de veces de ejecución del procesamiento desde S1110 a S1440 alcanza un número predeterminado de veces. Tenga en cuenta que también es posible emitir el parámetro θ actualizado en S1140 y las transformaciones inversas $f_K^{-1}(x;\theta_K)$, $f_{K-1}^{-1}(z_{K-1};\theta_{K-1})$, ..., $f_2^{-1}(z_2;\theta_2)$, $f_1^{-1}(z_1;\theta_1)$.

En lo sucesivo, el promedio $m_{i\_0,j}$ y la varianza $s_{i\_0,j}^2$ del dominio D_j de los datos de entrada que se introducirán en la transformación inversa $f_{i\_0}^{-1}$ se denominarán cantidad de estadísticas calculadas a partir de los datos de dominio del dominio D_j.

Según la realización de la presente invención, es posible aprender la distribución $q(x;\theta)$ de probabilidad de los datos x de entrada generados a partir de datos de dominio de P tipos de dominios D_j. Al realizar el aprendizaje utilizando AdaFlow en función de un Flujo de Normalización, es posible realizar el aprendizaje de forma estable sin datos de pares.

[Octava realización]

La octava realización descrita a continuación se refiere a un ejemplo que no está abarcado por las reivindicaciones, pero es útil para entender la presente invención.

El aparato 1200 de transformación de datos transforma datos de dominio de un dominio D_{j_0} en datos de dominio de un dominio D_{j_1} (donde j_0, j_1 son números enteros que satisfacen $1 \leq j_0, j_1 \leq P$ y $j_0 \neq j_1$) utilizando el parámetro θ del modelo aprendido, aprendido en el aparato 1100 de aprendizaje de distribuciones de probabilidad, y el promedio $m_{i\_0,j}$ y la varianza $s_{i\_0,j}^2$ (j = 1, ..., P) del dominio D_j de los datos de entrada que se introducirán en la transformación inversa $f_{i\_0}^{-1}$. En lo sucesivo, el dominio D_{j_0} se denominará dominio de origen de la transformación, y el dominio D_{j_1} se denominará dominio de destino de la transformación.

El aparato 1200 de transformación de datos se describirá a continuación con referencia a la Fig. 19 y la Fig. 20. La Fig. 19 es un diagrama de bloques que ilustra una configuración del aparato 1200 de transformación de datos. La Fig. 20 es un diagrama de flujo que ilustra el funcionamiento del aparato 1200 de transformación de datos. Como se ilustra en la Fig. 19, el aparato 1200 de transformación de datos incluye la unidad 1110 generadora de datos de entrada, una unidad 1220 de cálculo de la variable latente, una unidad 1230 de cálculo de datos de salida, una unidad 1240 generadora de datos de dominio, y la unidad 1190 de grabación. La unidad 1190 de grabación es un componente que graba la información necesaria para el procesamiento del aparato 1200 de transformación de datos según corresponda. La unidad 1190 de grabación, por ejemplo, graba el parámetro θ (es decir, el parámetro θ aprendido) de la distribución $q(x;\theta)$ de probabilidad aprendida utilizando el aparato 1100 de aprendizaje de distribuciones de probabilidad. Además, la unidad 1190 de grabación, por ejemplo, graba el promedio $m_{i\_0,j}$ y la varianza $s_{i\_0,j}^2$ (j = 1, ..., P) del dominio D_j de los datos de entrada que se introducirán en la transformación inversa $f_{i\_0}^{-1}$.

El funcionamiento del aparato 1200 de transformación de datos se describirá de acuerdo con la Fig. 20. En S1110, la unidad 1110 generadora de datos de entrada recibe los datos s de dominio del dominio D_{j_0} de origen de la transformación que son entrada y un identificador j_0 de dominio del dominio D_{j_0} de origen de la transformación (que se denominará identificador del dominio de origen de la transformación) como entrada, genera los datos x de entrada a partir de los datos s de dominio y emite los datos x de entrada.

En S1220, la unidad 1220 de cálculo de la variable latente recibe los datos x de entrada generados en S1110 y el identificador j_0 del dominio de origen de la transformación como entradas, calcula la variable latente z_0 correspondiente a los datos x de entrada a partir de los datos x de entrada utilizando el parámetro θ aprendido y el promedio m_{i_0,j_0} y

la varianza S_{i_0,j_0}^2 del dominio D_{j_0} de los datos de entrada que se introducirán en la transformación inversa $f_{i\_0}^{-1}$ y emite la variable latente z_0. La variable latente z_0 correspondiente a los datos x de entrada se calcula utilizando la siguiente expresión que utiliza las transformaciones f_i (i = 1,..., K) usadas en el aparato 1100 de aprendizaje de distribuciones de probabilidad.

[Fórmula 37]

$$z_0 = f_1^{-1}(f_2^{-1}(...(f_K^{-1}(x))...)) \qquad ...(29)'$$

En este momento, se utilizan el parámetro θ aprendido y el promedio m_{i_0,j_0} y la varianza s_{i_0,j_0}^2 del dominio D_{j_0} de los datos de entrada que se introducirán en la transformación inversa $f_{i\_0}^{-1}$. La unidad 1220 de cálculo de la variable latente es diferente de la unidad 1120 de estimación de la variable latente del aparato 1100 de aprendizaje de distribuciones de probabilidad en este punto.

En S1230, la unidad 1230 de cálculo de datos de salida recibe el identificador j_1 del dominio de destino de la transformación que es un identificador del dominio de destino de la transformación y la variable latente z_0 calculada en S1220 como entradas, calcula los datos x' de salida correspondientes a la variable latente z_0 a partir de la variable latente z_0 utilizando el parámetro θ aprendido y el promedio m_{i_0,j_1} y la varianza s_{i_0,j_1}^2 del dominio D_{j_1} de los datos de entrada que se introducirán en la transformación inversa $f_{i\_0}^{-1}$, y emite los datos x' de salida. Los datos x' de salida correspondientes a la variable latente z_0 se calculan utilizando la siguiente expresión que utiliza las transformaciones f_i (i = 1,..., K) usadas en el aparato 1100 de aprendizaje de distribuciones de probabilidad.

[Fórmula 38]

$$x' = f_K(f_{K-1}(...(f_1(z_0))...)) \qquad ...(21)''$$

En este momento, se utilizan el parámetro θ aprendido y el promedio m_{i_0,j_1} y la varianza s_{i_0,j_1}^2 del dominio D_{j_1} de los datos de entrada que se introducirán en la transformación inversa $f_{i\_0}^{-1}$. Tenga en cuenta que sólo es necesario utilizar una red neuronal obtenida cambiando la salida de la red neuronal NN que realiza la unidad 1220 de cálculo de la variable latente a la entrada, y cambiando la entrada de la red neuronal NN a la salida, como la red neuronal que realiza la unidad 1230 de cálculo de datos de salida.

Un aspecto del procesamiento por parte de la unidad 1220 de cálculo de la variable latente y de la unidad 1230 de cálculo de datos de salida se ilustra en la Fig. 21. La Fig. 21 ilustra un aspecto en el que los datos de entrada se transforman en una variable latente, y la variable latente se transforma en datos de salida utilizando cinco transformaciones $\{f_i(z)\}_{i=1}^5$ utilizadas en la descripción de los <Antecedentes técnicos Aquí, $f_1^{-1(j\_0)}$, y $f_1^{(j\_1)}$ son funciones que utilizan el promedio m_{i_0,j_0} y la varianza S_{i_0,j_0}^2 del dominio D_{j_0}, y $f_4^{-1(j\_0)}$ y $f_4^{(j\_1)}$ son funciones que utilizan el promedio m_{i_0,j_1} y la varianza s_{i_0,j_1}^2 del dominio D_{j_1}.

En S1240, la unidad 1240 generadora de datos de dominio recibe los datos x' de salida calculados en S1230 como entrada, genera los datos s' de dominio transformados, que son datos del dominio D_{j_1} de destino de la transformación a partir de los datos x' de salida y emite los datos s' de dominio transformados.

Tenga en cuenta que un componente que incluya la unidad 1220 de cálculo de la variable latente y la unidad 1230 de cálculo de datos de salida se denominará unidad 1205 de transformación de datos. En otras palabras, la unidad 1205 de transformación de datos es un componente que transforma los datos de entrada generados a partir de los datos de dominio, del dominio de origen de la transformación, en datos de salida que se convierten en una fuente de generación de los datos de dominio del dominio de destino de la transformación.

Según la realización de la presente invención, es posible transformar, mutuamente, datos entre dominios.

[Novena realización]

La novena realización descrita a continuación se refiere a un ejemplo que no está abarcado por las reivindicaciones pero es útil para entender la presente invención.

En la séptima realización y en la octava realización, se ha proporcionado una descripción suponiendo que P es un número entero de 1 o mayor, y se utilizan datos de dominio de P tipos de dominios. Aquí, se describirá un aparato 1300 de transformación de datos en un caso donde P = 2.

El aparato 1300 de transformación de datos se describirá a continuación con referencia a las Fig. 22 y a la Fig. 23. La Fig. 22 es un diagrama de bloques que ilustra una configuración del aparato 1300 de transformación de datos. La Fig. 23 es un diagrama de flujo que ilustra el funcionamiento del aparato 1300 de transformación de datos. Como se ilustra en la Fig. 22, el aparato 1300 de transformación de datos incluye una unidad 1310 generadora de datos de entrada, una unidad 1320 de cálculo de la variable latente, una unidad 1330 de cálculo de datos de salida, una unidad 1340 generadora de datos de dominio, y una unidad 1190 de grabación. La unidad 1190 de grabación es un componente que graba la información necesaria para el procesamiento del aparato 1300 de transformación de datos según corresponda. La unidad 1190 de grabación, por ejemplo, graba el parámetro θ (es decir, el parámetro θ aprendido) de la distribución

$q(x;\theta)$ de probabilidad aprendida utilizando el aparato 1100 de aprendizaje de distribuciones de probabilidad.

En lo sucesivo, el dominio de origen de la transformación se denominará primer dominio, el dominio de destino de la transformación se denominará segundo dominio, y se expresan, respectivamente, como D_1 y D_2. Además, en la unidad 1320 de cálculo de la variable latente, se establecen el promedio $m_{i\_0,1}$ y la varianza $s_{i\_0,1}^2$ del primer dominio D_1 de los datos de entrada que se introducirán en la transformación inversa $f_{i\_0}^{-1}$. De manera similar, en la unidad 1330 de cálculo de datos de salida, se establecen el promedio $m_{i\_0,2}$ y la varianza $s_{i\_0,2}^2$ del segundo dominio D_2 de los datos de entrada que se introducirán en la transformación inversa $f_{i\_0}^{-1}$.

El funcionamiento del aparato 1300 de transformación de datos se describirá de acuerdo con la Fig. 23. En S1310, la unidad 1310 generadora de datos de entrada recibe los datos s de dominio del primer dominio como entrada, genera los datos x de entrada a partir de los datos s de dominio y emite los datos x de entrada. En lo sucesivo, estos datos de entrada también se denominarán datos de entrada correspondientes a los datos de dominio del primer dominio.

En S1320, la unidad 1320 de cálculo de la variable latente recibe los datos x de entrada, generados en S1310 como entrada, calcula la variable latente z_0 correspondiente a los datos x de entrada a partir de los datos x de entrada utilizando el parámetro θ aprendido y emite la variable latente z_0.

En S1330, la unidad 1330 de cálculo de datos de salida recibe la variable latente z_0 calculada en S1320 como entrada, calcula los datos x' de salida correspondientes a la variable latente z_0 a partir de la variable latente z_0 utilizando el parámetro θ aprendido y emite los datos x' de salida.

En S1340, la unidad 1340 generadora de datos de dominio recibe los datos x' de salida calculados en S1330 como entrada, genera los datos s' de dominio del segundo dominio a partir de los datos x' de salida y emite los datos s' de dominio. En lo sucesivo, estos datos de salida también se denominarán datos de salida correspondientes a los datos de dominio del segundo dominio.

Tenga en cuenta que un componente que incluya la unidad 1320 de cálculo de la variable latente y la unidad 1330 de cálculo de datos de salida se denominará unidad 1305 de transformación de datos. En otras palabras, la unidad 1305 de transformación de datos es un componente que transforma los datos de entrada correspondientes a los datos de dominio del primer dominio en los datos de salida correspondientes a los datos de dominio del segundo dominio.

Como puede entenderse a partir de la descripción anterior, puede decirse que la unidad 1320 de cálculo de la variable latente realiza el cálculo utilizando una función (en lo sucesivo, denominada primera función) que transforma la variable latente en datos de entrada y que tiene una función inversa, y la unidad 1330 de cálculo de datos de salida realiza el cálculo utilizando una función (en lo sucesivo, denominada segunda función) que transforma la variable latente en datos de salida y que tiene una función inversa. Entonces, la primera función y la segunda función se derivan de una función predeterminada que transforma la variable latente z_0 en la variable x. Esta función predeterminada es una función obtenida utilizando una unión del primer dominio y del segundo dominio. Más específicamente, la función predeterminada es una función obtenida como función expresada como $x = f_K(f_{K-1}(...(f_1(z_0))...))$ utilizando transformaciones $f_i(i = 1, ..., K, K$ es un número entero de 1 o mayor, existen transformaciones inversas f_i^{-1} para las transformaciones f_i, y, al menos, una transformación inversa $f_{i\_0}^{-1}$ $(1 \leq i_0 \leq K)$ de las transformaciones $f_i(i = 1, ..., K)$ es una normalización adaptativa por lotes) a través del aprendizaje que utiliza datos de dominio, que son un elemento de la unión del primer dominio y del segundo dominio como datos de aprendizaje. Además, la primera función se deriva de la función predeterminada descrita anteriormente utilizando una cantidad de estadísticas (específicamente, el promedio $m_{i\_0,1}$ y la varianza $s_{i\_0,1}^2$ del primer dominio D_1 de los datos de entrada que se introducirán en la transformación inversa $f_{i\_0}^{-1}$) calculada a partir de los datos de dominio del primer dominio incluido en la unión, y la segunda función se deriva de la función predeterminada utilizando una cantidad de estadísticas (específicamente, el promedio $m_{i\_0,2}$ y la varianza $s_{i\_0,2}^2$ del segundo dominio D_2 de los datos de entrada que se introducirán en la transformación inversa $f_{i\_0}^{-1}$) calculada a partir de los datos de dominio del segundo dominio incluido en la unión.

[Ejemplo de aplicación: aplicación al problema de detección supervisada de anomalías]

Es posible aplicar el aparato 1300 de transformación de datos al problema de detección supervisada de anomalías. Aquí, la detección supervisada de anomalías es un marco que aprende un modelo de detección de anomalías a partir de una gran cantidad de datos normales y de una pequeña cantidad de datos de anomalías y realiza la detección de anomalías utilizando este modelo de detección de anomalías.

Si bien es posible aprender un modelo con mayor precisión a medida que la cantidad de datos de anomalías es mayor, es difícil recopilar una gran cantidad de datos de anomalías. Por lo tanto, se prepara una pluralidad de dominios que se desea establecer como objetivos de detección de anomalías, y se recopila la mayor cantidad de datos posible, y el parámetro θ del modelo se aprende a partir de los datos utilizando el aparato 1100 de aprendizaje de distribuciones de probabilidad. Luego, los datos de anomalías del primer dominio se transforman en los datos de anomalías del segundo dominio, diferente del primer dominio, utilizando el aparato 1300 de transformación de datos. Mediante esta técnica, es posible crear, artificialmente, datos de anomalías que son difíciles de recopilar, de modo que es posible aumentar el número de datos de anomalías, lo que permite aprender un modelo de detección de anomalías con mayor precisión.

Por ejemplo, el parámetro θ se aprende utilizando el aparato 1100 de aprendizaje de distribuciones de probabilidad mientras se establece un conjunto de sonidos emitidos desde uno o más equipos diferentes del equipo objetivo de detección de anomalías como el primer dominio y se establece un conjunto de sonidos del equipo objetivo de detección de anomalías como el segundo dominio. Luego, el sonido del equipo objetivo de detección de anomalías se genera como los datos de dominio del segundo dominio a partir del sonido anómalo emitido desde uno o más equipos diferentes del equipo objetivo de detección de anomalías, que son datos de dominio del primer dominio, utilizando el aparato 1300 de transformación de datos. Se espera que el sonido generado del equipo objetivo de detección de anomalías sea un sonido anómalo. Además, el modelo de detección de anomalías del segundo dominio se aprende utilizando el sonido generado del equipo objetivo de detección de anomalías como datos de aprendizaje.

Tenga en cuenta que se considera que la mayor parte del sonido emitido es sonido normal. Debido a que es preferible utilizar la mayor cantidad de datos posible tras el aprendizaje, sería mejor utilizar el sonido del primer dominio para el aprendizaje, independientemente de si el sonido es un sonido normal o un sonido anómalo.

Según la realización de la presente invención, es posible transformar, mutuamente, datos entre dominios. Como resultado, por ejemplo, es posible generar, eficientemente, datos de anomalías que se utilizarán para el aprendizaje de un modelo de detección de anomalías.

<Resultado experimental>

El aprendizaje se realiza mientras se establecen un conjunto de "fotografías de paisajes" y un conjunto de "paisajes" como dominios, y se emplea Glow, descrito en la bibliografía 8 de Referencia no de patente, como una arquitectura de Flujo de Normalización, y utilizando una red neuronal AdaFlow, en la que se reemplaza la Normalización de Activación de Glow con AdaBN. Se recopilan 400 datos de imágenes de las fotografías de paisajes y 400 datos de imágenes de los paisajes, y se hace que la red neuronal AdaFlow realice el aprendizaje utilizando estos datos de imágenes, y se realiza la transformación de datos. Específicamente, las "fotografías de paisajes" se transforman en "paisajes" o los "paisajes" se transforman en "fotografías de paisajes".

(Bibliografía 8 de Referencia no de patente: Diederik P. Kingma, Prafulla Dhariwal, et al., "Glow: Generative Flow with Invertible 1x1 Convolutions", arXiv: 1807.03039, https://arxiv.org/abs/1807.03039)

Podría confirmarse que a través de este experimento se generan datos de imágenes con alta calidad, es decir, transformación del dominio sin datos de pares utilizando AdaFlow.

<Apéndice>

La Fig. 24 es una vista que ilustra un ejemplo de una configuración funcional de un ordenador que realiza cada aparato descrito anteriormente. El procesamiento en cada aparato descrito anteriormente puede ser realizado por una unidad 2020 de grabación, que hace que se lea un programa para hacer que el ordenador funcione como cada aparato descrito anteriormente, y una unidad 2010 de control, una unidad 2030 de entrada, una unidad 2040 de salida, o similar, que lo hacen funcionar.

El aparato de la presente invención incluye una unidad de entrada a la que puede conectarse un teclado, o similar, una unidad de salida a la que puede conectarse una pantalla de cristal líquido, o similar, una unidad de comunicación a la que puede conectarse un aparato de comunicación (por ejemplo, un cable de comunicación) que puede realizar una comunicación con una entidad de hardware externa, una CPU (Unidad Central de Procesamiento, que puede incluir una memoria caché, un registro, o similar), una RAM y una ROM que son memorias, un aparato de almacenamiento externo, que es un disco duro, y un bus que conecta esta unidad de entrada, unidad de salida, unidad de comunicación, CPU, RAM, ROM, y el aparato de almacenamiento externo para poder intercambiar datos entre ellos, por ejemplo, como una entidad de hardware única. Además, según sea necesario, también es posible proporcionar un aparato (unidad), o similar, que pueda realizar una lectura en, y una escritura desde, un medio de grabación como un CD-ROM, en la entidad de hardware. Ejemplos de entidades físicas que incluyen dichos recursos de hardware pueden incluir un ordenador de propósito general.

En el aparato de almacenamiento externo de la entidad de hardware, se almacena un programa que es necesario para realizar las funciones y datos descritos anteriormente, o similares, que son necesarios para el procesamiento de este programa (el aparato no se limita al aparato de almacenamiento externo, y un programa puede almacenarse en, por ejemplo, una ROM que es un aparato de almacenamiento de sólo lectura). Además, los datos, o similares, obtenidos a través del procesamiento de estos programas se almacenan en una RAM, en un aparato de almacenamiento externo, o similares, según corresponda.

En la entidad de hardware, cada programa almacenado en el aparato de almacenamiento externo (o la ROM, o similar), y los datos necesarios para el procesamiento de cada programa se leen en una memoria según sea necesario, y la ejecución interpretativa y el procesamiento se realizan en la CPU según corresponda. Como resultado, la CPU realiza funciones predeterminadas (componentes respectivos indicados anteriormente como partes, unidades, o similares).

La presente invención se define por las reivindicaciones adjuntas y no se limita a las realizaciones descritas anteriormente. Además, el procesamiento descrito en la realización descrita anteriormente puede ejecutarse en paralelo, o individualmente, de acuerdo con el rendimiento de procesamiento de los aparatos que ejecutan el procesamiento o según sea necesario, además de ejecutarse en orden cronológico de acuerdo con el orden de descripción.

Como se describió anteriormente, en un caso donde las funciones de procesamiento en la entidad de hardware (el aparato de la presente invención) descrita en las realizaciones descritas anteriormente se realizan con un ordenador, el contenido de procesamiento de las funciones que deben proporcionarse en la entidad de hardware se describe con un programa. Luego, al ser ejecutado este programa por el ordenador, se realizan las funciones de procesamiento en la entidad de hardware.

El programa que describe este contenido de procesamiento puede grabarse en un medio de grabación legible por ordenador. Como medio de grabación legible por ordenador, por ejemplo, puede utilizarse cualquier medio, como un aparato de grabación magnético, un disco óptico, un medio de grabación magnetoóptico y una memoria semiconductora. Específicamente, por ejemplo, es posible utilizar un aparato de disco duro, un disco flexible, una cinta magnética, o similar, como el aparato de grabación magnética, y utilizar un DVD (Disco Versátil Digital), un DVD-RAM (Memoria de Acceso Aleatorio), un CD-ROM (Memoria de Sólo Lectura de Disco Compacto), un CD-R (Grabable)/RW (Regrabable), o similar, como el disco óptico, utilizar un MO (disco Magneto-Óptico), o similar, como el medio de grabación magnetoóptico, y utilizar una EEP-ROM (Memoria de Sólo Lectura Programable y Borrable Electrónicamente), o similar, como la memoria semiconductora.

Además, este programa se distribuye, por ejemplo, mediante un medio de grabación portátil como un DVD y un CD-ROM en los que se graba el programa que se vende, regala, presta, o similar. Aún más, también es posible emplear una configuración donde este programa se distribuya almacenándose el programa en un aparato de almacenamiento de un ordenador servidor y transfiriéndose desde el ordenador servidor a otros ordenadores a través de una red.

Un ordenador que ejecuta un programa de este tipo, por ejemplo, primero almacena una vez un programa grabado en el medio de grabación portátil o un programa transferido desde el ordenador servidor en el dispositivo de almacenamiento del propio ordenador. Luego, tras la ejecución del procesamiento, este ordenador lee el programa almacenado en el aparato de almacenamiento del propio ordenador y ejecuta el procesamiento de acuerdo con el programa leído. Además, como otra forma de ejecución de este programa, el ordenador puede leer, directamente, un programa del medio de grabación portátil y ejecutar el procesamiento de acuerdo con el programa y, además, ejecutar, secuencialmente, el procesamiento de acuerdo con el programa recibido cada vez que el programa se transfiere desde el ordenador servidor a este ordenador. Además, también es posible emplear una configuración donde el procesamiento descrito anteriormente se ejecuta mediante el llamado servicio de tipo ASP (Proveedor de Servicios de Aplicación) que realiza funciones de procesamiento sólo mediante una instrucción de ejecución y adquisición de un resultado sin que se transfiera el programa desde el ordenador servidor a este ordenador. Tenga en cuenta que se supone que el programa en el formulario incluye información que se utilizará para su procesamiento por parte de un ordenador electrónico, y que es equivalente a un programa (no un comando directo al ordenador, sino datos, o similares, que tienen una propiedad que especifica el procesamiento del ordenador).

Además, mientras que, en esta forma, la entidad de hardware está constituida por un programa predeterminado que se ejecuta en el ordenador, al menos, parte del contenido de procesamiento puede realizarse con hardware.

La descripción anterior de las realizaciones de la presente invención se presenta con fines ilustrativos y descriptivos. La descripción no pretende proporcionar una descripción exhaustiva, ni limitar la invención a la forma estricta descrita. La invención se expone en el conjunto adjunto de reivindicaciones.

REIVINDICACIONES

1. Un aparato de aprendizaje de distribución de probabilidad que comprende

una unidad (150) de aprendizaje configurada para desde un sonido normal emitido desde una o más piezas de equipos diferentes del equipo objetivo de detección de anomalías, en lo sucesivo denominado sonido normal aprender, una primera distribución de probabilidad $q_1(x;\theta)$ que indica distribución del sonido normal, comprendiendo la unidad (105) de aprendizaje:

una unidad (110) generadora de datos de entrada configurada para generar datos de entrada x_i (i = 1, ..., N) a partir de sonido normal para aprender s_i (i = 1, ..., N) que es la entrada,

una unidad (120) de estimación de variable latente configurada para estimar una variable latente $z_{0,i}$ (i = 1, ..., N) correspondiente a los datos de entrada x_i a partir de los datos de entrada x_i generados a partir del sonido normal para el aprendizaje s_i (i = 1, ..., N) que es de entrada utilizando el parámetro θ de la primera distribución de probabilidad $q_i(x;\theta)$,

una unidad (130) de cálculo de la función de pérdida configurada para calcular un valor de una función L(0) de pérdida que se utilizará para la optimización del parámetro θ de la primera distribución de probabilidad qi(x;θ) a partir de la variable latente $z_{0,i}$ (i = 1, ..., N),

una unidad (140) de actualización de parámetros configurada para actualizar el parámetro θ de la primera distribución de probabilidad $q_1(x;\theta)$ con el fin de optimizar el valor de la función de pérdida L(θ), y

una unidad (150) de determinación de las condiciones de convergencia configurada para determinar las condiciones de convergencia establecidas de antemano como condiciones de terminación de la actualización de parámetros, producir la salida de la primera distribución de probabilidad $q_1(x;\theta)$ utilizando el parámetro θ actualizado por la unidad (140) de actualización de parámetros en un caso donde se satisfacen las condiciones de convergencia, y repetir el procesamiento de la unidad (110) de generación de datos de entrada, la unidad (120) de estimación de la variable latente, la unidad (130) de cálculo de la función de pérdida y la unidad (140) de actualización de parámetros en un caso en el que no se satisfacen las condiciones de convergencia,

en donde una variable x de la primera distribución de probabilidad $q_1(x;\theta)$ es una variable que indica datos de entrada generados a partir del sonido normal emitido desde la una o más piezas de equipo diferentes del equipo objetivo de detección de anomalías,

en donde la variable x se expresa como $x = f_K(f_{K-1}(...(f_1(z_0))...))$ utilizando transformación es f_i(i = 1, ..., K), en donde K es un número entero de 1 o mayor, y existen transformaciones inversas f_i^{-1} para las transformaciones f_i y una variable latente z_0,

$q_0(z_0)$ se establece como una distribución de probabilidad de la variable latente z_0,

una densidad de probabilidad $q_1(x;\theta)$ de probabilidad de los datos x de entrada se calcula utilizando la densidad $q_0(z_0)$ de probabilidad de la variable latente $z_0 = f_1^{-1}(f_2^{-1}(...(f_K^{-1}(x))...))$ correspondiente a los datos x de entrada

al menos una transformación inversas de las transformaciones fi(i = 1, ..., K) es una normalización adaptativa por lotes.

2. El aparato de aprendizaje de distribución de probabilidad según la reivindicación 1,

en donde, al menos, una transformación inversa de las transformaciones f_i(i = 1, ..., K) es una transformación lineal, y

una matriz correspondiente a la transformación lineal se expresa como un producto de una matriz triangular inferior y una matriz triangular superior o como un producto de una matriz triangular inferior, una matriz diagonal y una matriz triangular superior.

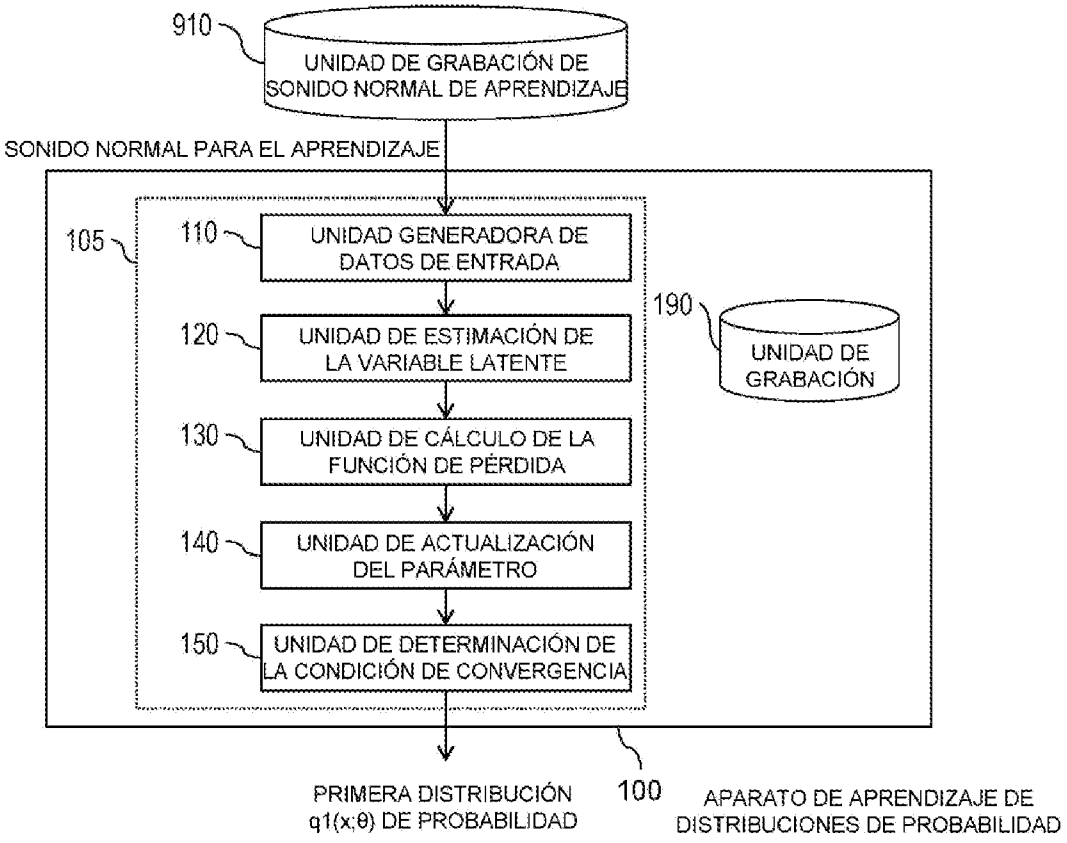
910
UNIDAD DE GRABACIÓN DE SONIDO NORMAL DE APRENDIZAJE
SONIDO NORMAL PARA EL APRENDIZAJE
105
110
UNIDAD GENERADORA DE DATOS DE ENTRADA
120
UNIDAD DE ESTIMACIÓN DE LA VARIABLE LATENTE
130
UNIDAD DE CÁLCULO DE LA FUNCIÓN DE PÉRDIDA
140
UNIDAD DE ACTUALIZACIÓN DEL PARÁMETRO
150
UNIDAD DE DETERMINACIÓN DE LA CONDICIÓN DE CONVERGENCIA
190
UNIDAD DE GRABACIÓN
PRIMERA DISTRIBUCIÓN q1(x;θ) DE PROBABILIDAD
100
APARATO DE APRENDIZAJE DE DISTRIBUCIONES DE PROBABILIDAD

FIG. 1

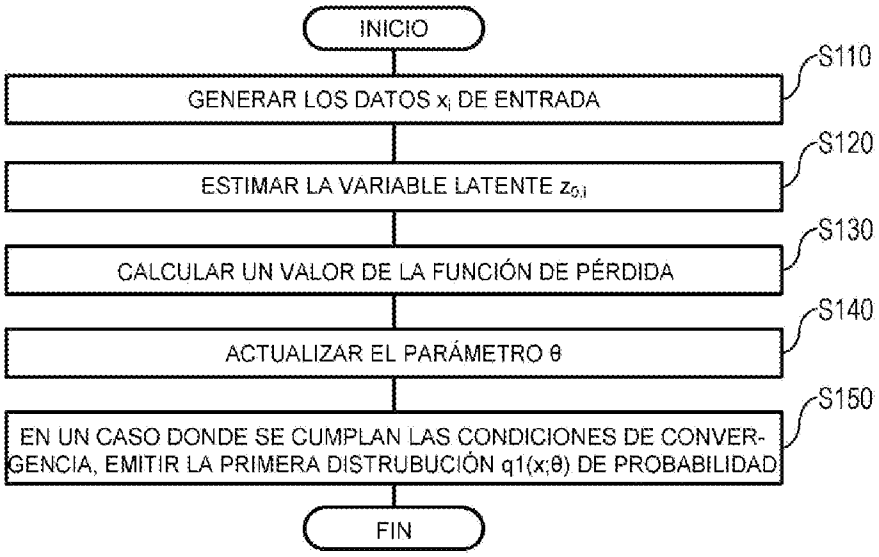
INICIO
S110
GENERAR LOS DATOS x_i DE ENTRADA
S120
ESTIMAR LA VARIABLE LATENTE $z_{\theta,i}$
S130
CALCULAR UN VALOR DE LA FUNCIÓN DE PÉRDIDA
S140
ACTUALIZAR EL PARÁMETRO θ
S150
EN UN CASO DONDE SE CUMPLAN LAS CONDICIONES DE CONVERGENCIA, EMITIR LA PRIMERA DISTRUBUCIÓN q1(x;θ) DE PROBABILIDAD
FIN

FIG. 2

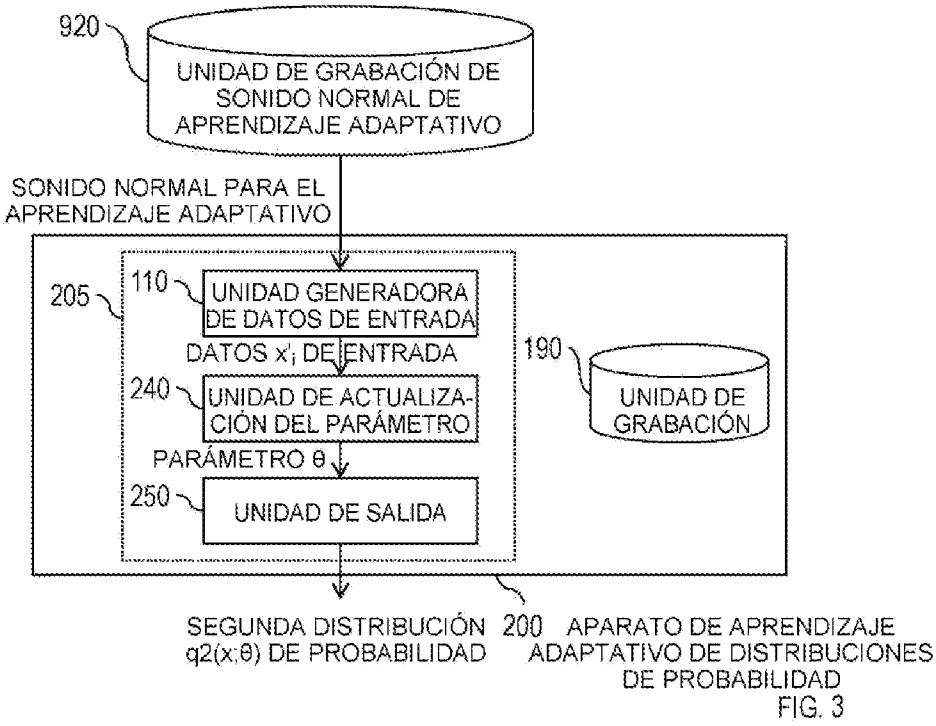
920
UNIDAD DE GRABACIÓN DE SONIDO NORMAL DE APRENDIZAJE ADAPTATIVO
SONIDO NORMAL PARA EL APRENDIZAJE ADAPTATIVO
205
110
UNIDAD GENERADORA DE DATOS DE ENTRADA
DATOS x'i DE ENTRADA
240
UNIDAD DE ACTUALIZACIÓN DEL PARÁMETRO
PARÁMETRO θ
250
UNIDAD DE SALIDA
190
UNIDAD DE GRABACIÓN
SEGUNDA DISTRIBUCIÓN q2(x;θ) DE PROBABILIDAD
200
APARATO DE APRENDIZAJE ADAPTATIVO DE DISTRIBUCIONES DE PROBABILIDAD

FIG. 3


INICIO
S110
GENERAR LOS DATOS x'i DE ENTRADA
S240
ACTUALIZAR EL PARÁMETRO θ
S250
EMITIR LA SEGUNDA DISTRIBUCIÓN q2(x;θ) DE PROBABILIDAD
FIN

FIG. 4

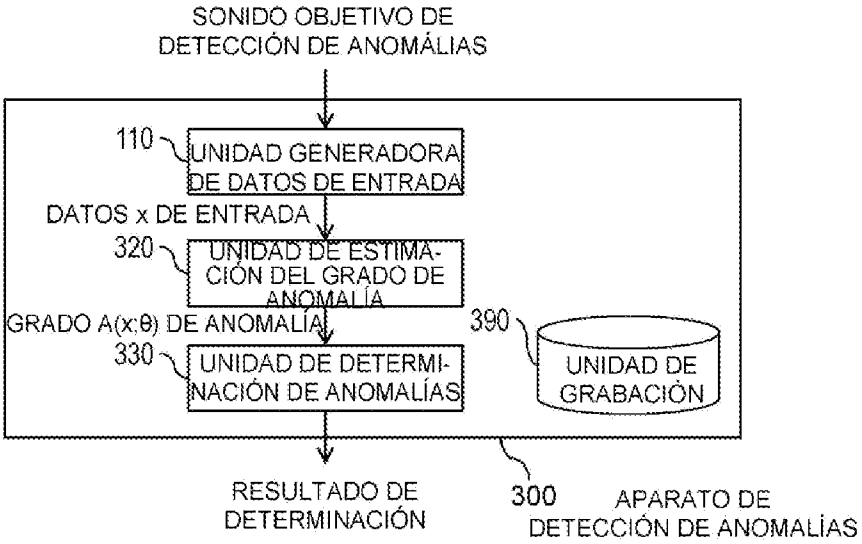
SONIDO OBJETIVO DE DETECCIÓN DE ANOMÁLIAS
110
UNIDAD GENERADORA DE DATOS DE ENTRADA
DATOS x DE ENTRADA
320
UNIDAD DE ESTIMACIÓN DEL GRADO DE ANOMALÍA
GRADO A(x;θ) DE ANOMALÍA
330
UNIDAD DE DETERMINACIÓN DE ANOMALÍAS
390
UNIDAD DE GRABACIÓN
RESULTADO DE DETERMINACIÓN
300
APARATO DE DETECCIÓN DE ANOMALÍAS

FIG. 5


INICIO
S110
GENERAR LOS DATOS x DE ENTRADA
S320
ESTIMACIÓN DEL A(x;θ) GRADO DE ANOMALÍA
S330
GENERAR UN RESULTADO DE DETERMINACIÓN
FIN

FIG. 6

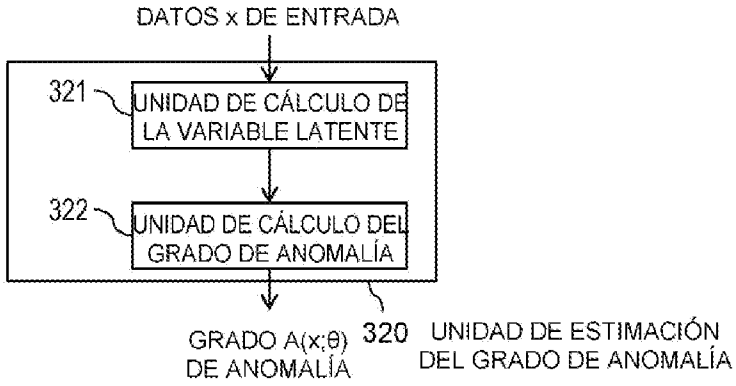
DATOS x DE ENTRADA
321
UNIDAD DE CÁLCULO DE LA VARIABLE LATENTE
322
UNIDAD DE CÁLCULO DEL GRADO DE ANOMALÍA
GRADO A(x;θ) DE ANOMALÍA
320
UNIDAD DE ESTIMACIÓN DEL GRADO DE ANOMALÍA

FIG. 7

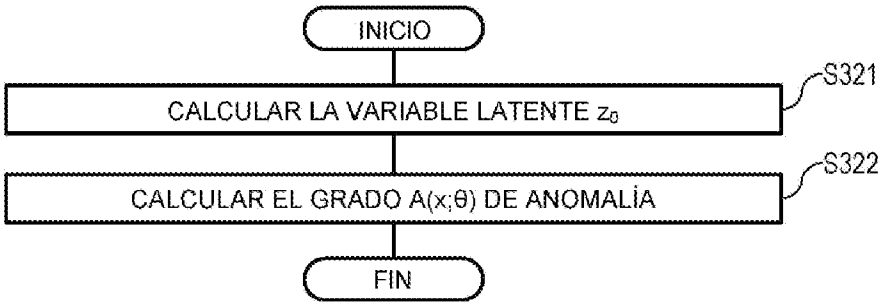
INICIO
S321
CALCULAR LA VARIABLE LATENTE z_0
S322
CALCULAR EL GRADO A(x;θ) DE ANOMALÍA
FIN

FIG. 8

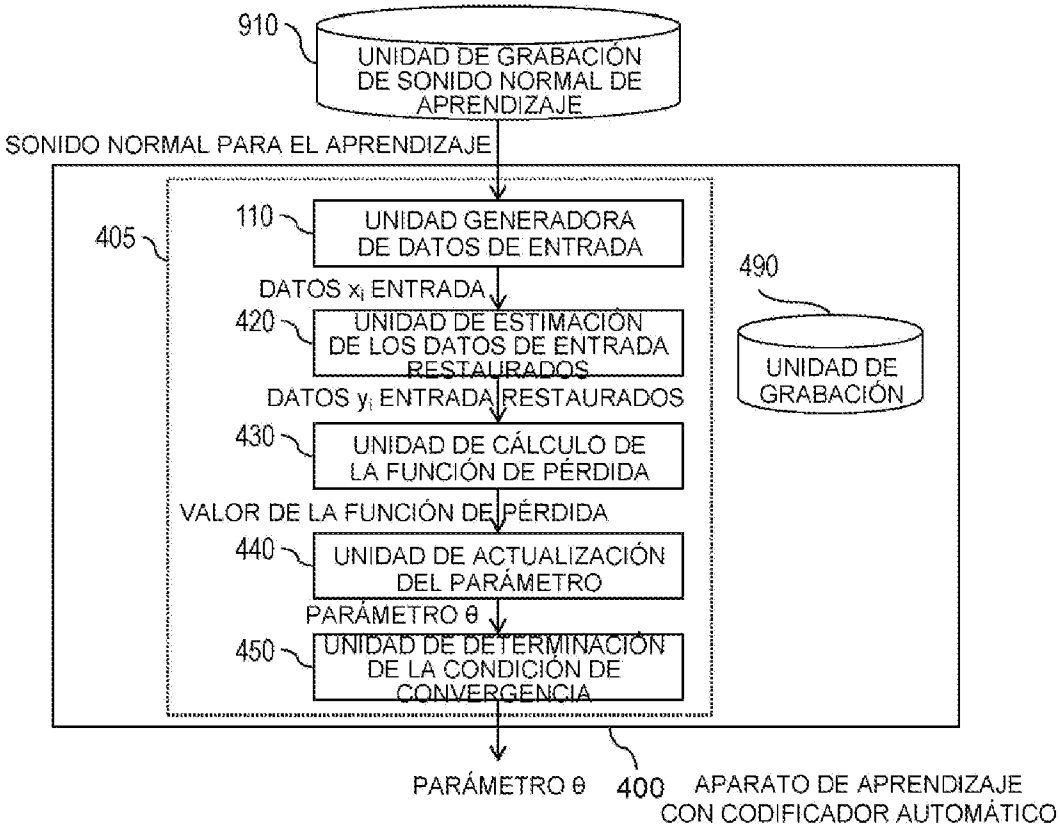
910
UNIDAD DE GRABACIÓN DE SONIDO NORMAL DE APRENDIZAJE
SONIDO NORMAL PARA EL APRENDIZAJE
405
110
UNIDAD GENERADORA DE DATOS DE ENTRADA
DATOS x_i ENTRADA
420
UNIDAD DE ESTIMACIÓN DE LOS DATOS DE ENTRADA RESTAURADOS
DATOS y_i ENTRADA RESTAURADOS
430
UNIDAD DE CÁLCULO DE LA FUNCIÓN DE PÉRDIDA
VALOR DE LA FUNCIÓN DE PÉRDIDA
440
UNIDAD DE ACTUALIZACIÓN DEL PARÁMETRO
PARÁMETRO θ
450
UNIDAD DE DETERMINACIÓN DE LA CONDICIÓN DE CONVERGENCIA
490
UNIDAD DE GRABACIÓN
PARÁMETRO θ
400
APARATO DE APRENDIZAJE CON CODIFICADOR AUTOMÁTICO

FIG. 9

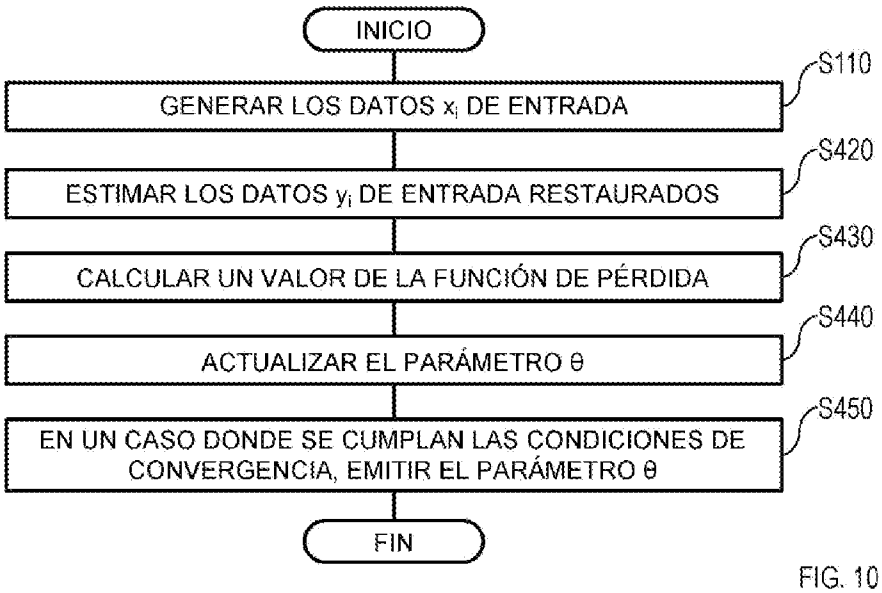
INICIO
S110
GENERAR LOS DATOS x_i DE ENTRADA
S420
ESTIMAR LOS DATOS y_i DE ENTRADA RESTAURADOS
S430
CALCULAR UN VALOR DE LA FUNCIÓN DE PÉRDIDA
S440
ACTUALIZAR EL PARÁMETRO θ
S450
EN UN CASO DONDE SE CUMPLAN LAS CONDICIONES DE CONVERGENCIA, EMITIR EL PARÁMETRO θ
FIN

FIG. 10

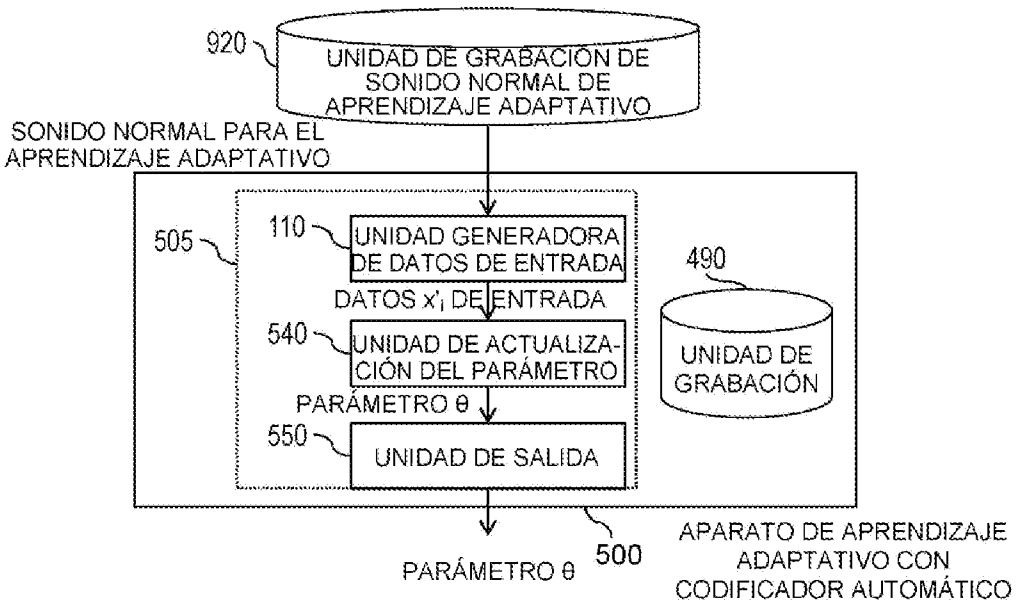
920
UNIDAD DE GRABACIÓN DE SONIDO NORMAL DE APRENDIZAJE ADAPTATIVO
SONIDO NORMAL PARA EL APRENDIZAJE ADAPTATIVO
505
110
UNIDAD GENERADORA DE DATOS DE ENTRADA
DATOS x'i DE ENTRADA
540
UNIDAD DE ACTUALIZACIÓN DEL PARÁMETRO
PARÁMETRO θ
550
UNIDAD DE SALIDA
490
UNIDAD DE GRABACIÓN
PARÁMETRO θ
500
APARATO DE APRENDIZAJE ADAPTATIVO CON CODIFICADOR AUTOMÁTICO

FIG. 11

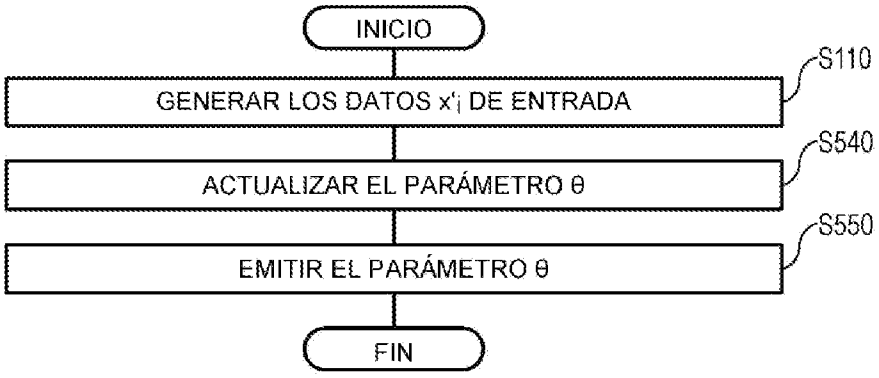
INICIO
S110
GENERAR LOS DATOS x'i DE ENTRADA
S540
ACTUALIZAR EL PARÁMETRO θ
S550
EMITIR EL PARÁMETRO θ
FIN

FIG. 12

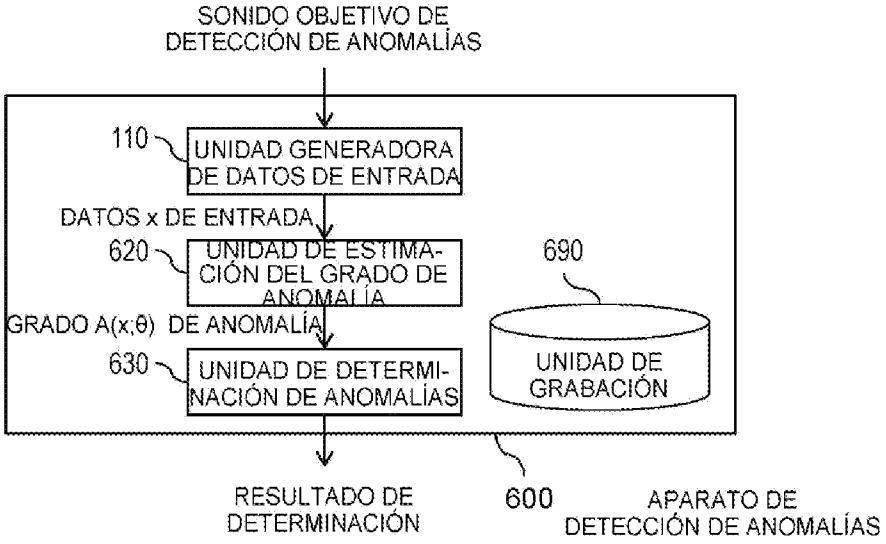
SONIDO OBJETIVO DE DETECCIÓN DE ANOMALÍAS
110
UNIDAD GENERADORA DE DATOS DE ENTRADA
DATOS x DE ENTRADA
620
UNIDAD DE ESTIMACIÓN DEL GRADO DE ANOMALÍA
690
GRADO A(x;θ) DE ANOMALÍA
630
UNIDAD DE DETERMINACIÓN DE ANOMALÍAS
UNIDAD DE GRABACIÓN
RESULTADO DE DETERMINACIÓN
600
APARATO DE DETECCIÓN DE ANOMALÍAS

FIG. 13

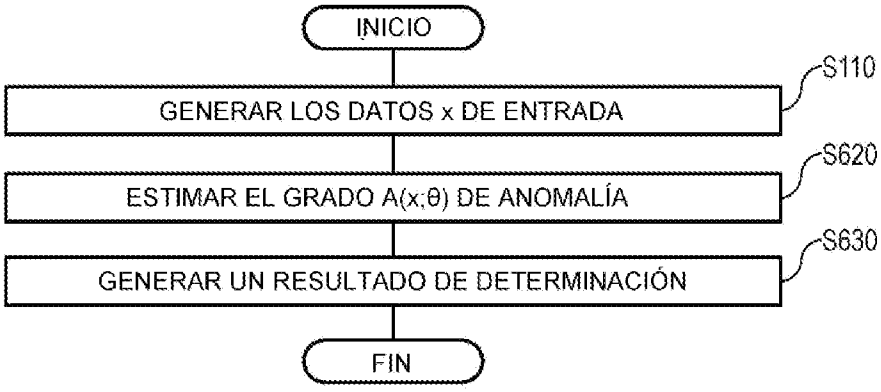
INICIO
S110
GENERAR LOS DATOS x DE ENTRADA
S620
ESTIMAR EL GRADO A(x;θ) DE ANOMALÍA
S630
GENERAR UN RESULTADO DE DETERMINACIÓN
FIN

FIG. 14

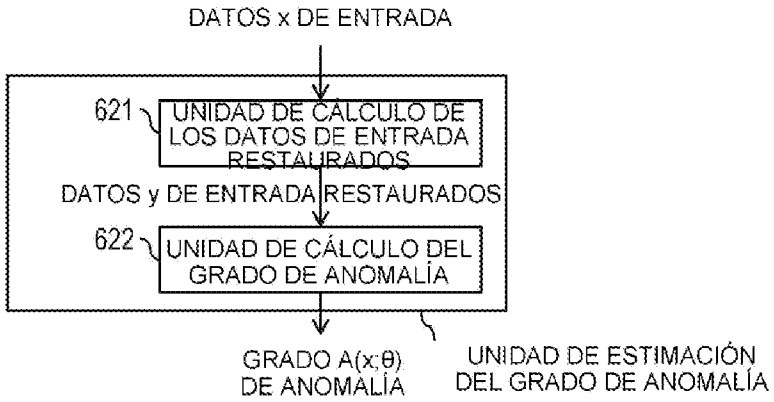
DATOS x DE ENTRADA
621
UNIDAD DE CÁLCULO DE LOS DATOS DE ENTRADA RESTAURADOS
DATOS y DE ENTRADA RESTAURADOS
622
UNIDAD DE CÁLCULO DEL GRADO DE ANOMALÍA
GRADO A(x;θ) DE ANOMALÍA
UNIDAD DE ESTIMACIÓN DEL GRADO DE ANOMALÍA

FIG. 15

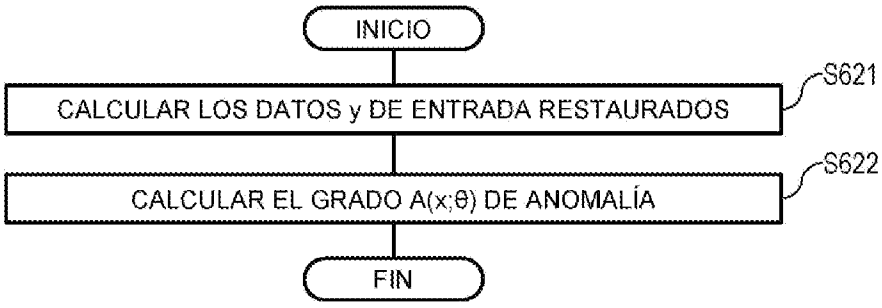
INICIO
S621
CALCULAR LOS DATOS y DE ENTRADA RESTAURADOS
S622
CALCULAR EL GRADO A(x;θ) DE ANOMALÍA
FIN

FIG. 16

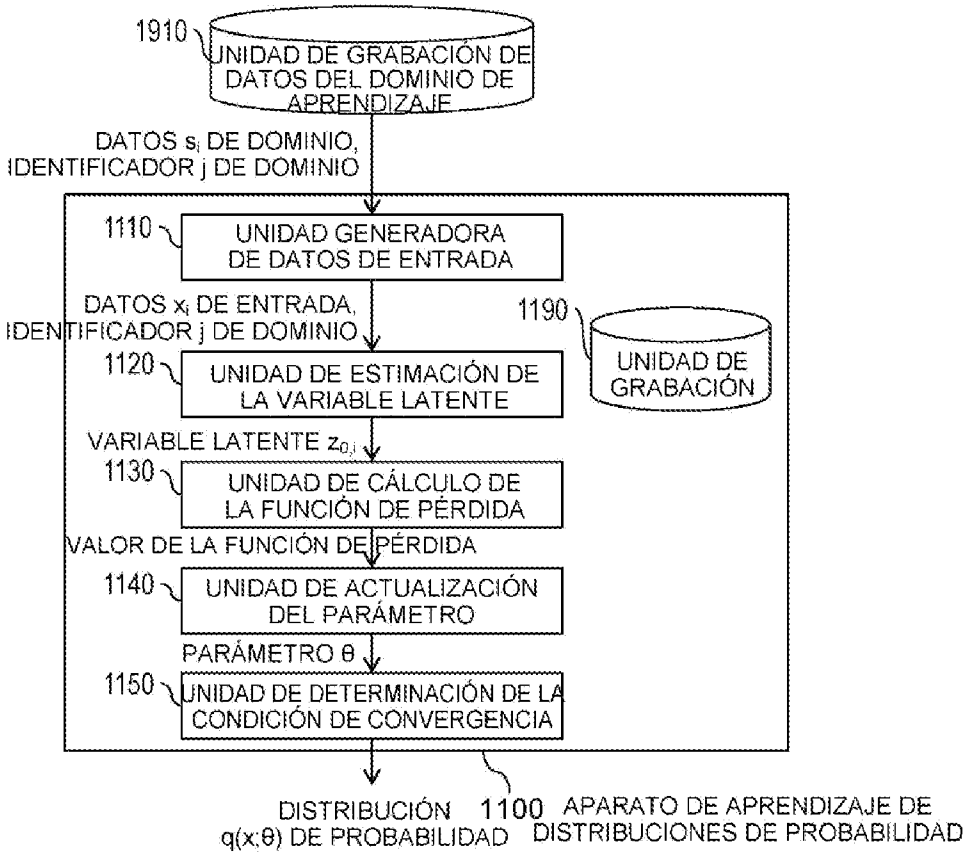
1910
UNIDAD DE GRABACIÓN DE DATOS DEL DOMINIO DE APRENDIZAJE
DATOS s_i DE DOMINIO, IDENTIFICADOR j DE DOMINIO
1110
UNIDAD GENERADORA DE DATOS DE ENTRADA
DATOS x_i DE ENTRADA, IDENTIFICADOR j DE DOMINIO
1190
UNIDAD DE GRABACIÓN
1120
UNIDAD DE ESTIMACIÓN DE LA VARIABLE LATENTE
VARIABLE LATENTE $z_{0,i}$
1130
UNIDAD DE CÁLCULO DE LA FUNCIÓN DE PÉRDIDA
VALOR DE LA FUNCIÓN DE PÉRDIDA
1140
UNIDAD DE ACTUALIZACIÓN DEL PARÁMETRO
PARÁMETRO θ
1150
UNIDAD DE DETERMINACIÓN DE LA CONDICIÓN DE CONVERGENCIA
DISTRIBUCIÓN q(x;θ) DE PROBABILIDAD
1100
APARATO DE APRENDIZAJE DE DISTRIBUCIONES DE PROBABILIDAD

FIG. 17

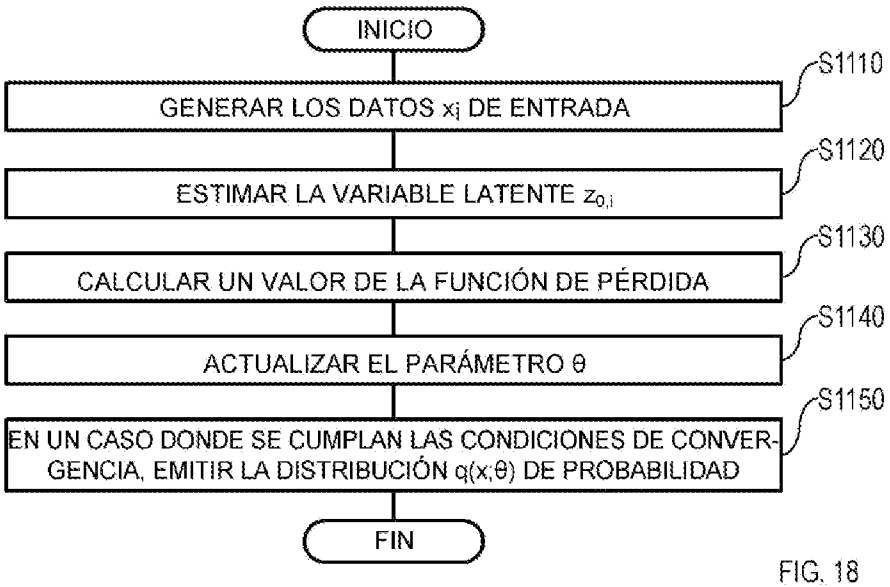
INICIO
S1110
GENERAR LOS DATOS x_i DE ENTRADA
S1120
ESTIMAR LA VARIABLE LATENTE $z_{0,i}$
S1130
CALCULAR UN VALOR DE LA FUNCIÓN DE PÉRDIDA
S1140
ACTUALIZAR EL PARÁMETRO θ
S1150
EN UN CASO DONDE SE CUMPLAN LAS CONDICIONES DE CONVERGENCIA, EMITIR LA DISTRIBUCIÓN q(x;θ) DE PROBABILIDAD
FIN

FIG. 18

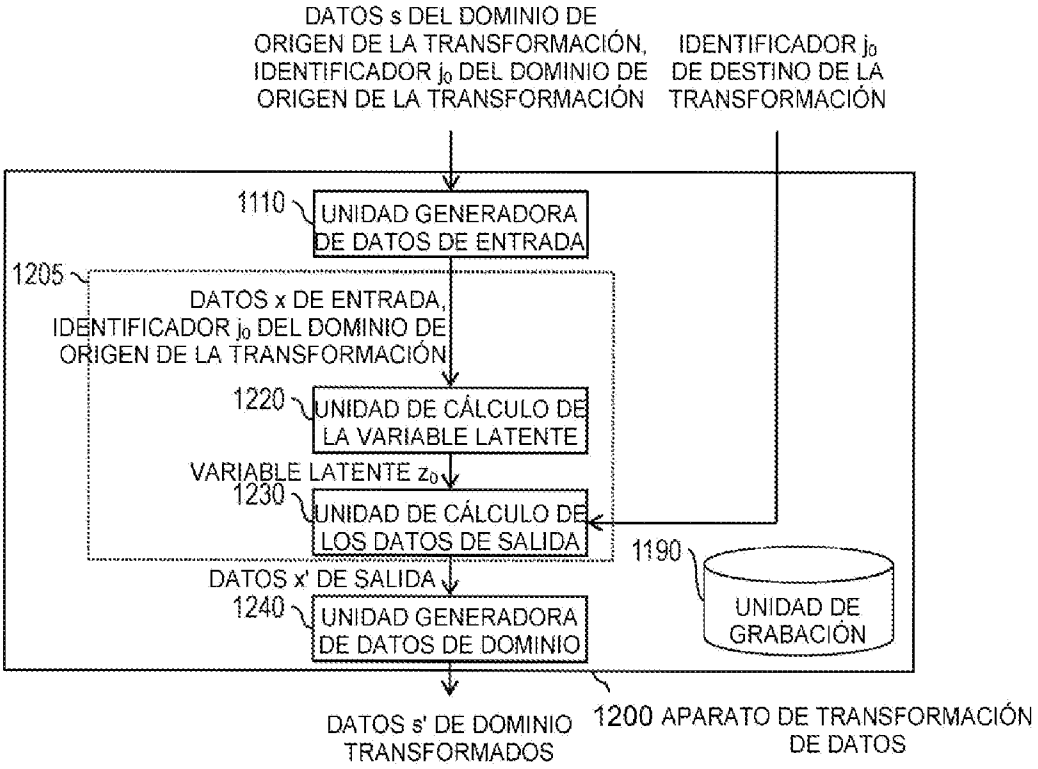
DATOS s DEL DOMINIO DE ORIGEN DE LA TRANSFORMACIÓN, IDENTIFICADOR j_0 DEL DOMINIO DE ORIGEN DE LA TRANSFORMACIÓN
IDENTIFICADOR j_0 DE DESTINO DE LA TRANSFORMACIÓN
1110
UNIDAD GENERADORA DE DATOS DE ENTRADA
1205
DATOS x DE ENTRADA, IDENTIFICADOR j_0 DEL DOMINIO DE ORIGEN DE LA TRANSFORMACIÓN
1220
UNIDAD DE CÁLCULO DE LA VARIABLE LATENTE
VARIABLE LATENTE z_0
1230
UNIDAD DE CÁLCULO DE LOS DATOS DE SALIDA
1190
UNIDAD DE GRABACIÓN
DATOS x' DE SALIDA
1240
UNIDAD GENERADORA DE DATOS DE DOMINIO
DATOS s' DE DOMINIO TRANSFORMADOS
1200 APARATO DE TRANSFORMACIÓN DE DATOS

FIG. 19

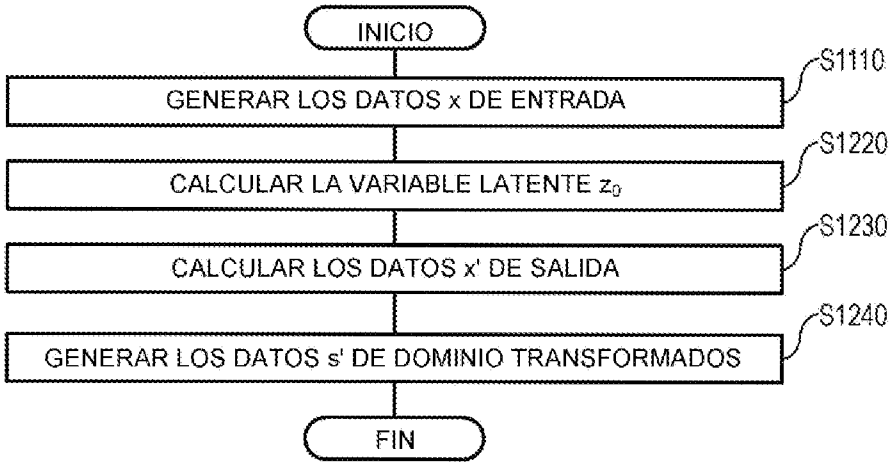
INICIO
GENERAR LOS DATOS x DE ENTRADA
S1110
CALCULAR LA VARIABLE LATENTE z_0
S1220
CALCULAR LOS DATOS x' DE SALIDA
S1230
GENERAR LOS DATOS s' DE DOMINIO TRANSFORMADOS
S1240
FIN

FIG. 20

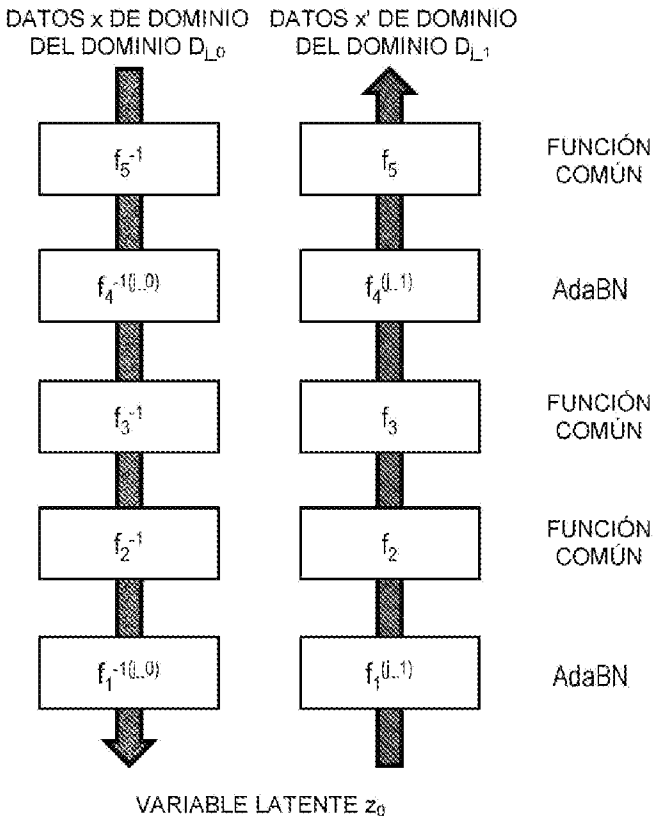
DATOS x DE DOMINIO DEL DOMINIO D_{L0}
DATOS x' DE DOMINIO DEL DOMINIO D_{L1}
f_5^{-1}
f_5
FUNCIÓN COMÚN
$f_4^{-1(0)}$
$f_4^{(1)}$
AdaBN
f_3^{-1}
f_3
FUNCIÓN COMÚN
f_2^{-1}
f_2
FUNCIÓN COMÚN
$f_1^{-1(0)}$
$f_1^{(1)}$
AdaBN
VARIABLE LATENTE z_0

FIG. 21

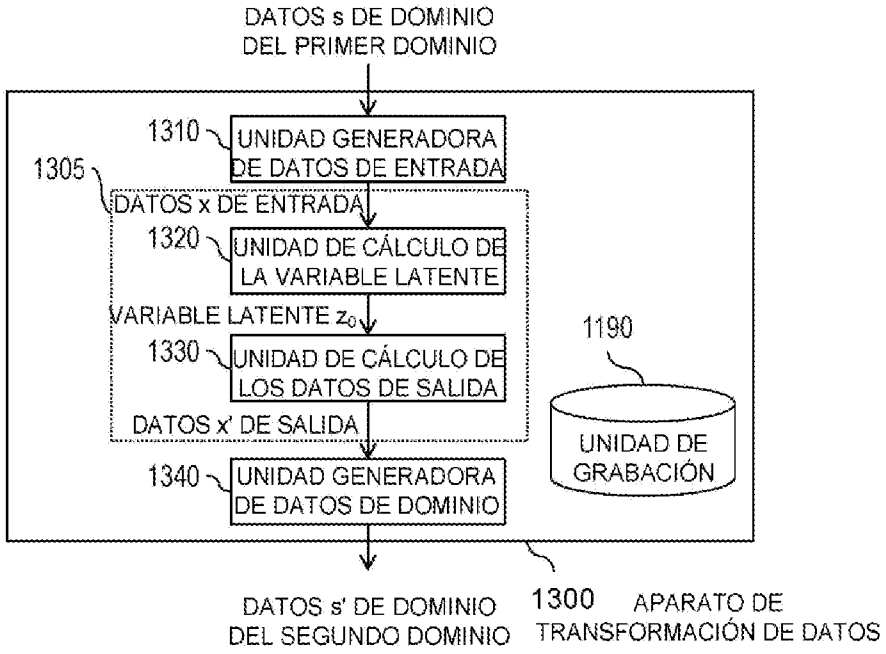
DATOS s DE DOMINIO DEL PRIMER DOMINIO
1310
UNIDAD GENERADORA DE DATOS DE ENTRADA
1305
DATOS x DE ENTRADA
1320
UNIDAD DE CÁLCULO DE LA VARIABLE LATENTE
VARIABLE LATENTE z_0
1330
UNIDAD DE CÁLCULO DE LOS DATOS DE SALIDA
DATOS x' DE SALIDA
1340
UNIDAD GENERADORA DE DATOS DE DOMINIO
1190
UNIDAD DE GRABACIÓN
DATOS s' DE DOMINIO DEL SEGUNDO DOMINIO
1300 APARATO DE TRANSFORMACIÓN DE DATOS

FIG. 22

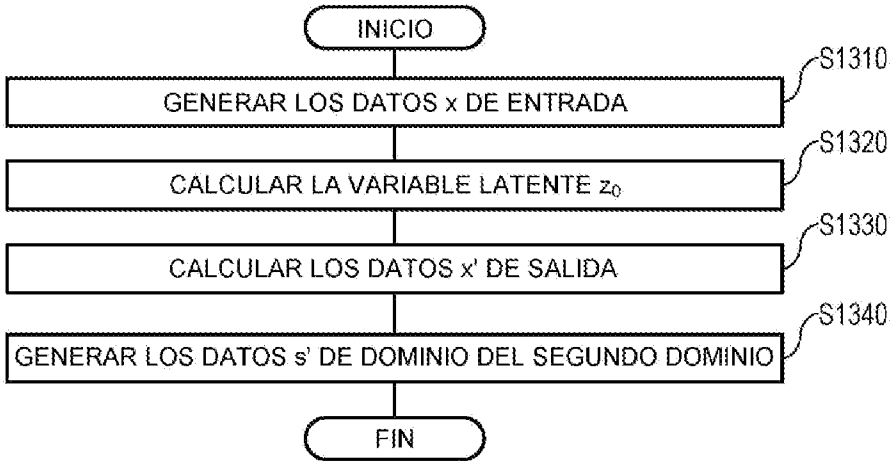
INICIO
S1310
GENERAR LOS DATOS x DE ENTRADA
S1320
CALCULAR LA VARIABLE LATENTE z_0
S1330
CALCULAR LOS DATOS x' DE SALIDA
S1340
GENERAR LOS DATOS s' DE DOMINIO DEL SEGUNDO DOMINIO
FIN

FIG. 23

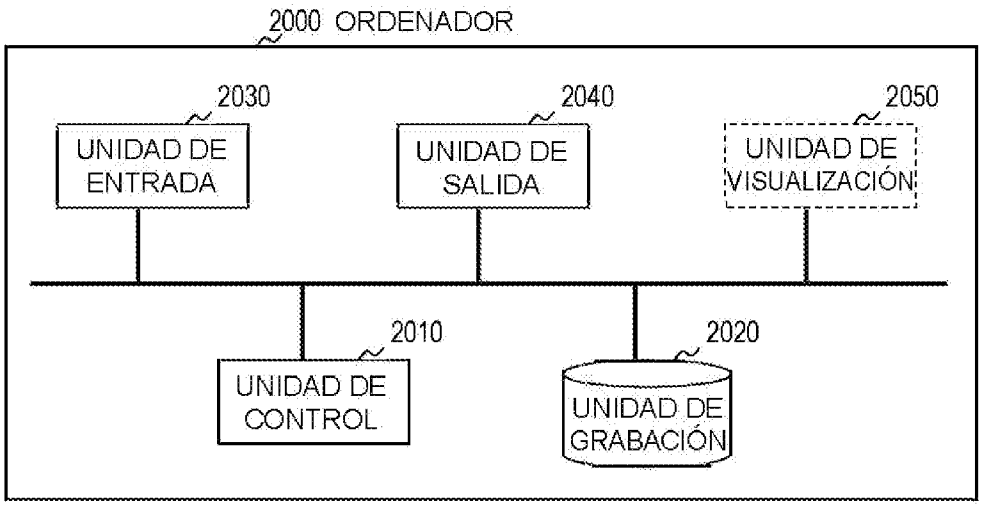
2000 ORDENADOR
2030
UNIDAD DE ENTRADA
2040
UNIDAD DE SALIDA
2050
UNIDAD DE VISUALIZACIÓN
2010
UNIDAD DE CONTROL
2020
UNIDAD DE GRABACIÓN

FIG. 24