

# CarePet – Un dispensador de alimento auxiliado con un sistema de Visión Computacional

**Kevin Michelle Contreras González**

Estudiante de octavo semestre en la Facultad de Ingeniería Mecánica y Eléctrica, en la carrera de Ingeniería en Tecnología de Software

Universidad Autónoma de Nuevo León

4 de Mayo de 2015

## RESUMEN

CarePet es un sistema de Visión Computacional que busca auxiliar a los dueños de mascotas con el problema de cómo alimentar a sus mascotas de manera automatizada y sin supervisión humana.

## INTRODUCCIÓN

Gracias a los avances de la tecnología se ha logrado conseguir muchos beneficios los cuales afectan a diversos campos de la vida humana. Incluso si algún campo de la vida cotidiana no ha sido afectada por estos avances (algo que considero que es imposible hoy en día), es muy probable que en algún futuro la tecnología influirá en esos de la vida.

Todo lo relacionado a las mascotas del hogar, en especial a perros y gatos, comienza a ser una vía para experimentar nuevas tecnologías, y esto queda demostrado gracias a iniciativas como Bistro [1] o CleverPet [2], las cuales son financiadas con *crowdfunding*, es decir, con la participación de gente quienes vía Internet [3] están interesados en dicho proyecto y donan dinero para que se complete la realización de los mismos.

## JUSTIFICACIÓN

Como parte de mi clase de Visión Computacional, impartida por la doctora Elisa Schaeffer, decidí crear un sistema que aproveche estas tecnologías para decidir si se debe alimentar o no a nuestra mascota en base a la información que dicho sistema obtenga a través de imágenes de un dispensador de comida de mascota, ya sea obtenidas de manera estática con cámaras o (como es deseado) de manera dinámica con una cámara web.

## DESARROLLO

La manera en la que abordé este problema fue la siguiente:

- Colocar una cámara alrededor de un contenedor de comida para mascota de tal manera que pueda tomar una fotografía y se pueda visualizar el contenedor y sobre todo su interior.
- Dicha fotografía sería procesada por un programa escrito en Python, la cual decidiría si hay suficiente comida o no para justificar una nueva reposición de la misma.

La razón por la que elegí a Python como lenguaje de programación para este proyecto fue porque tengo

familiaridad para trabajar con imágenes gracias a la librería PIL[4], pues ya he implementado [5] distintas funcionalidades de Visión Computacional en este lenguaje, como detección de bordes, detección de figuras, entre otras.

Para resolver este problema pensaba en utilizar mis propias funciones implementadas [5], en específico las funciones de detección de bordes y detección de figuras, pero encontré que la irregularidad de las figuras y algunos fallos en mis propias implementaciones me impedían resolver esta problemática con cierto éxito.



Imagen de un contenedor con croquetas de perro



Resultado de mi detector de figuras aplicado a la imagen de arriba.

No obstante, se me ocurrió la idea de preprocesar mis imágenes para que sólo exista una serie de colores únicos, es decir, que no existan variaciones de los mismos.

La manera en la lograría esta “discretización” de colores sería con remplazar los colores originales de dicha imagen con algún otro color de una lista de colores ya predefinidas, eligiendo al color que más se le aproxime en dicha lista. La proximidad de un color con otro color sería calculado con la distancia euclidiana, pues es un cálculo que me ha parecido muy útil en distintas situaciones cuando necesito comparar objetos.

De acuerdo a lo que he investigado en Internet este proceso es conocido como segmentación de

colores, el cuál ya ha sido investigado [6] por muchos científicos y no es algo nuevo que he inventado por casualidad.

A pesar de mi ligera decepción decidí aprovechar mi idea de preprocesar la imagen pero limitándolo al dominio de mi problema: decidir si hay suficiente alimento de mascotas en un contenedor. Los colores de las croquetas de perros y gatos son por lo general cafés, con algunas tonalidades claras o rojizas. Si lo único que estoy buscando son croquetas, ¿no sería lógico buscar en una imagen aquellos pixeles que sean “cafés” o que se le parezcan mucho?

Por supuesto que mi solución estaría arruinada si el contenedor llega a ser de color café, no obstante preferí ignorar esta posible complicación en favor de lograr alguna solución a mi problema. Y creo que los resultados lo demuestran.



Resultado de mi método que segmenta los colores de una imagen de acuerdo a si son lo suficientemente “café”

Como se puede observar ahora se pueden visualizar mucho mejor las croquetas, aunque existe todavía algo de ruido pero considero que los resultados representan mucho mejor a la realidad.

El siguiente paso de este sistema sería discriminar la imagen de acuerdo a algún parámetro, y aunque en un principio pensé en contar las croquetas en la imagen, me di cuenta que tendría los mismos problemas que comenté al principio, así que seguí el consejo de un amigo [7] y sólo me preocupé en contar la frecuencia de pixeles negros, o visto de otra manera, calcular porcentaje de pixeles de la nueva imagen que aparentemente son croquetas.

En mi sistema elegí que si el porcentaje de pixeles negros no es igual o mayor a 55%, entonces se imprimiría en pantalla que hay insuficiente comida.

## CONCLUSIONES

A pesar de que logré implementar el sistema de Visión Computacional con cierto éxito, estoy un poco decepcionado conmigo mismo pues no logré llevar a cabo mis propuestas iniciales, que era utilizar una cámara web para monitorear en tiempo real un contenedor e implementar un dispositivo físico que simule el dispensador de comida automático.

Reconozco que la razón principal por la que no se logró completar todas las metas del proyecto antes de esta fecha fue debido a la falta de presupuesto para comprar algunos materiales, pues ciertos problemas económicos y personales me impidieron a invertir en dicho material.

Creo que trabajo futuro que se puede realizar y que es pausable a mediano plazo es utilizar una cámara web y realizar mejoras al sistema para evitar los problemas que consideré con los contenedores de color café.

También he pensado en la posibilidad de conseguir un dispensador de comida automático y modificarlo con mi sistema, para evitar tener que diseñar las partes mecánicas más complicadas. No obstante, este trabajo (tanto conseguir el dispensador como modificarlo) sería para un futuro muy lejano pues no tengo los medios, ni las intenciones, de hacerlo realidad.

## REFERENCIAS

1. Página de Bistro en Indiegogo (<https://www.indiegogo.com/projects/bistro-a-smart-feeder-recognizes-your-cat-s-face>)
2. Página de CleverPet en KickStarter (<https://www.kickstarter.com/projects/1453211280/cleverpet-a-console-that-teaches-and-feeds-your-do>)
3. Entrada de la Wikipedia en inglés de crowdfunding (<http://en.wikipedia.org/wiki/Crowdfunding>)
4. Mi GitHub personal ([https://github.com/KevinMichelle/Vision\\_FIME\\_2015](https://github.com/KevinMichelle/Vision_FIME_2015))
5. Manual de la librería PIL (Python Imaging Library) para Python (<http://effbot.org/imagingbook/pil-index.htm>)
6. Báez Rojas, J.J. (2004), “Segmentación de imágenes de color”
7. Mi compañero Victor Ríos Martínez, también estudiante de la FIME
8. Propuesta inicial del proyecto CarePet ([https://github.com/KevinMichelle/Vision\\_FIME\\_2015/blob/master/CarePet/CarePet.pdf](https://github.com/KevinMichelle/Vision_FIME_2015/blob/master/CarePet/CarePet.pdf))