

CLASIFICACIÓN DE CUBIERTAS EN IMÁGENES MULTIESPECTRALES LANDSAT-TM MEDIANTE REDES NEURONALES SOBRE PC

J. Martín, J. M. Torres, J. Peón, L. Fernandes.

Laboratorio de Teledetección y SIG, Dep. de Física Aplicada, Universidad de Vigo.

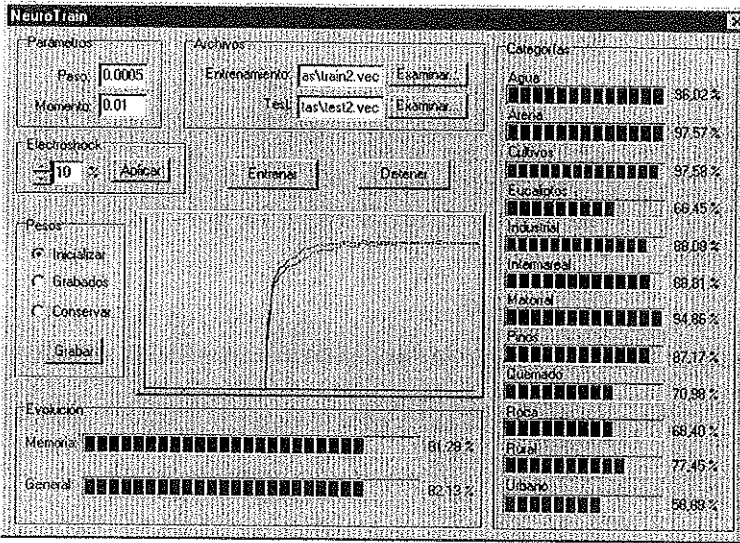


Fig. 1. Interfaz de usuario para el control y monitorización del entrenamiento de la red neuronal para la clasificación en doce clases de terreno.

Resumen: Al abordar la elaboración de mapas temáticos a partir de imágenes de teledetección provenientes del sensor Thematic Mapper del Landsat5, se escoge como opción idónea el uso de una red neuronal con cierta capacidad fuzzy como solución para la especial dificultad que plantean las características geográficas del territorio gallego. De antemano es de esperar de un clasificador de estas características una complejidad computacional elevada, que podría desaconsejar su implementación en un PC. Sin embargo, la espectacular evolución que éstos han experimentado en los últimos tiempos, así como su bajo coste y el elevado parque, que garantiza una gran compatibilidad, hacen recomendable el intentar implementar herramientas eficaces desarrolladas específicamente para este tipo de máquinas. Hemos comprobado cómo una adecuada selección de los medios, métodos y recursos permite obtener magníficos resultados en un PC de características medias, con prestaciones similares a las de una estación de trabajo de gama baja, pero a un coste mucho menor, tanto de plataforma como de desarrollo.

INTRODUCCIÓN

Hoy en día, cualquier aplicación de teledetección requiere de gran capacidad de computación que pueda con la ingente cantidad de datos que se manejan así como que realice de forma eficaz el elevado número de cálculos que éstas requieren, ya sea aritmética de enteros (habitual en tratamiento de imagen) como de punto flotante. Generalmente se tiende a confiar directamente a costosas estaciones de trabajo (aún cuando sean de gama baja) el grueso de estas tareas, relegando a los PC a meros terminales para trabajos menores o simple acceso a las capacidades de la estación. Esto quizá se deba a inercia de los tiempos en que los PC no se acercaban ni con mucho a las posibilidades de una estación de trabajo. Hoy en día las cosas han cambiado, y a la hora de equipar un laboratorio de teledetección, o al diseñar una aplicación, puede resultar muy conveniente

dirigir la mirada hacia los PC. En el primer caso, con el mismo presupuesto, el número de plataformas operativas se puede multiplicar por cuatro o más, y en el segundo el número de potenciales usuarios se multiplica por varios miles. La ventaja en coste es indiscutible, pero es lógico preguntarse si realmente los PC han alcanzado ya en prestaciones el extremo inferior de la gama de estaciones de trabajo, y para resolver la duda nada mejor que una aplicación de altos vuelos que implica tanto técnicas de procesado de imagen como cálculo complejo y manejo de grandes cantidades de datos, como es la elaboración de mapas temáticos a partir de imágenes Landsat TM mediante redes neuronales.

Cuando por primera vez nos enfrentamos a la

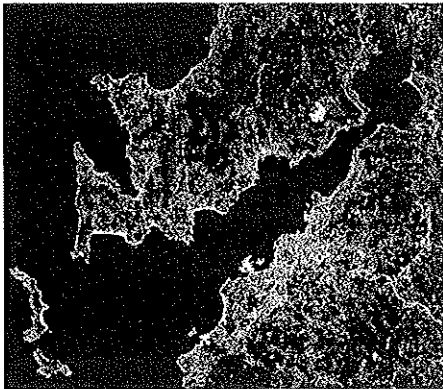


Fig. 2. La Ría de Vigo en las bandas en el visible (3, 2 y 1) de una imagen Landsat TM.

elaboración de mapas temáticos del territorio gallego, nos encontramos con una serie de problemas asociados a las singulares características geográficas de Galicia, como la desordenada orografía, la variada línea de costa, la dispersión de la población y el minifundismo llevado hasta el extremo del absurdo. Cada una de estas características aporta una serie de inconvenientes que requieren cuidadosas medidas correctoras y exigen un clasificador realmente selectivo y robusto. En su día desarrollamos un sistema [6] basado en redes neuronales siguiendo los trabajos de Bishof et al. [2] con especial consideración respecto a todas estas dificultades. El sistema fue desarrollado en una estación de trabajo Sun dotada de un procesador Sparc20, enteramente programado en C, y requirió de un elevado tiempo de aprendizaje de las interioridades de la interfaz de usuario (OpenWindows) y del uso de las herramientas de desarrollo (X11 y diversas librerías complementarias), así como de desarrollo propiamente dicho. Aparte de los resultados finales, muy satisfactorios, con

porcentajes de acierto en la clasificación superiores al 92% [6], las prestaciones computacionales se pueden considerar francamente buenas, incluyendo tiempos de entrenamiento de la red de en torno a una o dos horas, y tiempos de clasificación de una imagen de alrededor de 2,5 millones de píxeles (en cada una de las siete bandas) de unas cinco horas. Pero el uso del sistema requiere de cierto grado de conocimiento del tipo concreto de estación de trabajo por parte del usuario; además la portabilidad a otra máquina requiere la recompilación de todos los códigos fuente, así como la adaptación de ciertas partes del código, dependiendo del sistema de destino. Más aún, para una mejor prestación del sistema, sería conveniente aumentar en 32 Mbytes la memoria de la estación de trabajo, y sólo esa ampliación tiene un coste superior al de un PC de gama alta completo.

Por todo ello se planteó el desarrollo del sistema para PC adaptando los planteamientos y aprovechándonos de la experiencia adquirida. Los resultados obtenidos, así como el sistema en sí, se describen a continuación.

DESCRIPCIÓN

El sistema consta de tres partes bien diferenciadas. En primer lugar, una herramienta de tratamiento y visualización de imágenes que permite su preprocesado, etiquetado, georreferenciado y preparación para convertir los niveles digitales de las siete bandas de una imagen Landsat TM en vectores adecuados como entrada a una red neuronal.

El etiquetado de las imágenes es una parte fundamental del proceso, pues consiste en la selección en la imagen de puntos de cada uno de los tipos de terreno que servirán para el entrenamiento de la red y la validación de sus resultados. Si la información suministrada a la red es errónea, como es lógico sus prestaciones se verán sensiblemente mermadas. Así pues, los errores en el etiquetado constituyen una importante fuente de ruido que es conveniente minimizar dotando al operador de medios adecuados que faciliten su trabajo. Para ello se incluyen capacidades de manipulación de los histogramas, visualización de combinaciones de bandas adecuadas y georreferenciado de las imágenes. Asimismo se incorporó al sistema la posibilidad de un *feedback* desde el entrenamiento de la red hacia esta herramienta, que permite visualizar las etiquetas que estén resultando especialmente difíciles de asimilar a la red, para facilitar la detección de etiquetas erróneas y una mejor comprensión del funcionamiento correcto o no del sistema en cada caso.

A continuación del proceso de etiquetado, sin duda el más laborioso para el operador, entra en

juego la red neuronal. Una red neuronal es un algoritmo un tanto especial desde el punto de vista computacional. Su nombre viene de la semejanza con el sistema nervioso central (SNC), pues los primeros desarrollos surgieron como un intento de modelar el funcionamiento del SNC más que como una nueva algoritmia [8]. Una red neuronal de tipo perceptrón multicapa (MLP) consta de un número de nodos (o neuronas) ordenados en capas consecutivas de forma que los de cada capa se conectan con los de la siguiente mediante unas conexiones que se caracterizan por su peso, que es un factor multiplicativo que determina la influencia de la salida del nodo de la capa anterior en la entrada del nodo de la capa siguiente [5]. La salida de cada nodo se obtiene como resultado de aplicar una función no lineal [3], llamada de activación, a la suma ponderada de las entradas al nodo. Esta salida sirve de entrada a las capas consecutivas hasta alcanzar la última, cuyas salidas se consideran la salida de la red [5].

Se demuestra matemáticamente que un MLP de tres capas (entrada, oculta y de salida) es un aproximador universal, es decir, es capaz de comportarse exactamente igual que cualquier función matemática por compleja que sea, aún cuando ello implique un número de nodos infinito [4]. La ventaja de las redes neuronales es su condición de cajas negras, de forma que no necesitamos ninguna expresión analítica para el sistema que queremos simular: nos basta con un conjunto de entradas y sus salidas correspondientes para el aprendizaje de la red, que consiste en la adaptación de los pesos hasta lograr que se comporte como el propio sistema, o lo más aproximadamente posible. Existen para ello diversos algoritmos [7], aunque todos basados en un mismo principio, la retropropagación del error cometido por la red en cada par de vectores de entrada y salida. El más popular, y a nuestro entender el más robusto, es el denominado Backpropagation [10][9], que es el que hemos adoptado para nuestro sistema.

Para la clasificación de la imagen, las entradas son los vectores espectrales de cada píxel, es decir, las siete muestras de la firma espectral que obtiene el sensor TM, y la salida es la clase a la que pertenece dicho píxel. Para ello, nuestra red consta de tantos nodos en la capa de salida como clases quiera distinguir el operador (12 en el ejemplo de la figura 3), y el objetivo del entrenamiento es que el nodo correspondiente a la clase correcta para cada píxel presente la máxima salida.

En cuanto a las entradas, resulta conveniente dar información a la red no sólo de la firma espectral de cada píxel, sino también de la de sus píxeles vecinos, lo cual resulta de máxima importancia al caracterizar clases como las costeras y urbanas dispersas. Para ello obtenemos lo que denomina-

mos las bandas de textura, que resultan de aplicar un sencillo cálculo a cada una de las bandas de la imagen: se obtiene para cada píxel la varianza de la matriz 5x5 de valores digitales de sus vecinos más próximos, ponderados en función de la distancia al píxel en cuestión, y a continuación se resta este valor al nivel digital del propio píxel. De este modo se obtienen siete nuevas "bandas" en las que se aprecia cierta agrupación de los píxeles por clases de terreno, que es el tipo de información adicional que nos interesa. Así pues, el vector de entrada consta de 14 componentes para cada píxel. Cada una de estas componentes sirve de entrada a cada uno de los 14 nodos de la capa de entrada. Obviamente, en caso de utilizar una imagen de otro sensor, por ejemplo de SPOT, o las bandas de varios sensores combinados, este número sería diferente. El número de nodos en la capa intermedia, u oculta, queda a elección del operador, según las prestaciones que vaya obteniendo en cada caso concreto [1].

Durante el entrenamiento de la red se dispone de un control total del proceso, con monitorización continua en forma de gráficas, barras de progresión y valores numéricos de porcentaje de acierto en la clasificación tanto del conjunto de entrenamiento como del de validación, así como de cada clase separadamente (figura 1). Se pueden cambiar en cualquier momento los parámetros de aprendizaje (el tamaño de paso y el factor de momento [9]) para conseguir la mejor forma en la curva de error, sin detener el proceso, así como alterar los pesos en la red para una mejor búsqueda en la superficie de error para paliar el problema de los mínimos locales [6].

Una vez que se dispone de la red más idónea para la clasificación, se alcanza la última etapa, consistente en la clasificación de la imagen completa, que consiste en ir pasando por la red píxel a píxel las bandas de imagen, incluidas las texturas, e ir obteniendo la clase correspondiente para cada píxel. Pero no sólo se guarda esa información, sino que se aprovechan las especiales características de la red para una mejor clasificación: se almacena para cada píxel la clase que da mayor salida, junto con el nivel de ésta, así como la siguiente clase y su nivel correspondiente, como segunda clase más probable. Los niveles de cada una de las clases se emplean entonces como un valor de confianza en la clasificación de cada píxel, permitiendo luego un suavizado del mapa temático mucho más preciso que un simple filtro de mayoría.

El suavizado del mapa temático es necesario para una verdadera utilidad del producto resultante, pues de él se espera una clara agrupación del terreno en zonas de tipo de cubierta predominante; una simple clasificación píxel a píxel resulta en un mapa desordenadamente puntuado de diferentes

clases mezcladas por todo el terreno. Generalmente se recurre a un filtro con un tamaño de ventana adecuado, que asigna a cada píxel la clase que es mayoritaria en su entorno. De esta forma se está perdiendo, a nuestro entender, una valiosa información. Nuestro enfoque *fuzzy* implica la consideración de las dos clases más probables de cada uno de los píxeles vecinos junto con su nivel de confianza, observándose unos resultados francamente superiores a los del filtro de mayoría.



Fig. 3. Mapa temático obtenido de la clasificación de la imagen de la figura 2. Se diferencian las 12 clases que se observan a la derecha de la fig. 1.

RESULTADOS Y CONCLUSIONES

El sistema se realizó empleando herramientas de desarrollo que hoy en día son el medio más popular para conseguir aplicaciones profesionales de forma rápida y eficaz en PC, las llamadas herramientas de desarrollo rápido de aplicaciones (RAD), que se pueden considerar la generación siguiente a las herramientas visuales. Estas herramientas, de fácil aprendizaje y sin embargo gran potencia, superior incluso a los compiladores "desnudos" habituales, suponen una gran productividad a bajo coste, como hemos podido comprobar durante el desarrollo de este sistema: el tiempo de desarrollo ha pasado de medirse en meses en la estación de trabajo a medirse en semanas en el PC. Los resultados, además, son mucho más profesionales en lo que a interfaz de usuario se refiere. El operador enseguida se halla cómodo con el sistema al encontrarse en un entorno (Windows 95) que le es familiar e incluso amigable.

Pero no sólo eso, sino que nos hemos encontrado con unas prestaciones superiores a las de la estación de trabajo: los tiempos de entrenamiento de la red han bajado hasta el orden de pocos minutos, y la clasificación de una imagen de 2,5 millo-

nes de píxeles con una red 14-30-12 tarda exactamente 83 minutos en un Pentium a 100 MHz con 16 Mb de RAM, que hoy en día se puede considerar un PC modesto. En un Pentium a 166 MHz con 32 Mb de RAM dichos tiempos caen mucho más, realizando la clasificación de una imagen completa en menos de una hora.

De todo ello podemos concluir que el PC es una opción perfectamente viable a la hora de escoger plataformas para este tipo de aplicaciones, e incluso más recomendable que otro tipo de máquinas en principio más sofisticadas y desde luego mucho más costosas. Sólo encontramos un inconveniente, que es la flagrante falta de robustez de Windows95 como sistema operativo, aunque parece en vías de ser solucionado ante la aparición de su sucesor, que combinaría la amigabilidad de Windows 95 con la robustez de Windows NT.

REFERENCIAS

- [1] E.B. Baum, E.Haussler, 1989, "What Net Size gives Valid Generalization?", *Neural Computation*, nº1, págs. 154 a 160.
- [2] H. Bischof, W. Schneider, A.J. Pinz, 1992, "Multispectral Classification of Landsat Images Using Neural Networks", *IEEE Transactions on Geosciences and Remote Sensing*, vol. 30, nº3, págs. 482 a 490.
- [3] G. Cybenko, 1989, "Approximation by Superpositions of a Sigmoidal Function", *Mathematics of Control, Signals and Systems*, vol.2.
- [4] K. Funahashi, 1989, "On the Approximate Realization of Continuous Mappings by Neural Networks", *Neural Networks*, págs. 183 a 192.
- [5] D.R. Hush, B.G. Horne, 1993, "Progress in Supervised Neural Networks: What's New Since Lippmann?", *IEEE Signal Processing Magazine*, nº1, págs. 8 a 39.
- [6] J. Martín, 1996, "Clasificación de cubiertas en imágenes multiespectrales Landsat TM", Proyecto fin de carrera, ETSI Telecomunicación, Universidad de Vigo.
- [7] M. Riedmiller, 1994, "Advanced Supervised Learning in Multilayer Perceptrons", *Journal of Computer Standards and Interfaces, Special Issue on Neural Networks*.
- [8] F. Rosenblatt, 1961, *Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms*, Washington D.C., Spartan Books.
- [9] D.E. Rumelhart, G.E. Hinton, R.J. Williams, "Learning Internal Representations by Error Propagation", *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, vol.1, cap. 8, págs 318 a 362, Cambridge, MIT Press.
- [10] P.J. Verbois, 1974, *Beyond Regression: New Tools for Prediction and Analysis in the Behavior Science*, Tesis Doctoral, Harvard University.