



# Interfaz gráfica para la detección de distanciamiento social usando visión artificial en software libre

**ALEXANDER ESTEBAN ESPINOSA VALDEZ**

**Universidad Magdalena**

Facultad de Ingeniería  
Programa de Ingeniería Electrónica  
Santa Marta, Colombia  
2021



---

# **Interfaz gráfica para la detección de distanciamiento social usando visión artificial en software libre**

**ALEXANDER ESTEBAN ESPINOSA VALDEZ**

Trabajo presentado como requisito parcial para optar al título de:  
**Ingeniero Electronico**

Director:  
Jorge Gómez Rojas, PhD

Procesamiento de Señales e Imágenes  
Grupo de Investigación en Desarrollo Electrónico y Aplicaciones Móviles,  
GIDEAM

Universidad del Magdalena  
Facultad de Ingeniería  
programa de Ingeniería Electrónica  
Santa Marta, Colombia  
2021

---

# Nota de aceptación:

Aprobado por el Consejo de Programa en cumplimiento de los requisitos exigidos por la Universidad del Magdalena para optar al título de Ingeniero electrónico

---

Jurado

---

Jurado

Santa Marta, \_\_\_\_ de \_\_\_\_ del \_\_\_\_\_

---

# Resumen

La visión artificial consiste en tomar imágenes bidimensionales, generalmente obtenidas del mundo tridimensional, para luego hacer un análisis sistemático mediante técnicas y algoritmos de procesamiento de imágenes cuyo propósito es la de generar un modelo aproximado de lo que está representado la imagen. por consiguiente, en este campo se encuentra con diferentes tipos de detectores de objetos que dependiendo de la eficacia y su velocidad se acondicionan a cada aplicación, para la implementación de distanciamiento social de dos personas, se utilizan las siguientes técnicas; la diferencia de centroide y el método de clasificación por distancia euclidiana. para así tener un mejor resultado y que se mantenga el cumplimiento de esta norma

**Palabras claves:** Distanciamiento social, visión artificial, interfaz, código abierto.

## ABSTRACT

Artificial vision consists of taking two-dimensional images, generally obtained from the three-dimensional world, and then making a systematic analysis using image processing techniques and algorithms whose purpose is to generate an approximate model of what the image is represented. Therefore, in this field, there are different types of object detectors that, depending on their efficiency and their speed, are conditioned to each application, for the implementation of social distancing of two people, the following techniques are used; the centroid difference and the Euclidean distance classification method. in order to have a better result and to maintain compliance with this standard

**Keywords:** Social distancing, computer vision, interface, open source.

---

# Contenido

	Pág.
<b>Resumen .....</b>	<b>II</b>
<b>Contenido.....</b>	<b>III</b>
<b>Lista de figuras.....</b>	<b>IV</b>
<b>Lista de tablas .....</b>	<b>V</b>
<b>Introducción.....</b>	<b>1</b>
<b>1. Objetivos .....</b>	<b>3</b>
1.1 Objetivo general.....	3
1.2 Objetivos especificos.....	3
<b>2. Planteamiento del problema.....</b>	<b>4</b>
<b>3. Metodologia .....</b>	<b>5</b>
3.1 Analisis de procesamiento de video.....	5
3.2 Metodo utilizado para la deteccion.....	5
3.3 Tecnica para la estimacion de distanciamiento de personas.....	6
3.4 Calculo de distancia por pixel.....	7
<b>4. Resultados .....</b>	<b>8</b>
4.1 Presentacion del software .....	8
4.2 Funcionamiento interaz .....	11
<b>5. Conclusiones.....</b>	<b>16</b>
<b>A. Anexo: Manual de usuario .....</b>	<b>17</b>
<b>6. Bibliografia .....</b>	<b>18</b>

---

# Lista de figuras

	<b>Pág.</b>
<b>Figura 1:</b> Visualización de una cuadrícula compuesta por píxeles.....	6
<b>Figura 2:</b> Medición de altura .....	7
<b>Figura 3:</b> Interfaz principal .....	9
<b>Figura 4:</b> Interfaz Configuración de cámara.....	10
<b>Figura 5:</b> Interfaz ajuste de medida mm/píxel .....	11
<b>Figura 6:</b> Ventana de la interfaz para calcular valor mm/píxel.....	12
<b>Figura 7:</b> Ventanas de la interfaz .....	13

---

# Lista de tablas

**Pág.**

**Tabla 1:** Relacion de error..... 14

---

# Introducción

Para enfrentar el crecimiento en la curva del COVID-19 se deben mantener el uso adecuado de elementos de bioseguridad personal, la vigilancia activa para la búsqueda oportuna de casos y practicar el distanciamiento social [1]. De estas prácticas, la que mejor resultados ha entregado es el mantener el distanciamiento social [2]–[5].

Sin embargo, estos elementos no siempre son considerados por las personas. Por ello, es necesario que exista una supervisión constante por parte de las autoridades sanitarias o de quien haga sus veces al interior de una organización.

La infraestructura digital de América Latina, en términos generales, no es la más adecuada para implementar estrategias tecnológicas para enfrentar pandemias como en algunas ciudades inteligentes [6]. Al inspeccionar el índice de desarrollo del ecosistema digital [7], América Latina y el Caribe se encuentra por debajo de América del Norte y Europa. A pesar de estar por encima de África y Asia Pacífico, la tasa de crecimiento anual está alrededor del 6.25%.

Actividades como el teletrabajo y la telemedicina han mostrado su eficiencia durante los periodos de aislamiento en Colombia [8].

La Organización Mundial de la Salud (OMS) manifestó que todos los países deben equilibrar la relación de la protección de la salud, la prevención de perturbaciones económicas y sociales y el respeto de los derechos humanos [9].

Es necesario controles para el distanciamiento social en escenarios cerrados como las escuelas y universidades [9].

Seguimiento mediante apps, usando Wifi o Bluetooth desde teléfonos inteligentes como herramienta de interconexión de aplicaciones, brindan una mayor penetración en la población [5], [10]–[13]. Sin embargo, en superficies amplias con alta densidad de usuarios hacen que esta solución sea limitada y poco aplicable.

Plataformas tecnológicas han sido empleadas para captar información sobre comportamientos sociales permitiendo focalización de nichos en enfermedades virales y muertes por razones infecciosas [14].

---

Estrategias de videovigilancia pueden hacer perder la privacidad de una persona, sin embargo, con estrategias de nubosidad intencionada se puede mantener oculta la identidad de personas que no son de interés [15].

Desarrollos en áreas tecnológicas como la visión por computador permite entrenar a computadoras para interpretar y comprender datos visuales como imágenes o videos digitales [12]. Debido a los avances recientes en el aprendizaje de máquina como el reconocimiento de patrones y el aprendizaje profundo, la visión por computador ha permitido a las computadoras identificar y clasificar objetos con exactitud y precisión.

La visión por computador es una estrategia con alta probabilidad de ralentizar la curva de una pandemia mediante la conservación del distanciamiento social. Por ejemplo, la visión por computador permite convertir las cámaras de vigilancia en cámaras inteligentes que además de monitorear a las personas también pueden detectar, reconocer e identificar si las personas cumplen con los requisitos de distanciamiento social o no. Se puede aprovechar la tecnología de visión por computador para el monitoreo de lugares públicos y el monitoreo y detección de personas de alto riesgo (personas en cuarentena y personas con síntomas) [16][5].

En este artículo se presenta una herramienta desarrollada en software libre que permite la vigilancia del distanciamiento social en un entorno controlado.

---

# **1. Objetivos**

## **1.1 Objetivo general**

Desarrollar una interfaz gráfica para la detección de distanciamiento social usando visión artificial.

## **1.2 Objetivos específicos**

Evaluar los diferentes algoritmos de detección.

Diseñar una interfaz gráfica de usuario para la visualización de procesamiento de imágenes.

Divulgar los resultados obtenidos

---

## 2. Planteamiento del problema

A nivel mundial se desarrollan diversas tecnologías que tratan de minimizar la velocidad de contagio de la COVID-19 durante el proceso de finalización de cuarentena en los diferentes países. El distanciamiento social es una de las herramientas principales para combatir la propagación del SARS-CoV-2 mientras se progresa en el hallazgo científico de una cura[1]. Las aplicaciones de visión artificial, que son capaces de detectar calor corporal, distanciamiento social, etc. pueden ayudar a prevenir la expansión del contagio. Por consiguiente, es necesario el desarrollo de una interfaz gráfica para monitorear, de manera remota y segura, el cumplimiento de las normas de distanciamiento social establecidas por el gobierno local.

Actualmente las medidas de prevención se llevan a cabo directamente por el personal encargado de la seguridad del recinto, exponiendo su integridad a un posible contagio, ya que, no se ha divulgado el recurso tecnológico comercial de bajo costo que permita hacer un monitoreo remoto para cumplir con el distanciamiento social.

A raíz de dicha necesidad se plantea una interfaz gráfica para monitorear remotamente el correcto cumplimiento del distanciamiento social en un recinto controlado, logrando identificar cuantitativamente las violaciones que se efectúan esta restricción. A su vez aportándole información de control al personal encargado de vigilar o atender dicho recinto.

El desarrollo de esta interfaz se hará para brindar una alternativa de solución aplicada a la necesidad del cumplimiento de las normas generada por el estado a su vez generando futuras investigaciones con el método establecido en dicha aplicación.

---

## 3. Metodología

La metodología aplicada está dividida en tres fases. En la primera, se describe el análisis y el procesamiento de videos. Posteriormente, se describe la fase de elección del método para la detección de personas y ajuste de mm/pixel para finalmente, se expone la técnica para la detección de distanciamiento.

### 3.1 Análisis y procesamiento de videos

Para el análisis y procesamiento de videos e imágenes lo hacemos por medio de *Opencv*, una biblioteca de código abierto creada por Intel. Esta herramienta está diseñada con un enfoque computacional eficiente para aplicaciones en tiempo real de visión por computadora [17]. La implementación se realizó por medio del lenguaje de programación Python, dada su versatilidad, interfaz completa y un manejo poco complejo. Esta biblioteca nos permite la manipulación, ajuste y marcado de imágenes donde se logra proyectar los resultados visuales del proceso de detección de distanciamiento social.

Además, cuenta con otras librerías que permiten darle forma geométrica a contornos, añadir texto y manipular la cámara. Es importante destacar el módulo de Deep Neural Networks (dnn).

### 3.2 Método utilizado para la detección de personas

Dentro del campo de visión por computador, se encuentran diferentes tipos de detectores. Los tres detectores con mejores resultados en desarrollos previos son: R-CNN [18] con todas sus alternativas, Fast R-CNN, Faster R-CNN como también SSD (Single Shot Detector) y YOLO (You Only Look Once).

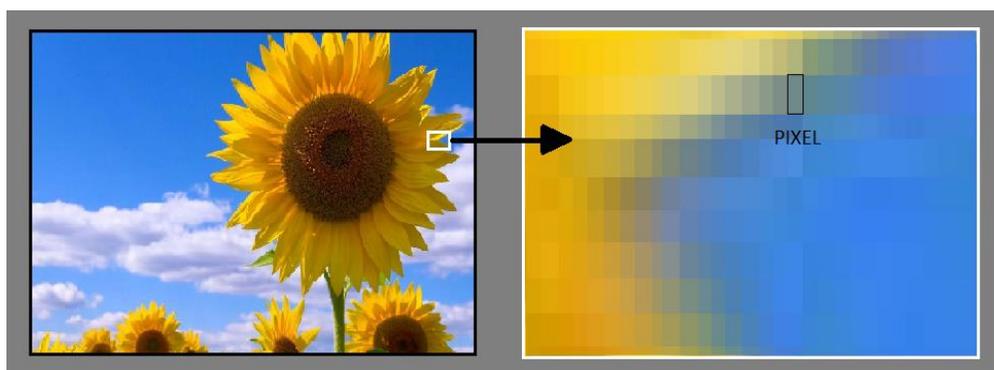
Este último es considerado el más adecuado ya que utiliza *deep learning* y CNN para detectar objetos. La ventaja que proporciona, además de trabajar con una sola red de convolución [19], es la velocidad de detección que tiene. Este método permite encontrar más rápido los objetos en tiempo real

---

El uso de YOLO en el algoritmo implementado permite un proceso rápido con menos coste computacional, logrando ejecutar este algoritmo en equipos de gama media con pequeñas pérdidas y una mayor cantidad en el número de cuadros por segundo [20] con respecto a los detectores que lo asemejan. Además, la gran comunidad ha permitido acceso a los diferentes conjuntos de entrenamiento que son de uso universal, ya que se pueden aplicar en diferentes ambientes sin perder su eficiencia.

### 3.3 Técnica para la estimación de distanciamiento entre personas.

Una imagen digitalizada se puede entender como una cuadrícula compuesta por píxeles de un tamaño PxP que corresponden a la cantidad de píxeles horizontales y verticales respectivamente [21], tal como se muestra en la figura 1.



*Fig. 1. Visualización de una cuadrícula compuesta por píxeles.*

*(Fuente: Autor).*

Para la estimación del distanciamiento de dos personas se utiliza el método de clasificación por Distancia Euclidiana[22]. El objetivo de esto es obtener precisión y valores simultáneos entre la imagen de entrada y la imagen de entrenamiento. Matemáticamente, permite encontrar un número positivo que indica la separación de dos puntos. Para la aplicación desarrollada, por ser un plano, es empleada la distancia euclidiana en 2D. Para determinar distancia euclidiana en un plano se emplea la ecuación (1):

---

$$D = \sqrt{\sum_{i=0}^N (x_1 - X_2)^2} \quad (1)$$

donde:

D: Valor de distancia

x1: Patrón de imagen de entrada

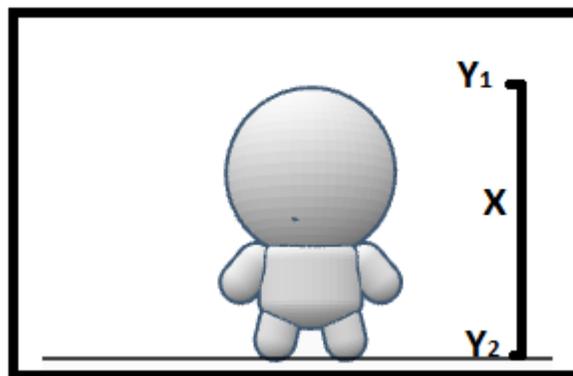
x2: Patrón de imagen de entrenamiento

Posteriormente, utilizamos la distancia euclidiana dada en cantidad de píxeles para determinar el espacio entre 2 centroides en metros.

### 3.4 Cálculo de distancia por píxel.

En esta sección se calcula la distancia la cual representa cada píxel en nuestro ambiente.

Para obtener la medida del píxel se procede a utilizar una persona de prueba, el cual se toma como referencia su estatura de tamaño X (en centímetros). previamente conocido, por medio del método de detección se calcula el tamaño en píxeles dentro de la imagen (Fig 2).



*Fig. 2. Medición de altura.  
(Fuente: Autor).*

---

Luego de obtener los resultados anteriores se procede a dividir la altura X de la persona entre la cantidad de pixeles que corresponden a la altura de la detección, como se observa en la ecuación (2), para este último se obtiene con la sustracción de los puntos  $y_2 - y_1$  ya que la magnitud de los pixeles en el eje vertical aumenta a medida que desciende a un punto inferior.

$$\text{mm/píxel} = (X * 10) / (y_2 - y_1) \quad (2)$$

Donde:

X la altura de la persona está en centímetros

Luego de obtener esta medida se multiplica por la distancia euclidiana entre los diferentes centroides obtenidos para determinar si supera o no el distanciamiento social (2 metros)

## 4. Resultado

Se desarrolló un software que se compone de tres interfaces que facilitan la interacción y recolección de recursos para el estudio.

### 4.1 Presentación del software:

interfaz 1: Es la ventana principal del software como se muestra en la figura 3 y se compone de:

1. Permite escoger la forma de procesamiento, en una ventana desplegable; imagen, vídeo o cámara en tiempo real.
2. Botón que da acceso a la segunda interfaz para la configuración de cámara implementado para procesamiento en tiempo real.
3. Botones de inicio y parada para procesamiento de video o cámara en tiempo real.
4. Permite buscar el archivo de imagen o video por medio del botón búsqueda en archivos. Al escoger el archivo se visualiza su dirección en el equipo.
5. Campo para visualizar el número de personas que violan el distanciamiento social en un cuadro.

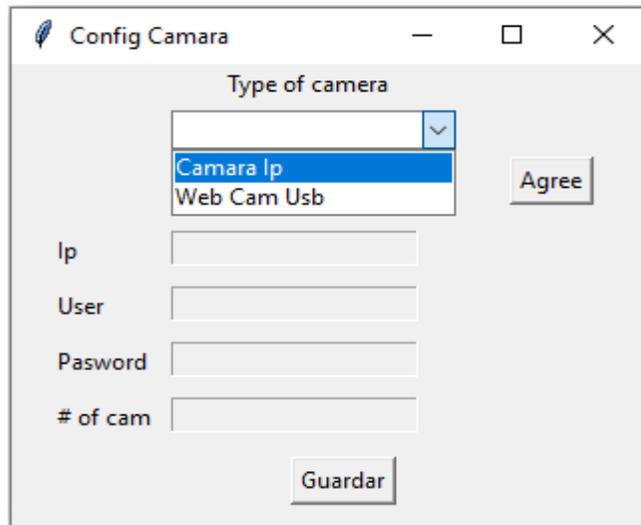
- 
6. Campo para visualizar el número de personas que se detectan en un cuadro.
  7. Botón que permite acceder a una tercera interfaz donde se realiza el cálculo de mm/pixel
  8. Zona de visualización de cuadro no procesado(Izquierda) y cuadro procesado (derecha).



*Fig. 3. Interfaz principal.*

*(Fuente: Autor).*

**Interfaz 2 :** La ventana mostrada en la figura 4, es utilizada principalmente cuando el propósito es hacer procesamiento en tiempo real, en la cual se puede configurar la conexión de cámara USB y cámaras Ip; Para estas últimas debe asegurarse estar conectadas a la misma red.

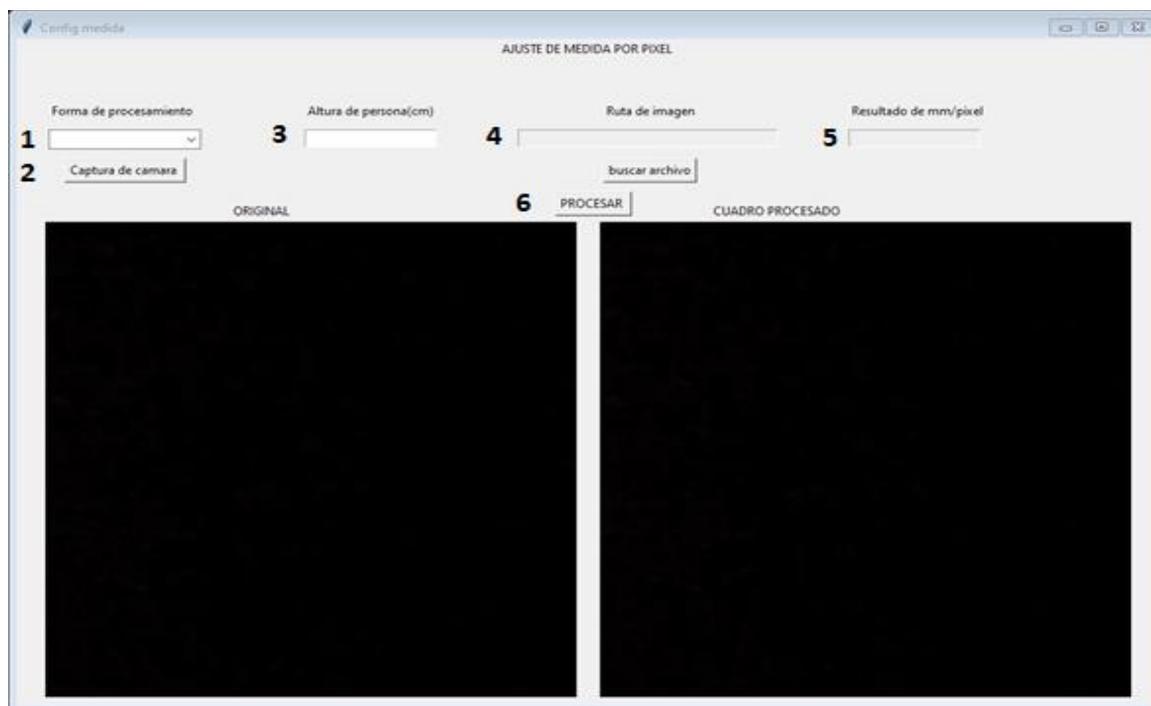


*Fig. 4. Interfaz Configuración de cámara.  
(Fuente: Autor).*

**Interfaz 3:** como se observa la figura 5, es la ventana para ajustar la medida de milímetro por pixel el cual depende del ambiente en que se use la aplicación.

1. Permite escoger la forma de procesamiento en una ventana desplegable imagen, vídeo o cámara en tiempo real.
2. Botón que corresponde a la captura en tiempo real si se elige como forma de procesamiento captura en vivo en el anterior ítem. La captura se visualiza en el cuadro inferior izquierdo.
3. Campo en el cual se debe ingresar la altura en centímetros (cm) de la persona referencia; El formato aceptado es un número entero.
4. Permite buscar el archivo de imagen o video por medio del botón búsqueda en archivos. Al escoger el archivo se visualiza su dirección en el equipo.
5. Campo para visualizar el resultado del cálculo de la medida mm/pixel que se almacenará para posteriormente ser utilizado en el cálculo de distancias.

6. Botón para iniciar el procesamiento de imagen para el cálculo de la medida mm/pixel, este debe ser presionado cuando se obtiene la configuración del ítem (1) y (3), en el caso de escoger el ítem (1) captura en tiempo real se debe realizar las instrucciones del ítem (2).



*Fig. 5. Interfaz ajuste de medida mm/pixel.  
(Fuente: Autor).*

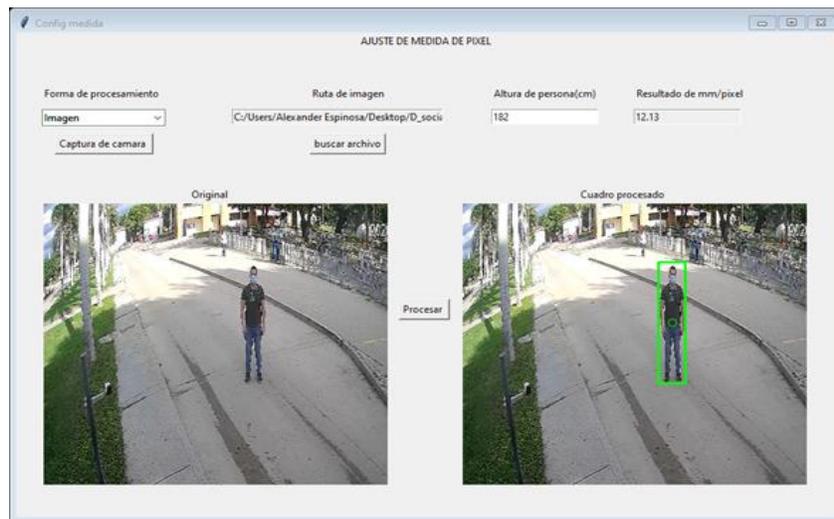
## 4.2 Funcionamiento interfaz:

En esta sección, se muestra el procedimiento y resultados de los modelos presentados en las pruebas para la clasificación de detección de personas en distanciamiento social. Los ensayos de detección se realizaron con Darknet-53 de ImageNet como red troncal y el conjunto de datos entrenados con COCO. La interfaz se ejecuta en una máquina de cómputo con sistema operativo Windows 10, de gama media con GPU NVIDIA GeForce

MX 110, con RAM de 8Gb y puesta a prueba en una cámara de seguridad instalada con una altura de 2.7 metros con vista diagonal a la entrada vehicular de la Universidad Del Magdalena.

En el ambiente de estudio se observa una persona de 182 cm de altura como punto de referencia inicial, donde, su medida dentro de la imagen corresponde a 150 píxeles y aplicando la ecuación (2) se obtuvo un valor de 12.13 mm/píxel.

En la figura 6 se muestra la primera ventana de la interfaz que nos permite calcular el valor que representa un píxel en milímetros, por medio de una imagen o captura en tiempo real ajustada a 500X500 píxeles.



*Fig. 6. Ventana de la interfaz para calcular valor mm/píxel.  
(Fuente: Autor).*

Con el valor obtenido se examina en diferentes medidas, variando la profundidad con referencia al punto inicial, se verifica con 3 medidas básicas en el distanciamiento social de 2 personas separadas 100 cm, 150 cm y 200 cm en cada punto, obteniendo la distancia en píxeles como se muestra en la figura 7.

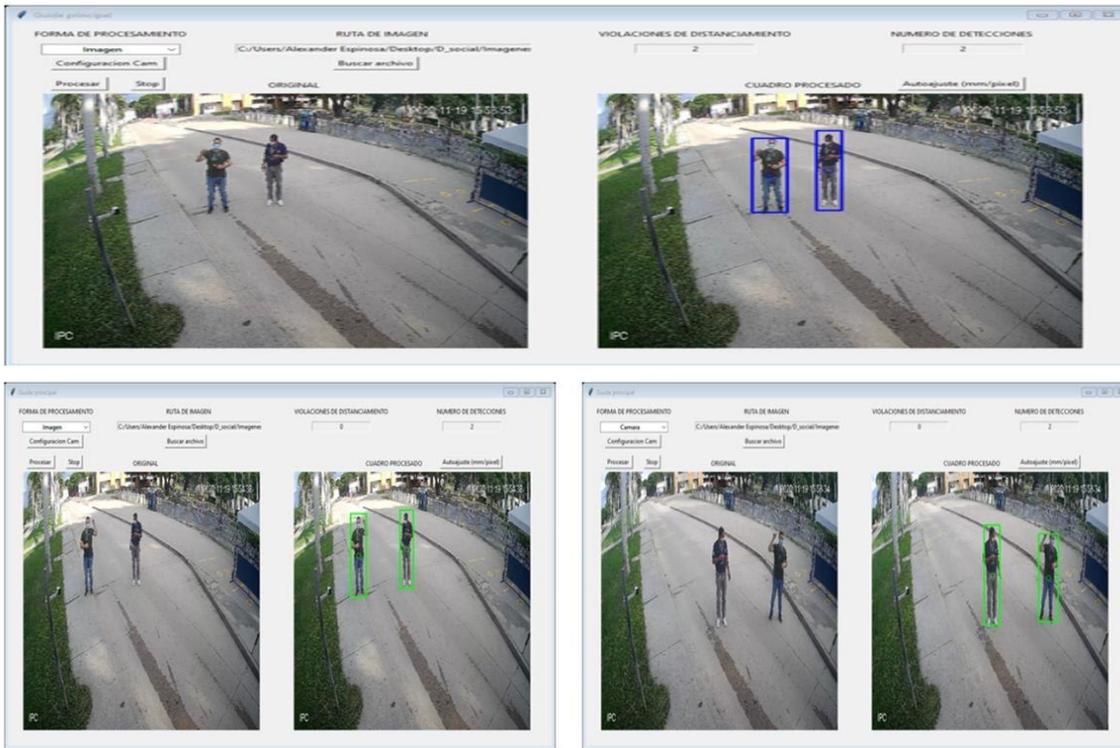


Fig. 7. Ventanas de la interfaz .  
(Fuente: Autor).

En la tabla 1, se presentan la relación de error con respecto a las medidas obtenidas con un flexómetro del centroide 1 al centroide 2 de las personas presentadas en las pruebas, donde se presenta un error promedio de 13.10% para los puntos paralelos a la posición de ajuste de píxel, además, en los puntos examinados entre la posición inicial de referencia y la cámara se obtuvo un promedio de eficiencia de 88.4% en el cálculo de sus medidas, mientras las medidas obtenidas alejados de la cámara su error aumenta hasta un 42.26%

Profundidad	Distancia Real	Distancia Calculada	Porcentaje de error
Posición inicial	100 cm	89.51	10.49%
	150 cm	128.34	14.44%
	200 cm	171,24	14,38%
100 cm del punto inicial al opuesto de la cámara	100 cm	75.39	24.61%
	150 cm	117.34	21.77%
	200 cm	154.21	22.89%
200 cm del punto inicial al opuesto de la cámara	100 cm	54.21	45.79%
	150 cm	85.23	43.18%
	200 cm	124.35	37.82%
200 cm del punto inicial hacia la cámara	100 cm	94.4	5.6%
	150 cm	128.77	14.15%
	200 cm	169.9	15.05%

Tabla 1. Relación de error

La interfaz implementada para la detección del distanciamiento por medio de videovigilancia en la Universidad Del Magdalena, logra una eficiencia del 88.4% , ya que, por tener una vista diagonal y una altura de 2.7m se llega a considerar mucho la posición donde se encuentra las personas, como también la iluminación, la profundidad que se presente con respecto al lente de la cámara, debido a que la detección se ve afectada por su localización espacial alterando la ubicación del centroide y a su vez el cálculo de distanciamiento de personas. Con estos resultados obtenidos se dio la iniciativa de divulgar un artículo científico en la revista Mundo Fesc, como también un manual de usuario para que cualquiera persona sin conocimiento del tema, pueda usar el software y pueda navegar, desplazarse por cada una de las interfaces existentes, sin complicación alguna, por medio de un equipo con las características mencionadas anteriormente.

## 5. Conclusiones

El algoritmo YOLOv3 permite una detección eficiente con una respuesta en poco tiempo, ideal para realizar aplicación de análisis de imágenes en tiempo real en equipos de baja capacidad.

En el caso de establecer si se viola o no el distanciamiento social, el cálculo de la separación por medio de la distancia euclidiana no tiene en cuenta la profundidad en la que se encuentran los objetos a medir. Afectando la precisión de detección del distanciamiento.

El método propuesto nos permite adaptar un sistema de videovigilancia en un detector de violaciones de distanciamiento social después de realizar una calibración del sistema para el ambiente en el que se implemente, la eficiencia del sistema depende del proceso de ajuste de la medida por pixel que puede variar por diferentes factores como: profundidad, iluminación, vista, resolución y postura.

El método utilizado en la interfaz, para la detección nos arroja un porcentaje aproximado de eficiencia del 88.4% en el cálculo de medidas a una altura de 2.7m

Se presenta un documento donde se muestra los pasos necesarios para su correcta instalación y funcionamiento donde también se describe la metodología empleada para su ejecución

# **Anexo: Manual de usuario**

Archivo en el cual indica cómo usar la interfaz desarrollada

# Bibliografía

1. J. Bedford *et al.*, “COVID-19: towards controlling of a pandemic,” *Lancet*, vol. 395, no. 10229, pp. 1015–1018, 2020, doi: 10.1016/S0140-6736(20)30673-5.
2. J. Y. Noh, H. Seong, J. G. Yoon, J. Y. Song, H. J. Cheong, and W. J. Kim, “Social distancing against COVID-19: Implication for the control of influenza,” *J. Korean Med. Sci.*, vol. 35, no. 19, pp. 10–14, 2020, doi: 10.3346/JKMS.2020.35.E182.
3. S. Kissler, C. Tedijanto, M. Lipsitch, and Y. Grad, “Social distancing strategies for curbing the COVID-19 epidemic,” *medRxiv*, vol. 46, no. 4, pp. 385–395, 2020, doi: 10.31857/s0132342320040272.
4. N. S. Punn, S. K. Sonbhadra, and S. Agarwal, “Monitoring COVID-19 social distancing with person detection and tracking via fine-tuned YOLO v3 and Deepsort techniques,” pp. 1–10, 2020, [Online]. Available: <http://arxiv.org/abs/2005.01385>.
5. C. T. Nguyen *et al.*, “Enabling and Emerging Technologies for Social Distancing: A Comprehensive Survey,” pp. 1–42, 2020, [Online]. Available: <http://arxiv.org/abs/2005.02816>.
6. M. Gupta, M. Abdelsalam, and S. Mittal, “Enabling and Enforcing Social Distancing Measures using Smart City and ITS Infrastructures: A COVID-19 Use Case,” pp. 1–5, 2020, [Online]. Available: <http://arxiv.org/abs/2004.09246>.
7. CAF, C.-S. tecnica del ELAC, D. P. and Law, and Te. A. S. LLC, “La Digitalización En América Latina Frente Al Covid-19,” *Corporación Andin. Fomento, 2020. Nac. Unidas, 2020*, 2020.
8. J. R. Márquez Velásquez, “Teleconsulta en la pandemia por Coronavirus: desafíos para la telemedicina pos-COVID-19,” *Rev. Colomb. Gastroenterol.*, vol. 35, no. Supl. 1, pp. 5–16, 2020, doi: 10.22516/25007440.543.
9. R. M. Viner *et al.*, “School closure and management practices during coronavirus outbreaks including COVID-19: a rapid systematic review,” *Lancet Child Adolesc. Heal.*, vol. 4, no. 5, pp. 397–404, 2020, doi: 10.1016/S2352-4642(20)30095-X.
10. Y. B. Liang, D. M. Li, J. Y. Yao, W. Jin, C. B. Gong, and R. Y. Liu, “Progresses in investigation and research on phycotoxins and toxic microalgae in the coastal waters of China,” *Oceanol. Limnol. Sin.*, vol. 50, no. 3, pp. 511–524, 2019, doi: 10.11693/hyhz20181000233
11. C. Menni *et al.*, “Real-time tracking of self-reported symptoms to predict potential COVID-19,” *Nat. Med.*, 2020, doi: 10.1038/s41591-020-0916-2.
12. A. Toliyat, “Tracking Citizen ’ s Concerns during COVID-19 Pandemic,” in *21st Annual International Conference on Digital Government Research*, 2020, pp. 322–323, doi: 10.1145/3396956.3397000.
13. Y. Mejova and K. Kalimeri, “COVID-19 on Facebook Ads : Competing Agendas around a Public Health Crisis,” in *ISI Foundation*, 2020, pp. 22–31.

14. F. Shahid, S. H. Ony, T. R. Albi, S. Chellappan, A. Vashistha, and A. B. M. Alim Al Islam, "Learning from Tweets: Opportunities and Challenges to Inform Policy Making during Dengue Epidemic," *Proc. ACM Human-Computer Interact.*, vol. 4, no. CSCW1, pp. 1–27, 2020, doi: 10.1145/3392875.
15. M. Grambow, J. Hasenbug, and D. Bermbach, "Public video surveillance: Using the fog to increase privacy," *M4IOT 2018 - Proc. 2018 Work. Middlew. Appl. Internet Things, Part Middlew. 2018 Conf.*, pp. 11–14, 2018, doi: 10.1145/3286719.3286722.
16. A. Torabi, G. Massé, and G. A. Bilodeau, "An iterative integrated framework for thermal-visible image registration, sensor fusion, and people tracking for video surveillance applications," *Comput. Vis. Image Underst.*, vol. 116, no. 2, pp. 210–221, 2012, doi: 10.1016/j.cviu.2011.10.006.
17. Opencv.org. Acerca de. <https://opencv.org/about/> (consultado 10/08/20)
18. S. Ren, K. He, R. Girshick y J. Sun, "R-cnn más rápido: hacia el tiempo real detección de objetos con redes de propuesta de región ", en *Avances en sistemas de procesamiento de información neuronal*, 2015, págs. 91–99
19. J. Redmon, A. Farhadi, YOLOv3, An Incremental Improvement, 2018
20. N. Singh, S. Kumar and S. Agarwal ,Monitoring COVID-19 social distancing with person detection and tracking via fine-tuned YOLO v3 and Deepsort techniques, 2020
21. A. Cosa, J. Fernandez, La imagen digital, Revista Española Ortodoncia ,2005
22. F. Dwi Aferi , T. Waluyo and R. Erfa Saputra, Cotton Texture Segmentation Based On Image Texture Analysis Using Gray Level Co-occurrence Matrix (GLCM) And Euclidean Distance, 2018