

# Visión por Computadora "Trabajo Final, Modelo Bi-linear "

2018

Sebastián Correa Echeverri

## 1 Introducción

Se propone como proyecto final de la asignatura Visión por computadora del Master en inteligencia artificial reconocimiento de formas e imágenes digital, implementar un modelo de identificación de modelos de automóviles usando modelos bi-lineal.

Las tareas de reconocimiento FINE-GRAINED generalmente implican discriminación entre categorías que tienen una estructura compartida pero difieren en formas sutiles, distinguiendo entre una "Toyota prius" y un "Toyota corola". Esto requiere el reconocimiento de atributos altamente localizados bajo los cambios de pose, punto de vista, iluminación y otros factores.

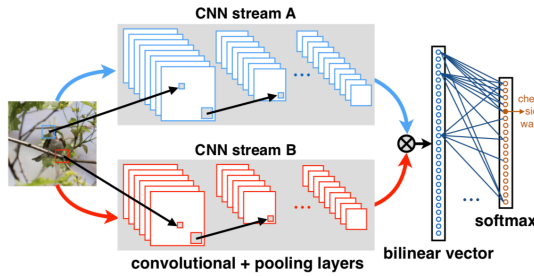


Ilustración 1 Imagen de clasificación usando una B-CNN

Un modelo bi-lineal para la clasificación de imágenes consiste en un cuádruple  $B = (f_A, f_B, P, C)$ . Donde  $f_A$  y  $f_B$  son funciones de función,  $P$  es una función de agrupamiento y  $C$  es una función de clasificación. Una función característica es un mapeo  $f: L \times I \rightarrow R^c \times D$  que toma una imagen  $I$  y una ubicación  $L$  y emite una característica de tamaño  $c \times D$ . Las salidas se combinan en cada ubicación usando el producto externo de la matriz, es decir, la combinación de características bilineales de  $f_A$  y  $f_B$  en una ubicación  $l$  dada por:

$$bilinear(l, I, f_A, f_B) = f_A(l, I)^T f_B(l, I)$$

Tanto  $f_A$  como  $f_B$  deben tener la misma dimensión de característica  $c$  para ser compatibles. [1]

### 1.1 Datos

Los datos fueron entregados para el proyecto y cuenta con un total de 1575 imágenes de 20 diferentes modelos de coches, distribuidas en 791 imágenes para el entrenamiento y 784 imágenes para el conjunto de test.

Cada imágenes están en escala de colores (RGB) y son de 250 por 250 píxeles.

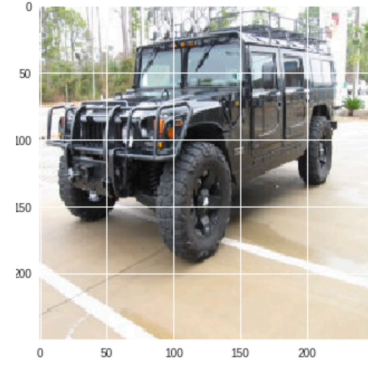


Ilustración 2 Imagen de ejemplo base de datos

## 2 Procedimiento

Para la construcción del modelo, se pre cargará una red neuronal VGG-16 pre entrenada con el data set 'ImageNet' para formar una red bi-lineal. Para esto se diseño una serie de experimentos. Entrenamiento inicial para toda la red incluyendo la VGG-16. Entrenamiento sin actualizar los pesos de la VGG-16 por el gradiente, solamente se entrenara la parte fully conected. Posterior a esto se entrenaran los pesos de la VGG-16 y se evaluara el procedimiento. Se realizaran pruebas sin y con learning rate aneling, diferentes optimizadores y learning rates, a si mismo como diferentes puntos de corte de la VGG-16.

Para todos los experimentos se aplico data-aumentation del 20% de movimiento horizontal y vertical, una rotación hasta de 20

grados, un zoom range 1.0 y 1.2, y un flip horizontal.

### 3 Resultados

A continuación se presentan los resultados de los procedimientos de la sección 2, inicialmente se realizaron pruebas con el entrenamiento completo de la red VGG-16 y los optimizadores Adam, SGD y RMSprop. Learning rate de 0.0001 y 0.001 con un total de 50 épocas.

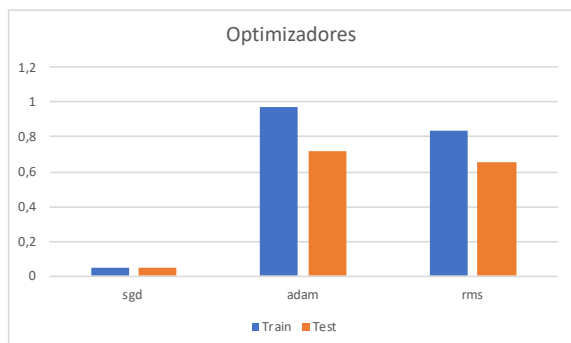


Ilustración 3 Gráfica resultado de ejecución del modelo con diferentes optimizadores.

Al realizar los diferentes se eligieron los optimizadores Adam y RMS, como se puede observar en Ilustración 3, el optimizador SGD con la configuración previamente mencionada no logra superar ni el 10% de accuracy.



Ilustración 4 Gráfica resultados entrenamiento con pesos congelados RMS y Adam.

Finalmente se puede observar en la Ilustración 4 que el entrenamiento de la red con el optimizador RMSprop presenta mejores resultados que el optimizador Adam.

Se realizaron experimentos seleccionando diferentes puntos de la VGG-16 como extractor de características pre entrenado. Se seleccionan los bloque 5 y 4 antes del

maxpooling de la arquitectura de la VGG-16 para obtener las características antes de realizar la multiplicación entre ambas redes. Se puede observar en la Ilustración 5 los resultados de los experimentos.

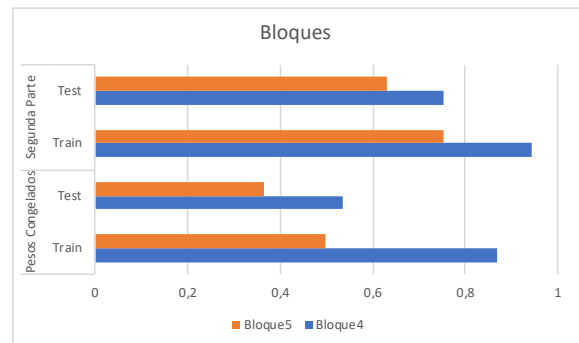


Ilustración 5 Gráfica resultado de la selección del bloque de partida.

### 4 Conclusiones

Se obtuvo resultados satisfactorios con las arquitecturas y experimentos propuestos. Debido a la limitada cantidad de datos podría decirse que no se obtuvo una mejor solución, una continuación del proyecto sería realizar el entrenamiento con una significativa cantidad de datos para el entrenamiento.

Es importante identificar un optimizador optimo para la tarea en la que se esta trabajando, RMSprop demostró un buen comportamiento fine tuning. Se podría mejorar los resultados del SGD si se realizar mas pruebas con sus diferentes parámetros, momentum y decay, debido a tiempo y disposición de equipos se decidió continuar con los optimizadores que presentaban un buen resultado.

### 5 Bibliografía

- [1] L. T.-Y. A. R. y S. M. , «Bilinear CNN Models for Fine-grained Visual Recognition,» *Transactions on Pattern Analysis and Machine Intelligence*, p. 14, 2017.