

Artificial vision algorithm on Raspberry, to remotely monitor an infant while it remains in its cradle

Cristopher Darío Vidal Lora, Ing¹, Pedro Huamani-Navarrete, Dr², Christian Tinoco Plasencia, Mag³, Elizabeth Puelles Bulnes, Mag⁴

^{1,2,3,4}Ricardo Palma University, Perú, cvl_1310@hotmail.com, phuamani@urp.edu.pe, christian.tinoco@urp.edu.pe, maria.puellesb@urp.edu.pe

Abstract- This article shows the algorithmic implementation of artificial vision techniques in a Raspberry Pi 3 Model B+ microcomputer, for monitoring during the day and real time of an infant who was represented by a doll, and while he remains in his cradle; so that, direct supervision is maintained, providing peace of mind and security to parents. For it, it was decided to serially integrate an ATMEGA 328p controller to control the actuators and acquire the variable weight of the infant, an Arduino module for connection with a distraction toy located in the cradle itself, a Shield Sim900 GSM module for sending text messages in order to alert about the current situation of the infant, and a Raspberry microcomputer for the image processing stage, which were acquired through a video camera in charge of focusing and monitoring the infant's cradle, and to know the position of the infant and thus carry out an action regarding each particular detection case (position ulna dorsal and ventral, side and fallen). Secondly, in the image processing stage, the Template Matching technique was used to find areas like a template, and was complemented with the Normalized Difference function, which delivered better detection results in terms of five other functions.

Keywords—Raspberry, Arduino, Artificial vision, Template Matching, OpenCV.

Digital Object Identifier (DOI):

<http://dx.doi.org/10.18687/LACCEI2022.1.1.761>

ISBN: 978-628-95207-0-5 ISSN: 2414-6390

Algoritmo de visión artificial en Raspberry, para monitorear de forma remota a infante mientras permanece en su cuna

Cristopher Darío Vidal Lora, Ing¹, Pedro Huamaní-Navarrete, Dr², Christian Tinoco Plasencia, Mag³, Elizabeth Puelles Bulnes, Mag⁴

^{1,2,3,4}Ricardo Palma University, Perú, cvl_1310@hotmail.com, phuamani@urp.edu.pe, christian.tinoco@urp.edu.pe, maria.puellesb@urp.edu.pe

Resumen- Este artículo comprende la implementación algorítmica de técnicas de visión artificial en un Raspberry Pi 3 Model B+, para el monitoreo diurno y en tiempo real de un infante, representado por un muñeco, mientras que este permanece en su cuna; de tal forma que, se mantenga una supervisión directa brindando tranquilidad y seguridad hacia los padres. Para ello, se optó por integrar serialmente un controlador ATMEGA 328p para controlar los actuadores y adquirir la variable peso del infante, un módulo Arduino para la conexión con un juguete distractor localizado en la propia cuna, un módulo Shield Sim900 GSM para el envío de mensajes de textos con el fin de alertar sobre la situación actual del infante, así como un microordenador Raspberry Pi 3 Model B+ para la etapa de procesamiento de imágenes. En cuanto a las imágenes, fueron adquiridas a través de una cámara de video encargada de enfocar y monitorear la cuna del infante para conocer la posición de este, y así realizar una acción referente a cada caso de detección particular (posición de cúbito dorsal y ventral, posición de lado y caído). Por otro lado, en la etapa de procesamiento de imágenes se utilizó la técnica de Template Matching para hallar áreas similares a una plantilla, y se complementó con la función Diferencia Normalizada, la cual entregó los mejores resultados de detección en cuanto a otras cinco funciones también utilizadas en este trabajo.

Palabras Claves—Raspberry, Arduino, Visión Artificial, Template Matching, OpenCV.

I. INTRODUCCIÓN

La población actual día a día se incrementa en todo el mundo, debido al mayor número de nacimientos que decesos; y son muchos los casos donde el padre y la madre acostumbran a trabajar fuera del hogar, sea por el propósito de mejorar la economía o enfrentar las carencias de la familia. Por lo cual, ante la ausencia de ambos en casa, es costumbre contratar a una tercera persona para que brinde el cuidado y la mejor atención al infante; por ello, los padres buscan la seguridad que, al encontrarse ausentes de casa, el hijo se encuentre en buenas manos. Complementariamente a ello, tengan la libertad de poder realizar otras actividades mientras dejan al hijo descansando en su propia cuna.

Sin embargo, actualmente el índice de accidentes de infantes se genera mayormente por descuido de los padres o de las personas encargadas, al no tenerlos bajo un monitoreo constante mientras que se encuentran en la cuna. Pues, mientras que el infante descansa existe un riesgo potencial, ya que estudios médicos revelan que la mala postura al dormir los puede conllevar a la asfixia o atragantamiento en caso de

vómito o regurgitación. Por tal razón, es importante prevenir el Síndrome de Muerte Súbita del lactante (SMSL), a pesar de que existe mucha controversia en hacer dormir al infante decúbito dorsal (supino), ventral (prono) o lateral (lado) [1].

Igualmente, cuando los infantes tienen la fuerza suficiente para ponerse en pie, son propensos a sufrir golpes por caídas; además de ello, considerar la curiosidad que poseen al “morder” las barandas de las cunas hasta el punto en que llegan a ingerir partículas de pintura, e incluso de madera.

De esta manera, la problemática de esta investigación radica en la técnica y metodología a utilizar para implementar un prototipo electrónico original que, con apoyo de herramientas de visión artificial, permita el monitoreo de la posición de un infante mientras que se encuentra en su propia cuna; además, interactuar con juguetes distractores y generar alertas cuando se encuentre en una posición de riesgo, y así permitir que los padres o personas responsables de su cuidado acudan y eviten accidentes. Para ello, se implementó el prototipo electrónico fusionando señales de sensores de proximidad y peso con el microordenador Raspberry Pi 3 Model B+. Este último fue programado algorítmicamente con técnicas de procesamiento de imágenes digitales utilizando la librería OpenCV, y analizando imágenes capturadas de una cámara de video, así como también se contó con una comunicación inalámbrica utilizando el estándar GSM para alertar a través de mensajes de textos al teléfono celular de los padres, o de las personas encargadas del cuidado y supervisión del infante.

En cuanto a investigaciones relacionadas al cuidado de un infante en interiores, y utilizando técnicas especializadas, se encontraron diversos estudios. Como, por ejemplo, en [2] se investigó lo referente a la detección de objetos extraños, vómitos y gestos faciales basados en un video, y se propuso el uso de la tecnología “inteligente” diseñada en forma de reloj para el monitoreo. Además, dicho reloj tuvo filtros de ojo y de nodo cruzado, los cuales sirvieron para detectar la zona y las características oculares, así como su posición en la región correcta de la boca, utilizando un método geométrico para detectar condiciones peligrosas alrededor de esta.

Por otro lado, en [3], se diseñó el modelo y prototipo automático de la horquilla de oscilación de una cuna, basada en la repetición y simulación de los movimientos de una madre. Para ello, en primer lugar, se requirió utilizar un filtro de medias móviles para el ruido presente en los datos, luego la realización de una integración trapezoidal por dos veces consecutivas. Inmediatamente, la ruta ideal para el movimiento de balanceo en la vuelta fue lograda en la

Digital Object Identifier: (only for full papers, inserted by LACCEI).
ISSN, ISBN: (to be inserted by LACCEI).
DO NOT REMOVE

consideración de las trayectorias de los temas. Por lo cual, el modelo matemático de dicha ruta fue derivado utilizando el método de guarnición de la curva cuadrada, y permitiendo que la horquilla diseñada en Solid Works siga dicho modelo.

Y como también, en [4] se propuso un sistema integral ergonómico para el monitoreo y control de ambientes para bebés, el cual ubicó a los sensores de temperatura y humedad de manera estratégica para la realización de dichas tareas. Sin embargo, para ello se requirió de un subsistema de cancelación de ruido para un esquema ergonómico encargado de monitorear y controlar las condiciones del dormitorio del bebé, además de la temperatura, humedad y luminosidad.

De la misma manera, en [5] se planteó una solución móvil de bajo costo para monitorear el movimiento, la temperatura y la frecuencia cardíaca del bebé, pero utilizando un servidor central donde se analizó la data recibida. Mientras que en [6], solamente se planteó el caso de monitorear al bebé e identificar la ubicación exacta de este, para luego emitir una alarma en el caso de encontrarse en una posición de riesgo. Para ello, se optó por distribuir un arreglo de 16 interruptores tipo membrana para cubrir la cuna completamente.

No obstante, además de almacenar valores ambientales como temperatura y humedad, también es posible reconocer el estado de ánimo del bebé a través de la captura y análisis de las imágenes utilizando una red neuronal convolucional, un Raspberry y el software Matlab [7]. Alternativamente, existen otras investigaciones donde monitorean la temperatura periódicamente, pero en el interior de la incubadora; por lo cual, eso determina una exigencia en el ambiente de forma más esterilizada [8].

Por otro lado, en [9] se propuso un algoritmo mejorado de coincidencia de puntos característicos basado en la técnica Template Matching rectangular, donde se tuvo coordenadas bidimensionales que apoyaron a la determinación de la coincidencia. De la misma forma, en [10] también se planteó el uso de la técnica Template Matching pero con la variante de extraer la información de diferencia de imágenes, lo cual permitió reducir las interferencias de fondos complejos cuando se trató de seguimientos de objetivos.

Es así como, potencialmente, con las anteriores experiencias, la propuesta de esta investigación estima que los resultados servirán como base para mejorar el diseño algorítmico, y la implementación de sistemas más complejos en cuanto se refiere al monitoreo, cuidado y alerta de infantes. Y para ello, [11] promueve la ley de la investigación científica e innovación tecnológica, que permite la realización de investigaciones y aportes en el avance tecnológico del país, bajo una rigurosa supervisión de las entidades públicas.

II. MARCO TEÓRICO

Se optó por emplear diversas técnicas de procesamiento digital de imágenes, y un determinado hardware con las características suficientes para abordar y solucionar la problemática existente. Así tenemos.

A. Visión artificial

Es la adquisición, procesamiento, clasificación y reconocimiento de imágenes digitales, con el fin de simular los procesos visuales del ser humano. Para ello, su

implementación en un prototipo combinando una cámara y un procesador, logra inspeccionar el color y/o forma de ciertos objetos [12], con la programación algorítmica correspondiente en el hardware utilizado [13].

De esta forma, la elección del tipo de cámara, distancia al objeto, método de codificación, y entre otros puntos más, resultan importantes en el momento de la adquisición de las imágenes. Complementariamente, la etapa de extracción de características también resulta importante para la identificación del objetivo [14].

B. Módulo Shield Sim900 GSM

Módulo que posee tecnología GMS/GPRS de la marca SIMCOM, y permite la comunicación con el microcontrolador a través de una interfaz serial y comandos AT. Además, soporta las 4 bandas de frecuencias internacionales de GSM, lo cual le permite una compatibilidad con la mayoría de los operadores de telefonía a nivel global. Cuenta con un regulador de voltaje y una ranura para insertar la SIM Card [15].

C. Sensor Celda de Carga

Es un transductor que convierte una fuerza en una señal eléctrica, a través de una o más galgas internas y configuradas en un puente de Wheatstone. En este trabajo se optó por la celda de carga que tolera hasta un máximo de 20 Kilogramos [16].

D. Transmisor Hx711

Es una interfaz entre la celda de carga y el microcontrolador utilizado, que permite interpretar el peso de una manera sencilla, convirtiendo la lectura analógica a digital por medio de un conversor A/D de 24 bits. Además, posee una comunicación serial mediante 2 pines (Clock y Data) con el microcontrolador [17].

E. Open CV

Es una biblioteca libre destinada a la visión artificial y desarrollada por Intel. Es multiplataforma, conteniendo en ella más de 500 funciones dentro de la gama del área de procesamiento de visión artificial.

F. VNC (Virtual Network Computing) Viewer y PuTTY

Es un sistema de cliente ultradelgado y gratuito, basado en un protocolo de pantalla simple que es independiente de la plataforma. Permite acceder a un solo escritorio desde distintos lugares simultáneamente, lo que hace posible compartir aplicaciones con un estilo de trabajo cooperativo y soportado por computadora. Por otro lado, PuTTY es un cliente del protocolo SSH (Secure Shell) capaz de conectarse a servidores remotos iniciando una sesión en ellos. Posee soporte para conexiones de puerto serie local y por su origen en inglés, significa Puerto Único de Tipo Terminal [18].

H. Microcontrolador ATMEGA 328p

Es un Chip microcontrolador creado por ATMEL, de alto rendimiento, y con la capacidad de leer mientras escribe. Además, posee 1KB de memoria EEPROM, 2KB de SRAM, 23 líneas de E/S, 32 registros de proceso general, puerto serial, y con respuesta promedio de 1 MIPS en un solo ciclo de reloj [19].

I. Raspberry Pi 3 Model B+

Es un microordenador de placa única que cuenta con un procesador de 1.4GHz, una conectividad inalámbrica de doble banda a 2.4GHz y 5GHz, un puerto Ethernet a 300 Mbits/s, y con conectividad Bluetooth 4.2 [20].

J. Cámara PI NOIR

Es un periférico fabricado por Raspberry para otorgar la función de captura de imágenes a 5 megapíxeles, pero con ausencia del filtro infrarrojo en el sensor. Por este motivo, se ha convertido en la herramienta perfecta para la captura de imagen en condiciones muy pobres de luz, y a un precio realmente bajo; además, permite grabar vídeos en alta definición de hasta 1080p a 30 cuadros por segundo.

III. DISEÑO DE INGENIERÍA

El prototipo diseñado e implementado en este trabajo de investigación, pertenece al Proyecto de Tesis sustentado para la obtención del Título de Ingeniero Electrónico en la Universidad Ricardo Palma, Lima, Perú. Para ello, se realizó un estudio previo de las técnicas necesarias para la implementación y desarrollo del procesamiento de imágenes, con apoyo de la librería Open CV, y su posterior implementación en hardware, así como la integración total. Por lo cual, a continuación, en la Fig. 1, se muestra el diagrama de bloques pictográfico de todo el proyecto segmentado en cuatro partes, y según la funcionalidad que cada una desarrolla; sin embargo, para la explicación de este, se hizo la subdivisión en dos etapas. La primera relacionada al hardware donde se explican las conexiones entre el controlador, el Raspberry Pi 3 Model B+ y los actuadores, mientras que la segunda etapa relacionada al software encargada de explicar la programación tanto para el controlador como para la tarea del procesamiento de imágenes.

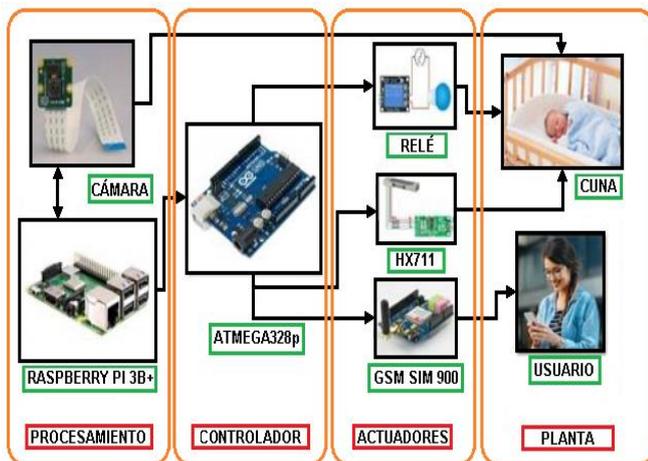


Fig. 1 Diagrama de bloques pictográfico del proyecto implementado.

De la figura anterior, la última parte fue denominada “planta” y corresponde tanto al usuario final (padre, madre o persona supervisora), así como a la propia cuna donde se encuentra el infante. No obstante, considerando la restricción existente para realizar las pruebas directamente con un infante debido al permiso que deberían otorgar los padres, se optó por emplear un muñeco de tal forma que represente al

infante con un peso aproximado de 15 Kg, y una dimensión de 20 x 40 cm.

A. Etapa de conexión de hardware

Para esta etapa, se utilizó un relé con 6 terminales de conexión como parte de los actuadores, para que permita la activación y desactivación del pin digital de salida del microcontrolador ATMEGA 328p (controlador); asimismo, tal relé fue utilizado para activar el funcionamiento de un juguete distractor que fue temporizado para un periodo de 3 minutos continuos. Y como también, se utilizaron sus borneras del estado Normalmente Abierto (NA), debido a que al activarse el interruptor del relé por medio de la señal de control (IN), se procedió a cerrar el circuito, permitiendo que el juguete distractor se active para entretener al infante hasta la llegada de la persona que se encuentra al cuidado de él.

Asimismo, previo a la utilización de la celda de peso con capacidad máxima de 20 Kg. se tuvo que calibrar el “0” posteriormente de la tara, la cual corresponde a la estructura sobre la celda. Esta última, está compuesta por una plancha de madera montada sobre la propia celda y el colchón de la cuna. El principio de funcionamiento de la celda se basa en la deformación de su masa, variando así su resistencia; por ello, cuando el infante se posicionaba sobre la celda, permitía conocer su posición cercana a la baranda, lo que ocasionaba la emisión de una alerta a los padres a través de un mensaje de texto. La variación de voltaje, que es la salida de la celda, se dirigía hacia los terminales del Transductor Hx711, este componente posee un reloj síncrono que posibilitaba la transmisión de datos con el ATMEGA 328p, para obtener el peso actual medido. En la Fig. 2, se observa el diagrama de conexiones de la celda de peso, siendo el terminal 1 la alimentación de 5 VDC, los terminales 2 y 3 correspondientes a la lectura de variación del voltaje de la celda; y, el terminal 4 la tierra.

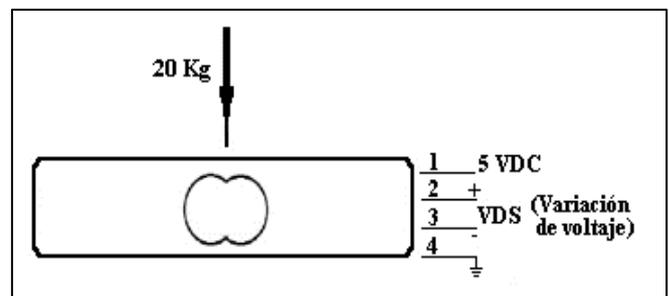


Fig. 2 Diagrama de conexión de la celda de peso.

Luego, en la Fig. 3 se visualiza el diagrama de conexión del transductor Hx711 (sentido de la señal va de izquierda a derecha). Donde, el pin “+E” va conectado al terminal 1 de la celda, el pin “-E” al del terminal 4, el pin “+A” al del terminal 3, y el pin “-A” al del terminal 4. Mientras que los terminales de salida “SCK” (reloj síncrono) y “DT” (transmisión de datos), se conectaron con el ATMEGA 328p por los pines A0 y A1 que corresponden a las entradas analógicas, respectivamente.

Por otro lado, el módulo Shield Sim900 GSM requirió de un SIM CARD de cualquier operador. Asimismo, después de apreciar el parpadeo de una luz azul por un periodo de 5 segundos, se confirmó la conexión a la red móvil. De la

misma manera, este módulo contó con una alimentación de 5 VDC por medio de un cable de color blanco, mientras que por el cable de color negro se realizó la conexión a tierra (GND).

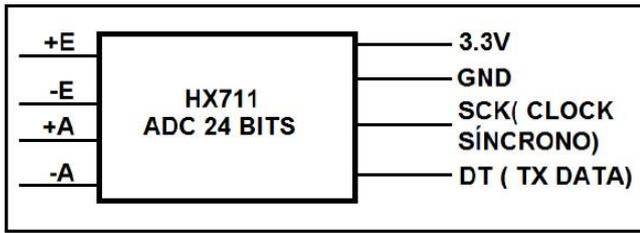


Fig. 3 Diagrama de conexión del Transductor Hx711.

Por otro lado, la comunicación Serial Tx/Rx se realizó por medio de los pines 2 y 3 del propio módulo. Y, el pin 8 se utilizó para comandar la activación del Relé. Luego, una vez conocida la función de cada terminal, se procedió a la integración de los tres componentes anteriormente comentados. Para ello, la Fig. 4 muestra la asignación de pines al ATMEGA 328p para el diagrama circuital correspondiente.

ATMEGA 328p						
Pines	ATMEGA328P	Función	Balanza1	Balanza2	SIM900	RELÉ
1	RST	Reset				
2	D0/RX	Serial rx				
3	D1/TX	Serial tx				
4	D2	GPIO2			PIN2	
5	D3	GPIO3			PIN3	
6	D4	GPIO4				IN
7	VCC	5VDC	E+	E+		VCC
8	GND	Gnd	E-	E-		GND
9	XTAL1	Crystal				
10	XTAL2	Crystal				
11	D5	GPIO5				
12	D6	GPIO6				
13	D7	GPIO7				
14	B0	GPIO8			PIN8	
15	B1	GPIO9				
16	B2	GPIO10				
17	B3	GPIO11				
18	B4	GPIO12				
19	B5	GPIO13				
20	AREF+	5VDC	VCC	VCC	VCC	
21						
22	AREF-	Gnd	GND	GND	GND	
23	C0	Analog0	CLK			
24	C1	Analog1	DT			
25	C2	Analog2		CLK		
26	C3	Analog3		DT		
27	C4	Analog4				
28	C5	Analog5				

Fig. 4 Asignación de pines para el microcontrolador ATMEGA 328p.

B. Etapa de programación del software

Adicionalmente a la conexión, el controlador ATMEGA 328p fue programado para controlar los actuadores y la comunicación serial con el Raspberry Pi 3 Model B+. Entonces, para la programación se utilizó el comando que proporciona la comunicación serial entre el Raspberry y el ATMEGA 328p. La lógica de activación de los actuadores corresponde al comando “IF” y al valor del carácter recibido; de esta manera, de acuerdo con el resultado del procesamiento de la imagen, se configuró la programación para que el Raspberry envíe un carácter de texto simbolizado por una letra del abecedario, de tal forma que active la emisión de la alerta por parte del módulo Shield Sim900 GSM. Asimismo, la activación del juguete distractor de la cuna fue de la mano con la emisión de la alerta.

A continuación, en la Fig. 5 se aprecia el diagrama de flujo del programa implementado, donde se observa la etapa del procesamiento, de la activación, y de la comunicación con el usuario.

Por otro lado, la programación del Raspberry Pi 3 Model B+, se realizó en un archivo del tipo “Script” en donde se encuentra la secuencia de pasos utilizados para establecer el resultado entre las 4 posiciones del infante con la función de “Diferencia Normalizada”, y el algoritmo del Template Matching. Además, para distinguir los resultados de cada posición, se realizó una comparación de cada resultado, hallando el máximo y mínimo valor localizado. Una vez conocida la posición del infante, se procedió a enviar el carácter por medio de una comunicación serial entre ambos dispositivos, momento en el cual, se otorgó el funcionamiento al ATMEGA 328P para la activación de los actuadores y el mensaje a enviar por medio del Shield Sim900 GSM.

En cuanto a la programación del algoritmo de Visión Artificial, se utilizaron seis funciones distintas que se complementaron con la técnica de Template Matching; de las cuales, se seleccionó solo una, a través de pruebas, con la finalidad de corroborar su eficacia. Las funciones utilizadas se listan a continuación.

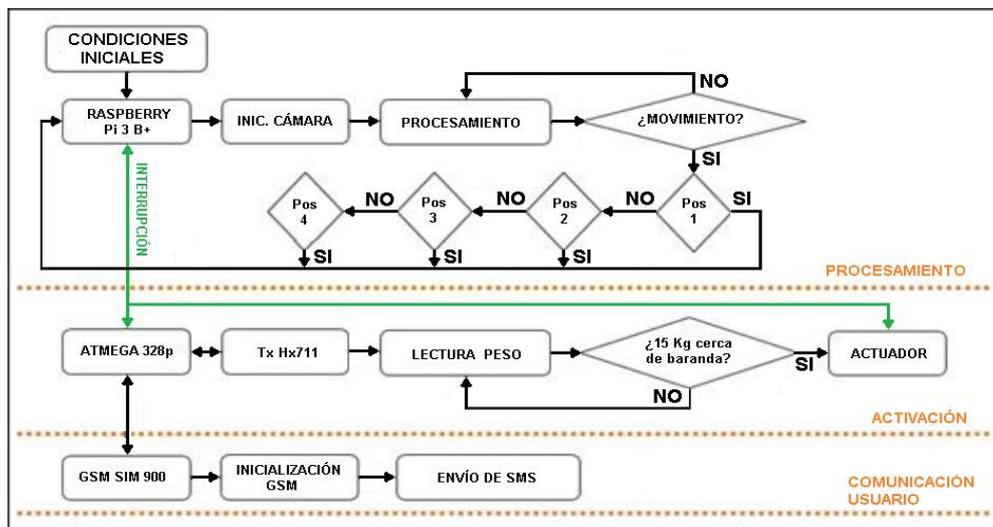


Fig. 5 Diagrama de flujo del programa implementado.

- Suma de Diferencias Cuadradas o SSD (CV_TM_SQDIFF). Esta función también es conocida como Distancia Euclidiana Simple, pero elevado al cuadrado. Y consiste en tomar cada par de píxeles y restarlos, posteriormente sumando todos los cuadrados de cada resultado. Ver la ecuación (1).

$$R(x, y) = \sum_{x', y'} (T(x', y') - I(x + x', y + y'))^2 \quad (1)$$

Donde:

T: es la imagen Template o muestra que se tiene que encontrar o comparar con la imagen capturada en tiempo real, conteniendo al infante.

I: es la imagen capturada en tiempo real de la posición del infante.

(x,y): son las coordenadas de un pixel en la imagen capturada.

(x',y'): son las coordenadas de un pixel de la imagen Template o muestra.

R: es el resultado obtenido de la sumatoria.

- Suma de Diferencias Cuadradas Normalizada (CV_TM_SQDIFF_NORMED). Es una función que no es muy utilizada en la práctica, pues la normalización se realiza con el denominador que se aprecia y es el mismo para todas las demás funciones normalizadas. Ver la ecuación (2).

$$R(x, y) = \frac{\sum_{x', y'} (T(x', y') - I(x + x', y + y'))^2}{\sqrt{\sum_{x', y'} (T(x', y'))^2 * \sum_{x', y'} (I(x + x', y + y'))^2}} \quad (2)$$

Asimismo, este factor es la suma de la imagen plantilla o Template al cuadrado, con la “ventana” de la imagen al cuadrado. La “ventana” es el valor del pixel resultante de la suma de las coordenadas de ambas imágenes (Template y captura) al cuadrado. Las variables de la ecuación (2) son las mismas que fueron nombradas en la ecuación anterior e igual a las siguientes, por lo que se omite redundar la descripción de estas.

- Correlación Cruzada (CV_TM_CCORR). Básicamente es un producto de punto, que toma cada par de píxeles y los multiplica, para luego proceder a sumar todos los productos. Ver la ecuación (3)

$$R(x, y) = \sum_{x', y'} (T(x', y') * I(x + x', y + y'))^2 \quad (3)$$

- Correlación Cruzada Normalizada (CV_TM_CCORR_NNORMED). Tiene el mismo proceder que la Correlación Cruzada, pero tal cual se aprecia en su fórmula se divide por la “ventana” de la imagen. Ver la ecuación (4).

$$R(x, y) = \frac{\sum_{x', y'} (T(x', y') * I(x + x', y + y'))}{\sqrt{\sum_{x', y'} (T(x', y'))^2 * \sum_{x', y'} (I(x + x', y + y'))^2}} \quad (4)$$

- Coeficiente Cruzado (CV_TM_CCOEFF). Similar a la Correlación Cruzada, pero se encuentra normalizada con sus propias covarianzas. Cabe recordar que la covarianza es un valor indicativo del grado de variación entre dos variables aleatorias respecto a sus medias. Este dato sirve para poder hallar el coeficiente de correlación lineal y

estimar la dependencia entre ambas variables. Ver la ecuación (5).

$$R(x, y) = \sum_{x', y'} (T'(x', y') * I'(x + x', y + y')) \quad (5)$$

Donde:

$$T'(x', y') = T(x', y') - \frac{1}{w * h} * \sum_{x'', y''} T(x'', y'')$$

$$I'(x + x', y + y') = I(x + x', y + y') - \frac{1}{w * h} * \sum_{x'', y''} I(x + x'', y + y'')$$

w y h : son los valores del ancho y alto de la imagen Template, respectivamente.

T': corresponde a la covarianza de los valores de la imagen Template.

I': corresponde a la covarianza de los valores de la imagen capturada en tiempo real.

- Coeficiente Cruzado Normalizado (CV_TM_CCOEFF_NORMED). Se rige bajo la misma lógica que el Coeficiente Cruzado, salvo que se encuentra normalizado. Ver la ecuación (6).

$$R(x, y) = \frac{\sum_{x', y'} (T'(x', y') * I'(x + x', y + y'))}{\sqrt{\sum_{x', y'} (T'(x', y'))^2 * \sum_{x', y'} (I'(x + x', y + y'))^2}} \quad (6)$$

Además, para la realización de las pruebas y el adecuado funcionamiento del algoritmo “Template Matching”, se optó por emplear las siguientes librerías del Open CV: `#!/usr/bin/env Python, import os, import time, import cv2, import picamera, import numpy as np, y from matplotlib import pyplot as plt.`

Por otro lado, la cámara PiNoir del Raspberry Pi 3 Model B+ tuvo una secuencia de encendido, iniciando con una previsualización, luego un retardo de 10 segundos, seguido de la captura y el almacenamiento en un archivo de extensión *.JPG en el mismo Raspberry. El retardo se debió al tiempo requerido para variar las posiciones del infante, el cual fue representado por un muñeco para facilitar la recreación del monitoreo en distintas situaciones cercanas a la realidad.

Todas las imágenes capturadas en tiempo real (denominadas con la variable “img”), tuvieron que ser comparadas con la imagen template (denominada por la variable “template”), la cual estuvo almacenada en la base de datos del propio Raspberry. También se optó por utilizar el comando `template.shape` para identificar las coordenadas de ancho (w) y alto (h) de la imagen. Luego, se planteó en una variable “methods” las seis funciones sometidas a prueba: de las cuales solamente se eligió una para el algoritmo final, porque se debe de establecer un patrón de resultados para realizar el discernimiento de las posiciones; sin embargo, en el momento de la ejecución del Script, se evaluaron todas las funciones de manera secuencial.

Por otro lado, en una variable denominada “res” se almacenó el resultado del “Matching”, donde se evaluó la imagen “template” dentro de la imagen “img” y utilizando la función correspondiente. Es así que, para conocer los valores necesarios del resultado, se contó con las variables “min_val” que representa al mínimo valor del resultado con la función utilizada, “max_val” que representa al máximo valor del resultado con la función utilizada, “loc_min” que representa

al mínimo valor de la coordenada de posición del resultado encontrado con la función utilizada, y “loc_max” que representa al máximo valor de la coordenada de posición del resultado encontrado con la función utilizada.

Entonces, de acuerdo con las ecuaciones (1) y (2), las únicas que utilizaron la diferencia de valores fueron CV_TM_SQDIFF y CV_TM_SQDIFF_NORMED, a diferencia de los 4 restantes que multiplicaron los valores. Es por eso que, para localizar cuánto concuerda la vecindad del píxel con el de la plantilla, para los dos métodos mencionados, se tomó la menor localización; por otro lado, para los métodos restantes se tomó el mayor. Asimismo, en una variable definida como “bottom_right” se almacenó la suma de las coordenadas (w,h) con la ubicación encontrada y definidas como variable “top_left”. El cálculo de “bottom_right” corresponde a la suma desde el punto o píxel con más similitud a la plantilla con el largo y alto de la misma.

Para una mayor comprensión del proceso de delimitación del resultado en la imagen “template”, la Fig. 6 muestra un círculo que representa la vecindad exacta o más parecida a la plantilla, por lo que desde el píxel con mayor similitud nace el rectángulo indicando cuál es el área correspondiente al resultado. Es decir, desde el punto de vista visual, es la porción de la imagen correspondiente a la plantilla. De igual manera, en la Fig. 7 se muestran las imágenes obtenidas después de utilizarse los 6 métodos con dos capturas diferentes como “template”.

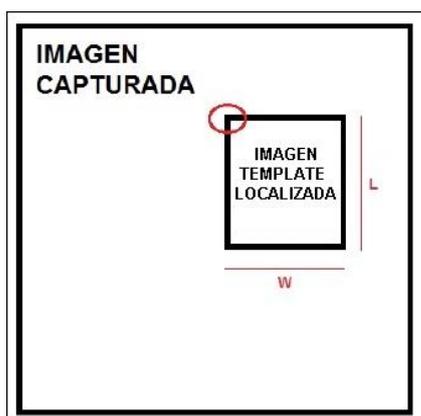


Fig. 6 Asignación de la delimitación del resultado.



Fig. 7 Representación del infante por medio de un muñeco de juguete.
a) Posicionado de lado b) Caído.

De la misma forma, en la Fig. 8 se representa una captura de imagen que corresponde a la posición actual del infante en el interior de su cuna.



Fig. 8 Captura de imagen del muñeco en el interior de la cuna, representando a un infante.

TABLA I
RESULTADOS NUMÉRICOS DE LAS FUNCIONES UTILIZADAS

	NOMBRE DE LA FUNCIÓN	CARACTE_RÍSTICA	RESULTADOS NUMÉRICOS
1	CV2.TM_CCOEFF	Min_Loc	(1725,3601)
		Max_Loc	(1329,1673)
		Min_Val	-379005824.0
		Max_Val	331438048.0
2	CV2.TM_CCOEFF_NO RMED	Min_Loc	(1732,3598)
		Max_Loc	(1329,1673)
		Min_Val	-0.42348253727
	Max_Val	0.998888671398	
3	CV2.TM_CCORR	Min_Loc	(2887,583)
		Max_Loc	(886,1786)
		Min_Val	2559664384.0
	Max_Val	12678572032.0	
4	CV2.TM_CCORR_NOR MED	Min_Loc	(0,572)
		Max_Loc	(1329,1673)
		Min_Val	0.648834586143
		Max_Val	0.999968886375
5	CV2.TM_SQDIFF	Min_Loc	(1329,1673)
		Max_Loc	(2887,582)
		Min_Val	738950.0
		Max_Val	7714269184.0

En la Tabla I, mostrada anteriormente, se comparten los resultados numéricos correspondiente a las 5 funciones propuestas y tomando como base la plantilla referente al infante posicionado de lado en su propia cuna. Tales resultados corresponden al límite de detección (vértice

superior izquierdo y el inferior derecho), así como también se señalan los valores mínimos y máximos correspondiente al método seleccionado.

Luego, en las Fig. 9 y Fig. 10, se muestran dos de los resultados obtenidos al ejecutar el comando CV2.matchTemplate().

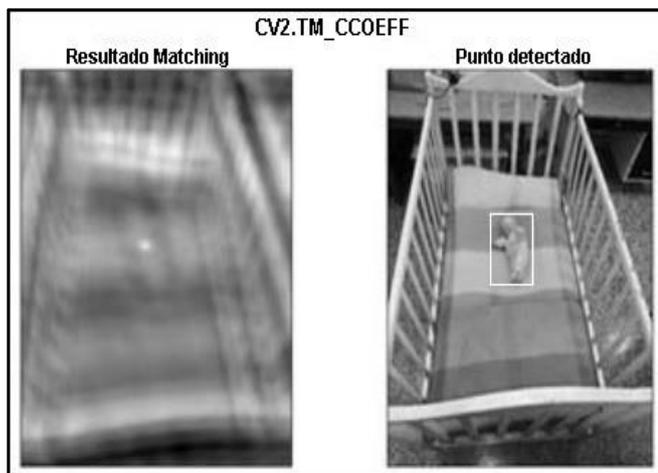


Fig. 9 Resultado de la aplicación de la función CV2.TM_CCOEFF.

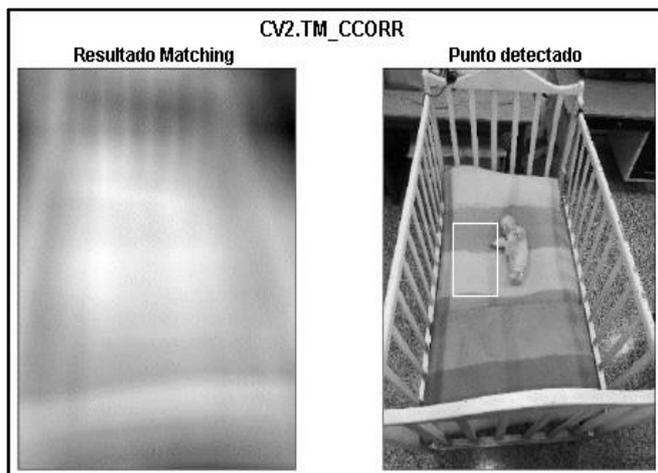


Fig. 10 Resultado de la aplicación de la función CV2.TM_CCORR.

Por lo cual, de los resultados anteriormente obtenidos, se optó por seleccionar el de la función CV_TM_SQDIFF_NORMED, debido a que está basada en la diferencia normalizada y hace posible establecer la lógica comparativa para determinar la posición del infante.

IV. RESULTADOS

De esta manera, a continuación, en la Tabla II, se muestran los resultados numéricos de las pruebas realizadas al utilizar la función CV_TM_SQDIFF_NORMED, y para el caso de las cuatro posiciones evaluadas con el infante representativo.

Igualmente, se observa que no existe falla alguna con los resultados obtenidos, debido a que se detectaron todas las posiciones correctas del infante representativo cuando se utilizó la técnica Template Matching y la función correspondiente. Por lo cual, el siguiente paso consistió en enviar un carácter al microcontrolador ATMEGA 328p a través de la comunicación serial, para activar el módulo

Shield Sim900 GSM y así hacer llegar el mensaje de alerta. A continuación, la asignación de caracteres se observa en la Tabla III.

TABLA II. RESULTADOS NUMÉRICOS DE LA FUNCIÓN CV2.TM_SQDIFF_NORMED

	FUNCIÓN CV2.TM_SQDIFF_NORMED	CARACTERÍSTICAS	RESULTADOS NUMÉRICOS
1	Posición Cúbito Ventral	Min_Loc	(1361,1697)
		Max_Loc	(0,394)
		Min_Val	5.067116945e-5
		Max_Val	1.0
2	Posición Cúbito Dorsal	Min_Loc	(1449,1861)
		Max_Loc	(0,459)
		Min_Val	5.1713996072e-5
		Max_Val	1.0
3	Posición Caído	Min_Loc	(2229,2337)
		Max_Loc	(2648,345)
		Min_Val	7.185455615e-5
		Max_Val	1.0
4	Posición de Lado	Min_Loc	(1329,1673)
		Max_Loc	(2613,0)
		Min_Val	6.2358507421e-5
		Max_Val	1.0

TABLA III. ASIGNACIÓN DE CARACTERES REFERENTE A LA POSICIÓN DEL INFANTE

CARACTER	POSICIÓN	MENSAJE
A	Lado	-----
B	Cúbito Ventral	“El infante se encuentra boca abajo”
C	Caído	“El infante se ha caído”
D	Cúbito Dorsal	“El infante se encuentra boca arriba”

Seguidamente, en la Fig. 11 se muestra una fotografía de la integración del prototipo propuesto en este trabajo. En dicha fotografía se observa la cuna de madera (color blanco, con una colcha de franjas bicolors, y con un tamaño de 138x110x80 cm), las celdas de peso, el ATMEGA 328p, el microordenador Raspberry Pi 3 Model B+, el módulo Shield Sim900 GSM, el juguete distractor, y el soporte donde se ubicó la cámara de video. Asimismo, el juguete distractor seleccionado cuenta con sonido y un foco giratorio, pero es establecido arbitrariamente por el usuario final.

Referente a los resultados alcanzados en cuanto al envío de los caracteres (A, B, C y D), se afirma que fue satisfactoria en su totalidad. Y, en cuanto a las pruebas realizadas con la celda de pesaje, también se obtuvieron resultados positivos.

A continuación, en la Fig. 12 se muestran algunas de las pruebas realizadas en cuanto a la posición y el tipo de detección logrado. Y, con la finalidad de llevar una

contrastación del funcionamiento correcto del prototipo en conjunto, fue necesario realizar pruebas en posiciones que no son iguales a la base de datos Template. Es por ello que, se procedió a ubicar al muñeco en diversas posiciones para identificar la robustez y margen de error del algoritmo.



Fig. 11 Fotografía del prototipo propuesto, incluyendo sus principales partes.

ARCHIVO	POSICIÓN	DETECCIÓN	ARCHIVO	POSICIÓN	DETECCIÓN
Abajo 1		SI	Arriba 3		SI
Abajo 2		SI	Arriba 4		SI
Abajo 3		SI	De lado 1		SI
Abajo 4		SI	De lado 2		SI
Arriba 1		SI	De lado 3		SI
Arriba 2		SI	De lado 4		SI

Fig. 12 Resultados de pruebas para distintas posiciones del muñeco que representa al infante.

V. CONCLUSIONES

Se diseñó y se programó el prototipo de monitoreo diurno en tiempo real utilizando técnicas de visión artificial, con la finalidad de reconocer la posición del infante mientras se encuentra en su cuna. Sin embargo, es necesario resaltar que el mayor reto fue establecer la comunicación entre el Raspberry Pi 3 Model B+ y el ATMEGA 328p. Además, se logró que el prototipo actúe de forma autónoma y correctamente, al comprobar el total funcionamiento del monitoreo.

Asimismo, se programó el algoritmo de procesamiento de imagen basado en la técnica Template Matching, teniendo como función determinante la “Diferencia Normalizada” la cual permitió reconocer la posición del infante. Para ello, se utilizó el lenguaje de programación Python, la librería OpenCV, y la cámara Pi NOIR

A la par, se utilizó una comunicación serial para establecer el enlace en el controlador ATMEGA 328p, el

módulo Arduino, el módulo Shield Sim900 GSM, y el microordenador Raspberry Pi 3 Model B+. Además, la programación del ATMEGA 328p se realizó en lenguaje C, y fue el encargado de recibir la data enviada por el Raspberry, para luego, proceder con la activación de los actuadores: juguete distractor de cuna y módulo Shield Sim900 GSM.

Y como también, se logró desarrollar el envío de alertas mediante mensajes de texto hacia un teléfono celular, utilizando el módulo Shield Sim900 GSM, y de igual manera se procedió con la activación de un juguete con la idea de distraer al infante, ante un posible peligro mientras se encuentra en una cuna de tamaño estándar.

Finalmente, se recomienda que el sensor de peso sea sometido a una calibración previa a su uso, con la finalidad de establecer un adecuado factor de pesaje y así brindar la información confiable. De la misma manera, se debe recargar periódicamente el Sim CARD para que siempre disponga de un saldo y permita el envío de los mensajes de texto. Y como también, considerar un color apropiado para las prendas del infante de tal forma que no se asemejen al color de la ropa de cama de la cuna, porque influye en el reconocimiento de la posición.

REFERENCIAS

- [1] Bbmundo. (02 marzo 2022). La posición adecuada para dormir a tu bebé. Marzo 2022. [Online]. Available: <https://www.bbmundo.com/bebes/cero-seis/la-posicion-adeuada-para-dormir-a-tu-bebe/>
- [2] O. Wei-Liang, Y. Sheng-Min, Ch. Jia-Wei and F. Chih-Peng, “Video based vomit and facial foreign object detections for baby watch and safety,” in 1st International Conference on Orange Technologies (ICOT), Tainan, Taiwan, 2013.
- [3] F. Elmas, A. Yilmaz and M. Garip, “Rocking motion of the baby sleeping on the mother’s lap: Modeling and prototype automatic swing cradle design,” in Electric Electronics, Computer Science, Biomedical Engineerings’ Meeting (EBBT), Istanbul, Turkey, 2017.
- [4] S. Brangui, M. El-Kihal and Y. Salih-Alj, “An ergonomic comprehensive system for monitoring and control of baby environments,” in 6th International Conference on Information and Communication Systems (ICICS), Amman, Jordan, 2015.
- [5] E. Saadatian, S. Priya, Ch. Lihui, O. Newton, N. Hideaki, A. Cheok and A. Perakum, “Low cost infant monitoring and communication system,” in IEEE Colloquium on Humanities, Science and Engineering, Penang, 2011.
- [6] S. Wipulasundara, “A novel concept for remotely monitoring babies,” in Moratuwa Engineering Research Conference (MERCon), Moratuwa, Sri Lanka, 2017.
- [7] A. Sallah and P. Sundaravadeivel, “Tot-Mon: A real-time internet of things based affective framework for monitoring infants,” in IEEE Computer Society Annual Symposium on VLSI (ISVLSI), Limassol, Cyprus, 2020.
- [8] K. Khotimah, M. Imam and S. Wahyu, “Infant incubator temperature controlling and monitoring system by mobile phone based on Arduino,” in International Seminar on Research of Information Technology and Intelligent Systems (ISRITI), Yogyakarta, Indonesia, 2019.
- [9] Z. Liu, Y. Guo, Z. Feng and S. Zhang, “Improved rectangle template matching based feature point matching algorithm,” in Chinese Control and Decision Conference (CCDC), Nanchang, China, 2019.
- [10] Ch. Xiu and X. Pan, “Tracking algorithm based on the improved template matching,” in 29th Chinese Control and Decision Conference (CCDC), Chongqing, China, 2017.
- [11] CONCYTEC. (18 febrero 2022). Ley N° 30309, Beneficios Tributarios que promueven la I+D+i. Marzo 2022. [Online]. Available: <https://vinculate.concytec.gob.pe/ley-no-30309-ley-que-promueve-la-idi/>
- [12] J. García. (12 diciembre 2021). Arduino+Matlab: adquisición de datos. Julio 2011. [Online]. Available:

<https://wechoosethemoon.es/2011/07/15/arduino-matlab-adquisicion-de-datos/>

- [13] R. Gimeno, “Estudio de técnicas de reconocimiento facial,” Universitat Politècnica de Catalunya, Barcelona, España, 2010.
- [14] T. Rojas, W. Sanz and F. Arteaga, “Sistema de visión por computadora para la detección de objetos esféricos a través de la transformada de Hough”, Revista Ingeniería UC, vol. 15, no. 1, pp. 77–87. Abril 2008.
- [15] Naylamp. (14 diciembre 2021). Shield Sim900 GSM/GPRS. 2021. [Online]. Available: <https://naylampmechatronics.com/ardusystem-shields/146-shield-sim900-gsm-gprs.html>
- [16] Naylamp, (14 diciembre 2021). Celda de Carga 20Kg. 2021. [Online]. Available: <https://naylampmechatronics.com/sensores/157-celda-de-carga-20kg.html>
- [17] Naylamp, (14 diciembre 2021). Módulo HX711 Transmisor de Celda de Carga. 2021. [Online]. Available: <https://naylampmechatronics.com/fuerza-peso/147-modulo-hx711-transmisor-de-celda-de-carga.html>
- [18] Hostinger Tutoriales, (15 diciembre 2021). ¿Cómo funciona el SSH?. Noviembre 2021. [Online]. Available: <https://www.hostinger.es/tutoriales/que-es-ssh>
- [19] Course Hero, (28 febrero 2022). Atmega 328p. 2022 [Online]. Available: <https://www.coursehero.com/file/45703651/ATmega328pdocx/>
- [20] S. Monk. Raspberry Pi. 200 ejercicios prácticos, editorial Anaya Multimedia, 2014.