

Asignatura	Datos del alumno	Fecha
Sistemas Multiagente y Percepción Computacional	Apellidos: Bocanegra Capera	
	Nombre: Oscar David	

Actividades

Trabajo: Instalación de OpenCV

Objetivos

Instalar la herramienta OpenCV y probar alguna de sus rutinas de visión artificial

Descripción del trabajo

En este trabajo se trata instalar la herramienta OpenCV. Se valorará también que se modifique el código de algunos de los tutoriales que se ofrecen en su sitio o en otros lugares de la red.

Los ficheros de instalación y mucha otra documentación y herramientas se encuentran disponibles en el sitio: <http://opencv.org/>

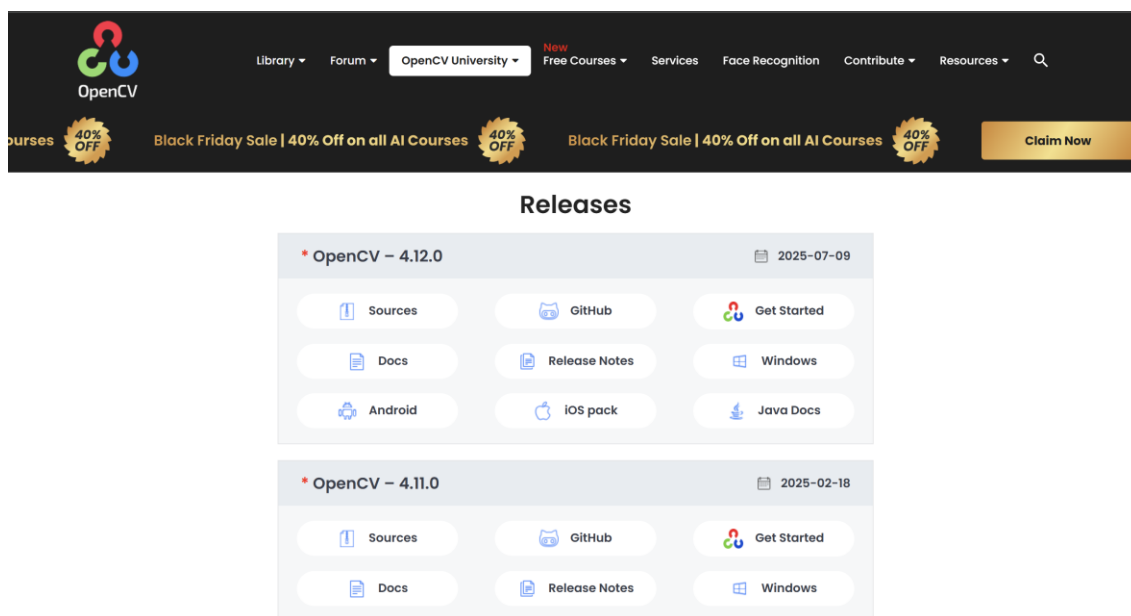
Entrega del trabajo

La entrega consistirá en un fichero en formato word que muestre con capturas de pantalla y texto el procedimiento de instalación y ejecución, así como el código modificado que se ha probado y su resultado. La extensión del documento no debe exceder las 10 páginas (Georgia 11, interlineado 1,5).

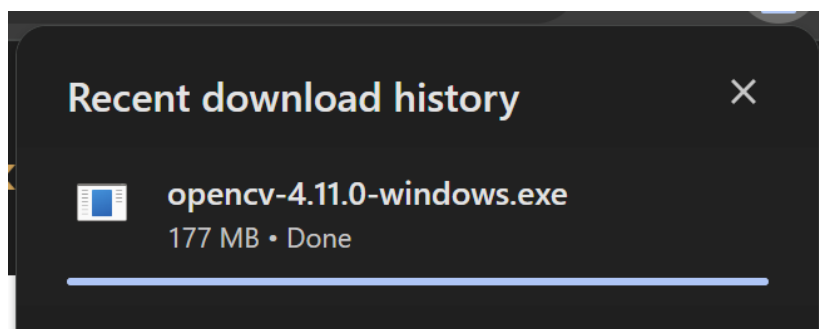
Asignatura	Datos del alumno	Fecha
Sistemas Multiagente y Percepción Computacional	Apellidos: Bocanegra Capera	
	Nombre: Oscar David	

Para el desarrollo de esta actividad lo primero que debemos tener en cuenta es el IDE que vamos a realizar el cual ya debe de contener java para una correcta ejecución, y para este caso el IDE que estaré usando será NetBeans.

Teniendo esta información clara lo que hacemos es ir a la pagina oficial de OpenCV



Y para este caso instalaremos la version 4.11.0, la cual es una version anterior a la ultima, esto debido a que la librería anterior nos estaria brindado una mayor estabilidad o menos problemas que puede que suceda si instalamos la ultima version.

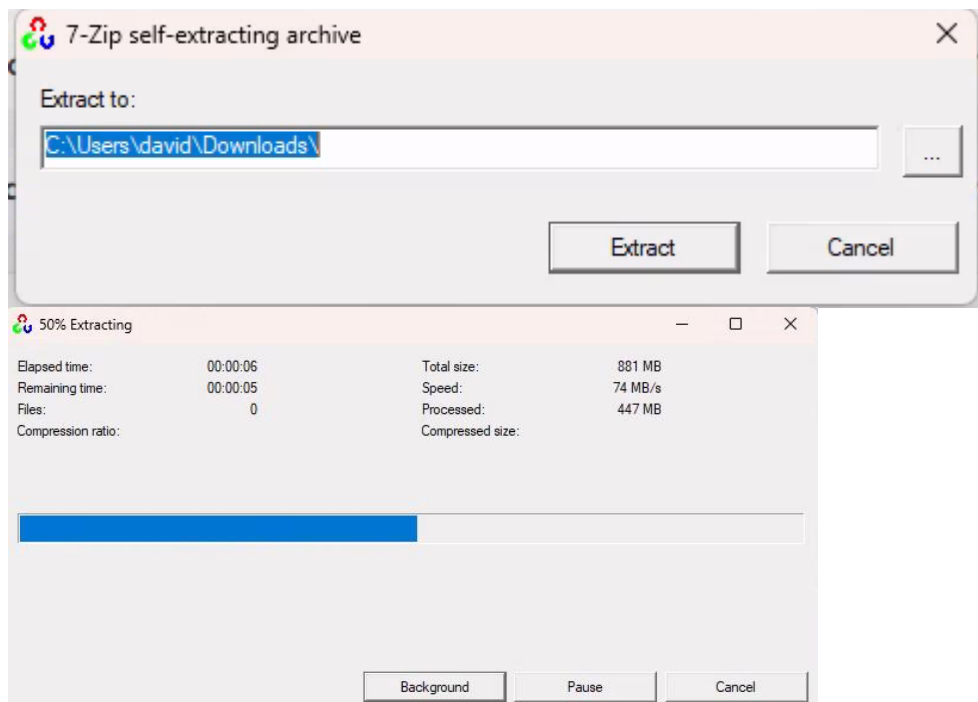


Y como veremos en la siguiente imagen ya esta instalado opencv



Y el siguiente paso es ejecutar este archivo como administrador, para asi de esta manera lograr descargar el archivo de manera definitiva

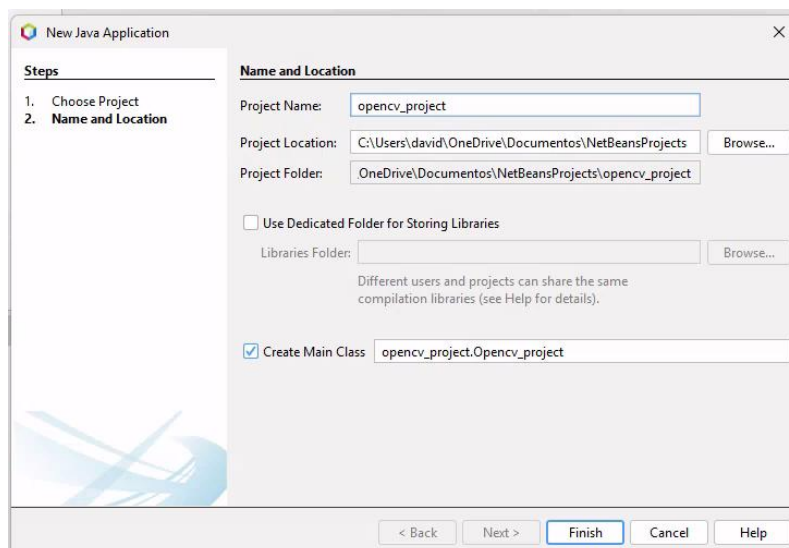
Asignatura	Datos del alumno	Fecha
Sistemas Multiagente y Percepción Computacional	Apellidos: Bocanegra Capera	
	Nombre: Oscar David	



Y ejecutando el archivo lograremos ver una nueva carpeta que se crea después de la extracción del archivo mencionado anteriormente



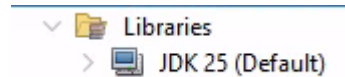
Con esto ya realizado lo que pasamos es abrir el IDE (NetBeans), para poder configurarla esta librería en el IDE, y poder hacer uso de esta misma para los próximos proyectos que requiramos de dicha librería



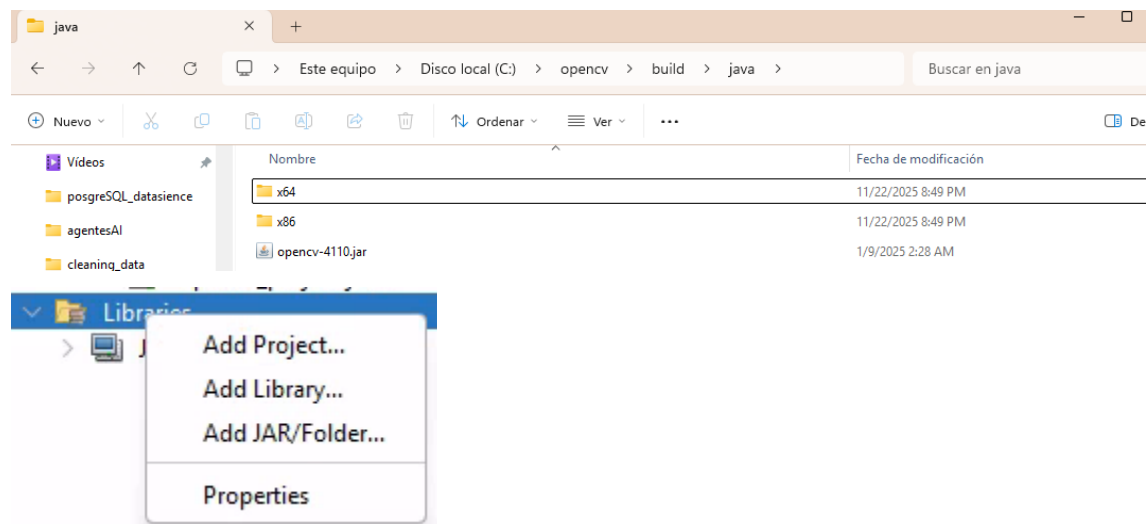
Por ende, como vemos en la imagen anterior lo que hacemos es crear un nuevo proyecto en Java el cual se nombra `opencv_project`.

Asignatura	Datos del alumno	Fecha
Sistemas Multiagente y Percepción Computacional	Apellidos: Bocanegra Capera	
	Nombre: Oscar David	

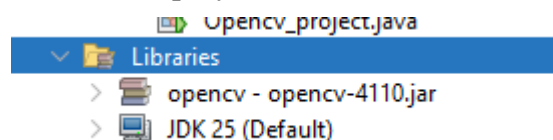
Ahora hecho esto lo que se realiza es configurar la librería como se ve en la siguiente imagen



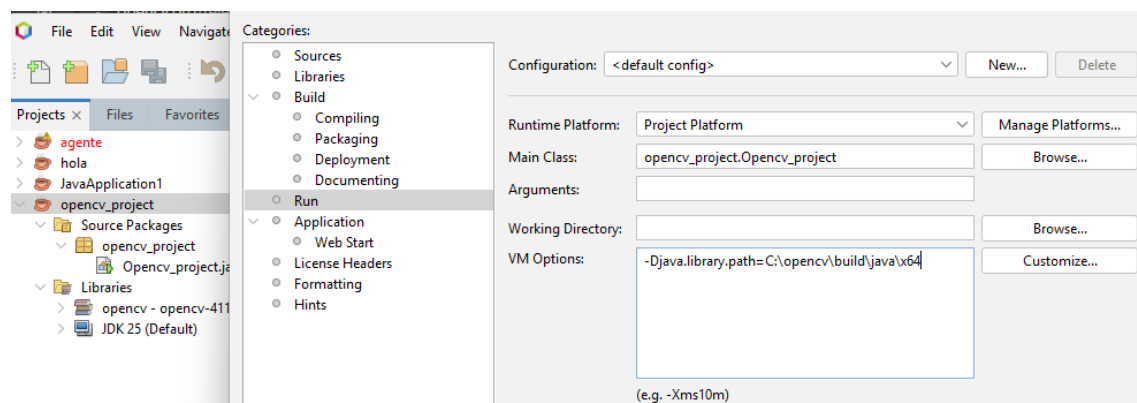
Ubicamos el archivo .jar en la carpeta que fue creada al realizar la extracción



Le decimos que añadir librería y le pasamos el directorio de en donde se encuentra el archivo .jar, aceptamos el añadir la librería y terminara viendose así la carpeta de librerías del proyecto



Hecho esto lo que hacemos es ir a las propiedades del proyecto y ubicamos el archivo .dll, esto es debido a que es el corazón nativo de OpenCV, y si no se lo indicamos al proyecto lo más probable es que nos falle cuando lo vayamos a correr



Asignatura	Datos del alumno	Fecha
Sistemas Multiagente y Percepción Computacional	Apellidos: Bocanegra Capera	
	Nombre: Oscar David	

Despues de esto ejecutamos un pequeño codigo el cual se encargara basicamente de devolver la version de la librería y con este cpmprobamos que lo este leyendo de manera adecuada

```

L  */
  public class Opencv_project {

      /**
       * @param args the command line arguments
       */
      public static void main(String[] args) {
          // TODO code application logic here
          System.loadLibrary(Core.NATIVE_LIBRARY_NAME);
          System.out.println(Core.VERSION);
      }
  }

```

Y vemos que el output es efectivamente la librería

```

4.11.0
BUILD SUCCESSFUL (total time: 0 seconds)

```

Ademas de esto le añadi la foto de un perro para hacer una comprobacion mas profunda a la librería en donde, carga una imagen de prueba en una matriz, luego de esto, el objetivo de implementar rutinas de visión, se utilizó la clase Imgproc para asi realizar dos operaciones secuenciales: la cual la primera fue una conversión del espacio de color BGR a escala de grises para simplificar la información de la imagen, y la segunda, hace aplicación del filtro de Canny para detectar discontinuidades en la intensidad de los píxeles para asi, finalmente las imágenes procesadas se guardan en el directorio local mediante Imgcodecs para su visualización.

Asignatura	Datos del alumno	Fecha
Sistemas Multiagente y Percepción Computacional	Apellidos: Bocanegra Capera	
	Nombre: Oscar David	

```

public static void main(String[] args) {
    // TODO code application logic here
    try {
        System.loadLibrary(Core.NATIVE_LIBRARY_NAME);
        System.out.println("Versión de OpenCV cargada: " + Core.VERSION);

        String rutaImagen = "perro.jpg";
        Mat src = Imgcodecs.imread(rutaImagen);

        if (src.empty()) {
            System.out.println("Error: No se encuentra 'perro.jpg'. Ponla en la carpeta del proyecto (no en src).");
            return;
        }
        System.out.println("Imagen cargada correctamente.");

        Mat gris = new Mat();
        Imgproc.cvtColor(src, gris, Imgproc.COLOR_BGR2GRAY);

        Mat bordes = new Mat();
        Imgproc.Canny(gris, bordes, 100, 200);

        Imgcodecs.imwrite("resultado_gris.jpg", gris);
        Imgcodecs.imwrite("resultado_bordes.jpg", bordes);

        System.out.println("Proceso terminado! Revisa la carpeta del proyecto para ver las nuevas imágenes.");
    } catch (Exception e) {
        System.out.println("Ocurrió una excepción: " + e.getMessage());
        e.printStackTrace();
    }
}

```

Y adjunto las 3 imágenes



La anterior es la imagen original, y las siguientes 2 las que el código devuelve, cumpliendo con su objetivo



Asignatura	Datos del alumno	Fecha
Sistemas Multiagente y Percepción Computacional	Apellidos: Bocanegra Capera	
	Nombre: Oscar David	



Junto con el output por consola

```
Version de OpenCV cargada: 4.11.0
Imagen cargada correctamente.
◆Proceso terminado! Revisa la carpeta del proyecto para ver las nuevas im genes.
BUILD SUCCESSFUL (total time: 0 seconds)
```

Conclusiones

El desarrollo de esta actividad fue bastante interesante, ya que aprender una librer a de agentes inteligentes, y que es asociada con la inteligencia artificial es algo bastante retador e interesante a mi parecer.

Adem s de esto el ver de lo que es capaz una Librer a como esta es bastante asombroso ya que se puede evidenciar filtros de preprocesamiento en escala de grises y extracci n de caracter sticas de las im genes.