



Visión Artificial
Instalación Python 3.11+ OpenCV
Semestre 2024-01



Muchos sistemas operativos incluyen por defecto una distribución de Python. Sin embargo, se recomienda instalar la distribución Anaconda, la cual incluye la mayoría de los paquetes necesarios para el desarrollo de esta clase.

Para el desarrollo de este curso se hará uso de la distribución para Python3 de Anaconda, llamada Anaconda3.

1. INSTALACIÓN EN WINDOWS

Se accede a la distribución de anaconda desde el siguiente enlace:

<https://www.anaconda.com/download/>

Free Download

Everything you need to get started in data science on your workstation.

- ✓ Free distribution install
- ✓ Thousands of the most fundamental DS, AI, and ML packages
- ✓ Manage packages and environments from desktop application
- ✓ Deploy across hardware and software platforms

 Download

Get Additional Installers



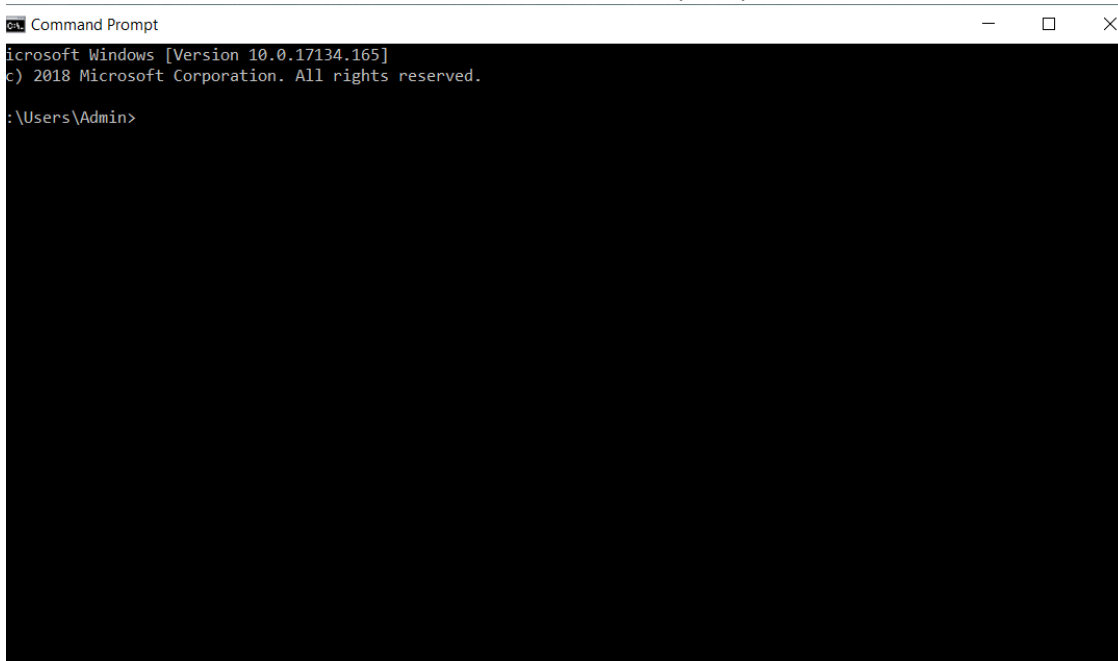
Una vez dentro, hacemos clic en la opción que cumpla su sistema operativo. Si su ordenador no es de 64 bits, pueden usar el instalador de 32. Una vez descargado el archivo, procedemos a instalarlo como cualquier `.exe`.

Para manipular los comandos que nos brinda anaconda debemos de abrir la aplicación llamada "Anaconda prompt". Existe otra manera en donde directamente añadimos anaconda al PATH en windows, pero esto podría crear conflicto con otras aplicaciones.

2. CREANDO UN AMBIENTE E INSTALANDO OPENCV.

Instalar OpenCV es muy simple, pero es recomendable hacerlo en un ambiente aislado (*environment*) a fin de no crear conflictos en nuestra distribución de Python base. Para ello seguimos los siguientes pasos:

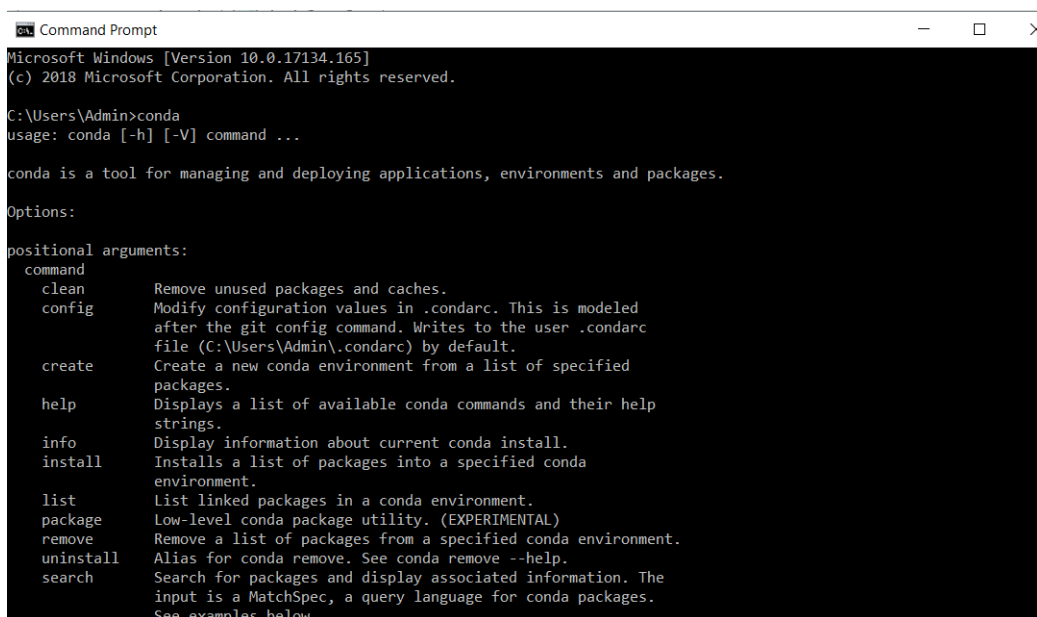
- Abrimos una ventana de línea de comandos (cmd).



```
Command Prompt
Microsoft Windows [Version 10.0.17134.165]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Admin>
```

- Escribimos el comando `conda` para verificar que nuestra distribución de Anaconda funcione correctamente.



```
Command Prompt
Microsoft Windows [Version 10.0.17134.165]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Admin>conda
usage: conda [-h] [-V] command ...

conda is a tool for managing and deploying applications, environments and packages.

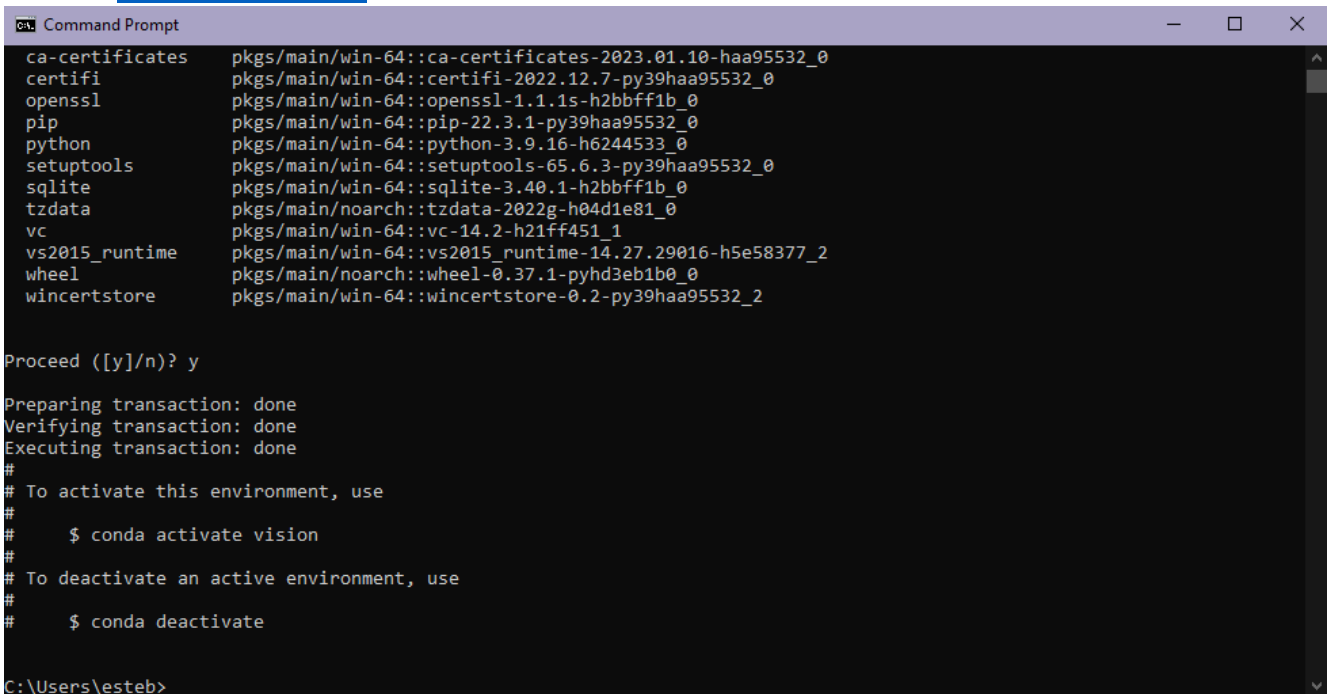
Options:
positional arguments:
  command
    clean      Remove unused packages and caches.
    config     Modify configuration values in .condarc. This is modeled
               after the git config command. Writes to the user .condarc
               file (C:\Users\Admin\condarc) by default.
    create     Create a new conda environment from a list of specified
               packages.
    help       Displays a list of available conda commands and their help
               strings.
    info       Display information about current conda install.
    install    Installs a list of packages into a specified conda
               environment.
    list       List linked packages in a conda environment.
    package    Low-level conda package utility. (EXPERIMENTAL)
    remove     Remove a list of packages from a specified conda environment.
    uninstall  Alias for conda remove. See conda remove --help.
    search     Search for packages and display associated information. The
               input is a MatchSpec, a query language for conda packages.
               See examples below.
```

- Creamos un nuevo ambiente con Python 3.11 utilizando la siguiente instrucción desde el ~~terminal~~ **cmd**

```
conda create --name vision python=3.11.5
```

ATENCIÓN: Los corchetes son para indicar que en esa posición se coloca el nombre que deseamos para nuestro ambiente de Anaconda. Deben omitirse al momento de ejecutar el comando. Asimismo, es recomendable evitar el uso de tildes y otros caracteres especiales, una excepción común cuando se tienen nombres compuestos por varias palabras es el carácter *underscore* (`_`). Ej: vision_artificial.

NOTA: Existen otras opciones que podemos utilizar al momento de crear un ambiente, para saber cuáles son, pueden acceder a la documentación en el siguiente enlace: <https://conda.io/docs/user-guide/tasks/manage-environments.html>



```

C:\Users\esteb>conda create --name vision python=3.11.5

pkgs/main/win-64::ca-certificates-2023.01.10-haa95532_0
pkgs/main/win-64::certifi-2022.12.7-py39haa95532_0
pkgs/main/win-64::openssl-1.1.1s-h2bbff1b_0
pkgs/main/win-64::pip-22.3.1-py39haa95532_0
pkgs/main/win-64::python-3.9.16-h6244533_0
pkgs/main/win-64::setuptools-65.6.3-py39haa95532_0
pkgs/main/win-64::sqlite-3.40.1-h2bbff1b_0
pkgs/main/noarch::tzdata-2022g-h04d1e81_0
pkgs/main/win-64::vc-14.2-h21ff451_1
pkgs/main/win-64::vs2015_runtime-14.27.29016-h5e58377_2
pkgs/main/noarch::wheel-0.37.1-pyhd3eb1b0_0
pkgs/main/win-64::wincertstore-0.2-py39haa95532_2

Proceed ([y]/n)? y

Preparing transaction: done
Verifying transaction: done
Executing transaction: done
#
# To activate this environment, use
#
#   $ conda activate vision
#
# To deactivate an active environment, use
#
#   $ conda deactivate
#

C:\Users\esteb>
```

- Escribir la letra “y”, y presionar la tecla “Enter” para proceder con la creación del ambiente.
- Una vez creado el ambiente, podemos activarlo usando siguiente comando:

```
conda activate vision
```

- Y podemos desactivarlo usando siguiente comando:

```
conda deactivate
```

- En cualquier caso, volvamos a activar nuestro ambiente de Anaconda con el comando mencionado anteriormente. Nos podemos dar cuenta de que nuestro ambiente está activo porque al *prompt* de la línea de comando tiene como prefijo el nombre de nuestro ambiente entre paréntesis.

```
Command Prompt
openssl      pkgs/main/win-64::openssl-1.1.1s-h2bbff1b_0
pip          pkgs/main/win-64::pip-22.3.1-py39haa95532_0
python       pkgs/main/win-64::python-3.9.16-h6244533_0
setuptools   pkgs/main/win-64::setuptools-65.6.3-py39haa95532_0
sqlite       pkgs/main/win-64::sqlite-3.40.1-h2bbff1b_0
tzdata       pkgs/main/noarch::tzdata-2022g-h04d1e81_0
vc           pkgs/main/win-64::vc-14.2-h21ff451_1
vs2015_runtime pkgs/main/win-64::vs2015_runtime-14.27.29016-h5e58377_2
wheel        pkgs/main/noarch::wheel-0.37.1-pyhd3eb1b0_0
wincertstore pkgs/main/win-64::wincertstore-0.2-py39haa95532_2

Proceed ([y]/n)? y
Preparing transaction: done
Verifying transaction: done
Executing transaction: done
#
# To activate this environment, use
#
#   $ conda activate vision
#
# To deactivate an active environment, use
#
#   $ conda deactivate

C:\Users\esteb>conda activate vision
(vision) C:\Users\esteb>
```

- A continuación, instalaremos las librerías necesarias y OpenCV. Esta es una de las librerías más completas de visión artificial para Python, C++ y Java. Para ello, estando dentro de nuestro ambiente, escribiremos los comandos siguientes:

```
pip install notebook
pip install scikit-image
pip install pandas
pip install numpy
pip install matplotlib
pip install opencv-contrib-python
```

NOTA: el commando `pip install` (perteneciente a la librería pip) puede ser utilizado para instalar otras librerías como skimage, sklearn, matplotlib, scipy, entre otras, aunque la mayoría de estas vienen por defecto en el ambiente base de Anaconda.

- La línea de comando puede ser utilizada también para programar, aunque en nuestro caso la utilizaremos para comprobar que la instalación de Open CV se haya completado exitosamente. Para ello, ejecutamos el siguiente comando dentro de nuestro ambiente:

```
python
```

- Ejecutar el comando anterior activa el modo de Python en línea de comando, que nos debería dejar algo como lo siguiente:

```
Select Command Prompt - python
Collecting pyparsing>=2.2.1
  Using cached pyparsing-3.0.9-py3-none-any.whl (98 kB)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\esteb\anaconda3\envs\vision\lib\site-packages (from matplotlib) (2.8.2)
Collecting kiwisolver>=1.0.1
  Downloading kiwisolver-1.4.4-cp39-cp39-win_amd64.whl (55 kB)
    ----- 55.4/55.4 kB ? eta 0:00:00
Collecting cycler>=0.10
  Using cached cycler-0.11.0-py3-none-any.whl (6.4 kB)
Collecting fonttools>=4.22.0
  Downloading fonttools-4.38.0-py3-none-any.whl (965 kB)
    ----- 965.4/965.4 kB 8.7 MB/s eta 0:00:00
Requirement already satisfied: six>=1.5 in c:\users\esteb\anaconda3\envs\vision\lib\site-packages (from python-dateutil>=2.7->matplotlib) (1.16.0)
Installing collected packages: pyparsing, kiwisolver, fonttools, cycler, contourpy, matplotlib
Successfully installed contourpy-1.0.7 cycler-0.11.0 fonttools-4.38.0 kiwisolver-1.4.4 matplotlib-3.6.3 pyparsing-3.0.9

(vision) C:\Users\esteb>pip install opencv-contrib-python
Collecting opencv-contrib-python
  Downloading opencv_contrib_python-4.7.0.68-cp37-abi3-win_amd64.whl (44.9 MB)
    ----- 44.9/44.9 MB 10.6 MB/s eta 0:00:00
Requirement already satisfied: numpy>=1.17.0 in c:\users\esteb\anaconda3\envs\vision\lib\site-packages (from opencv-contrib-python) (1.24.2)
Installing collected packages: opencv-contrib-python
Successfully installed opencv-contrib-python-4.7.0.68

(vision) C:\Users\esteb>python
Python 3.9.16 (main, Jan 11 2023, 16:16:36) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

- En este modo es posible ejecutar líneas de código, pero nuestra mayor preocupación es saber si Open CV se instaló correctamente, por lo que simplemente utilizaremos el siguiente comando:

```
import cv2
```

- En caso de que nuestra instalación se haya realizado correctamente, no aparecerá ningún mensaje:

```
Command Prompt - python
  Using cached pyparsing-3.0.9-py3-none-any.whl (98 kB)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\esteb\anaconda3\envs\vision\lib\site-packages (from matplotlib) (2.8.2)
Collecting kiwisolver>=1.0.1
  Downloading kiwisolver-1.4.4-cp39-cp39-win_amd64.whl (55 kB)
    ----- 55.4/55.4 kB ? eta 0:00:00
Collecting cycler>=0.10
  Using cached cycler-0.11.0-py3-none-any.whl (6.4 kB)
Collecting fonttools>=4.22.0
  Downloading fonttools-4.38.0-py3-none-any.whl (965 kB)
    ----- 965.4/965.4 kB 8.7 MB/s eta 0:00:00
Requirement already satisfied: six>=1.5 in c:\users\esteb\anaconda3\envs\vision\lib\site-packages (from python-dateutil>=2.7->matplotlib) (1.16.0)
Installing collected packages: pyparsing, kiwisolver, fonttools, cycler, contourpy, matplotlib
Successfully installed contourpy-1.0.7 cycler-0.11.0 fonttools-4.38.0 kiwisolver-1.4.4 matplotlib-3.6.3 pyparsing-3.0.9

(vision) C:\Users\esteb>pip install opencv-contrib-python
Collecting opencv-contrib-python
  Downloading opencv_contrib_python-4.7.0.68-cp37-abi3-win_amd64.whl (44.9 MB)
    ----- 44.9/44.9 MB 10.6 MB/s eta 0:00:00
Requirement already satisfied: numpy>=1.17.0 in c:\users\esteb\anaconda3\envs\vision\lib\site-packages (from opencv-contrib-python) (1.24.2)
Installing collected packages: opencv-contrib-python
Successfully installed opencv-contrib-python-4.7.0.68

(vision) C:\Users\esteb>python
Python 3.9.16 (main, Jan 11 2023, 16:16:36) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import cv2
>>>
```

- Para salir del modo de ejecución de código de Python, basta con escribir el comando `exit()`
- Esto concluye el tutorial de instalación de Python y OpenCV. Para los que están acostumbrados a la línea de comando hay buenas noticias: Anaconda incluye una aplicación llamada Anaconda Navigator que permite realizar la mayoría de las tareas explicadas anteriormente: <https://www.youtube.com/watch?v=cX6l3IzWewc>
- Es imprescindible leer a continuación los tutoriales de Jupyter Notebook y Spyder3 incluidos en el resto de la documentación.