



Influencia del tamaño de la tesela en el reconocimiento y extracción de caminos en ortoimágenes

Comparación y análisis de resultados en función de tres tamaños de teselas en tareas de Aprendizaje Profundo.

N.º del tema de las jornadas: 1. Herramientas y tecnologías

Resumen:

Las tareas de aprendizaje profundo, como son el reconocimiento o la segmentación semántica, utilizan redes neuronales artificiales con arquitecturas que se van actualizando y perfeccionando con el paso del tiempo. Estas arquitecturas cuentan con un gran número de parámetros que consumen recursos y tiempo de procesamiento. El número de parámetros varía proporcionalmente con el tamaño de las imágenes y de la red escogida. Por tanto, surge la necesidad de saber cómo modificar la arquitectura o características del entrenamiento sin que ello afecte a la calidad de los resultados.

En esta ponencia, se presentan algunos de los desafíos que han surgido para el análisis de los resultados en función del tamaño de las teselas partiendo de la preparación del conjunto de datos de estudio. El análisis de resultados se realizó en Excel tras haber realizado las tareas de aprendizaje profundo en un entorno Python, apoyado en librerías como Keras, TensorFlow y librerías de CUDA (para las tarjetas gráficas).

En el marco del proyecto SROADEx (PID2020-116448GB-I00) y a partir de los tiles generados de 16 hojas del MTN50, se prepararon los directorios de trabajo para cada tamaño de las teselas (256x256, 512x512 y 1024x1024 píxeles). La creación de los conjuntos de datos se realizó mediante un script en Python que dividía el conjunto de imágenes en tres grupos: entrenamiento, validación y test. Una vez preparados, se pasaba a la fase de entrenamiento, en la cual surgió la primera problemática pues, además del tiempo que empleaba cada red para realizar su respectiva tarea, se observó que durante la ejecución de dichos procesos se agotaba la memoria RAM integrada en la tarjeta gráfica. Como se dispone de un servidor que contaba con cuatro GPUs Tesla V100, se ha hecho uso de las librerías de NVIDIA-CUDA para paralelizar el procesamiento de las redes, siendo capaces de realizar distintas tareas simultáneamente.

Solventado este punto, se llegó al segundo desafío, ya que en tamaños superiores distintos a 256x256 píxeles debían de realizarse modificaciones en los *batch size* (número de imágenes que recorren la red en una iteración) ya que, con la misma configuración de parámetros, en 512x512 o 1024x1024 las redes se saturaban.



La solución que mejor funcionó para las tareas de segmentación semántica y reconocimiento en 512, y segmentación semántica en 1024x1024, se basó en reducir el *batch-size* o número de imágenes que recorrerán la red en cuestión durante una iteración. Esta reducción consistió en dividir entre cuatro dicho batch para las redes de 512, pues en una tesela de este tamaño cabrían cuatro imágenes del tamaño de tesela más bajo con el que se trabaja. El procedimiento es análogo para el tamaño superior.

Si bien esta solución parecía funcionar para todos los casos, en la última tarea, la de reconocimiento de imágenes en 1024x1024, no funcionó por falta de memoria. Se propusieron dos alternativas, que fueron la de eliminar una de las últimas capas de la arquitectura, que falló rápidamente en las pruebas, y la de reducir un 25% el número de neuronas artificiales con el que trabajaría la red. Esta última finalmente se pudo ejecutar sin problema y devolvió resultados que concordaban con los obtenidos en otras arquitecturas.

Palabras claves

Reconocimiento, Segmentación Semántica, Conjuntos de datos, Entrenamiento, Aprendizaje profundo, Viales, Ortoimágenes

Autores

Víctor Fernández Valladolid

victor.fernandez.valladolid@alumnos.upm.es

Universidad Politécnica de Madrid

Francisco Serradilla

fserra@eui.upm.es

Universidad Politécnica de Madrid

Pablo Barreira González

pbarreira@mitma.es

D.G. Instituto Geográfico Nacional

Cira Calimanut-Ionut

ionut.cira@upm.es

Universidad Politécnica de Madrid