	Universidad Tecnológica de México	Proyecto final	Agosto, 2023
	Dirección Académica de Sistemas Computacionales e Informática		Ciclo: 22-3

Misión y Filosofía Institucional de la Universidad Tecnológica de México (UNITEC)



MISIÓN INSTITUCIONAL


Generar y ofrecer servicios educativos en los niveles medio superior, superior y posgrado; conjugando educación científica y tecnológica así como la innovación sobre una base de humanismo; promoviendo una actitud de aprendizaje permanente, una cultura basada en el esfuerzo y un espíritu de superación; combinando la profundidad en el estudio de cada disciplina con una visión amplia de la empresa, la sociedad y la vida; procurando elevar permanentemente la calidad académica; adecuando nuestros procesos educativos a las diversas necesidades de los estudiantes, y aprovechando eficientemente los recursos de la Institución para dar acceso a grupos más amplios de la sociedad.

FILOSOFÍA INSTITUCIONAL

La filosofía de la Universidad Tecnológica de México constituye los principios que guían y dan sentido a nuestra vida institucional. Expresa nuestra razón de ser, nuestros valores, nuestras convicciones y nuestro deseo de contribuir, a través de la educación, al engrandecimiento de México. Como mexicanos, deseamos el bienestar del país y estamos conscientes del gran potencial de sus recursos, de su cultura y de su voluntad de progreso. Sabemos que para incidir en su desarrollo tenemos que hacerlo en el de las personas: nuestros educandos, nuestros maestros, nuestra comunidad universitaria en general.

VISIÓN INSTITUCIONAL

Institución educativa particular de mayor tamaño en el país, reconocida como una de las mejores por su calidad e integridad académica, innovación, formación profesional y la competitividad de sus egresados.

	Universidad Tecnológica de México	Proyecto final	Agosto, 2023
	Dirección Académica de Sistemas Computacionales e Informática		Ciclo: 22-3

Universidad Tecnológica de México



CARRERA

Ingeniería en Sistemas computacionales

NOMBRE DEL PROYECTO

VCSM

(VIDEOJUEGO CON SENSOR DE MOVIMIENTO)

INTEGRANTES

- HINOJOSA GONZALEZ GAEL - 18126540
- ESTRADA MARTINEZ BRYAN FRANCISCO - 21993618
- RODRIGUEZ JUAREZ JOHAN - 21099903

PROFESOR


Raymundo Soto Soto

ASIGNATURA

Procesamiento Digital de Imágenes


EQUIPO

2

	Universidad Tecnológica de México	Proyecto final	Agosto, 2023
	Dirección Académica de Sistemas Computacionales e Informática		Ciclo: 22-3


INDICE

<u>1. INTRODUCCION</u>	<u>4</u>
<u>2. PLANTEAMIENTO</u>	<u>5</u>
<u>3. OBJETIVO</u>	<u>5</u>
<u>4. MARCO TEORICO</u>	<u>6</u>
<u>5. ANTECEDENTES</u>	<u>8</u>
<u>6. METODOLOGIA</u>	<u>9</u>
<u>7. DIAGRAMA DE FLUJO</u>	<u>10</u>
<u>8. HERRAMIENTAS</u>	<u>11</u>
<u>9. CODIGO</u>	<u>12</u>
<u>10. EXPLICACION DEL CODIGO</u>	<u>20</u>
<u>11. PRUEBAS</u>	<u>21</u>
<u>12. CONCLUSIONES</u>	<u>23</u>
<u>13. BIBLIOGRAFIAS</u>	<u>24</u>

	Universidad Tecnológica de México	Proyecto final	Agosto, 2023
	Dirección Académica de Sistemas Computacionales e Informática		Ciclo: 22-3

INTRODUCCION

El reconocimiento de movimiento, también conocido como detección de movimiento, es una tecnología fascinante que ha revolucionado numerosas aplicaciones en campos como la seguridad, la salud, el entretenimiento y la automatización. Esta innovadora área se enfoca en identificar y rastrear los cambios en la posición de objetos, personas o animales a lo largo del tiempo. Al aprovechar sensores avanzados, técnicas de procesamiento de imágenes y algoritmos sofisticados, el reconocimiento de movimiento permite capturar, analizar y comprender los patrones de desplazamiento, gestos y acciones. Desde sistemas de vigilancia que alertan sobre intrusiones hasta dispositivos portátiles que monitorizan la actividad física, el reconocimiento de movimiento ha transformado nuestra capacidad para interactuar con el mundo digital y físico de maneras antes inimaginables. En este contexto, exploraremos cómo funciona esta tecnología, sus aplicaciones más relevantes y su impacto en diversos sectores de la sociedad moderna.

	Universidad Tecnológica de México	Proyecto final	Agosto, 2023
	Dirección Académica de Sistemas Computacionales e Informática		Ciclo: 22-3

PLANTEAMIENTO

En la actualidad ya existen varios videojuegos de este tipo, pero lo que nosotros queremos


hacer es algo que ayude a todas las personas ya que no solo será un videojuego, en este caso así será por el tema de tiempo y presupuesto, pero si se pudiera avanzar al siguiente paso se crearía una interfaz capaz de controlar todo lo que sea necesario como una computadora, televisión, etc. Con el movimiento de tus manos, esto será útil para la casi todo el mundo ya que cualquiera lo podría usar, y en un futuro agregar actualizaciones para

que más gente lo pueda usar.

OBJETIVO

Nuestro objetivo es crear más medios de entretenimiento y ayuda a la vez, ya que con este


proyecto se podrá realizar ambos y sin necesidad de algún accesorio adicional más que una cámara que logre detectar el movimiento. Puede que en la actualidad ya haya varios proyectos, aplicaciones así, pero lo que nosotros buscamos es innovar e darles más oportunidades a los que no pueden probar este tipo de aplicaciones ya sea por discapacidad u otra razón.

	Universidad Tecnológica de México	Proyecto final	Agosto, 2023
	Dirección Académica de Sistemas Computacionales e Informática		Ciclo: 22-3

MARCO TEORICO


El reconocimiento de movimiento es uno de los campos más estudiados en el área de Visión por Computador, ya que puede ser aplicado en distintas actividades de la vida cotidiana, como por ejemplo videovigilancia, detección de acciones en deporte, entre otros. En este proyecto de titulación realizamos una investigación detallada del estado del arte referente a los métodos para la extracción de movimiento y semántica en una secuencia de imágenes, partiendo desde las bases es decir definiendo Inteligencia Artificial, Redes Neuronales, Aprendizaje Profundo, para después revisar los distintos métodos y algoritmos propuesto en los últimos años, desde el uso de flujo óptico, descriptores de movimientos, entre otros más que se describen a lo largo del documento, así mismo se da una completa revisión a los distintos datasheets para el reconocimiento de movimiento, para finalmente poner a prueba una red basada en estos principios, que cumpla de manera eficiente el reconcomiendo de movimiento y semántica, realizando el respectivo entrenamiento, y experimentos para determinar la precisión de la red.

Python es un lenguaje de programación de código abierto, creado por Guido van Rossum en

	Universidad Tecnológica de México	Proyecto final	Agosto, 2023
	Dirección Académica de Sistemas Computacionales e Informática		Ciclo: 22-3

1991. Se trata de un lenguaje orientado a objetos, fácil de interpretar y con una sintaxis que permite leerlo de manera semejante a como se lee el inglés. Es un lenguaje interpretado, esto significa que el código de programación se convierte en bytecode y luego se ejecuta por el intérprete, que, en este caso, es la máquina virtual de Python. Pygame es un conjunto de módulos del lenguaje Python que permiten la creación de videojuegos en dos dimensiones de una manera sencilla. Está orientado al manejo de sprites. Gracias al lenguaje, se puede prototipar y desarrollar rápidamente. Esto se puede comprobar en las competencias que se disputan en línea, donde es cada vez más usado. Los resultados pueden llegar a ser profesionales. También puede utilizarse para crear otros programas multimedia o interfaces gráficas de usuario. Pygame está basado en la librería SDL (2.24.2 en la versión actual)


Visual Studio Code es un editor de código popular que se puede usar con Python. Tiene una serie de extensiones que facilitan el desarrollo de juegos con Python, como para escribir scripts y módulos.

	Universidad Tecnológica de México	Proyecto final	Agosto, 2023
	Dirección Académica de Sistemas Computacionales e Informática		Ciclo: 22-3

ANTECEDENTES

La interacción entre los usuarios y las computadoras ha sido objeto de estudio en el campo de la interfaz humano-computadora. Los antecedentes teóricos de esta área pueden ayudarte a comprender cómo diseñar una interfaz intuitiva y efectiva para tu videojuego. Esto incluye teorías sobre diseño de interacción, usabilidad y experiencia del usuario.

Los sensores de movimiento de manos, como los acelerómetros y giroscopios, permiten capturar los movimientos y gestos de los jugadores. Es importante comprender cómo funcionan estos sensores y cómo se pueden utilizar para rastrear y reconocer los movimientos de las manos con precisión. Puedes explorar la literatura científica y técnica relacionada con los sensores de movimiento y las técnicas de seguimiento de gestos

	Universidad Tecnológica de México	Proyecto final	Agosto, 2023
	Dirección Académica de Sistemas Computacionales e Informática		Ciclo: 22-3

METODOLOGIA

a) Universo de Estudio: El universo de estudio para un proyecto de videojuego con sensor de movimiento de manos podría abarcar diferentes aspectos, desde la tecnología utilizada hasta los elementos del juego y la experiencia del usuario.

b) Procedimiento para la Recolección de la información: Después de finalizar el proyecto vamos a hacer pruebas de este invitando a gente a probarlo ya que siento que esta sería la manera más eficiente de poder reunir información.

c) Plan de análisis de los resultados: Al recolectar la información necesaria o máxima, procederemos a analizarla y ver que se tiene que cambiar o mejorar y si el tiempo es suficiente se realizarán los cambios, si no se guardarán para otra versión.


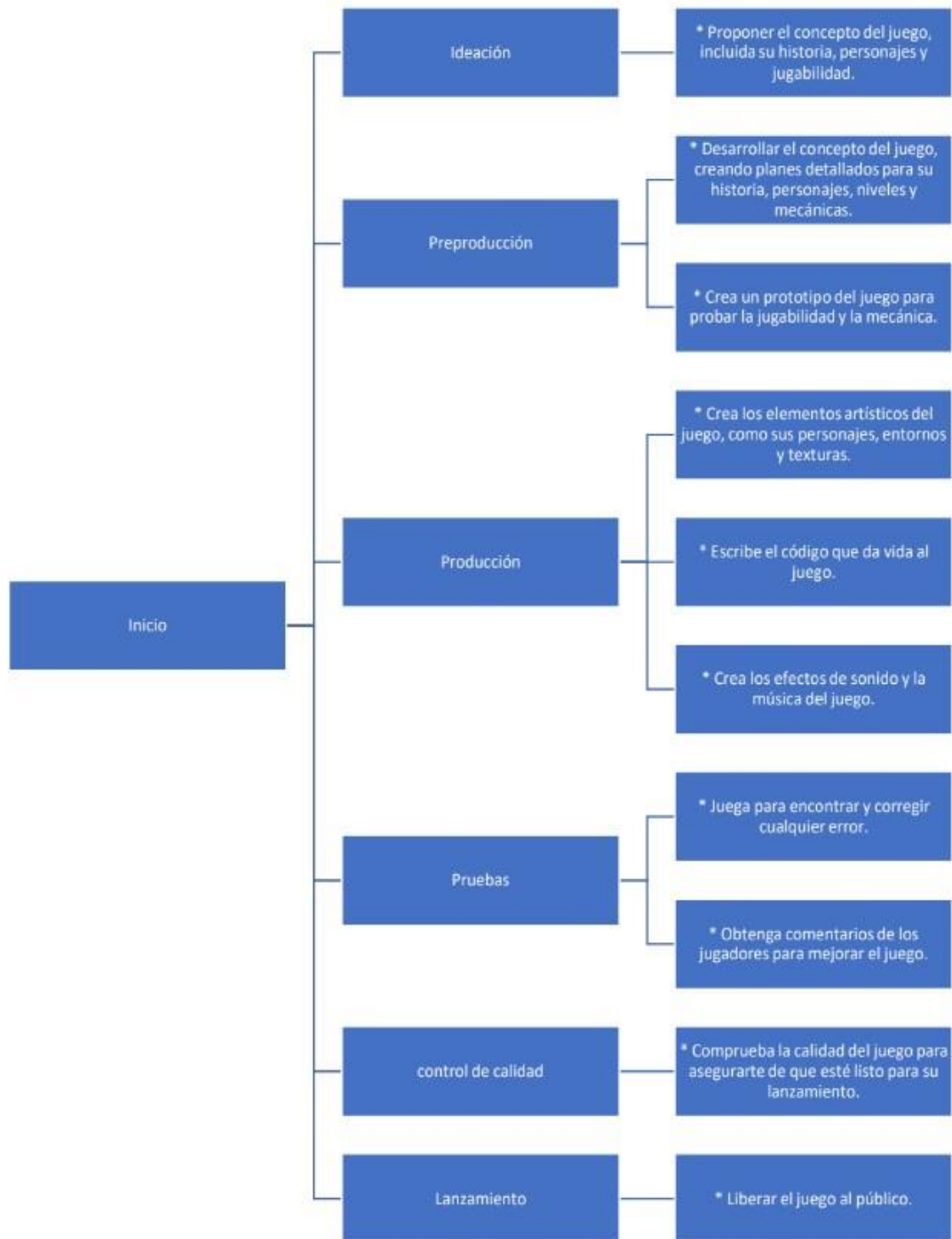

	Universidad Tecnológica de México	Proyecto final	Agosto, 2023
	Dirección Académica de Sistemas Computacionales e Informática		Ciclo: 22-3

DIAGRAMA DE FLUJO



	Universidad Tecnológica de México	Proyecto final	Agosto, 2023
	Dirección Académica de Sistemas Computacionales e Informática		Ciclo: 22-3

HERRAMIENTAS

▣ Visual Studio Code


Es un editor de código popular que se puede usar con Python. Tiene una serie de extensiones que facilitan el desarrollo de juegos con Python, como para escribir scripts y módulos.

▣ Pygame

Es una biblioteca de Python gratuita y de código abierto para crear aplicaciones multimedia, incluidos videojuegos. Es fácil de aprender y usar, y admite una amplia gama de dispositivos de entrada, incluidos los controladores de movimiento. Pygame es una buena opción para los principiantes que quieren crear videojuegos sencillos.

▣ Python

Es un lenguaje de programación de código abierto, creado por Guido van Rossum en 1991. Se trata de un lenguaje orientado a objetos, fácil de interpretar y con una sintaxis que permite leerlo de manera semejante a como se lee el inglés.

	Universidad Tecnológica de México	Proyecto final	Agosto, 2023
	Dirección Académica de Sistemas Computacionales e Informática		Ciclo: 22-3


CODIGO

```
#Snake con sensor de movimiento
#Snake with motion sensor
# Importar las bibliotecas necesarias
import math import random import
pygame import tkinter as tk from
tkinter import messagebox import
mediapipe as mp import cv2

# Configurar detección de manos de Mediapipe mp_hands
= mp.solutions.hands

hands = mp_hands.Hands(static_image_mode=False, max_num_hands=1,
min_detection_confidence=0.7) mpDraw =
mp.solutions.drawing_utils

# Definir la clase "cube" que representa una celda en el juego
class cube(object):
    rows = 20 # Número de filas en la cuadrícula w = 500
# Ancho de la ventana def
__init__(self,start,dirnx=1,dirny=0,color=(255,0,0)):
    self.pos = start
self.dirnx = 1 self.dirny
= 0 self.color = color
def move(self, dirnx, dirny):
    self.dirnx = dirnx
self.dirny = dirny
    self.pos = (self.pos[0] + self.dirnx, self.pos[1] +
self.dirny) def draw(self, surface, eyes=False):
```

	Universidad Tecnológica de México	Proyecto final	Agosto, 2023
	Dirección Académica de Sistemas Computacionales e Informática		Ciclo: 22-3

```

        dis = self.w // self.rows

i = self.pos[0]          j =
self.pos[1]

        # Dibujar un cuadrado en la posición de la celda
pygame.draw.rect(surface, self.color, (i*dis+1,j*dis+1, dis-2,
dis-2))          if eyes:
                centre = dis//2
radius = 3

                circleMiddle = (i*dis+centre-radius,j*dis+8)
circleMiddle2 = (i*dis + dis -radius*2, j*dis+8)

        # Dibujar los "ojos" del cubo
pygame.draw.circle(surface, (0,0,0), circleMiddle,
radius)


        pygame.draw.circle(surface, (0,0,0), circleMiddle2,
radius)

# Definir la clase "snake" que representa a la serpiente en el
juego
class snake(object):

    body = [] # Lista para almacenar las partes del cuerpo de la
serpiente    turns = {} # Diccionario para almacenar giros
def __init__(self, color, pos):

    self.color = color
self.head = cube(pos)
self.body.append(self.head)
self.dirnx = 0          self.dirny =
1

```

	Universidad Tecnológica de México	Proyecto final	Agosto, 2023
	Dirección Académica de Sistemas Computacionales e Informática		Ciclo: 22-3

```

def move(self,
cam):
    #keys = pygame.key.get_pressed() # Obtener las teclas
    presionadas
    for event in pygame.event.get():
if event.type == pygame.QUIT:
        pygame.quit()


    # Capturar la imagen de la cámara y procesarla para
    detectar manos          success, img = cam.read()
    imgg = cv2.flip(img, 1)

    # Convertir el fotograma de BGR a RGB
    image_rgb = cv2.cvtColor(imgg, cv2.COLOR_BGR2RGB)
    results = hands.process(image_rgb)

    # Obtener las coordenadas de la mano
    if results.multi_hand_landmarks:
        for
        hand_landmarks in results.multi_hand_landmarks:
            #
            Get the size of the hand tracking window
            image_height, image_width, _ = imgg.shape

            # Draw circles for all the landmarks in red
            for landmark in hand_landmarks.landmark:
                x =
                int(landmark.x * image_width)
                y =
                int(landmark.y * image_height)
                cv2.circle(imgg, (x, y), 5, (0, 0, 255), -1)

```

	Universidad Tecnológica de México	Proyecto final	Agosto, 2023
	Dirección Académica de Sistemas Computacionales e Informática		Ciclo: 22-3

```

# Calculate the average coordinates of all the
landmarks

total_x = 0
total_y = 0
num_landmarks = len(hand_landmarks.landmark)
for landmark in hand_landmarks.landmark:
    total_x += landmark.x
total_y += landmark.y

average_x = total_x /
num_landmarks
average_y = total_y /
num_landmarks

# Dibujar un círculo en la posición del centro de la
mano (verde)
center_x = int(average_x *
image_width)
center_y = int(average_y *
image_height)

cv2.circle(imgg, (center_x, center_y), 10, (0,
255, 0), -1)

# Mapear las coordenadas del centro de la mano al
movimiento de snake

self.dirnx = int((average_x * cube.rows) -
self.head.pos[0])


self.dirny = int((average_y * cube.rows) -
self.head.pos[1])

# Limitar la velocidad de movimiento

self.dirnx = max(-1, min(self.dirnx, 1))
self.dirny = max(-1, min(self.dirny, 1))

# Guardar la posición anterior de la cabeza

```

	Universidad Tecnológica de México	Proyecto final	Agosto, 2023
	Dirección Académica de Sistemas Computacionales e Informática		Ciclo: 22-3

```
prev_head_pos = self.head.pos
```

```
#Mover la serpiente
```

```
self.head.move(self.dirnx, self.dirny)
cv2.imshow("Seguimientos de manos", imgg)
```

```
#mover resto del cuerpo
for i,
c in enumerate(self.body[1:], start=1):
    temp_pos = c.pos
    c.pos = prev_head_pos
prev_head_pos = temp_pos
```

```
# Asegurarse de que las posiciones estén dentro de los
límites de la ventana
```

```
c.pos = (c.pos[0] % cube.rows, c.pos[1] %
cube.rows)
```

```
def
```

```
reset(self, pos):
    self.head
```

```
= cube(pos)
    self.body = []
```

```
self.body.append(self.head)
```

```
self.turns = {}
    self.dirnx
```

```
= 0
    self.dirny = 1
```

```
def
```


```
addCube(self):
```

```
    tail = self.body[-1]
```

```
dx, dy = tail.dirnx, tail.dirny
```

```
    if dx == 1 and dy ==
```


```
0:
```


	Universidad Tecnológica de México	Proyecto final	Agosto, 2023
	Dirección Académica de Sistemas Computacionales e Informática		Ciclo: 22-3

```

        self.body.append(cube((tail.pos[0]-1,tail.pos[1])))
elif dx == -1 and dy == 0:
        self.body.append(cube((tail.pos[0]+1,tail.pos[1])))
elif dx == 0 and dy == 1:
        self.body.append(cube((tail.pos[0],tail.pos[1]-1)))
elif dx == 0 and dy == -1:
        self.body.append(cube((tail.pos[0],tail.pos[1]+1)))
        self.body[-1].dirnx =
dx        self.body[-1].dirny =
dy
        def draw(self, surface):
for i, c in enumerate(self.body):
        c.draw(surface, i == 0)
        def drawGrid(w, rows,
surface):
        sizeBtwn = w // rows
        x = 0        y = 0
for l in range(rows):
x = x + sizeBtwn
y = y + sizeBtwn
        pygame.draw.line(surface, (255,255,255),
(x,0),(x,w))        pygame.draw.line(surface, (255,255,255),
(0,y),(w,y))
        def redrawWindow(surface):
global rows, width, s, snack
surface.fill((0,0,0))
        s.draw(surface)
snack.draw(surface)

```

	Universidad Tecnológica de México	Proyecto final	Agosto, 2023
	Dirección Académica de Sistemas Computacionales e Informática		Ciclo: 22-3

```

drawGrid(width,rows, surface)
pygame.display.update()

def randomSnack(rows,
item):

    positions = item.body
    while
True:
x = random.randrange(rows)
y = random.randrange(rows)
        if len(list(filter(lambda z:z.pos == (x,y), positions))) >
0:                continue                else:                break


    return (x,y)

def message_box(subject, content):
root = tk.Tk()    root.attributes("-
topmost", True)    root.withdraw()
    messagebox.showinfo(subject, content)
try:
    root.destroy()
except:    pass

def main():    global width, rows, s, snack
width = 500    rows = 20    win =
pygame.display.set_mode((width, width))    s =
snake((255,0,0), (10,10))    snack =
cube(randomSnack(rows, s), color=(0,255,0))    cam =
cv2.VideoCapture(0)    flag = True

pygame.init()

```

	Universidad Tecnológica de México	Proyecto final	Agosto, 2023
	Dirección Académica de Sistemas Computacionales e Informática		Ciclo: 22-3


```

        clock = pygame.time.Clock()
        while
flag:
            pygame.time.delay(50)
clock.tick(10)
            s.move(cam)            print("Snake
position:", s.head.pos)
print("Snack position:", snack.pos)
if s.head.pos == snack.pos:
            s.addCube()            snack =
cube(randomSnack(rows, s), color=(0,255,0))
print("Snack eaten! New snack position:", snack.pos)
            for x in
range(len(s.body)):
                if s.body[x].pos in list(map(lambda z:z.pos,
s.body[x+1:]]):
                    print('Score: ', len(s.body))
message_box('You Lost!', 'Play again...')
                    s.reset((10,10))
break

redrawWindow(win)

pass
pygame.quit()
main()

```

	Universidad Tecnológica de México	Proyecto final	Agosto, 2023
	Dirección Académica de Sistemas Computacionales e Informática		Ciclo: 22-3

EXPLICACION DEL CODIGO

Este código es una implementación de un juego de serpiente (Snake) en Python, utilizando la biblioteca de juegos `pygame` junto con la detección de manos proporcionada por la biblioteca `mediapipe` y la biblioteca de visión por computadora `OpenCV`.

Importación de bibliotecas: Se importan las bibliotecas necesarias, como `math`, `random`, `pygame`, `tkinter`, `mediapipe` y `cv2`.

Configuración de la detección de manos de Mediapipe: Se establece la configuración para la detección de manos utilizando la biblioteca `mediapipe`.

Clase `cube`: Representa una celda en el juego, que puede ser parte del cuerpo de la serpiente o el bocadillo (comida) que la serpiente debe consumir. Esta clase tiene métodos para mover y dibujar la celda en la ventana del juego.


Clase `snake`: Representa a la serpiente en el juego. Tiene métodos para mover la serpiente de acuerdo con la detección de manos, agregar celdas al cuerpo, dibujar la serpiente en la ventana y reiniciar el juego en caso de perder.

Funciones `drawGrid` y `redrawWindow`: Estas funciones se encargan de dibujar la cuadrícula y actualizar la ventana del juego con la serpiente y otros elementos.

Función `randomSnack`: Genera una posición aleatoria para el bocadillo (comida) en la cuadrícula, asegurándose de que no esté en la misma posición que una parte del cuerpo de la serpiente.

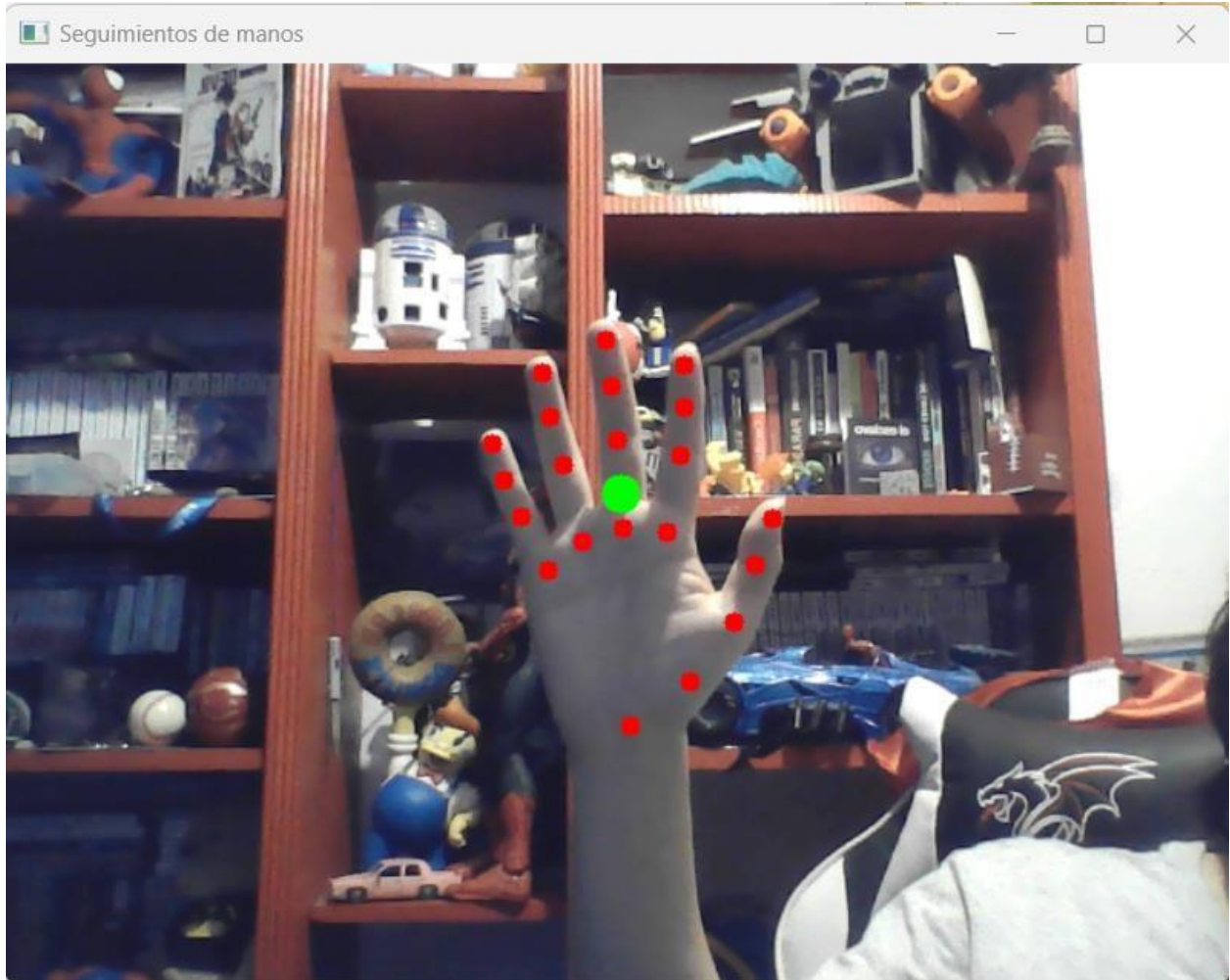
Función `main`: Esta función es el punto de entrada principal del juego. Configura la ventana de pygame, crea la serpiente y el bocadillo inicial, inicia la cámara para la detección de manos, y maneja la lógica del juego, como el movimiento de la serpiente, la detección de colisiones y la actualización de la ventana.

Bucle de juego: El bucle principal del juego se ejecuta continuamente, actualizando el movimiento de la serpiente y la posición del bocadillo en cada iteración. También maneja eventos como la colisión de la serpiente con ella misma o con los bordes de la ventana.


	Universidad Tecnológica de México	Proyecto final	Agosto, 2023
	Dirección Académica de Sistemas Computacionales e Informática		Ciclo: 22-3

PRUEBAS

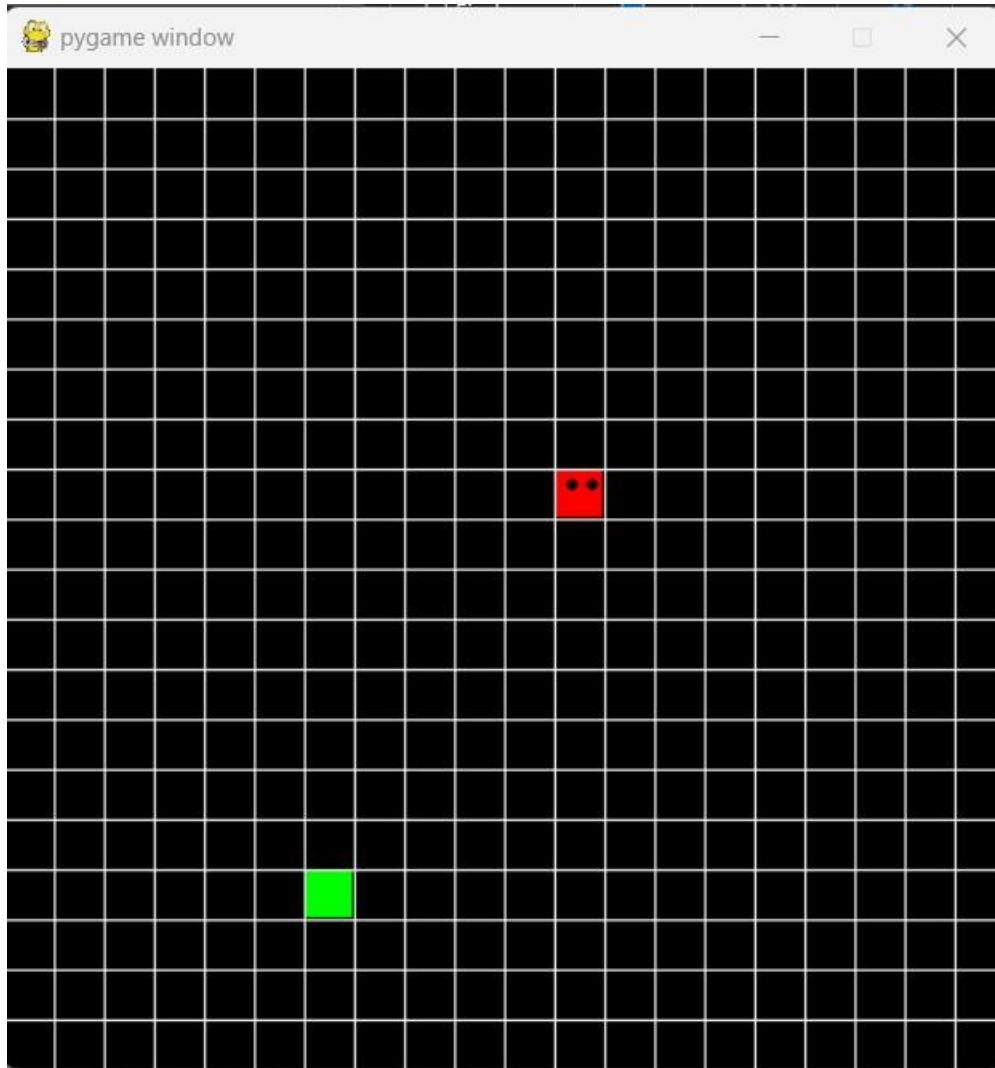
Seguimiento de mano.




En esta imagen se muestra el seguimiento de la mano, el esqueleto de la mano y el punto de referencia para que se mueva el personaje en el juego Snake.

	Universidad Tecnológica de México	Proyecto final	Agosto, 2023
	Dirección Académica de Sistemas Computacionales e Informática		Ciclo: 22-3

Juego en Ejecución.



En esta imagen se observa el juego ya en funcionamiento junto con la parte de seguimiento de manos para que el punto rojo se mueva por toda la pantalla.

	Universidad Tecnológica de México	Proyecto final	Agosto, 2023
	Dirección Académica de Sistemas Computacionales e Informática		Ciclo: 22-3

CONCLUSIONES

ESTRADA MARTINEZ BRYAN FRANCISCO


El proyecto fue un gran reto al tener que implementar diversos conocimientos obtenidos con anterioridad cómo también adquirir nuevos para desarrollarlo plenamente, concluye con la meta y objetivo alcanzado poder realizar un videojuego con sensor de movimiento desde 0, este proyecto a futuro tiene muchas posibilidades de ser más grande

RODRIGUEZ JUAREZ JOHAN

Este proyecto fue de mucha ayuda, porque durante el proceso pude comprender mejor el cómo funciona el reconocimiento de movimiento, también a como implementarlo que en nuestro caso fue a un videojuego, a pesar de todos los inconvenientes al final se logró terminar y presentar el proyecto como lo teníamos planeado y al igual que en nuestra presentación, espero que este tipo de implementación a futuro pueda llevarse más allá, implementándolo en otro tipo de dispositivos.

HINOJOSA GONZALEZ GAEL

En este proyecto logramos llenarnos de mas conocimiento sobre diferentes lenguajes que en este caso fue Python, fue un gran reto, pero al final se logro completar, claro que se puede mejorar y hay varias ideas que se pueden implementar en el futuro, pero por ahora lo dejaremos ahí, igual gracias a este proyecto logramos trabajar como equipo y presentar fluidamente, fue de gran ayuda y como mencione anteriormente se puede convertir en algo grande.

	Universidad Tecnológica de México	Proyecto final	Agosto, 2023
	Dirección Académica de Sistemas Computacionales e Informática		Ciclo: 22-3

BIBLIOGRAFÍAS

- Javired, & Javired. (2022). ¿Qué son los sensores de movimiento y cómo funcionan? Electrositio. <https://electrositio.com/que-son-los-sensores-de-movimiento-y-comofuncionan>
- HMI - the interface between process and operators. (s. f.). <https://www.aveva.com/es-es/solutions/operations/hmi/>
- Biometrics: Concepts, Methodologies, Tools, and Applications. By Ajith Abraham, Fei-Fei Li, and Rama Chellappa. Springer, 2012.
- Cerebrovascular Ultrasound: Theory, Practice and Future Developments. By Peter D. Steinbach. Springer, 2011
- <https://github.com/raymundosoto/movimiento-de-objetos-en-un-videojuego--condeteccion-de-manos-usando-mediapipe-y-pygame/blob/main/readme.md>

Nuestro repositorio en GitHub

- <https://github.com/kanikass/proyectgm.git>