

Inteligencia Artificial Aplicada a la Domótica

Ana Gabriela Silva, Juan Camilo Gil, Juan Sebastián Cadavid



CONCEPTOS CLAVE

La habilidad de los ordenadores para hacer actividades que normalmente requieren inteligencia humana.

INTELIGENCIA
ARTIFICIAL

IoT(Internet of Things) describe la red de objetos físicos que llevan incorporados sensores, software y otras tecnologías con el fin de conectarse e intercambiar datos con otros dispositivos y sistemas a través de internet.

IOT

La domótica agrupa un conjunto de técnicas que emplean la electrónica, la informática y los automatismos industriales.

DOMÓTICA

La inteligencia artificial(IA) se refiere a sistemas o máquinas que imitan la inteligencia humana para realizar tareas y pueden mejorar iterativamente a partir de la información que recopilan. La IA la podemos encontrar en varias aplicaciones, una de ella es la Domótica, la cual es el conjunto de tecnologías aplicadas al control y la automatización inteligente de la vivienda.

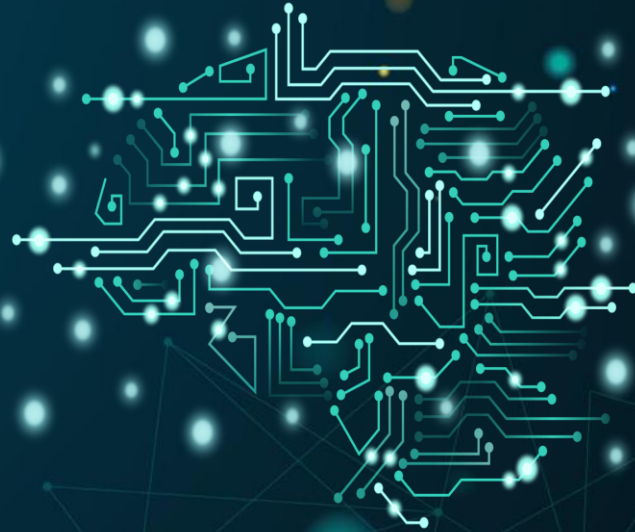
PROBLEMA

El problema a solucionar es la falta de seguridad para la accesibilidad hacia los lugares residenciales, facilitando el ingreso y personalización del hogar a cada miembro de la familia. Las principales implementaciones están dirigidas al acceso de la vivienda, llevándolo un paso mas adelante, el cual es hacer que al momento que detecte a una persona dentro de una habitación, esta se adapte de acuerdo a como tenga personalizado el perfil.



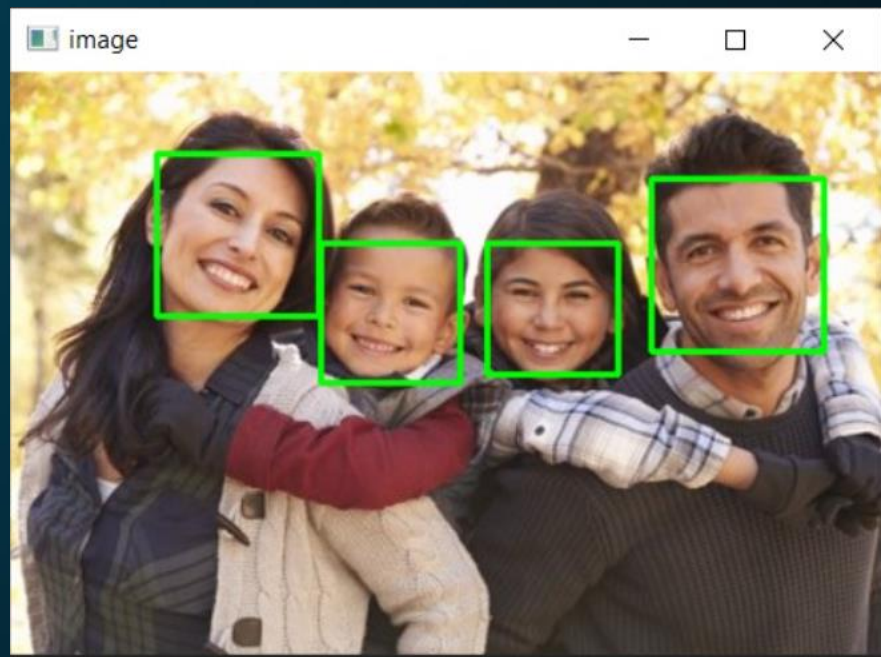
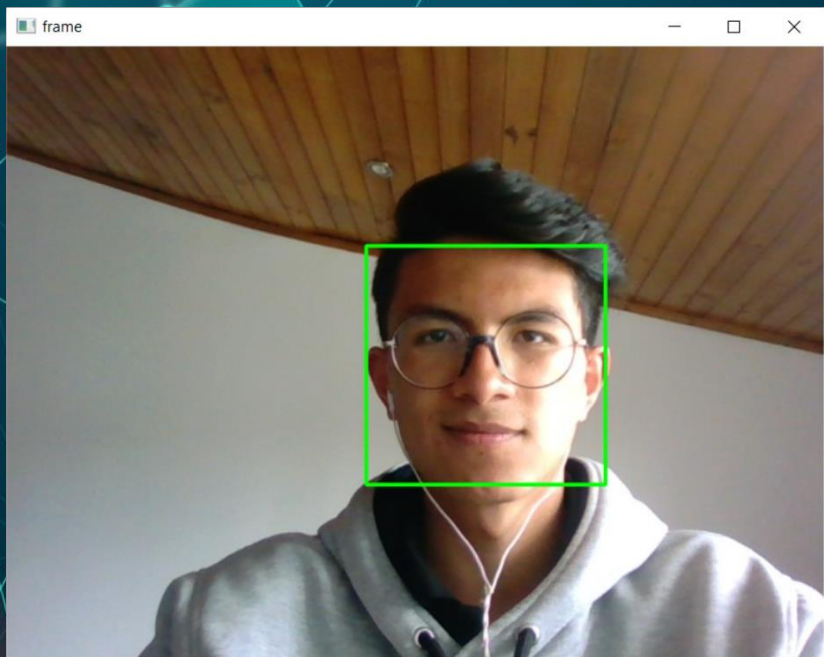
PROTOTIPO

Se implementó por medio de un software desarrollado en Python, debido a que actualmente es de los lenguajes de programación que cuenta con mas librerías que ayuda a terminar con éxito esta implementación.



```
1 import cv2
2 import numpy as np
3
4 #Reconocimiento facial en fotos
5
6 cascPath = "haarcascade_frontalface_default.xml"
7 faceCascade = cv2.CascadeClassifier(cascPath)
8
9 image = cv2.imread("familia.jpg")
10 gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
11
12 faces = faceCascade.detectMultiScale(gray,
13                                     scaleFactor=1.1,
14                                     minNeighbors=5,
15                                     minSize=(30,30),
16                                     maxSize=(200,200))
17
18 for (x,y,w,h) in faces:
19     cv2.rectangle(image,(x,y),(x+w,y+h),(0,255,0),2)
20
21 cv2.imshow("image", image)
22 cv2.waitKey(0)
23 cv2.destroyAllWindows()
```

```
1 import cv2
2 import numpy as np
3
4 #Reconocimiento facial en videos
5
6 cap = cv2.VideoCapture(0)
7
8 cascPath = "haarcascade_frontalface_default.xml"
9 faceCascade = cv2.CascadeClassifier(cascPath)
10
11 while True:
12     ret, frame = cap.read()
13     gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
14
15     faces = faceCascade.detectMultiScale(gray,1.3,5)
16
17     for (x, y, w, h) in faces:
18         cv2.rectangle(frame, (x, y), (x + w, y + h), (0, 255, 0), 2)
19
20     cv2.imshow("frame", frame)
21
22     if cv2.waitKey(1) and 0xFF == ord("q"):
23         break
24
25 cap.release()
26 cv2.destroyAllWindows()
```





GRACIAS

<https://github.com/Juank544/AREP-Seminario>

CREDITS: This presentation template was created by **Slidesgo**, including icons by **Flaticon**, and infographics & images by **Freepik**.