

DEEP LEARNING

El *Deep Learning* es una forma especializada de aprendizaje automático (*machine learning*). En estos modelos de redes neuronales el procesamiento se realiza en un número alto de capas de neuronas o elementos de proceso. De ahí surge su denominación (*deep*), por la mayor profundidad a la que se realiza el aprendizaje, que ahora consiste en ajustar un número mucho mayor de parámetros. Los modelos de aprendizaje profundo se denominan redes neuronales profundas (*deep nets*) frente a las redes neuronales con sólo una o dos capas ocultas (*shallow nets*).

Los modelos de aprendizaje profundo requieren grandes cantidades de datos de entrenamiento y un alto poder de procesamiento, ninguno de los cuales estaba fácilmente disponible para los programadores hasta hace poco en la que los algoritmos y el hardware son capaces de tratar grandes cantidades de datos (*big data*) y de realizar computación en la nube o en la niebla (*cloud/fog computing*). Los modelos de *Deep Learning* requieren grandes cantidades de datos etiquetados.

Alexnet es una red preentrenada para clasificar 1000 objetos y disponible para la comunidad. AlexNet es una red neuronal convolucional, diseñada por Alex Krizhevsky, y publicada con Ilya Sutskever y Geoffrey Hinton (<https://en.wikipedia.org/wiki/AlexNet>).

Enunciado

Se plantea realizar un programa que use la red ALEXNET y realice la clasificación de algún objeto en una instantánea capturada con la webcam.

Implementación

Deep_alexnet1.m

% Clasificación de una foto con ALEXNET

```
clear all;

% Conexión con la webcam
camera = webcam(1);

% Carga de la Deep Net preentrenada
nnet = alexnet;

preview(camera);

disp('Pulse una tecla para tomar una foto... ');
pause;

% Captura de una instantánea
picture = camera.snapshot;

% Reescalado de la imagen para alexnet
picture = imresize(picture,[227,227]);

% Clasificación con la red nnet
label = classify(nnet, picture);

% Mostrar la imagen
image(picture);
% Mostrar la etiqueta del objeto identificado
title(char(label));
drawnow;

clear camera;
```

