

Técnicas de visión artificial y su aplicación en el sector salud. Caso práctico: Conteo de huevecillos mosquito Aedes Aegypti.

Artificial vision techniques and their application in the health sector. Practical case: Aedes Aegypti mosquito egg count.

Gil Santana Esparza* (1).
Tecnológico Nacional de México, Instituto Tecnológico Superior de Pánuco.
gil.santana@itspanuco.edu.mx.

Armando Hernández Machuca (2). TecNM, I. T. S. de Pánuco, armando.hernandez@itspanuco.edu.mx.

*corresponding author.

Artículo recibido en febrero 25, 2023; aceptado en abril 05, 2023.

Resumen.

El presente artículo muestra el resultado de aplicar la técnica de procesamiento de imágenes en la visión artificial, derivado del proyecto en el cual uno de los objetivos fue desarrollar un algoritmo computacional para detección y conteo de huevecillos del mosquito Aedes Aegypti en ovitrampas del sector salud, aplicando la metodología para análisis automatizado de imágenes a través de visión artificial. Se presenta como resultado final un programa computacional que recibe como entrada una imagen de un pellón de ovitrampa, dicha imagen es procesada con el algoritmo que analiza y detecta los huevecillos presentes en el pellón, para proporcionar de salida la cantidad de huevecillos detectados en la imagen. Para finalizar, se concluye que el proceso automático del conteo de huevecillos con técnicas de visión artificial proporciona información inmediata y confiable que permite al sector salud tomar decisiones oportunas para implementar acciones de respuesta a la comunidad circundante de la región para prevenir la propagación del dengue.

Palabras clave: Procesamiento de imágenes, sector salud, visión artificial.

Abstract.

This article shows the result of applying the image processing technique in artificial vision, derived from the project whose objective was to develop a computational algorithm for detection and counting of Aedes Aegypti mosquito eggs in ovitraps in the health sector, applying the methodology for analysis. automated imaging through artificial vision. The final result is a computer program that receives as input an image of an ovitrap puff, said image is processed with the algorithm that analyzes and detects the eggs present in the puff, to provide output with the number of eggs detected in the image. . Finally, it is concluded that the automatic egg count process with artificial vision techniques provides immediate and reliable information that allows the health sector to make timely decisions to implement response actions to the surrounding community in the region to prevent the spread of dengue.

Keywords: Artificial vision, health sector, image processing.

1. Introducción.

Al observar la naturaleza nos damos cuenta de que las criaturas que dentro de ella habitan son capaces de realizar tareas que parece que lo hacen de manera perfecta. Dichas tareas no serían posibles si estas criaturas no contaran con los sentidos que les permitieran percibir su respectivo medio ambiente, así como las capacidades para poder procesar estas informaciones de manera conveniente. En lo particular, la visión es uno de los sentidos con el que casi cualquier criatura cuenta. Este sentido le permite a la criatura visualizar el mundo circundante facilitándole a la vez tomar las decisiones que más le convengan. Es decir, la percepción visual es necesaria no sólo por que ver es divertido, sino porque es necesaria para poder tener una idea mejor de lo que una acción puede llegar a representar (Russel, S. y Norving, P., 2004).

Podemos definir, entonces, percepción visual humana como la habilidad para procesar, interpretar y analizar información recibida del espectro de la luz visible mediante nuestros ojos. Sin embargo, con las nuevas tecnologías, esta definición se podría ampliar a otras longitudes de onda distintas de la luz visible, y esta nueva definición se aplicaría también a la percepción visual por computadora (Chacón, M. et al., 2015).

El dotar a una máquina de las capacidades para percibir y analizar ha sido, desde hace más de 50 años, uno de los retos científicos y tecnológicos con que la comunidad científica se ha enfrentado. Estas capacidades deberían quedar integradas en lo que actualmente se conoce como un sistema de visión artificial.

La visión por computadora puede ser definida como la capacidad con que una máquina debería contar para ser capaz de tratar y analizar una o más imágenes de su entorno, con el objeto de que dicha máquina pueda desarrollar de forma automática una tarea dada (Rodríguez, R. y Sossa, J., 2012).

Así también, el tratamiento o procesamiento digital de imágenes (TDI o PDI) es una de las etapas primarias del proceso de la visión por computadora. Se puede definir como un conjunto de procedimientos que se realizan sobre una imagen para su almacenamiento, transmisión o tratamiento. El PDI recibe como entrada una imagen y produce como salida otra imagen con las mismas dimensiones que la imagen de entrada, además puede ser utilizado para entregar un resultado de más alto nivel como podría ser la identidad de los objetos presentes en dicha imagen de entrada o incluso una interpretación de alto nivel de dicha imagen (Rodríguez, R. y Sossa, J., 2012).

Se puede decir también, que el procesamiento de imágenes es un conjunto de técnicas que toma una imagen de entrada y genera otra de salida. Y esto se puede hacer: Pixel a pixel: en el que cada pixel de la imagen de salida depende únicamente de un pixel de la imagen de entrada. O bien, por áreas de pixeles vecinos: en el que cada pixel de la imagen de salida depende de un conjunto de pixeles vecinos anexos a un determinado pixel de la imagen de entrada (Domínguez, T., 2021a).

Y dentro de las técnicas del procesamiento de imágenes se encuentra la segmentación, en la cual se puede aplicar filtrado basado en umbral como son el umbral fijo y el umbral adaptativo. En el umbral fijo, el umbral se representa con un valor entero en el rango de 0 a 255. El umbral adaptativo tiene variación de valor en cada punto, para calcularlo se toma un área determinada alrededor del punto de la imagen origen y se obtiene la media de la intensidad de los puntos que lo rodean, agregando un margen de seguridad. También se encuentran los filtros lineales, en los cuales para el cálculo de nivel de intensidad de cada pixel de la imagen destino, se utiliza una función lineal aplicada sobre el correspondiente pixel de la imagen origen. Algunos de los filtros lineales son, el filtro paso bajo que sirve para suavizar la imagen; el filtro paso alto, que se usa para realzar las zonas de la imagen aumentando su contraste; filtro de detección de bordes, que permiten identificar objetos con propiedades similares. Y por otro lado, se encuentran los filtros morfológicos, que procesan la imagen basándose en su forma (Domínguez, T., 2021b).

De tal manera que, el análisis de imágenes por computadora es un procedimiento que tiene como objetivo procesar, identificar e interpretar de forma automática los aspectos más relevantes presentes en una imagen que ha sido previamente digitalizada. Para ello, es necesario conocer la problemática abordada por el análisis automatizado de imágenes que tiene que ver con (Rodríguez, R. y Sossa, J., 2012):

- a) El reconocimiento de objetos: consiste en la determinación en forma automática de la identidad de los objetos en una imagen a partir de un conjunto de rasgos descriptivos extraídos a partir de una o más imágenes de dicho objeto. Desde el punto de vista de la clasificación, el reconocimiento de objetos puede ser visto como el problema de encontrar las clases a las cuales pertenecen cada uno de los objetos que se encuentran en la imagen.
- b) Detección de objetos: tiene que ver con la determinación de si un objeto dado se encuentra o no en una imagen. Para esto se puede utilizar, una descripción del objeto a detectar o una imagen recortada de dicho objeto.

Derivado de lo anterior, en el presente artículo se abordan los conceptos, elementos y técnicas computacionales de los sistemas de visión artificial con la intención de crear aplicaciones para uso habitual en los campos de la vida moderna, en este caso, desarrollando un algoritmo para detección y conteo de huevecillos del mosquito Aedes Aegypti en ovitrampas tratando de impactar con ello en el sector salud proporcionando información útil que apoye las estrategias de disminución del mosquito y en consecuencia con la disminución de la enfermedad transmitida (dengue).

2. Métodos.

Con base al análisis situacional previo, realizado en conjunto con personal de la Jurisdicción Sanitaria No.1 de Pánuco, Veracruz, en el cual se conoció el proceso de vigilancia y monitoreo de las ovitrampas instaladas en la zona, se ha detectado la necesidad de vigilar y cuantificar la cantidad de huevecillos producidos por el mosquito Aedes Aegypti transmisor de la enfermedad del dengue. El hecho de cuantificar los huevecillos captados en las ovitrampas le proporciona al sector salud información para determinar las estrategias a implementar para combatir la proliferación de dicho mosquito. Sin embargo, se percibe que este proceso de vigilancia y cuantificación se realiza de forma manual, por lo cual sería importante realizarlo con base a tecnologías emergentes que coadyuven en esta tarea para aumentar la optimización del tiempo y los niveles de confianza en el conteo y monitoreo. Para realizar lo anterior se hizo la propuesta de la técnica de visión artificial, estableciendo el periodo de enero a diciembre del 2021 como marco temporal para el desarrollo y validación de la herramienta computacional con técnicas de visión artificial para lograr el objetivo planteado.

Para ello, se ha optado en seguir las etapas para el análisis automatizado de imágenes mostrado en la Figura 1.

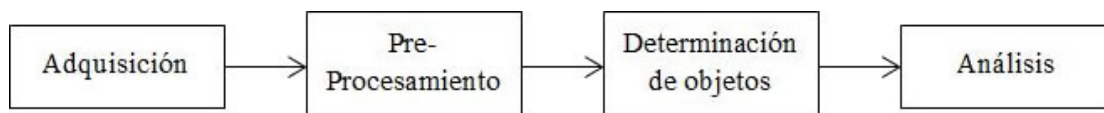


Figura 1. Etapas en un sistema de análisis de imágenes. Adaptado de Chacón, M. et al. (2015).

En la Figura 1, en la adquisición a través del hardware se obtiene una imagen digital que se almacena normalmente a través de algún formato característico de imágenes (TIFF, BMP, JPEG, etc.), de tal manera que la interpretación de la imagen se representa desde el punto de vista de su tratamiento computacional como una matriz numérica de dimensión $M \times N$, el contenido de esa matriz se refiere a valores enteros situados en localizaciones espaciales (x,y) denominados píxeles (Pajares, G. y De la Cruz, J. 2008). En el pre-procesamiento, se aplican acciones sobre la imagen que definen a la escena, las cuales corresponden al suavizado, umbralización, eliminación de ruido, análisis de textura, etc. En la determinación de objetos, se tienen acciones como definición de límites, definición de bordes, regiones, superficies, que están relacionadas con generar objetos presentes en la imagen. En el análisis, se entablan relaciones entre objetos para realizar la interpretación o descripción de la escena (Chacón, M. et al., 2015).

Se ha tomado como punto de partida el hecho de que, para una computadora, las imágenes no son más que una sucesión de puntos, o píxeles, dispuestos en filas y columnas sobre una pantalla. Cada uno de dichos píxeles se identifica por sus coordenadas y el nivel de intensidad con el que se visualiza. En las imágenes en blanco y negro

dicho nivel de intensidad es un valor en el rango 0 – 255, siendo 0 la oscuridad absoluta y 255 la máxima luminosidad. Entre medias estaría toda la escala de grises. En las imágenes en color cada pixel se compone de tres valores, correspondientes a la intensidad de los tres colores primarios: rojo, verde y azul. De tal manera que para el desarrollo del proyecto, del cual surge el presente artículo, se han aplicado los aportes conceptuales de los diversos autores consultados así como las cuatro fases: adquisición, pre-procesamiento, determinación de objetos y análisis de imágenes del mundo real.

3. Desarrollo.

En la adquisición, el mundo analógico de luces y colores que rodea a la imagen se ha traducido a datos binarios compuestos de ceros y unos. En el pre-procesamiento de la imagen se ha realizado extracción de información útil de ésta, aplicando funciones con las que se ha realizado el filtrado para centrarse en los objetos de interés. En la determinación de objetos, se han detectado los bordes descubriendo patrones de sus características determinando las superficies. En el análisis de la imagen y con base también en funciones y librerías del lenguaje se ha partido del paso anterior para realizar el reconocimiento de los objetos, identificándolos y haciendo el conteo de los mismos. El Algoritmo 1, presenta las líneas de código en lenguaje *Processing* de todas las acciones redactadas en las cuatro fases.

Adquisición.

Para la adquisición, como elemento *hardware*, se ha utilizado una cámara web convencional conectada a una laptop, la cámara de un dispositivo móvil *android* y una cámara microscópica 1000x. Y como elemento *software* se ha utilizado el lenguaje de programación *Processing* versión 3.5.4 acompañado de la librería de software *OpenCV*, que han permitido en su conjunto, desarrollar una aplicación de visión artificial en tiempo real. Para ello, se aplicó la clase *PImage* para la representación en la pantalla, obteniendo con dicha clase la información necesaria del ancho, alto y la matriz de valores de los pixeles de las imágenes tratadas. Así también, se utilizó la función *loadImage()* para crear objetos para la clase *PImage* trabajando en este caso, el formato jpg. Un ejemplo de esta fase de adquisición se muestra en la Figura 2 que contiene en formato jpg un pellón recolectado por el sector salud con huevecillos del mosquito *Aedes Aegypti*, y que se ha utilizado para aplicar el algoritmo de visión artificial desarrollado.

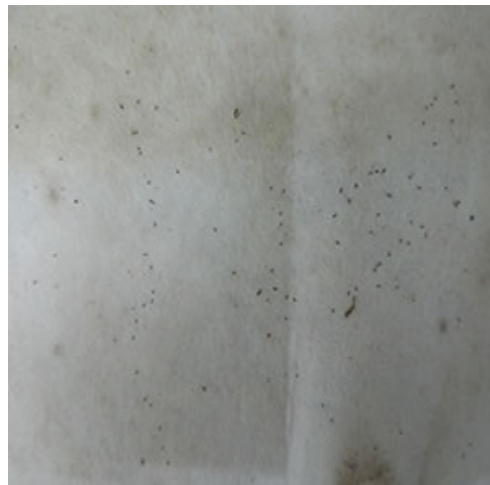


Figura 2. Pellón “A” recolectado por el Sector Salud con huevecillos del mosquito *Aedes Aegypti*. Capturada la imagen con cámara web convencional.

Pre-procesamiento.

Dentro de las técnicas que se aplicaron en el proyecto para el procesamiento de imagen se encuentra la binarización, en la cual lo que se hizo fue componer la imagen destino a partir de los pixeles de la imagen origen cuyo valor de

intensidad ha superado un determinado valor en el rango de 0 a 255. En este caso, la imagen destino será de color blanco en las zonas de la imagen origen cuyos pixeles tengan un valor mayor que el del umbral, dejando el resto en negro, logrando con la binarización que la imagen destino esté compuesta de blancos y negros puros (no hay grises). La aplicación de esta técnica en el desarrollo del proyecto fue muy útil ya que ha permitido seleccionar las regiones de la imagen que contienen los objetos sobre los que se requiere centrar la atención, descartando el resto, de tal manera que con dicha técnica se ha logrado un proceso de separación que se conoce técnicamente como segmentación. Para lograr lo anterior, se aplicó el tipo de filtrado basado en umbral fijo con valor de 80, como se muestra en la Figura 3.



Figura 3. Pellón “B”. Segmentación con umbral fijo en valor de 80. Capturada la imagen con una cámara microscópica digital 1000x.

Determinación de objetos.

Una vez aplicada la binarización de la imagen origen, se procedió a localizar los contornos de la imagen, y en última instancia, los objetos que contienen, para lo cual se creó un *ArrayList* (lista de datos para almacenar cada objeto de interés) con los contornos identificados aplicando los métodos de la clase *Contour* de *OpenCV*, dando como resultado la identificación como se muestra en la Figura 4.



Figura 4. Pellón “B”. Identificación de contornos de los huevecillos.

Una vez que ha sido posible identificar los posibles contornos pertenecientes a los huevecillos del mosquito Aedes Aegypti, y tomando en cuenta que, en este caso, lo que se busca es el conteo de los huevecillos, se determinó que la mejor técnica para discriminar y evitar falsos positivos, se aplicaría la identificación final aplicando cálculo de áreas con base a un análisis previo de rangos en los cuales se encuentran los tamaños de los huevecillos. Y para ello, se determinaron áreas rectangulares, aplicando el método *getBoundingBox()* para detectar las coordenadas de la esquina superior izquierda, así como al ancho y alto del rectángulo resultante, almacenando cada resultado en el *ArrayList*, dando como resultado lo mostrado en la Figura 5.

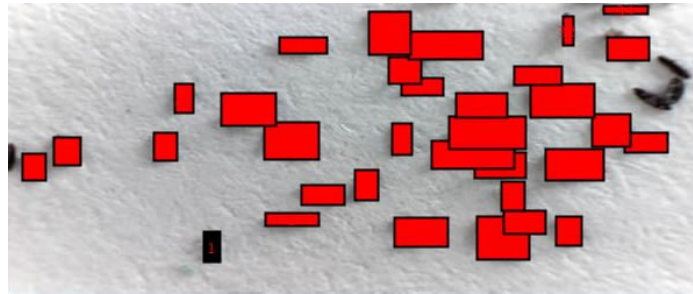


Figura 5. Pellón “B”. Identificación de áreas rectangulares.

Análisis.

Tomando como sugerencia las aportaciones de personal calificado del sector salud, que aportó datos estadísticos en relación al dato del área promedio y rango de tamaños reales de los huevecillos a contar, se procedió a recorrer el *ArrayList* resultante comparando cada área individual contra el rango de áreas probables, clasificando los tamaños de cada huevecillo como Chico o Mediano. Obteniendo los resultados mostrados en la Figura 6.



Figura 6. Pellón “B”. Conteo de huevecillos.

Posteriormente, se procedió a probar el algoritmo resultante con diversos pellones proporcionados por el sector salud, como el mostrado en la Figura 7 y Figura 8, en las cuales se aprecia el conteo por tamaño y el conteo total de huevecillos.

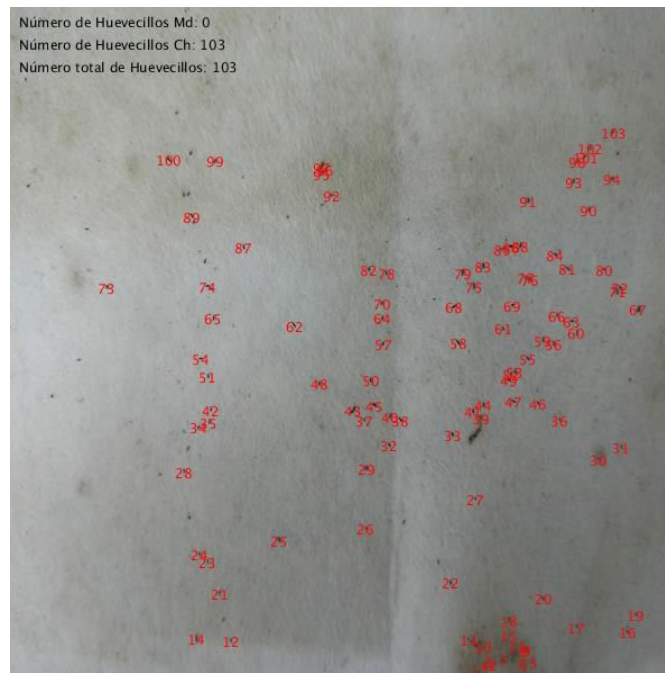


Figura 7. Pellón “A”. Conteo de huevecillos sin resaltar el contorno.

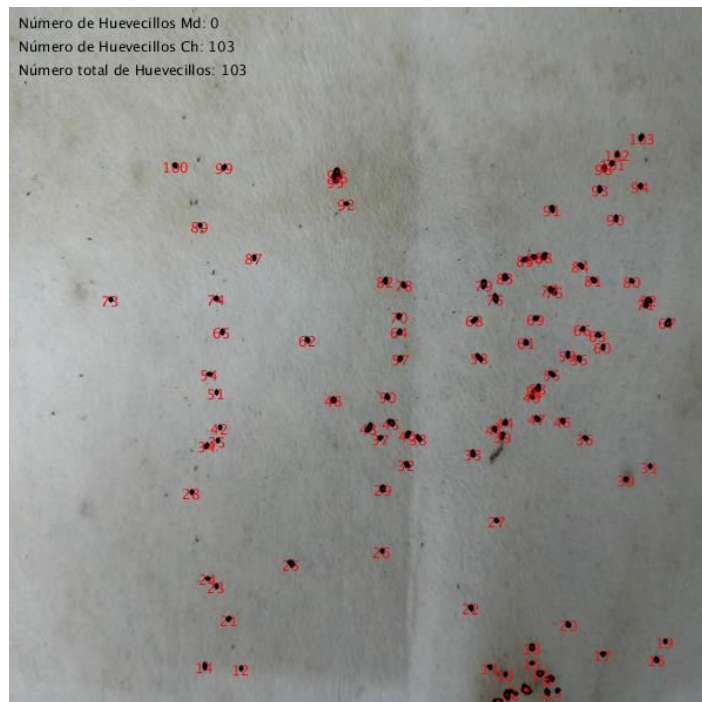


Figura 8. Pellón “A”. Conteo de huevecillos resaltando el contorno.

El algoritmo general que representa la idea del desarrollo se muestra a continuación:

Algoritmo1. Código en *Processing* v.3.5.4 que implementa visión artificial para conteo de huevecillos.

```
import gab.opencv.*;
import java.awt.Rectangle;

PImage img, imgUmbral;
OpenCV imgOpenCv;

ArrayList<Contour> contornos;
Rectangle rectangulo; //variable para enmarcar cada objeto
String tipoHuevecillo="";

float AREA_MINIMA = 100;
float AREA_MAXIMA = 2000;
float AREAMd = 500;
float AREAch = 100;
int numeroHuevCh, numeroHuevMd;

void setup()
{
  img = loadImage("huevecillos.jpg");
  size(850, 200);

  imgOpenCv = new OpenCV(this, img);

  strokeWeight(2);
  textSize(12);

  imgOpenCv.blur(20); //filtro paso bajo
  imgOpenCv.threshold(80); //filtro de umbral fijo
  imgUmbral = imgOpenCv.getOutput();
  contornos = imgOpenCv.findContours();
}

void draw()
{
  numeroHuevCh = 0;
  numeroHuevMd = 0;
  scale(0.7);
  image(img, 0, 0);
  image(imgUmbral, img.width, 0);

  for (Contour contorno : contornos) //recorre los contornos identificados
  {
    if(contorno.area() > AREA_MINIMA && contorno.area() < AREA_MAXIMA)
    {
      if(contorno.area() > AREAMd)
      {
        tipoHuevecillo = "Md";
        numeroHuevMd++;
      }
      else if(contorno.area() > AREAch)
      {
        tipoHuevecillo = "Ch";
        numeroHuevCh++;
      }
    }

    rectangulo = contorno.getBoundingBox(); //contiene los contornos-rectángulos
  }
}
```



```
        identificados
    rect(rectangulo.x, rectangulo.y, rectangulo.width, rectangulo.height); //dibuja o
        enmarca cada objeto en un rectángulo

    fill(255, 0, 0);
    textAlign(CENTER, CENTER);
    text(tipoHuevecillo, rectangulo.x+rectangulo.width/2,
        rectangulo.y+rectangulo.height/2);
    contorno.draw(); //dibuja el contorno de los objetos identificados
}
}
textAlign(LEFT);
fill(0);
text("Número de Huevecillos Md: "+str(numeroHuevMd), 10, 20);
text("Número de Huevecillos Ch: "+str(numeroHuevCh), 10, 40);
}
```

Conclusiones.

Con base a lo anterior, se resalta el hecho de que el conteo de huevecillos aplicando técnicas de visión artificial se realiza de manera más precisa en comparación con el método manual que actualmente sigue personal de la Jurisdicción Sanitaria, el cual consiste en contar a simple vista y manualmente cada huevecillo presente en el pellón. En algunos casos, para agilizar el conteo manual se divide el pellón en cuadrantes y se obtiene un promedio de huevecillos por cuadrante para así obtener un conteo total. Por otro lado, se resalta el margen de error, con la técnica de visión artificial se tiene que, por cada 100 huevecillos hay un margen de error de 2 huevecillos no contados, esto debido a variaciones en la imagen procesada o por empalme de los huevecillos contabilizando una sola ocurrencia. Y en forma manual, no está completamente detectado, debido a que son diferentes personas las que hacen el conteo y eso implica diferentes sesgos de visión y conteo, además de que el margen de error del conteo por promedio, puede ser mucho mayor y no está estandarizado. La velocidad de conteo manual es de varios días dependiendo de la cantidad de pellones, y la velocidad de conteo con visión artificial es inmediata. Se concluye que el proceso automático del conteo de huevecillos con técnicas de visión artificial puede proporcionar información automática y confiable en poco tiempo que permita al personal de la Jurisdicción Sanitaria tomar decisiones oportunas para implementar acciones de respuesta a la comunidad circundante con la intención de disminuir la propagación del mosquito Aedes Aegypti y los posibles casos de enfermedad (dengue) en la población.

Agradecimientos.

Los autores agradecen al Tecnológico Nacional de México por el financiamiento del proyecto 11289.21-PD ("Construcción de plataforma digital para vigilancia y monitoreo de ovitrampas") de la Convocatoria de Investigación Científica y Tecnológica 2021. Así como a la Jurisdicción Sanitaria No.1 de Pánuco, Ver. por el apoyo para el análisis de campo.

Referencias bibliográficas.

- Chacón, M., Sandoval, R. y Vega, J. (2015).** *Percepción visual aplicada a la robótica*. Cd. de México, México: Alfaomega Grupo Editor, S.A. de C.V.
- Domínguez, T. (2021a).** *Processing*. Cd. de México, México: Alfaomega Grupo Editor, S.A. de C.V.
- Domínguez, T. (2021b).** *Visión artificial*. Madrid, España: Marcombo, S.L.

Pajares, G. y De la Cruz, J. (2008). *Ejercicios resueltos de visión por computador*. Cd. de México, México: Alfaomega Grupo Editor, S.A. de C.V.

Rodríguez, R. y Sossa, J. (2012). *Procesamiento y análisis digital de imágenes*. Cd. de México, México: Alfaomega Grupo Editor, S.A. de C.V.

Rusell, S. y Norving, P. (2004). *Inteligencia artificial. Un enfoque moderno*. Madrid, España: Pearson Prentice Hall.

Información de los autores.



Gil Santana Esparza es Ingeniero en Sistemas Computacionales por el Instituto Tecnológico de Cd. Madero, Maestro en Comunicación Académica por la Universidad Autónoma de Tamaulipas, Maestro en Dirección de Ingeniería de Software por la Universidad ISU, Doctor en Educación por la Universidad del Golfo. Es docente en el área de Ingeniería en Sistemas Computacionales del Instituto Tecnológico Superior de Pánuco. Con Reconocimiento a Perfil Deseable Prodep. Participante en proyectos de investigación tanto financiados como internos como titular y colaborador. Asesor de proyectos académicos de Innovación Tecnológica. Ponente en congresos multidisciplinarios de investigación.



Armando Hernández Machuca es Ingeniero en Sistemas Computacionales por el Instituto Tecnológico Superior de Pánuco, Maestro en Sistemas de Información por la Universidad Autónoma de Tamaulipas. Es docente en el área de Ingeniería en Sistemas Computacionales del Instituto Tecnológico Superior de Pánuco. Con Reconocimiento a Perfil Deseable Prodep. Participante en proyectos de investigación tanto financiados como internos como titular y colaborador. Asesor de proyectos académicos de Innovación Tecnológica. Ponente en congresos multidisciplinarios de investigación.