

Prototipo de sistema embebido para la asistencia de personas con discapacidad visual en la acera pública

Diego Cruz Mondragón, Katia Santillán Álvarez, Víctor Hugo García Ortega,
Juan Humberto Sossa Azuela

Instituto Politécnico Nacional, Escuela Superior de Cómputo, Ciudad de México,
México
diegocruzmondragon@hotmail.com, kat_santillan@outlook.com

Resumen. Se plantea el desarrollo de un prototipo de sistema embebido para la asistencia en el desplazamiento de personas con discapacidad visual. Este sistema estará montado en un computador de placa reducida, el cual, sea capaz de tomar una fotografía del entorno en el que se encuentra el usuario del sistema, para posteriormente procesarla y detectar los posibles objetos que puedan interferir en su desplazamiento. Finalmente, mediante síntesis de voz, generar un comando de alerta que permita evitar los objetos identificados.

Palabras clave: asistencia a discapacitados, reconocimiento de formas, sistema embebido, discapacidad visual, electrónica digital.

Embedded System Prototype for the Assistance of Visually Impaired People on the Public Sidewalk

Abstract. The development of a prototype of the embedded system for the assistance in the displacement of people with visual disabilities is proposed. This system will be mounted on a reduced plate computer, which is capable of taking a picture of the environment in which the user of the system is located, to subsequently process it and detect possible objects that may interfere with its movement. Finally, by voice synthesis, generate an alert command that allows to avoid the identified objects.

Keywords: assistance for the disabled, recognition of forms, embedded system, visual impairment, digital electronics.

1. Introducción

De acuerdo con datos del INEGI, en 2014 el 6.4% de la población del país (7.65 millones de personas) reportaron tener al menos una discapacidad, de

este porcentaje, se estima que, alrededor del 35.5%, presentan algún tipo de discapacidad visual, total o parcial, lo que la convierte en la segunda causa de discapacidad visual, después de la motriz.

Las dificultades relacionadas con la visión representan retos para las personas que las padecen y están ligadas a la complejidad de enfrentarse a un entorno que no siempre se encuentra adaptado a sus necesidades, lo cual tiene como consecuencia, de acuerdo con la CIF, que la actividad y la participación de la persona disminuya [1].

2. Metodología

El desarrollo de soluciones basada en sistemas electrónicos embebidos de alta confiabilidad va en aumento y los requisitos de los entornos a los que se orienta el prototipo, son principalmente el coste, el tiempo y la confiabilidad, por lo tanto, se observa la necesidad de una metodología, como herramienta, que facilite el desarrollo de sistemas electrónicos embebidos de alta confiabilidad [2].

El prototipo generado abarca aspectos tanto de Software (SW) como de Hardware (HW), porque pretende dar solución a problemas reales donde resulta necesario la interacción de ambos.

Con base en lo anterior se decidió utilizar la metodología en V o de 4 niveles, ya que el realizar pruebas en cada una de las fases permitirá corregir los errores detectados con mayor antelación sin esperar hasta el termino de cada etapa para rectificar el correcto funcionamiento del sistema, asimismo, este modelo permite que las estimaciones de fechas y presupuesto para las pruebas unitarias y de integración sean determinadas con mayor precisión.

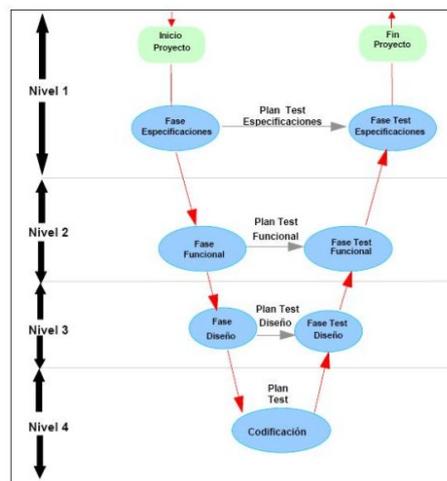


Fig. 1. El modelo representa, en forma de V, las relaciones temporales entre las distintas fases del ciclo de desarrollo de un proyecto.

En cada uno de los niveles mostrados en la Figura 1 existe una correspondencia, existe una fase correspondiente o paralela de verificación o validación. Esta estructura obedece al principio de que para cada fase del desarrollo debe existir un resultado verificable.

2.1. Sistemas embebidos

Los sistemas embebidos son dispositivos de propósito específico que han sido diseñados para realizar un conjunto reducido de operaciones.

2.2. Sensores

Son dispositivos que están capacitados para detectar acciones o estímulos externos y responder en consecuencia. Es decir, nos permiten captar la información del medio físico que nos rodea.

2.3. Visión artificial

La visión artificial es una disciplina científica que incluye métodos para adquirir, procesar y analizar imágenes del mundo real con el fin de producir información que pueda ser tratada por una máquina. Tiene como objetivo principal entender la historia detrás de una imagen [3].

2.4. Síntesis de voz

La síntesis de voz es la producción artificial del habla. El sistema computarizado que es usado con este propósito es llamado computadora de habla o sintetizador de voz y puede ser implementado en productos software o hardware [4].

2.5. Diseño del sistema

En la Figura 2 se muestra un diagrama a bloques con los módulos que componen el sistema embebido propuesto. Mientras que en la Figura 3 se tiene un diagrama a bloques que muestra la arquitectura general del prototipo desarrollado.

- El usuario porta el módulo 1.1, encargado de detectar los obstáculos presentes en el camino del usuario.
- El usuario porta un ordenador de bajo coste (Raspberry Pi 3), encargado de llevar a cabo el procesamiento de los algoritmos del sistema y el módulo 1.3, encargado de capturar las imágenes del camino del usuario.
- El usuario porta el módulo 1.2, encargado de dar una señal al usuario de la presencia de un obstáculo en su camino.
- El usuario porta un par de auriculares, por medio de los cuales recibirá el comando de voz generado por el sistema.

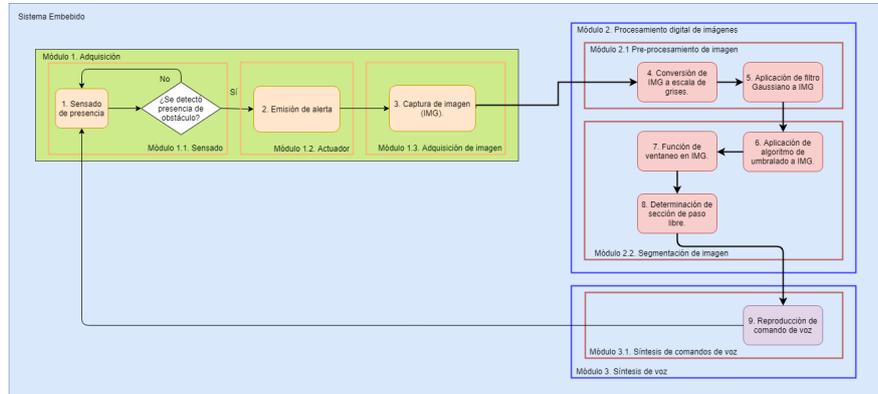


Fig. 2. Arquitectura del sistema embebido.

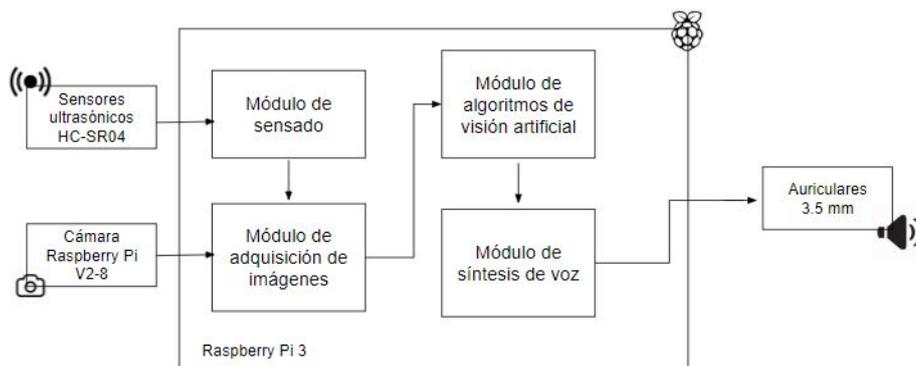


Fig. 3. Descripción general del prototipo.

Módulo 1: Adquisición

Este módulo está encargado de adquirir la información del medio necesaria para el funcionamiento del sistema. Está compuesto por el módulo 1.1. Sensado, el módulo 1.2. Actuador y el módulo 1.3. Adquisición de imagen.

Módulo 1.1: Sensado

En este módulo se realiza la detección de los objetos por medio de la red de sensores, en caso de que se detecte un objeto se procede con la captura de la fotografía para su posterior análisis, en caso contrario se continua sensando.

Módulo 1.2: Actuador

En caso de que uno o más sensores hayan detectado un objeto, se ejecutará este módulo cuyo propósito es activar el actuador que se ha implementado, comunicando al usuario la existencia de un obstáculo en su camino. Se contempla que

el usuario se detenga por 0.5 segundos, permitiendo que la cámara se estabilice y sea capturada la fotografía del camino del usuario.

Módulo 1.3: Adquisición de imagen

Este módulo realiza la adquisición y análisis de imágenes a través de un módulo de cámara para Raspberry Pi modelo V2-8

Módulo 2: Procesamiento digital de imágenes

Conversión RGB a Gray Una imagen RGB consta de 3 capas (rojo, verde, azul), siendo así una matriz tridimensional, por lo que al llevar a cabo esta conversión obtenemos como resultado una representación de la imagen en dos dimensiones, los valores oscilando entre el 0 y el 255 (enteros sin signo de 8 bits). Por lo tanto, los algoritmos aplicados se estarán aplicando a matrices bidimensionales. A través de programación paralela, es posible realizar un procesamiento simple píxel por píxel de una imagen de megapíxeles en milisegundos [5].

Filtro de suavizado Gaussiano El suavizado gaussiano reduce las variaciones de intensidad entre píxeles vecinos, eliminando ruido y detalles finos. La función realiza una convolución de un kernel gaussiano normalizado de 9x9 sobre la matriz de píxeles de la imagen

El kernel de convolución se calcula a partir de la siguiente función, considerando una varianza de 4.0:

$$f(x) = e^{-x^2/s^2}. \quad (1)$$

Módulo 2.2 Segmentación de imagen

Función de umbralado Esta función hace uso del método de umbralado de una imagen, el cual consiste en un proceso de reducción de la información de la misma, sustituyendo todos los píxeles cuya luminancia sea mayor que el coeficiente indicado con el valor de gris 255 (blanco) y sustituyendo todos los demás píxeles por el valor de gris 0 (negro). Para llevar a cabo este proceso se hace uso de un coeficiente de umbral de 50. Una vez que se ha llevado a cabo el proceso de umbralado en las 3 porciones de dimensiones iguales de la imagen, representadas por las etiquetas: “secciónIzquierda”, “secciónCentro” y “secciónDerecha”; dentro de cada una se lleva a cabo un conteo de la densidad de píxeles para determinar la o las secciones que se encuentran obstruidas o libres. Con base en este resultado se genera una instrucción de voz sintetizada por medio de la herramienta eSpeak [6].

Módulo 3: Síntesis de voz

Módulo 3.1 Síntesis de comandos de voz Una vez que ha finalizado la conversión binaria y el conteo de píxeles con valor 0 en cada una de las secciones propuestas, se verifican los valores de los contadores para generar un comando con la herramienta eSpeak.

Se considera que un cuadrante se encuentra obstruido si la densidad de píxeles negros dentro de éste es igual o mayor a 100, pues este valor representa la

cantidad mínima aproximada que un objeto como el descrito en las restricciones del escenario puede contener. Los comandos que reproduce el sistema se muestran en el cuadro 1 con los parámetros considerados para su generación: (Se referirá a los cuadrantes derecho, central e izquierdo con las abreviaciones: **CD**, **CC**, **CI**, respectivamente.)

Tabla 1. Tabla de instrucciones.

CI (px)	CC (px)	CD (px)	COMANDO	DESCRIPCIÓN
<100	<100	<100	"Paso libre, continúe con precaución."	Según los valores obtenidos, el camino se encuentra libre.
>=100	<100	<100	"Lado izquierdo obstruido."	CI está obstruido; CC y CD se encuentran libres.
<100	>=100	<100	"Centro obstruido."	CC está obstruido; CD y CI están libres.
<100	<100	>=100	"Lado derecho obstruido."	CD está obstruido; CC y CI están libres.
<100	>=100	>=100	"Lado izquierdo libre."	CI está libre; CC y CD están obstruidos.
>=100	>=100	<100	"Lado derecho libre."	CD está libre; CC y CI están obstruidos.
>=100	>=100	>=100	"No hay paso libre."	Todos los cuadrantes están obstruidos.

Pruebas Las pruebas correspondientes al módulo 1, nos ayudan a definir la distancia promedio a la cual los sensores detectan los objetos que se encuentran en el escenario.

Con base en la medida del bastón y la altura promedio de una persona a la altura de la cintura, se propone una distancia de 1 m para detectar los objetos, esto se puede observar en la Figura 4 recordando que cada una de las piernas llevara un módulo de adquisición como el mostrado en la Figura 5.

Se realizó una prueba tomando 50 muestras de sensado, adquiriendo la señal en la Raspberry y registrando los tiempos de obtención de respuesta.

Posteriormente se realizó una gráfica, Figura 6, de los datos obtenidos, donde puede observarse que el pico obtenido es de 0.00090 segundos.

Pruebas unitarias del módulo 1.3 - Adquisición de imagen Se encontró que a la altura de 1.20, con una inclinación de 25°, se obtuvieron las fotografías de mayor utilidad.

Una vez que se definió la altura y la inclinación a la que la cámara debe estar posicionada, se realizaron múltiples capturas para asegurar que todos los objetos que son considerados como obstáculos aparezcan en el área adecuada dentro de la fotografía.

Estas pruebas se muestran en las Figuras 7 y 8



Fig. 4. Medición de los objetos.

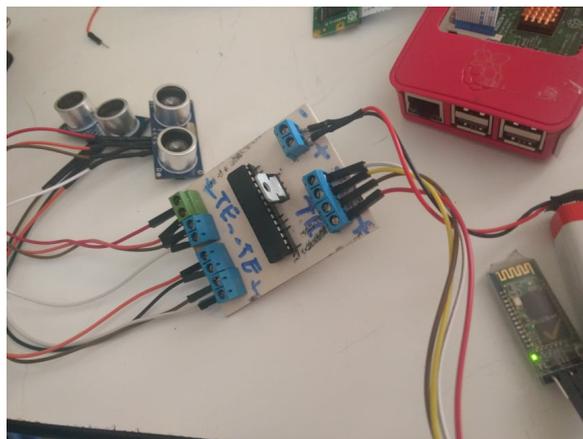


Fig. 5. Placa final del sistema embebido.

2.6. Pruebas unitarias del módulo 2 - Procesamiento de imagen

Una vez que se obtuvieron múltiples imágenes con los ajustes anteriormente mencionados, se seleccionaron aquellas imágenes que cumplen con las condiciones mencionadas en el escenario propuesto: Los objetos considerados como obstáculos serán aquellos con una luminosidad menor a 35% (color semi-oscuro a oscuro).

Posteriormente, éstas fueron utilizadas como la entrada del algoritmo de procesamiento de imágenes desarrollado e implementado en el proyecto. Ejemplo de entrada es la Figura 9, la salida en la Figura 10 y la instrucción generada en la Figura 11.

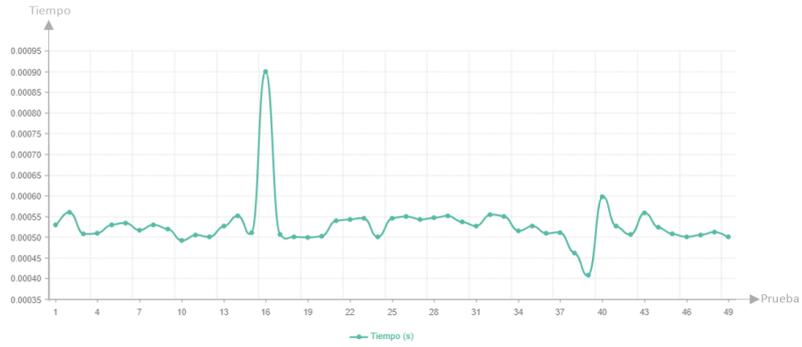


Fig. 6. Gráfica de rendimiento, tiempo de respuesta.



Fig. 7. Pruebas de distancia.

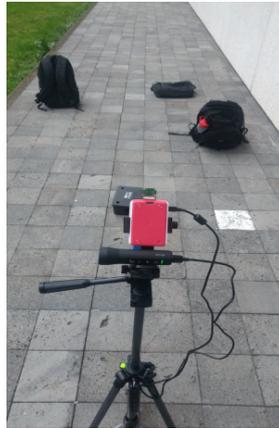


Fig. 8. Pruebas de altura.



Fig. 9. Imagen original.



Fig. 10. Resultado del algoritmo.

2.7. Pruebas de integración

En las figuras 12 y 13 se muestra el prototipo usado durante las pruebas y el modo en el que fue verificado su funcionamiento.



Fig. 13. Pruebas físicas del prototipo.



Fig. 14. Prueba con un bote de basura.



Fig. 15. Prueba con una mochila.



Fig. 16. Prueba con diversos objetos.

3. Resultados

Se logró desarrollar una red de sensores que detecta el 95.9% de los objetos que se encuentran en el paso del usuario a una distancia de 180 cm, desarrollar e

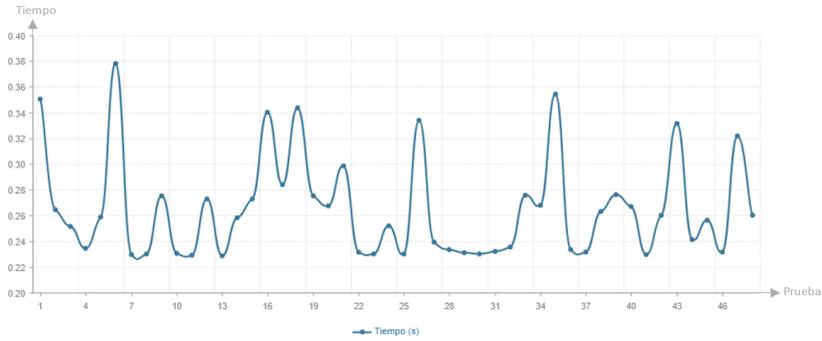


Fig. 17. Gráfica de rendimiento, tiempo de ejecución.

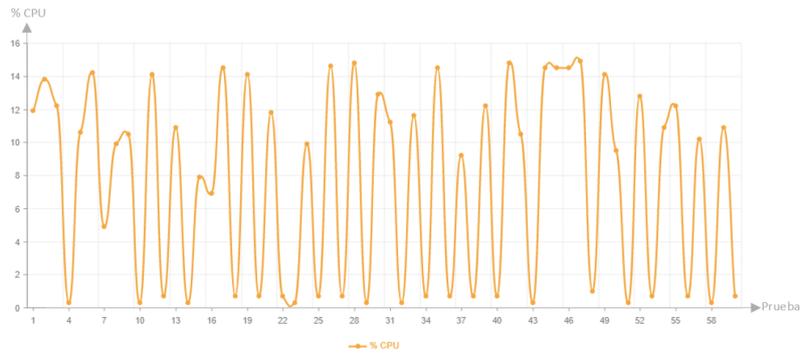


Fig. 18. Gráfica de rendimiento, CPU.

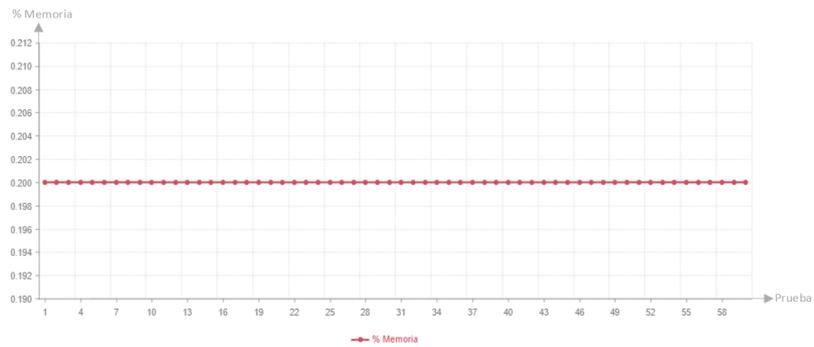


Fig. 19. Gráfica de rendimiento, memoria utilizada.

implementar un algoritmo, paralelizado e implementado en lenguaje C, usando técnicas de visión artificial capaz de determinar las zonas libres y obstruidas

en el camino del usuario a partir de una fotografía capturada por el sistema, generar y comunicar al usuario las instrucciones de asistencia correspondientes a través de la librería *espeak*, en un tiempo máximo de 0.38 segundos a partir de que se ha detectado del objeto, implementar comunicación inalámbrica entre el módulo de sensado y la Raspberry disminuyendo el tiempo total de ejecución del algoritmo en un 16 %.

Fue posible reducir el tiempo de ejecución del algoritmo, permitiendo al sistema comunicar instrucciones al instante.

Con las configuraciones aplicadas, fue posible expandir el escenario controlado, permitiéndonos llevar a cabo pruebas en distintos escenarios y con distintos objetos que cumplieran las especificaciones mencionadas en la definición del escenario.

4. Conclusiones

El diseño del prototipo se realizó con base en los requerimientos y objetivos que se plantearon en un inicio, se logró realizar el prototipo en un tamaño reducido en comparación con los sistemas actualmente existentes. El costo del prototipo es bajo y éste no depende de una conexión a internet para su funcionamiento.

Se logró desarrollar una red de sensores que detecta el 95.9 % de los objetos que se encuentran en el paso del usuario a una distancia de 180 cm.

Se logró desarrollar e implementar un algoritmo, paralelizado e implementado en lenguaje C, haciendo uso de técnicas de visión artificial, capaz de determinar las zonas libres y obstruidas en el camino del usuario a partir de una fotografía capturada por el sistema, emitiendo los comandos de voz pertinentes en tiempo casi real, obteniendo un pico de 0.38 segundos a partir de que un objeto ha sido detectado.

Se logró implementar comunicación inalámbrica entre el módulo de sensado y la Raspberry disminuyendo el tiempo total de ejecución del algoritmo en un 16 %.

Se logró diseñar físicamente el sistema de manera que no interfiera en el paso del usuario y que éste no sea dependiente de una red Wi-Fi.

Gracias a la modularización del prototipo desarrollado, éste puede ser fácilmente escalable, tanto en términos de hardware como de software, pues es posible trasladar el software desarrollado a una tarjeta de desarrollo con dimensiones menores, incorporar distintos sensores y de esta manera, obtener un mayor número de señales que nos permitan tener más información del medio. Por otro lado, es posible desarrollar e implementar nuevas etapas en el algoritmo de procesamiento de imágenes que nos permitan extraer más información de las imágenes para poder generar comandos más detallados.

Agradecimientos. Los autores agradecen a la Escuela Superior de Cómputo y al Centro de Investigación en Computación del Instituto Politécnico Nacional por el apoyo recibido, las herramientas y los conocimientos brindados para el desarrollo del presente trabajo.

Referencias

1. Rodríguez, E.: Diagnóstico sobre la situación de las personas con discapacidad en México. pp. 12–16 (2016)
2. Perez, A., Berreteaga, O., Ruiz de Olano, A.: Una metodología para el desarrollo de hardware y software embebidos en sistemas críticos de seguridad. <http://www.iiisci.org/journal/CV%24/risci/pdfs/C863GM.pdf> (2011)
3. Sossa, H., Morales, R: Procesamiento y análisis digital de imágenes. pp. 20–25 (2012)
4. Black, A., Taylor, P:, Síntesis de voz: Conversión texto a fonemas. <https://francocarlos.com/category/material-academico/> (2017)
5. Cook, J.: Three algorithms for converting color to grayscale. <https://www.johndcook.com/blog/2009/08/24/algorithms-convert-color-grayscale/> (2009)
6. Solé, L.: Binarización de las imágenes. <http://www.dimages.es/Tutorial%20A.I/segmentacion/binaria.htm/> (2018)