Programa de Visión artificial

|  |  |
| --- | --- |
| Carrera | * Ingeniería informática * Ingeniería industrial * Departamento de electrónica * Materia optativa |
| Plan de estudios | * PE10 informática, PE17 industrial |
| Revisión | * 2020 |
| Carga horaria | * 64 hs |
| Materias correlativas | * Álgebra II |

# Objetivos

Generar competencias en la aplicación y desarrollo de algoritmos de procesamiento de imágenes y visión artificial (computer vision). El alumno deberá poder desarrollar código con soluciones de visión artificial para problemas concretos como:

* Reconocimiento de patentes de automotores
* Reconocimiento de rostros
* Posicionamiento 3D

# Temario Analítico

## Introducción

Definición de imagen, procesamiento de imagen y visión artificial. Presentación de soluciones que muestran el estado del arte: Visual SLAM, Reconstrucción 3D, etc. Presentación de las primitivas clásicas de procesamiento de imagen: cambio de espacio de color, cambio de resolución, brillo, contraste, filtrado por colores, histograma de color, etc. Presentación de las primitivas clásicas de visión artificial: kernel, detección de bordes, vectorización, detección de esquinas, descriptores, etc.

## Biblioteca OpenCV

Historia de la biblioteca. Ventajas y limitaciones de cada lenguaje: javascript, python, java, c++. Compilado de OpenCV en Linux. El objeto Mat y la representación de imágenes en memoria. Objetos Scalar y vec3. Imágenes y videos. La interfaz del desarrollador de visión artificial: mostrar una imagen en una ventana, controlar parámetros con barra deslizante.

## Procesamiento de imágenes básico

Píxel y canal de color. Cambio de espacio de color. Brillo y contraste. Utilidades de dibujo sobre imágenes: líneas, marcas, figuras geométricas, texto. Anotaciones.

## Histogramas

Histograma de una imagen. Ecualización del histograma: balance automático. Aplicaciones de la comparación de histogramas. Backprojection.

## Imágenes binarias

Umbral. Operaciones morfológicas y su uso para eliminar ruido y para distinguir regiones. Componentes conectados. Contornos. Vectorización de contornos. Aproximación por polígonos. Objetos para dibujar contornos vectorizados. Momentos. Invariantes. Detección de formas.

## Machine learning

Regresión. Clasificación. Clustering. KNN. Decision tree. SVM. Entrenamiento de un clasificador de formas.

## Kernel

Filtros espaciales. Convolución de kernel. Difuminado. Gradiente. Filtro de Sobel. Varios kernels y sus características. Detección de bordes y líneas. Canny edge detector. Transformada de Hough: rectas, segmentos, circunferencias. Pirámides. Deblur.

## Segmentación

Segmentación: por colores, semántica. Algoritmo watershed. Algoritmo GrabCut.

## Análisis de movimiento

Flujo óptico. Flujo óptico denso y disperso. Tracking. Background subtraction.

## Puntos singulares y descriptores

Detección de puntos singulares (feature detection): FAST y otros algoritmos. Subpixel accuracy. Extracción de descriptores. Distancia. Invarianza: descriptores invariantes a la rotación y al escalamiento. Uso de pirámides para invarianza de escala. Uso de orientadores para invarianza de rotación. Descriptores vectorizados vs descriptores binarios. LBP. Uniform LBP. BRIEF. ORB. Macheo. Stitching. Reconocimiento de escenas. Bag of Words (BoW). Bag of Binary Words. Machine learning para la generación del diccionario de BoW. Reconocimiento de escenas con BoW.

## Calibración de cámara

Modelo de cámara estenopeica (pinhole camera) y epipolar (fisheye lens). Parámetros intrínsecos. Distorsiones radiales y tangenciales: coeficientes. Calibración de cámara. Antidistorsión de imagen y de puntos singulares. Estado del arte.

## Transformaciones geométricas

Cambio de tamaño. Pirámides. Recorte. Mapeo y warp. Transformaciones 2D: traslación, escala, rotación, transformación afín. Geometría proyectiva. Transformación afín y de perspectiva. Coordenadas homogéneas. Espacios proyectivos. Homografías.

## Métodos numéricos

M-estimators. Minimización por Gauss Newton y Levenberg Marquardt. Estimadores robustos. Métodos robustos: RANSAC, LMEDS. PROSAC y RHO para homografías.

## Geometría epipolar

Estereoscopía. Parámetros extrínsecos. Estructura. Matriz esencial. Matriz fundamental. Relación entre ambas matrices. Rectificación de imágenes estereoscópicas. Estructura a partir de estas matrices.

## Perspective N-point

Homografía. Rectificación de códigos QR. Nube de puntos: densa y espaciada. DLT: direct linear transformation. PNP: Perspective N-point algorithms. Algoritmo de Nister de los 5 puntos, algoritmo de los 8 puntos. Estado del arte. Algoritmos para relevamiento denso de escenas 3D: Remode.

## Structure from motion

SfM: estructura a partir del movimiento. VO: visual odometry. Visual SLAM: LSD-SLAM, ORB-SLAM. Estado del arte.

## Deep neural networks

Detección de objetos. Detección y reconocimiento de rostros. Facial landmarks.

## Reconocimiento óptico de caracteres

Reconocimiento de texto. Reconocimiento de códigos QR y de barras.

# Bibliografía

## Básica

* <https://docs.opencv.org/master/index.html>
* [www.PyImageSearch.com](http://www.pyimagesearch.com)
* [www.learnopencv.com](http://www.learnopencv.com)
* Learning OpenCV 3: Computer Vision in C++ with the OpenCV Library, Gary Bradski, Adrian Kaehler, Ed. O’Reilly, 2016
* Computer Vision: Algorithms and Applications, Richard Szeliski, 2010

## Consulta

### Libros

* Learning OpenCV 3 Computer Vision with Python, 2nd Edition
* Mastering Opencv with practical computer vision projects, 2012
* Multiple view geometry in computer vision, Hartley & Sizzerman, 2002