Universidad de Chile
Facultad de Ciencias Físicas y Matemáticas
Departamento de Ingeniería Eléctrica

Proyecto

Reconocimiento de objetos en góndolas de supermercados

Integrantes: Bastián Garcés G.
María José Liberona T.
Profesor: Claudio A. Pérez
Auxiliar: Jorge Zambrano I.
Ayudante: Gabriel Cubillos F.
Jhon Pilataxi
Ayudante de laboratorio: Juan Pérez C.

Fecha de entrega: 12 de mayo 2022
Santiago de Chile

Índice de Contenidos

Índice de Tablas

1. Introducción

La visión computacional o también conocido como visión por computador es un concepto que ha adquirido gran relevancia en los últimos años, debido a la gran cantidad de aplicaciones que posee como son: detección de defectos, reconocimiento de rostros, seguimiento de movimientos, entre otras. Una de las principales tareas de la visión computacional está enfocado en el reconocimiento, ya sea de personas, animales u objetos. Teniendo en consideración lo anterior, se ha decidido realizar el proyecto del curso EL7007 enfocado en el reconocimiento de objetos en góndolas de supermercados, donde se espera implementar un algoritmo capaz de discernir entre distintos tipos de productos.

El reconocimiento de objetos en góndolas de supermercados es implementado para resolver diferentes problemáticas, por ejemplo, el abastecimiento de las góndolas, aplicaciones utilizadas por personas en situación de discapacidad visual, automatización del inventario, entre otros. No obstante, el problema de reconocimiento de objetos conlleva la detección del objeto en la estantería y el reconocimiento de su clase, por esta razón se decidió utilizar la base de datos *SKU110K*[1] que cuenta con las detecciones de los diferentes productos, para así enfocarse netamente en la tarea de reconocimiento.

2. Estado del arte

Para la resolución de este problema se han propuesto diferentes metodologías en investigaciones de los últimos años, por ejemplo, los artículos *A deep learning pipeline for product recognition on store shelves* [2] y *Grocery product detection and recognition* [3] trabajan la detección y reconocimiento de objetos en góndolas de supermercado. A continuación, se describen brevemente ambos artículos.

2.1. Deep Learning for Retail Product Recognition: Challenges and Techniques

En este caso, el procedimiento de detección y reconocimiento de imágenes consta de 3 partes:

1. La primera parte consiste en la implementación de un sistema de detección mediante una CNN, la cual se encarga de extraer propuestas de región de la imagen de consulta. Para ello, se propone como red de detección la red *YoloV2*[1], ya que garantiza el rendimiento en tiempo real en una GPU y la disponibilidad de la implementación original.

2. Luego, cada propuesta detectada se recorta de la imagen de consulta y se ingresa a otra CNN (Embbeder) que calcula una representación de la imagen. Para la red principal de Embbeder se propone la red *VGG-16*[2], la cual se encuentra pre-entrenada en la tarea de clasificación *Imagenet-1000*. Cabe mencionar que el proceso de entrenamiento de Embbeder se lleva a cabo utilizando las imágenes de referencia de los productos.

3. Por último, se plantea realizar el reconocimiento del producto a través de una búsqueda de similitud KNN en una base de datos de representaciones pre-calculadas fuera de línea por el Embbeder en las imágenes de referencia.

2.2. Grocery product detection and recognition

Para el caso del presente articulo la metodología se divide en preselección, selección fina y post-procesamiento. A continuación se presenta la labor de cada una de las fases:

1. La etapa de preselección está destinada a seleccionar el conjunto inicial de ventanas candidatas sobre la información conjunta obtenida de la posición de las esquinas y la distribución de color de las imágenes en cuestión. Para la detección de esquinas se propone el uso de un detector de esquinas Harris, luego para cada esquina detectada se proponen 4 ventanas candidatas. Posteriormente se evalúa en cada una de estas ventanas candidatas la varianza de la escala de grises en la ventana en cuestión y el porcentaje de píxeles en primer plano. Luego, si se cumplen ambas condiciones, se calcula el histograma de color 3D calculado en el espacio de color YbCBr, se compara con el histograma de referencia del producto destino mediante la intersección de histogramas habiendo realizado previamente una normalización del color en las imágenes de entrenamiento y prueba, lo cual se puede lograr mediante el uso de un algoritmo CLAHE. Posterior a la intersección, se elige la ventana candidata cuyo histograma tiene una mayor intersección con el producto objetivo y dicha ventana pasa al postprocesamiento.

[1] https://github.com/pjreddie/darknet
[2] https://github.com/tensorflow/models/tree/master/research/slim

2. En la etapa de selección fina se aplican características más robustas para la selección de candidatos. Es en este punto en que se proponen dos métodos para la obtención de vectores de características, los que corresponden al algoritmo de Bolsa de palabras (Bag of Words) y una Red Neuronal Convolucional Profunda. El algoritmo de bolsa de palabras se utilizaría para obtener vectores de características para las imágenes de entrenamiento, dichos vectores son independientes de la distancia entre los objetos y la cámara. Luego para el caso de la red convolucional se utilizaría la red *AlexNet* ingresando imágenes reescaladas a 227x227 píxeles, en el articulo también se usan imágenes en escalas de grises para poder comparar el algoritmo de Bag of Words con la CNN (para ponerlos en igualdad de condiciones), luego se comparan los puntajes del vector obtenido de la red con los de referencia y se conservan aquellos con mayor score. Sin embargo, es necesario un postprocesamiento para elegir un único resultado como factible.

3. En la etapa de postprocesamiento se utilizan técnicas para filtrar los resultados, entre estas técnicas está el NMS o clustering.

3. Metodología

En esta sección se presenta la metodología implementada durante todo el proyecto, destacando un cambio en la base de datos y el problema a resolver.

3.1. Etapa inicial

Al comienzo del trabajo se buscaba implementar la metodología expuesta en la sección 2.1 utilizando a su vez la base de datos *GroZi-120* [4], la cual contiene imágenes y vídeos para la detección de 120 productos diferentes. Sin embargo, surgieron problemas al momento de utilizar el algoritmo descrito para la detección de los objetos, ya que este apuntaba al uso de la red *YoloV2* que está programada para la detección y reconocimiento sólo de 80 clases, por lo que no permitía detectar la mayoría de los objetos presentes en las góndolas. Así también, se probaron sin éxito otras redes preentrenadas para la detección, dentro de las cuales estaban *Cascade R-CNN*, *RetinaNet* y *ResNet50*, donde la principal dificultad fue la ignorancia en su utilización e instalación.

A partir de las dificultades presentadas anteriormente se decidió enfocar el trabajo solo a la etapa de reconocimiento, es decir, asumir que las detecciones ya estaban hechas. Por consiguiente, fue necesario cambiar la base de datos.

3.2. Etapa actual

En la continuación del trabajo se determinó el uso de la base de datos *SKU110K*, la cual contiene 11762 imágenes de estanterías de supermercados alrededor del mundo y provee las detecciones sobre estas mismas, pero no sus clases respectivas. No obstante al contener tantas imágenes existía una gran cantidad de productos diferentes, motivo por el cual se acotó el trabajo al reconocimiento de bebidas y productos relacionados a 20 clases diferentes. A continuación se exponen las clases escogidas:

- Clase 0: Botella de Agua
- Clase 1: Caja de Cocacola
- Clase 2: Caja de Cocacola Zero
- Clase 3: Botella de Cocacola Zero
- Clase 4: Botella de Fanta
- Clase 5: Caja de Pepsi
- Clase 6: Botella de Pepsi Light
- Clase 7: Botella de Pepsi Zero
- Clase 8: Botella de Pepsi
- Clase 9: Lata de Redbull Light
- Clase 10: Lata de Redbull
- Clase 11: Caja de Sprite

- Clase 12: Botella de Sprite Zero
- Clase 13: Botella de Sprite
- Clase 14: Botella de Starbucks
- Clase 15: Lata de Starbucks
- Clase 16: Botella de Cocacola
- Clase 17: Botella de Coke
- Clase 18: Caja de Redbull Light
- Clase 19: Caja de Redbull

Luego, con la finalidad de implementar un algoritmo preliminar se generó un conjunto de entrenamiento con 77 detecciones repartidos entre las diferentes clases, donde cabe destacar que las clases están des-balanceadas. Además, se creó un conjunto de prueba que posee 2 detecciones para cada clase, con el fin de evaluar las predicciones del modelo inspirado en la metodología de la sección 2.1 donde se consideran sólo los puntos 2 y 3 para el trabajo de reconocimiento.

Para la parte encargada de la extracción de vectores de características se emplearon dos situaciones, la primera consistía en utilizar únicamente las capas convolucionales de la red *VGG-16* para posteriormente convertir los datos a una dimensión, resultando en un vector de largo 25088. El segundo caso corresponde a utilizar el vector entregado por la red completa, lo que conllevó a obtener un vector de largo 1000.

La etapa de reconocimiento se basó en el método de búsqueda de similitud *K-Nearest Neighbors (KNN)*, que fue entrenada con los vectores de características resultantes del conjunto de entrenamiento.

4. Resultados y análisis

Para la evaluación del desempeño del proyecto, se pretende utilizar las métricas *Recall*, *Precision* y *F1-Score*. A continuación se presentan las métricas mencionadas anteriormente, donde en la Tabla 1 se observan estas mismas para el caso en que se extrajo el vector de características de la red completa.

Tabla 1: Resultados para vectores de características de largo 1000

	precision	**recall**	**f1-score**
0	0.20	0.50	0.29
1	0.00	0.00	0.00
2	0.25	0.50	0.33
3	0.18	1.00	0.31
4	1.00	0.50	0.67
5	0.00	0.00	0.00
6	0.00	0.00	0.00
7	0.00	0.00	0.00
8	0.00	0.00	0.00
9	0.25	0.50	0.33
10	0.50	0.50	0.50
11	0.00	0.00	0.00
12	0.00	0.00	0.00
13	0.00	0.00	0.00
14	0.00	0.00	0.00
15	0.00	0.00	0.00
16	0.00	0.00	0.00
17	0.67	1.00	0.80
18	1.00	0.50	0.67
19	1.00	0.50	0.67

accuracy	0.28

De la tabla anterior se logra observar que existen algunas clases que no lograron ser detectadas, dichas clases corresponden a las que tienen las 3 métricas en 0. Por otro lado existen algunas clases con un alto grado de predicción como lo son las clase 4, 17, 18 y 19 que poseen más de 0.5 en la métrica f1-score.

Luego, en la Tabla 2 se observan los resultados obtenidos para el caso en que se obtuvo el vector de características de las capas convolucionales de la red.

Tabla 2: Resultados para vectores de características de largo 25088

	precision	**recall**	**f1-score**
0	0.05	1.00	0.10
1	0.00	0.00	0.00
2	0.00	0.00	0.00
3	0.00	0.00	0.00
4	0.00	0.00	0.00
5	0.00	0.00	0.00
6	0.00	0.00	0.00
7	0.00	0.00	0.00
8	0.00	0.00	0.00
9	0.00	0.00	0.00
10	0.00	0.00	0.00
11	0.00	0.00	0.00
12	0.00	0.00	0.00
13	0.00	0.00	0.00
14	0.00	0.00	0.00
15	0.00	0.00	0.00
16	0.00	0.00	0.00
17	0.00	0.00	0.00
18	0.00	0.00	0.00
19	0.00	0.00	0.00
accuracy			0.05

Tras observar la tabla presentada anteriormente se logra observar que el algoritmo solo era capaz de predecir la clase 0 (Agua). Esta forma de predecir los resultados puede deberse al hecho de que los vectores de características utilizados para esta ocasión pueden poseer muchos atributos como 0, lo que desvirtúa la predicción, motivo por el cual sería recomendable hacer una reducción de características de cara al futuro del proyecto.

Al comparar ambas tablas, se puede identificar que al emplear el vector de características de largo 1000 se posee un rendimiento considerablemente mayor con un *accuracy* de 0.28, que al utilizar el vector de largo 25088 que tiene un *accuracy* de 0.05.

5. Conclusiones

A modo de conclusión se confirma que la metodología permite llegar a resultados tangibles, aunque no con los resultados esperados. Esta problemática puede recaer en la existencia de un grupo de entrenamiento muy pequeño y con un desbalance en las clases.

En un futuro se espera expandir la base de datos de entrenamiento y de prueba, con el fin de lograr una mayor robustez en el algoritmo y de esta forma incrementar el número de reconocimientos correctos. Además se propone incorporar un conjunto de validación que permita mejorar el modelo mediante la correcta elección de los hiperparámetros. También se espera emplear un preprocesamiento de las imágenes mediante técnicas de mejoramiento de contraste, brillo, color, entre otros. Otra de las posibles mejoras a considerar sería implementar un algoritmo de reducción y selección de características. Por último se propone probar otros métodos de extracción de características y clasificación

Referencias

[1] "SKU110K," Database. [Online]. Available: https://drive.google.com/file/d/1iq93lCdhaPUN0fWbLieMtzfB1850pKwd/edit. [Accessed: 09-05-2022].

[2] A. Tonioni, E. Serra, and L. Di Stefano, "ArXiv:1810.01733v3 [CS.CV] 27 Jan 2019," 27-Mar-2019. [Online]. Available: https://arxiv.org/pdf/1810.01733.pdf. [Accessed: 28-Mar-2022].

[3] A. Franco, D. Maltoni, and S. Papi, "Grocery product detection and recognition," Expert Systems with Applications, 21-Mar-2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417417301227. [Accessed: 28-Mar-2022].

[4] "120 database," GroZi. [Online]. Available: http://grozi.calit2.net/grozi.html. [Accessed: 31-Mar-2022].