Se requiere desarrollar un agente inteligente capaz de extraer y sumar los valores numéricos que están representados en los objetos asignados por el profesor tales como cartas de juegos de mesa (verificar la categoría de elementos asignados en el correo). Para esto el sistema debe implementar algoritmos de detección de polígonos para extraer los objetos de la mesa y métodos de Deep learning para la detección de los valores.

|  |  |
| --- | --- |
| **Ejemplo 1 – Cartas Uno** | **Ejemplo 2 - Cartas Uno** |
| El algoritmo realizó un proceso de extracción de objetos y determinó que las cartas sobre la mesa corresponden a los valores de 5 y 3, por tanto, la suma es 8. | El algoritmo realizó un proceso de extracción de objetos y determinó que las cartas sobre la mesa corresponden a los valores de 3 y 7, por tanto, la suma es 10. |

Es de aclarar que para efectos de facilidad en el proceso de entrenamiento y pruebas, se trabajará en un ambiente controlado, esto quiere decir que la superficie de la mesa debe tener una cartulina negra, (esto es para facilitar el proceso de extracción de los objetos).

Basado en lo anterior debe realizar lo siguiente:

1. **(Valor 1.0)** Desarrollar un algoritmo de visión artificial capaz de detectar la forma de los polígonos que están en tablero y extraerlos. **NOTA:** Siempre serán 2 ROI’s correspondientes a los 2 objetos.
2. **(Valor 0.5)** Crear un dataset de entrenamiento con las clases correspondientes a los números del 0 al 9 (El tamaño de las imágenes será de 128\*128 pixeles). **Nota:** Entre más imágenes tenga el dataset mejor será el rendimiento; para esto se aconseja, tomar fotografías de los objetos en diferentes posiciones. Ejemplo estructura del dataset.



1. **(Valor 0.8)** Implementar y entrenar mínimo 3 modelos de redes neuronales convolucionales, las cuales ayuden a predecir el valor que representa cada objeto en el tablero.
2. Realizar el proceso de pruebas en las cuales detalle la siguiente información:

* 1. **(Valor 0.4)** Muestre matrices de confusión de cada uno de los modelos
  2. **(Valor 0.4)** Llene la siguiente tabla.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| N° | Nombre modelo | Accuracy | Precision | Recall | F1 Score | Loss | Épocas de entrenamiento | Tiempo de respuesta |
| 1 |  |  |  |  |  |  |  |  |
| 2 |  |  |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |  |  |

* 1. **(Valor 0.4)** Realice un análisis comparativo de la tabla y determine cual es el modelo final que va a implementar en la solución, en el cual argumente por que elegirlo y cuales son los posibles escenarios en lo que el modelo puede fallar.

1. **(Valor 0.5)** Una vez tenga el modelo implementado debe realizar lo siguiente: cada vez que se pongan los 2 objetos sobre la mesa y se presione la tecla “p” se debe predecir el valor numérico que representan los objetos, y en la interfaz debe ir apareciendo el valor acumulado de todas las pruebas que han pasado por el sistema hasta ese momento. Ejemplo:

|  |  |
| --- | --- |
|  |  |
| Instante 1 | Instante 2 |

1. **(Valor 0.5)** Realice un video de máximo 5 minutos en el cual realice una corta explicación de la manera como implementó la solución del parcial, donde se muestre especialmente la estructura del proyecto, (mostrando algunas de las imágenes utilizadas), haga principal hincapié en lo mencionado en los literales 4 a, 4 b y 4 c.
2. **(Valor 0.5)** Sustentación ante el profesor, en la cual se validará el proceso implementado.

**Nota:** Si usted no desea realizar el algoritmo de extracción de los 2 elementos, puede realizar pruebas con un solo objeto, y en el literal 5 puede realizar la predicción de un solo valor e ir llevando en cuenta el acumulado de todas las pruebas realizadas hasta ese momento. (Si lo realiza de esta forma, la máxima nota a obtener sería de 4.0 en el parcial)

|  |  |  |
| --- | --- | --- |
|  |  |  |
| Instante 1 | Instante 2 | Instante 3 |