

Redes neuronales ópticas para rotación de imágenes

R. Cerdán
Inisel Espacio

RESUMEN

El presente artículo es una breve exposición aplicación de las redes neuronales ópticas a problema clásico en tratamiento de imágenes, es el de la rotación de imágenes digitalizadas.

PALABRAS CLAVE: redes neuronales, tratamiento de imágenes.

ABSTRACT

This article briefly exposes the application of cal neural nets to solve a now classical problem in image processing, namely, image rotation.

KEY WORDS: neural nets, image processing.

INTRODUCCIÓN

Una de las funciones más importantes dentro del área del tratamiento digital de imágenes la constituye la rotación de imágenes digitalizadas. Todo sistema de procesamiento digital de imágenes debe realizar las funciones básicas de muestreo (que incluye la rotación y el dimensionamiento) y la convolución bidimensional o filtrado de la imagen. Entre las aplicaciones básicas que encuentran estas funciones cabe citar la *corrección geométrica de imágenes* tomadas por satélites de observación terrestre.

El proceso de corrección de imágenes surge de la necesidad de compensar dichas imágenes frente a los errores inducidos por los sensores de recogida de datos, para lo cual se procede a una *rectificación geométrica y radiométrica* de la imagen. Básicamente, el proceso no tiene por objetivo sino la conversión a sistemas determinados de coordenadas de los datos recogidos por el sensor, para lo cual se efectúa el muestreo electrónico de la imagen.

Si bien los algoritmos existentes para efectuar estas correcciones son fijos, no lo es su implementación, que desde la aparición de la tecnología VLSI ha venido consistiendo en la fabricación de circuitos integrados especiales que efectúan las laboriosas operaciones de adición y diagonalización necesarias para ejecutar los algoritmos.

ALGORITMOS DE ROTACIÓN DE IMÁGENES

El escenario clásico es aquél en el que la interpretación correcta de una imagen de satélite de 512 por 512 requerirá la rotación, escalamiento y *warping* de la imagen.

Este proceso conlleva aplicar unos algoritmos que transformen los datos de entrada del modo deseado.

Conceptualmente hablando, la transformación global de una imagen, o su muestreo, requiere resolver el problema en dos pasos: en primer lugar, realizar una *transformación del sistema de coordenadas* inicial para a continuación proceder a una *interpolación de píxeles*. La interpolación es necesaria debido a que las posiciones de aquellos píxeles transformados no coincidirá con las posiciones iniciales de los mismos. Los valores de los nuevos píxeles se obtienen entonces interpolando los píxeles originales respecto a las posiciones colindantes a la nueva posición del píxel.

La interpolación es una operación de reconstrucción de la imagen que conlleva una *convolución discreta* en dos dimensiones, convolución que se efectúa sobre una función derivada de los datos de la imagen mediante una función de sopesamiento que se denomina *kernel de interpolación*. Así, si $p(x,y)$ es la imagen original muestreada, de dimensión $m \times n$, la imagen interpolada $g(x,y)$ se expresa de la siguiente forma:

$$g(x,y) = \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} c_{ij} W(x-x_i, y-y_j) \quad (1)$$

donde x_i e y_j son nodos de interpolación, W es el kernel de interpolación y g la función de interpolación es cuestión. Las c son los parámetros dependientes de los datos muestreados. Una de las propiedades fundamentales de las funciones de interpolación es que éstas deben coincidir con los datos muestreados en los nodos de interpolación o puntos de muestreo, de lo contrario no existirá convolución ya que el resultado no es reversible al esta-

do original. Expresado en términos físicos, diremos que la función de interpolación es simétrica respecto al tiempo y, por lo tanto, reversible.

Las funciones de interpolación más comunes son la *convolución cúbica*, la *bilineal* y la denominada «*nearest neighbor*» o punto más próximo. La precisión relativa de cada una de estas funciones de interpolación viene determinada por sus *tasas de convergencia*, es decir, por la rapidez con que el error de aproximación toma el valor cero a medida que decrece el incremento de muestreo (o función delta del muestreo). En estos términos, la convolución cúbica presenta el mejor rendimiento de las tres funciones de interpolación. La elección de una función u otra depende no sólo del rendimiento deseado, sino de la arquitectura física que soportará la implementación del algoritmo de rotación.

La complejidad física del circuito responsable de ejecutar el algoritmo determinará su coste de fabricación y, ergo, el de su adquisición.

Existe un diseño cuyo rendimiento es muy superior al de las pastillas de VLSI y que se basa en *redes neuronales ópticas*.

FORMALISMO MATEMÁTICO

El problema matemático a resolver por una red neuronal es idéntico al que debe resolver la solución VLSI, a saber: una transformación de coordenadas que determine la ubicación del píxel muestreado dentro del marco de la imagen original.

Expresado en términos de percepción espacial: se dice que un objeto ha rotado si y sólo si mentalmente somos capaces de reconocerlo como era antes de la rotación. En caso contrario, o tenemos un problema neurológico grave o simplemente el objeto que se nos presenta y se nos dice rotado no es en realidad un tal objeto sino otro diferente.

Una transformación afín que permita la rotación de una imagen, su escalamiento y su translación puede describirse mediante notación matricial del siguiente modo:

sean x, y las coordenadas originales y u, v las muestradas. Entonces

$$[u \ v \ 1] = [x \ y \ 1] * M \tag{2}$$

si la multiplicación por M^{-1} es tal que

$$[x \ y \ 1] = [u \ v \ 1] * M^{-1} \tag{3}$$

donde:

$$M = \begin{bmatrix} a & d & 0 \\ b & e & 0 \\ c & f & 1 \end{bmatrix}$$

$$M^{-1} = \begin{bmatrix} Dx_u & Dx_v & 0 \\ Dy_u & Dy_v & 0 \\ x_0 & y_0 & 1 \end{bmatrix}$$

son, respectivamente, las matrices de transformación de coordenadas regresivas e inversas. Es necesario además las condiciones siguientes:

$$\begin{aligned} Dx_u &= e/D \\ Dx_v &= -d/D \\ Dy_u &= -b/D \\ Dy_v &= a/D \\ x_0 &= (bf-ce)/D \\ y_0 &= (cd-af)/D \\ D &= ae-bd \end{aligned}$$

Este conjunto de condiciones, junto con la definición de las matrices, no hace sino expresar que la rotación en un eje cualquiera conlleva implicaciones en el segundo eje que no son idénticas, sino que vienen multiplicadas por un factor de escala con el fin de mantener la perspectiva y no violentar la percepción visual. Rotar un objeto no es girarlo sin más en el espacio: es necesario *reescalar* sus dimensiones.

La transformación afín anterior es suficiente para escalar, trasladar y rotar una imagen, pero no es suficiente para aplicaciones tipo Landsat, donde además se requiere un *warping no lineal*. Para ello puede utilizarse una función bidimensional de segundo orden que sirva para aproximar el *warping geométrico* deseado. Y sólo podemos aproximar esa operación porque, *per dejinitionem*, toda función lineal no admite soluciones exactas sino aproximadas. La solución de este tipo de funciones puede mecanizarse mediante VLSI de un modo sencillo utilizando sólo sumadores y registros sin necesidad de recurrir a multiplicadores.

Veamos la mecánica a seguir:

Sea una matriz de $m \times n$ píxels en el sistema de coordenadas $[x \ y]$ con valores de píxel en (i, j) , expresados como $p(i, j)$, definidos tal que $0 < i < m-1$ y $0 < j < n-1$ para representar la imagen original. El valor del píxel $p(x, y)$ en (x, y) se obtiene entonces por convolución utilizando la función kernel de interpolación bidimensional $W(x, y)$ tal que:

$$p(x, y) = \sum_{i=j=0}^{n-1} p(i, j) W(x-x_i, y-y_j) \tag{4}$$

que puede simplificarse interpolando sólo un número k de píxels colindantes, tal que:

$$p(x, y) = \sum_{i=j=0}^{k-1} p(x_i - x_j) W(x-x_i, y-y_j) \tag{5}$$

donde x_i e y_i son uno de los k enteros más próximos de x e y , respectivamente, resultando que $p(x, y)$ tiene un valor constante para aquellos píxels no definidos de la matriz original de píxels.

REDES NEURONALES OPTICAS

Una vez explicados los algoritmos de rotación de imágenes más usuales y el formalismo matemático general que los sustenta, veremos ahora cómo la solución física sobre la que deben implementarse presenta inconvenientes de rendimiento.

En primer lugar, hay que hacer notar que todo el tratamiento rotacional hasta aquí explicado se refiere a transformaciones píxel a píxel, lo que implica definir matrices para cada píxel y aplicar los algoritmos expuestos a cada píxel. Cuando las imágenes a tratar son las típicas de 512 x 512, este proceso deviene lento y computacionalmente costoso. La solución puede atacarse desde dos bandas:

- o bien se diseña un algoritmo nuevo que abarque a conjuntos de píxels
- o bien se implementan los algoritmos clásicos en arquitecturas nuevas más robustas y rápidas

El enfoque seguido aquí es un compromiso entre ambas soluciones, es decir: se utilizará un algoritmo rotacional nuevo, llamado de *variación espacial*, y se implementará en una arquitectura radicalmente distinta de las tradicionales basadas en VLSI: en *redes neuronales ópticas*.

No vamos a explicar aquí qué es una red neuronal óptica. Bastará definir las como arquitecturas de procesamiento no lineales altamente paralelas. Su paralelismo les confiere una alta velocidad de procesamiento. El hecho de que sean ópticas se refiere a que, a diferencia de las electrónicas, las señales son ópticas y se utilizan para su tratamiento dispositivos optoelectrónicos. Un ejemplo grosero de la diferencia entre un tratamiento de señal digital y uno óptico vendría dado por la solución que adoptan ambas tecnologías para un mismo problema, cual es hallar la transformada de Fourier de una imagen. Este problema tiene una solución tradicional mediante DSP que implementan el algoritmo de Tukey de un modo normalmente aceptable en términos de tiempo de respuesta.

El mismo dispositivo, implementado utilizando técnicas ópticas, consiste en iluminar una imagen mediante luz coherente y utilizar un dispositivo de correlación óptico. Así, la imagen de entrada sufre una transformación de Fourier mediante difracción mientras se propaga desde el plano focal frontal hasta el plano posterior de una lente de focalización. La transformada se multiplica por una función de filtrado que se visualiza en un modulador óptico espacial colocado en el plano de Fourier. El producto de la imagen por la función de filtrado se transforma inversamente al pasar al plano focal posterior de una segunda lente. En dicho plano se obtiene entonces la correlación de la imagen y la función de filtrado, es decir, se obtiene la transformada de Fourier de la imagen.

Técnicamente hablando diremos que la difracción se consigue mediante matrices de *microlentes binarias*, que funcionan con el principio físico de la difracción focalizando hacia un mismo punto

toda la luz que reciben, lográndose rendimientos ópticos del 99%. Las lentes tampoco representan problema, siendo el principal problema el modulador óptico. La tecnología actual permite fabricar ya moduladores ópticos de 256 x 256 píxels, lo que equivale a un DSP operando a 500 MOPS en la resolución del algoritmo FFT.

La arquitectura de un sistema como el descrito permite fabricar una neurona compuesta por nueve microlentes binarias capaz de procesar la luz emitida por nueve píxels, lo que correspondería a la matriz M . Una red compuesta por 30.000 de estas neuronas, directamente unidas a fotodetectores, bastaría para procesar una imagen de 512 x 512. A continuación de la red de microlentes y fotodetectores se inserta una matriz holográfica, que facilita las *conexiones sinápticas* y *los pesos entre neuronas*, dos parámetros esenciales en toda red neuronal. Además, dicho holograma puede codificar directamente la función kernel de interpolación que ataca directamente a la red neuronal.

Finalmente, la salida de la red neuronal consiste en una pantalla de visualización donde la imagen rotada se presenta.

Dado que la red así construida opera simultáneamente sobre todas las matrices de píxels de entrada de un modo altamente paralelo, la operación de rotación no requiere ninguna iteración siendo automática. No obstante, una red de microlentes de 30.000 elementos presenta ciertas dificultades de fabricación, especialmente en lo referente a su conexión con la matriz de fotodetectores. Reducir su número implica que la operación de rotación no puede aplicarse a toda la imagen, por lo cual dicha reducción sólo puede efectuarse a cambio de mejorar el algoritmo de rotación con el que operará la red neuronal. Por ello, es necesario aplicar un algoritmo robusto y más generalista que los clásicos.

Dicho algoritmo se discute en el siguiente apartado.

ALGORITMO DE VARIACIÓN ESPACIAL

El algoritmo de variación espacial es una técnica de tratamiento de imágenes que permite *contextualizar* conjuntos de píxels. La contextualización consiste en asignar a conjuntos de píxels un atributo, como por ejemplo «trigo», «algodón», «zona urbana». Es decir, tras el tratamiento de la imagen original mediante la transformada de Fourier y su correlación, el siguiente paso es analizar la imagen resultante para ~~entender~~ *entender* ~~su~~ *su* ~~carácter~~ *carácter* y el reconocimiento de formas no puede realizarse siguiendo el enfoque clásico de tratamiento píxel-a-píxel, sino atendiendo al contexto en el que existe un conjunto de píxels dado. El tratamiento píxel-a-píxel no tiene en

cuenta la información espacial, que es esencial para establecer categorías de dependencias entre pixels y la correlación bidimensional entre pixels.

Todos los algoritmos expuestos en el apartado segundo permiten procesar las imágenes en función de la rotación, ampliación, reescalamiento o corrección geométrica de los pixels. La aplicación de la arquitectura neuronal óptica sirve para acelerar este proceso, pero todavía estamos lejos de *entender la imagen*, su contenido, su contexto.

Existen esquemas de clasificación contextual de rendimiento aceptable, como la clasificación contextual multispectral basada en campos aleatorios de Markov, o el esquema basado en la relajación estocástica. El primer esquema, además, ha sido implementado en su versión neuro-óptica por técnicos de la NASA para análisis espectral de la llama de motores en ignición, con el fin de detectar elementos químicos presentes en la misma y determinar, mediante análisis espectral, anomalías en la fase de combustión que pudieran llevar a una explosión catastrófica del propulsor. Esta técnica se basa en un procedimiento recursivo que optimiza la estimación del estado de un campo aleatorio bi-estado de Markov.

En cualquier caso el corazón de ambos esquemas es la estimación de probabilidad máxima de las probabilidades de transición, técnica matemática que no sirve para reconocer fronteras entre regiones o segmentos de una imagen. Por el contrario, la técnica de variación espacial descansa sobre datos estadísticos locales, en vez de globales. El nombre del algoritmo viene precisamente de esta propiedad, es decir, de la estimación local de variaciones espaciales en vez de una estimación global de la probabilidad máxima, que requiere extrapolar a toda la imagen. Ahora bien, la utilización de datos locales dentro de la imagen requiere algún tipo de muestreo que defina la ventana de análisis sobre la que aplicar el algoritmo. Una ventana excesivamente pequeña nos llevaría de nuevo al tratamiento pixel-a-píxel; una ventana excesivamente grande implicaría contener varias regiones o segmentos, que no serían reconocidos.

El problema de la clasificación contextual puede formularse desde el modelo bayesiano puro en estos términos: dada una imagen digitalizada tipo Landsat o ERS-1, *¿cómo asignar etiquetas C a los pixels que rodean a un píxel determinado (i,j) manteniendo al mínimo las pérdidas?* Las pérdidas a las que nos referimos se derivan de que los datos de entrada para la clasificación contextual derivan a su vez de las medidas que efectuemos sobre los pixels que rodean al píxel (i,j) así como de las dependencias predefinidas entre pixels. Advertimos aquí que la solución a este problema no puede hacerse pixel a pixel por una sencilla razón: cuando pasemos al siguiente píxel lo único que habremos hecho es transferir el problema a este píxel. Y si alguien está pensando en utilizar el modelo de

Markov-Gibbs deberá explicar entonces cómo esquivar el ineludible uso de la simulación por Monte Carlo a la que, *per definitionem*, nos avoca este método.

La utilización de probabilidades de transición, que procede directamente de un tal modelo Markov-Gibbs, implica segmentar el intervalo de posibles valores que puede tomar un pixel entre, por ejemplo, 0 y 1. Seguidamente este intervalo se subdivide entre 0 y k , y entre k y 1, calculándose k mediante el método de selección de parámetros estadísticos locales, *hic et nunc.* el método de Monte Carlo. Seguidamente uno puede asignar una categoría a todos los pixels que estén comprendidos dentro del rango $0 < (ij) < k$ y denominarlos, por ejemplo, «trigo». Pero la pregunta que nos corta el paso es la siguiente: ¿qué ocurre cuando de lo que se trata es de calcular el valor de k para un conjunto de pixels y no sólo para un pixel? ¿Podría esta técnica diferenciar una imagen de agua a cero grados Celsius de otra donde se mostrase hielo a -1 grado Celsius? Haval! la respuesta es... no.

Sólo resta una posibilidad teórica, cual es particionar la imagen en regiones homogéneas, realizar una contextualización basada en una ventana de pixels y encontrar, mediante filtrado de ruido basado en estadística local, las variaciones espaciales y contextuales que se pretenden buscar. Finalmente, bastará que la salida de la red neuronal óptica sea tratada por un sistema experto que realice, mediante reglas de inferencia, la asignación a los conjuntos de pixels.

IMPLEMENTACIÓN FÍSICA DE LA ARQUITECTURA

A continuación se muestra una figura de la arquitectura física del sistema neuroóptico:

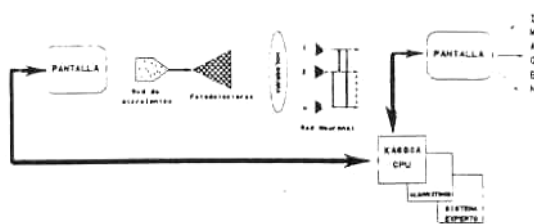


Figura 1. Arquitectura de la Red Neuroóptica

Los elementos principales son:

- la red de 30.000 microlentes en óptica binaria
- la red de fotodetectores
- la matriz holográfica de interconexión neuronal
- la red neuronal tri-estado
- la CPU K680A
- la unidad de visualización

La red de microlentes en óptica binaria, así como la red de fotodetectores, la matriz holográfica y la red neuronal ya se han descrito en apartados anteriores.

En cuanto al chip K680A se trata de la CPU donde se implementa el algoritmo de clasificación contextual y el sistema experto de asignación de etiquetas a los grupos de pixels, escrito en OPS5 y codificado en lenguaje máquina por razones de rendimiento.

Algoritmos residentes en la CPU

El algoritmo de clasificación contextual programado en ensamblador es el siguiente:

$$P_{kl(i,j)} = \frac{\text{Sigma-alef}}{\text{Sigma-beth}} \tag{6}$$

donde:

Sigma-alef = $W(i,j,u,v) d_{kl(u,v)}$ para (u,v) perteneciente a N

Sigma-beth = $W(i,j,u,v) dk_{(u,v)}$ para (u ,v) perteneciente a N

siendo:

N ventana de la cual se toman los datos de muestreo

(i,j) pixel central del bloque a procesar

(u,v) pixel de muestreo dentro de la ventana N

$d_{kl(u,v)}$ transición unidad desde la clase k a l

$dk_{(u,v)}$ transición unidad de la clase k

$W(i,j,u,v)$ función de sopesamiento que cuantifica la importancia del pixel (u ,v) respecto al pixel central (i,j)

Es evidente que si $W(i,j,u,v)$ es constante, entonces $P_{kl(i,j)}$ es invariante respecto de la posición, cual es el caso de la estimación de máxima probabilidad. Pero para suprimir la no-linearidad podemos considerar la función de sopesamiento exclusivamente en función del pixel (u,v), para lo cual basta con asignar valores positivos a esta función tal que su valor máximo ocurra cuando el pixel (u ,v) y el (i,j) sean el mismo pixel y mínimo cuando la distancia entre ambos pixels, dentro de la ventana, sea máxima, es decir:

$$W(u,v) = \frac{1}{D(u,v)} \tag{7}$$

siendo $D(u,v)$ la distancia entre el pixel (u,v) y el pixel central (i,j).

El siguiente algoritmo residente en la CPU es aquel que calcula la función de Sobel, que es el operador local de resaltamiento de bordes. El operador de Sobel se aproxima definiendo la vecindad de cada pixel en función de la ventana N del algoritmo de clasificación contextual. Generalmente se utiliza la típica matriz:

$$\begin{matrix} a_0 & a_1 & a_2 \\ a_7 & p & a_3 \\ a_6 & a_5 & a_4 \end{matrix}$$

tomando:

$$\begin{aligned} x &= (a_2+2a_3+a_4) - (a_0-2a_7+a_6) \\ y &= (a_0+2a_1+a_2) - (a_6+2a_5+a_4) \end{aligned} \tag{8}$$

$$\text{Sobel} = \text{abs}(x) + \text{abs}(y)$$

aunque esta ecuación es realmente una aproximación del valor real del operador Sobel, que es la raíz cuadrada de x al cuadrado más y al cuadrado.

La siguiente figura muestra una representación geométrica interna de la red neuronal donde, para un pixel central (i,j), se ha tomado una ventana N de 27 pixels. Los números situados sobre los pixels representan la función de sopesamiento respecto a dicho pixel central. Uno de los pixels representa el pixel de muestreo:

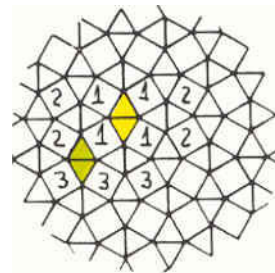


Figura 2. Distribución de pesos en la imagen.

La siguiente figura muestra la misma zona de la imagen con dos regiones hipotéticas marcadas, cuya frontera modifica el valor de la función de transición de 0 a 1:

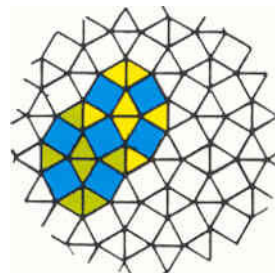


Figura 3. Detección de regiones homogéneas

CONCLUSIONES

La arquitectura aquí presentada es innovadora en dos aspectos. En primer lugar, la *algorítmica* utilizada trasciende al tratamiento clásico pixel-a-pixel, que es excesivamente local, sustituyéndose por un algoritmo nuevo que se apoya en elementos estadísticos semilocales condicionados por la función de distribución de energía *à lá Boltzmann*. En segundo lugar, dicha algorítmica es apropiada para su codificación en *arquitecturas neuronales ópticas*.

La sinergia entre la algorítmica, la arquitectura física neuronal y el enfoque puramente óptico del problema da como resultado una técnica potente, eficaz y rápida para tratar un problema usual en tratamiento de imágenes digitalizadas, cual es la corrección geométrica de dichas imágenes.

Desde el punto de vista del diseño, desarrollo y fabricación de una arquitectura semejante los elementos físicos están dentro del estado del arte, exceptuando la matriz de microlentes binarias. Probablemente el problema principal resida en el aprendizaje de la red, que requiere un enfoque específico no tratado en el presente artículo, así como en el diseño del *sistema experto*, cuyas reglas de producción son sólo válidas para los datos de entrada en los que se ha basado dicho aprendizaje.

BIBLIOGRAFIA

- CERDAN, R. (1987). On Neural Net Cortical Connectivity- Neurological Correlations For N-Net Learning Procedures, Position Paper, University Of Tarshiss, UoT Press.
- CERDAN, R. (1989). «Neurocomputing and Neural Nets Control Strategy», en Proceedings on Applied Artificial Intelligence, University of Tarshiss, UoT Press.
- MING-CHAUN, KAI-BOR YU (1982). «Spatial Varying Image Contextualization», en Methodologies for Intelligent Systems, Elsevier Publications.
- STROLL, Z.Z., SWARTZLANDER, E., ELDON, J. (1984). «VLSI for image rotation», en VLSI Signal Processing, IEEE Press.