

Normativa de la práctica libre

SISTEMAS DE VISIÓN ARTIFICIAL / VISIÓN ARTIFICIAL

Tal y como consta en la Guía Docente, todos los alumnos de la asignatura debéis realizar un trabajo final en el que se tendrán que aplicar aspectos concretos del temario. Supondrá el **40% de la calificación final** de la asignatura.

Para la solicitud de la práctica se ha habilitado un entregable en el aula virtual, denominado **Solicitud Práctica Libre**, donde cada grupo deberá subir un listado con las propuestas de práctica libre incluidas en este documento, por orden de preferencia (ver Anexo I al final de este documento). El estudiante también podrá proponer una práctica libre diferente, en cuyo caso deberá enviar un resumen de la propuesta (ver Anexo II al final de este documento). Los profesores realizarán la asignación final, de modo que la distribución de las prácticas sea lo más equilibrada posible.

Para la entrega de la práctica se habilitará un entregable en el aula virtual, denominado **Práctica Libre**, donde cada grupo deberá subir un fichero titulado **SVA-Apellido1Nombre1_Apellido2Nombre2.zip** que contenga:

- a) la presentación (ppt y pdf) y
- b) todos los ficheros implicados en su implementación (.m, .mlx, imágenes, etc.).

La **presentación** tendrá el siguiente esquema:

1. Portada (debe indicar: nombre de la asignatura, título del trabajo, nombre completo del autor o autores, titulación, correo electrónico y fecha de la entrega).
2. Índice.
3. Introducción (una breve descripción de los objetivos del trabajo).
4. Desarrollo.
5. Conclusiones.
6. Bibliografía.

En la calificación de la práctica libre se tendrá en cuenta la calidad global del programa, la adecuación y originalidad de la solución aportada y la presentación.

Cada grupo defenderá su práctica final en alguna de las fechas que se indican más adelante mediante una **presentación**, cuya duración máxima será de **8 min**, y posteriormente se abrirá un turno de **preguntas** de no más de **5 min**.

Fechas:

- **Solicitud Práctica Libre.** Antes de las 23:59h del día **17 de noviembre de 2019**.
- **Entrega Práctica Libre:**
 - Primer turno: Antes de las 23:59h del día **18 de diciembre de 2019**.
 - Segundo turno: Antes de las 23:59h del día **9 de enero de 2020**.
- **Fechas de Presentación:**
 - Primer turno: **19 y 20 de diciembre de 2019** de 15:00 a 17:00h Lab. OL8 y OL11. Los alumnos de los dos grupos (martes y jueves/lunes y viernes) podrán presentar indistintamente en cualquiera de las dos sesiones. Más adelante solicitaremos que nos indiquéis vuestras preferencias/necesidades en cuanto a fecha y hora de presentación para poder organizar los horarios.
 - Segundo turno: **10 de enero de 2020** en la hora y lugar indicado por la Escuela.

La entrega y presentación de la práctica en el primer turno se considerará una mejora, multiplicándose su nota por 1.1.

Las tareas a realizar deben ser:

- a) Experimentar con los ejemplos proporcionados por Matlab y con otros parecidos.
- b) Explicar lo más clara y detalladamente posible cada uno de los pasos involucrados en cada algoritmo, explicando con más detalle todas aquellas funciones que considere más importantes.
- c) Intentar realizar modificaciones y explicar en qué consisten los cambios.

Trabajos propuestos: (Nota: Se aconseja tener instalada la última versión: **Matlab 2019b)**

1. [Automatically Detect and Recognize Text in Natural Images](#). Este ejemplo muestra cómo detectar regiones en una imagen que contienen texto. Pruébalo con más imágenes, con matrículas de coches, etc. Opcional: una vez detecte el texto, conviértalo en voz (utilizando Matlab, google, herramientas online, etc.).
2. [Digit Classification Using HOG Features](#). Este ejemplo muestra cómo clasificar los dígitos usando características HOG y un clasificador SVM multiclase. Pruébalo con más imágenes y pruebe su validez también con letras.
3. [Recognize Text Using Optical Character Recognition \(OCR\)](#). Este ejemplo muestra cómo utilizar la función OCR del Computer Vision System Toolbox™ para realizar el reconocimiento óptico de caracteres. Pruébalo con más imágenes. Opcional: una vez detecte el texto, conviértalo en voz (utilizando Matlab, google, herramientas online, etc.).
4. [Face Detection and Tracking Using the KLT Algorithm](#). Este ejemplo muestra cómo detectar y seguir automáticamente una cara utilizando local features. Por defecto, el detector está configurado para detectar caras, pero se puede utilizar para detectar otros tipos de objetos. Intente modificarlo para que detecte otro tipo de objetos.
5. [Motion-Based Multiple Object Tracking](#). Este ejemplo muestra cómo realizar la detección automática y el seguimiento basado en el movimiento de objetos en movimiento en un vídeo desde una cámara fija. Grabe una secuencia en la que aparezca alguna persona en movimiento y aplique la misma técnica.
6. [Feature Based Panoramic Image Stitching](#). El ejemplo muestra cómo crear automáticamente una imagen panorámica. Utilice otros detectores y descriptores distintos a SURF indicando cuál funciona mejor. Utilice imágenes propias para crear una panorámica.
7. [Measuring Planar Objects with a Calibrated Camera](#). Este ejemplo muestra cómo medir el diámetro de las monedas en unidades “del mundo” usando una sola cámara calibrada. Calibre una webcam propia y mida otro objeto distinto.
8. [Detecting Cars Using Gaussian Mixture Models](#). Este ejemplo muestra cómo detectar y contar coches en una secuencia de vídeo utilizando un detector de primer plano basado en modelos de mezcla gaussianos (GMM). Grabe una secuencia en la que aparezca algún objeto o persona en movimiento y aplique la misma técnica.
9. [Tracking Pedestrians from a Moving Car](#). Este ejemplo muestra cómo seguir a los peatones utilizando una cámara montada en un coche en movimiento. Grabe una secuencia en la que aparezca alguna persona en movimiento y aplique la misma técnica.
10. [Train a Deep Learning Vehicle Detector](#). Este ejemplo muestra cómo entrenar un detector de vehículos basado en la visión utilizando el aprendizaje profundo. Pruébalo con otras imágenes de coches.
11. [Object Detection Using Deep Learning](#). Este ejemplo muestra cómo entrenar un detector de objetos utilizando el aprendizaje profundo y R-CNN (Regiones con redes neuronales convolucionales). Pruébalo con objetivos diferentes a la señal de stop.

12. [Semantic Segmentation Using Deep Learning](#). Este ejemplo muestra cómo entrenar una red de segmentación semántica utilizando el aprendizaje profundo. Utilice nuevas imágenes que contengan las clases más representadas en la red.

ANEXO I

Instrucciones para la solicitud de una de las prácticas libres propuestas:

- Indique el grupo de laboratorio, el puesto y los nombres y apellidos de los miembros del grupo.
- Indique con números del 1 al 12 el orden de preferencia de las prácticas, siendo el **1** la práctica con **mayor preferencia** y el 12 la de menor preferencia.

Grupo: Puesto

Apellidos Nombre:.....

Apellidos Nombre:.....

| Orden de Preferencia | Nombre de la Práctica Libre |
|----------------------|--|
| | Automatically Detect and Recognize Text in Natural Images. |
| | Digit Classification Using HOG Features. |
| | Recognize Text Using Optical Character Recognition (OCR). |
| | Face Detection and Tracking Using the KLT Algorithm. |
| | Motion-Based Multiple Object Tracking. |
| | Feature Based Panoramic Image Stitching. |
| | Measuring Planar Objects with a Calibrated Camera. |
| | Detecting Cars Using Gaussian Mixture Models. |
| | Tracking Pedestrians from a Moving Car. |
| | Train a Deep Learning Vehicle Detector. |
| | Object Detection Using Deep Learning. |
| | Semantic Segmentation Using Deep Learning. |

ANEXO II

Instrucciones para la propuesta de una nueva práctica libre:

- Indique el grupo de laboratorio, el puesto y los nombres y apellidos de los miembros del grupo.
- Breve resumen de la propuesta.

Grupo: Puesto

Apellidos Nombre:.....

Apellidos Nombre:.....

Resumen de la propuesta: