

TACHO CON SEGUIMIENTO INTELIGENTE

Escobar Fabian, Estigarribia Emmanuel, Ferrarese Martin, Tejerina Ezequiel, Vera Cristian

Universidad Nacional de La Matanza,
Departamento de Ingeniería e Investigaciones Tecnológicas,
Florencio Varela 1903 San Justo, Argentina

Resumen: Este trabajo de investigación intenta comprobar si es posible utilizar aprendizaje automático para que SmartTrash reconozca a una persona mediante imágenes captadas por una cámara integrada, y que la siga mientras ésta se encuentra limpiando, utilizando procesamiento en paralelo mediante GPU.

Palabras claves: SmartTrash

fabiannorbertoescobar@gmail.com, emmanuelestigarribia@hotmail.com,
martin.ferrarese@gmail.com, ezequiel.tejerina95@gmail.com, crygvera@gmail.com

1. Introducción

Esta investigación se basa en añadir a nuestro proyecto SmartTrash, que consiste en la automatización de un tacho de basura que puede funcionar tanto de forma independiente como también siendo controlado por una aplicación Android que se comunica vía Bluetooth, una funcionalidad extra que permita activar el modo “Limpieza”, en el cual el tacho seguirá a la persona que se encuentre cerca de forma automática.

Para esto, haremos uso del modulo GPU integrado en nuestro dispositivo Android, que a través del procesamiento de imágenes e información de forma paralela nos permitirá recolectar los datos necesarios para enviar los comandos necesarios al tacho para que persiga a la persona más cercana.

GPU, a diferencia de la secuencialidad que manejan las CPUs convencionales, es considerablemente más poderoso, sobre todo cuando se trata de operaciones que se pueden realizar en paralelo. Para sacar ventaja de esto, los algoritmos deben presentar alto grado de paralelismo o grandes requerimientos computacionales, como por ejemplo algoritmos de reconocimiento de personas.

2. Desarrollo

Para aplicar la funcionalidad propuesta en esta investigación, como primera medida, necesitaremos integrar al tacho un juego de cuatro ruedas con su mecanismo para su funcionamiento correcto y una cámara para obtener información del ambiente que rodea al tacho, para que, de esta manera, paralelamente se pueda procesar la misma permitiendo la generación de comandos para que el tacho avance por el lugar. La cámara enviara las imágenes captadas a la placa Arduino, y ésta, a su vez, al dispositivo Android para que sean procesadas.

Una vez activado el modo “Limpieza”, en la aplicación del dispositivo Android, la cámara comenzara el envío de imágenes hacia la placa Arduino.

Para lograr la implementación utilizaremos la librería [OpenCV](#), la cual contiene algoritmos de Computer Vision (nos permite enseñarle a la computadora a que interprete el significado del

mundo físico a través de la visión) y de aprendizaje automático. Estos algoritmos pueden usarse para reconocimiento facial, identificación de objetos, seguir los movimientos de una cámara, seguir objetos en movimiento, seguir movimiento ocular entre muchas otras cosas.

La librería se basa en la API OpenCV 2.x, la cual es esencialmente una API C++.

Tiene una estructura modular, de la cual utilizaremos varios de ellos: la funcionalidad núcleo (que define estructuras de datos básicas y funciones utilizadas por otros módulos), el módulo de procesamiento de imágenes, el módulo de video (incluye algoritmos de seguimiento de objetos entre otras cosas), el módulo de detección de objetos, y el módulo GPU (algoritmos acelerados mediante GPU de distintos módulos).

3. Explicación del algoritmo

El input del sistema de seguimiento es una imagen en escala de grises e 24x24, obtenida a través de la cámara a incluirse en el tachó. Luego la imagen es analizada por un proceso; el cual define si las imágenes recibidas obtienen las muestras que se definieron como válidas, utilizando un algoritmo de detección de objetos para esto utilizaremos una red neuronal. El objetivo de la red es obtener la dirección a la cual debe moverse el tachó.

La estructura de la red consiste en dos capas. La primera es una capa de representación (confeccionada a partir de las unidades capturadas por la cámara). Cada una de esas unidades pasa a la segunda capa, que es la capa de orientación (representa la orientación del tachó a la hora de capturar la imagen).

Durante el periodo de aprendizaje, se utilizan las muestras válidas y las muestras no válidas las cuales se utilizan para detectar si las unidades capturadas por la cámara contienen objetos que consideramos válidos.

Si luego de realizar otra captura la orientación del tachó no cambio, pero la imagen capturada si, el tachó continuará moviéndose en su dirección actual hasta la obtención de una nueva imagen. En el caso de que en la imagen capturada no se reconozca ningún objeto valido el tachó debe cambiar de orientación.

Utilizaremos la librería OpenCV que nos permitirá correr en segundo plano el procesamiento de la información. A continuación, pasaremos a explicare el funcionamiento interno de la misma:

OpenCV consiste en almacenar cada imagen en una matriz que contiene todos los valores de intensidad de los puntos de píxeles. Esos píxeles pueden ser almacenados de distintas formas seleccionando el espacio de color (cómo combinamos los componentes de color para codificar un color determinado) y el tipo de datos utilizado. En nuestro caso, utilizaremos el sistema de color RGB, que se basa en los colores Rojo, Verde y Azul, y nos permitirá tener una mayor precisión a la hora de comparar imágenes.

Para crear nuestra matriz, debemos indicarle el tamaño en filas y columnas, y el tipo de imagen a guardar. Con la función "imread" guardaremos la imagen capturada por la cámara en nuestra matriz, y de esta forma podremos analizar si la imagen contiene objetos válidos, e ir tomando la decisión de cambiar la orientación en tiempo real.

El GPU sería necesario para el procesamiento de la información y el manejo del tacho de forma paralela, pero principalmente será útil en el procesamiento de imágenes, materia en la cual es especialista.

4. Pruebas que pueden realizarse

Podemos realizar la prueba del algoritmo de varias formas, luego de hacer las respectivas implementaciones. Como primera prueba, podemos realizar un camino simple, por ejemplo, todo recto, la persona se acerca al tacho, activa el modo “Limpieza”, y camina en línea recta, y el tacho avanza de la manera correcta y esperada.

Para la segunda prueba, podemos realizar un camino más complejo, incluyendo movimientos en diagonal, y comprobar el correcto funcionamiento de modo “Limpieza”.

Como prueba final, podemos realizar un camino complejo, incluyendo movimientos en diagonal, movimientos rectos, movimientos en diferentes direcciones y corroborar que el tacho avanza correctamente siguiendo a la persona cuando esta en éste modo.

5. Conclusiones

A través de esta investigación intentamos demostrar que se puede implementar el aprendizaje automático a SmartTrash utilizando una cámara integrada y una red neuronal, de esta forma puede realizar seguimientos de personas de manera autónoma. Dicha implementación se puede considerar viable en cuanto a lo económico y tecnológico, ya que existen distintas herramientas que facilitan el desarrollo. Asimismo, las grandes empresas tecnológicas del mundo han profundizado y realizado prototipos que incluyen dichas funcionalidades, por lo que tendríamos una base firme de la cual apoyarnos a la hora de realizar esta implementación.

6. Referencias

<https://opencv.org>

<https://osl.ull.es/software-libre/opencv-libreria-vision-computador/>

https://docs.opencv.org/master/d9/df8/tutorial_root.html

https://docs.opencv.org/master/dc/d88/tutorial_traincascade.html