|  |
| --- |
|  |
| PRÁCTICA 1: DETECCIÓN DE OBJETOS |
| Visión artificial. Grado en Ingeniería Informática |
|  |
| **Jorge Aranda García**  **Jose Vicente Bañuls García** |
| **31/03/2018** |

|  |
| --- |
|  |

# Ejercicio 1: Detección de coches mediante punto de interés

El objetivo de este ejercicio es detectar coches en fotografías tomadas en distintas situaciones ,posiciones y calidad, a partir de unas fotografías de coches tomadas de frente y con un cierto nivel de calidad. Para ello hemos seguido un camino bastante progresivo. Primero comenzamos leyendo las imágenes de entrenamiento del algoritmo y reconociendo los puntos de interés y los descriptores de éstos, comprobando que este procedimiento se realizaba correctamente. Para ello se usan las funciones *"detect"* y *"compute"*  de la clase ORB.

A continuación estos puntos de interés se almacenan en una estructura de datos "*FLANN*", que es una librería pensada para realizar búsquedas aproximadas del vecino más cercano de forma rápida en grandes espacios.

Tras haber almacenado los puntos de interés de las imágenes de entrenamiento , se empieza a tratar las imágenes de test. Además de encontrar los puntos de interés y descriptores de estas imágenes hay que hacer algunas cosas más. Una vez encontrados los puntos de interés hay que buscar para cada uno de ellos, los vecinos más parecidos de entre los encontrados en las imágenes de entrenamiento(Al principio fueron 2 y al final 6). Una vez comparados los puntos, hay que crear una matriz de votación del tamaño que la imagen de test, para decidir dónde está el coche. Para ello, se coge el vector que va del punto de interés de la imagen de entrenamiento al centro de esa imagen y se suma al punto de interés. En la posición resultante se suma un voto. Tras haber realizado el mismo proceso con todos los puntos de interés, la posición de la matriz con mayor número (se saca con la función "*cv2.unravel\_index*") de votos sería el supuesto centro del coche. A pesar de que esta primera aproximación no es realista, ya empezamos a obtener los primeros resultados. Como se ha comentado anteriormente, esta aproximación no es realista puesto que para el cálculo de la matriz de votaciones no se ha tenido en cuenta ni el ángulo ni la escala de los puntos de interés. Por lo tanto ahora, al sumar el vector hay que escalar éste de tal forma que se corresponda con la escala del punto de interés de la imagen de test. Además, los puntos de interés tienen un ángulo que no tiene por qué coincidir, por lo que hay que orientar el vector en relación al ángulo del descriptor de la imagen de test. En este punto los resultado si que son realistas.