

---

## INSTITUTO TECNOLÓGICO DE MORELIA

DIVISIÓN DE ESTUDIOS PROFESIONALES  
DEPARTAMENTO DE SISTEMAS Y COMPUTACIÓN

### TESIS PROFESIONAL

**“CONSTRUCCIÓN DE UN SISTEMA DE MONITOREO DE  
BAJO COSTO PARA COLONIAS DE POLINIZADORES  
UTILIZANDO SENsoRES, MICROSENsoRES,  
PROCESAMIENTO DE IMÁGENES Y APRENDIZAJE DE  
MÁQUINAS”**

QUE PARA OBTENER EL TÍTULO DE:  
**INGENIERO EN TECNOLOGÍAS DE LA INFORMACIÓN Y  
COMUNICACIONES**

PRESENTA:

**BALTAZAR MUÑOZ LEDO HURTADO**

ASESOR:

**DR. ULISES OLIVARES PINTO**

*Dedicatoria*

*A Dios, por darme el conocimiento, a mis padres, por darme el ejemplo.*

## AGRADECIMIENTOS

Este proyecto fue parcialmente financiado por los siguientes proyectos de investigación: Universidad Nacional Autónoma de México (DGAPA-UNAM, PAPIME PE104318, PAPIIT 212714 and IV200418); y por proyectos del Consejo Nacional de Ciencia y Tecnología (CONACyT), México (CB-2009131008, SAGARPA-CONACYT 291333, LANASE 2015-250996, 2016-271449, 2017-280505 2018293701, Repositorios Institucionales CONACyT, 271432).

Agradezco a mis revisores que se tomaron el tiempo para leer este trabajo y a mi asesor, que compartió su experiencia y conocimientos conmigo.

Agradezco al tecnológico, que marcó la senda y el camino y el camino a seguir, y como si no fuera suficiente, puso ante mí, excelentes guías, a quienes siempre estaré agradecido por su tiempo y dedicación.

Sinceramente agradezco a mis compañeros (que al leer esto se sentirán identificados), quienes siempre han estado conmigo, de quienes y con quienes aprendí todo cuanto fue posible en este camino. Muchas cosas hemos pasado, pero ahora podemos decir orgullos que vemos solo ingenieros entre nosotros. Verdaderamente cada titulación es un logro de los 4.

Profundamente agradezco a mis padres, que han hecho de mi un hombre de bien y un profesionista, que, con el ejemplo, me han dado las lecciones más importantes, quienes tal vez no saben aún como encender una computadora y, sin embargo, sin todo lo que me han enseñado, yo no habría terminado este trabajo. Sé que al leer este párrafo se sentirán orgullos, no debería haber otro sentimiento de un parente hacia sus hijos, también sé que al leer las otras 100 páginas se sentirán confundidos, no debería haber otro sentimiento en una persona al leer por primera vez sobre abejas y redes convolucionales.

# Índice de contenido

|  |     |
|--|-----|
| Índice de contenido .....  | iv  |
| Índice de figuras .....  | vii |
| Índice de tablas .....   | ix  |
| Capítulo 1. Introducción .....   | 1   |
| 1.1 Antecedentes .....   | 1   |
| 1.2 Planteamiento del problema .....   | 3   |
| 1.3 Objetivos .....  | 5   |
| 1.3.1. Objetivo General .....  | 5   |
| 1.3.2. Objetivos Específicos .....   | 5   |
| 1.4 Hipótesis .....  | 6   |
| 1.5 Contribuciones .....   | 6   |
| Capítulo 2. Estado del arte .....  | 8   |
| 2.1 Introducción a la observación apícola .....                                | 8   |
| 2.1.1. Variables que afectan a las colonias .....                              | 8   |
| 2.2 Observación poblacional y de variables ambientales .....                   | 10  |
| 2.2.1 Monitoreo de variables específicas .....                                 | 10  |
| 2.2.2 Monitoreo de variables en conjunto .....                                 | 12  |
| 2.2.3 Métodos de conteo básicos .....  | 12  |
| 2.2.3.1. Soluciones electromecánicas .....                                     | 12  |
| 2.2.3.2. Microsensores y chips .....   | 13  |
| 2.2.4. Métodos para detección en imágenes .....                                | 15  |
| 2.2.4.1. Introducción al tratamiento de imágenes en ambientes biológicos ..... | 15  |
| 2.2.4.2. Enfoques en detección de imágenes .....                               | 17  |
| 2.2.4.2.1. Catalogación .....  | 17  |
| 2.2.4.2.2. Reacciones a variables externas .....                               | 18  |
| 2.2.4.2.3. Patrones de reconocimiento simple .....                             | 18  |
| 2.2.4.2.4. Detección de movimiento con asimilación a figuras simples .....     | 19  |
| 2.2.4.2.5. Clasificación .....   | 23  |

## Índices

|  |     |
|--|-----|
| 2.2.4.2.6. Aprendizaje de máquinas.....  | 24  |
| Capítulo 3. Arquitectura Propuesta.....  | 38  |
| 3.1. Campo, infraestructura en los apiarios .....  | 43  |
| 3.2. Comunicación Remota (salida a internet).....  | 67  |
| 3.3. Centro de datos (almacén central de información) .....  | 69  |
| Capítulo 4. Resultados .....   | 73  |
| 4.1. Red de sensores mediante IoT .....  | 73  |
| 4.1.1. Caracterización de sensores y asignación de rangos normales de operación. ....  | 74  |
| 4.1.2. Recolección de lecturas, envío y almacenamiento .....   | 74  |
| 4.1.2.1. Comparativa con dataloggers tradicionales.....  | 74  |
| 4.1.2.1.1. Organización de la información .....  | 75  |
| 4.1.2.1.2. Disponibilidad de la información .....  | 76  |
| 4.1.2.2. Comparativa con envíos por FTP.....   | 78  |
| 4.1.3. Desarrollo de mecanismos para garantizar la integridad de la información y notificar de fallas en agentes o sensores..... | 80  |
| 4.2. Detección de abejas utilizando aprendizaje de máquinas.....   | 81  |
| 4.2.1. Comparativa entre métodos .....   | 81  |
| 4.2.1.1. COCO Model.....   | 81  |
| 4.2.1.2. iNaturalist Model.....  | 84  |
| 4.2.1.3. Reconocimiento de color .....   | 87  |
| 4.2.1.4. Clasificación ImageNet .....  | 92  |
| 4.1.1.1. Detección de movimiento y clasificación .....   | 97  |
| 4.2.2. Complementación RFID-Aprendizaje de máquinas.....   | 102 |
| Capítulo 5. Conclusiones.....  | 103 |
| 5.1. Resumen.....  | 103 |
| 5.2. Conclusiones Finales .....  | 104 |
| 5.2.1. Implementación IoT .....  | 104 |
| 5.2.2. Procesamiento de imágenes .....   | 104 |
| 5.2.3. Trabajo futuro .....  | 105 |
| Capítulo 6. Bibliografía .....   | 107 |
| Capítulo 7. Apéndices .....  | 112 |
| 7.1. Red de sensores .....   | 112 |

## Índices

|        |                               |     |
|--------|-------------------------------|-----|
| 7.1.1. | Componentes de software ..... | 112 |
| 7.1.2. | Componentes de hardware.....  | 112 |
| 7.2.   | Aprendizaje de máquinas.....  | 113 |
| 7.2.1. | Componentes de software ..... | 113 |
| 7.2.2. | Componentes de hardware.....  | 114 |

# Índice de figuras

|   |    |
|---|----|
| Figura 1.1 Localización de chips en el cuerpo de una abeja .....  | 3  |
| Figura 2.1 Información sobre algunos pesticidas y efecto en las abejas (Agence France-Presse, 2018) . ....                                | 10 |
| Figura 2.2 Virus varroa en abejas. Fuente: (Bulacio, 2018).....   | 10 |
| Figura 2.3 Apidictor original (Estados Unidos Patente nº 2,806,082, 1957).....  | 11 |
| Figura 2.4 SmartHive web reporting Fuente: (Bromenshenk, y otros, 1996).....  | 12 |
| Figura 2.5 Diagrama de sensor led. Fuente (Octavio, Márquez, Ángela, Sebastian, & Giovanni, 2012) .....                                   | 13 |
| Figura 2.6 Abeja y comparación con chips (Commonwealth Scientific and Industrial Research Organisation (CSIRO), 2015).....                | 14 |
| Figura 2.7 Se etiqueta solo una muestra de la población .....   | 15 |
| Figura 2.8 Microscopio utilizado para clasificación de patrones en alas. Fuente: (Tom, Stefan, Volker, & Dieter, 2001) .....              | 18 |
| Figura 2.9 Abejas detectadas en tiempo real mediante etiquetas con letras. Fuente: (Chiu, En-Cheng, Joe-Air, & Ta-Te, 2012).....          | 19 |
| Figura 2.10 Píxeles blancos, movimiento; píxeles negros, fondo. Fuente: (Gang Jun Tu, 2016) .....   | 20 |
| Figura 2.11 Asimilación a figuras sencillas y marcas en centroides. Fuente: (Gang Jun Tu, 2016)....                                       | 21 |
| Figura 2.12 Posicionamiento en entorno real. Fuente: (Campbell, Mummert, & Sukthankar, 2008) .....  | 22 |
| Figura 2.13 Entrada de una colmena real. Fuente: (Campbell, Mummert, & Sukthankar, 2008) ....   | 23 |
| Figura 2.14 Resultados de la clasificación, distinción entre abejas y abejorros Fuente: (Dagher, 2016) .....                              | 24 |
| Figura 2.15 Patrón identificador, abejas etiquetadas en ambientes reales. Fuente: (Wario, Wild, Couvillon, Rojas, & Landgraf, 2015) ..... | 25 |
| Figura 2.16 Detección de un patrón en tiempo real. Fuente: (Wario, Wild, Couvillon, Rojas, & Landgraf, 2015) .....                        | 26 |
| Figura 2.17 Elementos de una neurona básica. Fuente: (Matich, 2001) .....   | 28 |
| Figura 2.18 Red neuronal simple con capas. Fuente: (García, 2013).....  | 30 |
| Figura 2.19 Supresión No Máxima de áreas con probabilidad de detección. Fuente: (Weiguang & Graham, 2016).....                            | 33 |
| Figura 3.1 Arquitectura de IoT, Fuente: (Khan, Khan, Zaheer, & Khan, 2012).....   | 39 |
| Figura 3.2 Arquitectura propuesta y flujo de la información.....  | 41 |
| Figura 3.3 Detalles del lago de datos y flujo de información .....  | 43 |
| Figura 3.4 Estación meteorológica (pluviómetro y anemómetro).....   | 45 |
| Figura 3.5 Piquera adaptada en entrada de un panal. Fuente: (Engelke, Marendy, Nguyen, De Souza Junior, & Williams, 2014) .....           | 48 |
| Figura 3.6 Capas comunes en una red neuronal convolucional (IBM Corporation, 2016) .....  | 51 |
| Figura 3.8 Aproximación a una arquitectura CNN real donde se repiten las capas. (IBM Corporation, 2016) .....                             | 52 |

## Índices

|   |    |
|---|----|
| Figura 3.9 Comparativa en milisegundos entre GPU y CPU .....  | 54 |
| Figura 3.10 Diagrama de inicialización de matrices .....  | 56 |
| Figura 3.11 Flujo de información para la segmentación de tensores de imágenes.....  | 57 |
| Figura 3.12 Flujo de datos principal del módulo "Learning Rate".....  | 58 |
| Figura 3.13 Entradas y salidas de los submódulos de optimizador .....   | 58 |
| Figura 3.14 Entradas y salidas generales del módulo FeatureExtractor .....  | 59 |
| Figura 3.15 Fragmento del módulo desglosado "FeatureExtractor" .....  | 60 |
| Figura 3.16 Desglose del módulo "ClassPredictor" .....  | 61 |
| Figura 3.17 Desglose del módulo "BoxPredictor" y flujo de información .....   | 61 |
| Figura 3.18 Rieles para el paso de polinizadores .....  | 63 |
| Figura 3.19 Cámara colocada en la tapa de la piquera .....  | 63 |
| Figura 3.20 Vista trasera general del prototipo de piquera .....  | 64 |
| Figura 3.21 Apreciación gráfica de los elementos principales de un LSTM (Mathworks, 2018) .....                                     | 65 |
| Figura 3.22 Subsección "Campo", Detalles de tecnologías, sensores y dispositivos utilizados .....                                   | 67 |
| Figura 3.23 Subsección "Comunicación Remota", Tecnologías y protocolos .....  | 69 |
| Figura 3.24 Subsección "Centro de Datos", Funcionamiento y flujo de la información en el servidor .....                             | 72 |
| <br>  |    |
| Figura 4.1 Los datos se guardan como un compendio y cada tupla se relaciona con su lugar de procedencia y tipo de información ..... | 75 |
| Figura 4.2 Los datos están en constante actualización y son accesibles en cualquier momento.....                                    | 76 |
| Figura 4.3 Los tiempo pueden ajustarse incluso en minutos .....   | 77 |
| Figura 4.4 Las publicaciones consisten en dos cadenas sencillas .....   | 78 |
| Figura 4.5 Tiempos representativos de envíos mediante FTP .....   | 79 |
| Figura 4.6 Tiempos representativos de publicaciones mediante protocolo MQTT .....   | 80 |
| Figura 4.7 Ejemplo simple de alerta cuando el valor obtenido sale del rango normal de operación .....                               | 81 |
| <br>  |    |
| Figura 4.8 Confusión del modelo entre objetos.....  | 82 |
| Figura 4.9 Detección de una persona en un área de agrupación de abejas .....  | 83 |
| Figura 4.10 Detección de abejas con modelo a la medida.....   | 83 |
| Figura 4.11 Detección de modelo ad hoc.....   | 84 |
| Figura 4.12 Imagen de abeja. Detección y porcentaje arrojado por iNaturalist model.....   | 85 |
| Figura 4.13 Detección de abeja y porcentaje de similitud arrojado por el modelo del proyecto ....                                   | 86 |
| Figura 4.14 Comparativa de tiempo por iteración .....   | 87 |
| Figura 4.15 El amarillo detectado en la abeja es el mínimo .....  | 88 |
| Figura 4.16 Debido a la menor cantidad de luz se aprecia una mejor reconstrucción de la abeja ..                                    | 88 |
| Figura 4.17 El entorno también se clasifica .....   | 89 |
| Figura 4.18 Abeja localizada mediante aprendizaje de máquinas .....   | 89 |
| Figura 4.19 Discernimiento entre objetos y entorno .....  | 90 |
| Figura 4.20 A pesar de los cambios de luz sigue detectado ya que se basa en las características...                                  | 90 |
| Figura 4.21 Imagen 1 de detección de color en ambiente no controlado .....  | 91 |
| Figura 4.22 Imagen 2 de detección de color en ambiente no controlado .....  | 91 |
| Figura 4.23 Imagen 3 de detección de color en ambiente no controlado .....  | 91 |

## Índices

|  |     |
|--|-----|
| Figura 4.24 Detección de varias abejas en ambiente no controlado .....                                   | 92  |
| Figura 4.25 Detección de abejas en ambiente no controlado .....  | 92  |
| Figura 4.26 Detección individual en ambiente no controlado.....  | 92  |
| Figura 4.27 Imagen base de los ejemplos anteriores.....  | 92  |
| Figura 4.28 Clasificaciones arrojadas y porcentaje de similitud .....                                    | 94  |
| Figura 4.29 El entorno no confunde la clasificación siempre que el objeto ocupe un plano principal ..... | 94  |
| Figura 4.30 El algoritmo delimita la zona de detección .....   | 95  |
| Figura 4.31 Múltiples detecciones con su porcentaje de similitud cada una .....                          | 95  |
| Figura 4.32 Abeja aislada y clasificada .....  | 96  |
| Figura 4.33 Mejor resultado obtenido con clasificador .....  | 96  |
| Figura 4.34 Aislamiento de movimiento en un círculo (pocas abejas) .....                                 | 97  |
| Figura 4.35 Circunferencia rodeando área grande de detección de movimiento .....                         | 98  |
| Figura 4.36 Se aísla el movimiento detectado .....   | 98  |
| Figura 4.37 Clasificación del aislamiento de movimiento.....   | 99  |
| Figura 4.38 Mejor tiempo de clasificación .....  | 100 |
| Figura 4.39 Tiempo de detección más común .....  | 100 |
| Figura 4.40 Peor tiempo de detección .....   | 101 |
| Figura 4.41 Gráfica de tiempos efectuados en 160 procesos de convolución completos.....                  | 101 |

## Índice de tablas

|  |    |
|--|----|
| Tabla 1 Comparativa entre trabajos relevantes..... | 37 |
|--|----|

# Capítulo 1. Introducción

## 1.1. Antecedentes

Las abejas son uno de los grupos más comunes de insectos, y son de gran importancia ecológica y económica gracias a sus hábitos alimenticios. La visita a las flores en busca de néctar y polen tiene como consecuencia la polinización de un gran número de plantas de interés para otros organismos (Nates-Parra, 2005)

La polinización, entendida como la transferencia de polen desde la parte masculina de una flor hasta la parte femenina de la misma u otra flor, es un proceso esencial para el mantenimiento de la viabilidad y la diversidad genética de las plantas con flor, además de mejorar la calidad y cantidad de semillas y frutos, así como de las características de la descendencia (Chautá-Mellizo, Campbel, Bonilla, Thaler, & Poveda, 2012). Puede ser realizada de forma abiótica, mediante el transporte del polen por el viento o el agua, o biótica, empleando para ello a animales como vectores en el transporte. (Bonilla, 2012)

Entre los aspectos más importantes de la polinización en el planeta de acuerdo con (United States Department of Agriculture, 2017) y (Organización de las naciones unidas para la alimentación y la agricultura, 2019) podemos destacar:

- Los polinizadores apoyan la biodiversidad: existe una correlación entre la diversidad de plantas y la diversidad de polinizadores.
- El aumento de los rendimientos y los cultivos de mayor calidad, son beneficios que los cultivadores y consumidores obtienen de una población de polinizadores saludables, nativos o administrados.

## Capítulo 1. - Introducción

- En todo el mundo, aproximadamente 1.000 plantas cultivadas para alimentos, bebidas, fibras, especias y medicamentos deben ser polinizadas por animales para producir los bienes de los que dependemos.
- La población de polinizadores de un área es un gran indicador de la salud general de un ecosistema.
- El 90 por ciento de la cosecha de manzanas de estados unidos es polinizada por las abejas.
- 3 de cada 4 cultivos que producen frutos o semillas para consumo humano dependen, al menos en parte, de los polinizadores.
- Los productos alimenticios que dependen de la polinización contribuyen a las dietas saludables y a una mejor nutrición.

Existen teorías sobre el comportamiento de las poblaciones de abejas y su reciente decrecimiento a nivel mundial, algunos investigadores teorizan que su declive no es un fenómeno nuevo, ya que se han observado casos similares a través de distintas épocas en la historia (Kulincevic, Rothenbuhler, & Rinderer, 1982), otros investigadores afirman que si la temperatura ambiental decrece lo suficiente y se mantiene así por un periodo de tiempo suficientemente largo o permanentemente, puede afectar a los individuos adultos y jóvenes (Stankus, 2008) e incluso aquellas que resisten inviernos largos se caracterizan por una descendencia menos eficiente en el mantenimiento de la colmena (Groh, Tautz, & Rossler., 2004). Un enfoque distinto de teorías asegura que la desaparición de las abejas puede deberse a la excesiva depredación y parasitación que afecta las colonias, desde microorganismos que afectan la salud de la abeja, hasta animales grandes como ratones, mapaches y osos (de estos últimos se ha comprobado que prefieren comerse las abejas y el panal más allá de la miel) (Stankus, 2008).

La tecnología ha jugado un papel crucial en la observación de este suceso, se han desarrollado mayormente sistemas que observan variables individuales, entre los

## Capítulo 1. - Introducción

más destacados estos son: El monitoreo de la temperatura y humedad (Jurijs, Aleksejs, & Egils, 2009) ha sido un factor común en el monitoreo de los panales, también se ha implementado visión en tiempo real de los apiarios, así como procesamiento de imágenes para identificación de los individuos (Chiu, En-Cheng, Joe-Air, & Ta-Te, 2012) y se ha trabajado también con microsensores que son colocados físicamente en la parte posterior de la abeja, arriba del nacimiento de las alas (Commonwealth Scientific and Industrial Research Organisation (CSIRO), 2015) (véase Figura 1.1).



Figura 1.1 Localización de chips en el cuerpo de una abeja

### 1.2. Planteamiento del problema

Evidencias recientes han documentado un alarmantemente declive global de abejas tanto silvestres como domesticadas, en campos de cultivo y en paralelo un significativo declive de la producción agrícola. Estos patrones de mortalidad de abejas melíferas han sido asociados a la aparición de un síndrome particular, llamado por sus siglas en inglés CCD (Colony Collapse Disorder) o Síndrome de Colapso de Colmena en español, se caracteriza por la súbita desaparición de las poblaciones de obreras y abandono de la cría. (Quesada Avedaño M. , 2010)

En México más del 85% de los cultivos dependen del servicio natural de polinización que ofrecen las abejas, uno de los más importantes polinizadores de las plantas, y su disminución tendría efectos negativos sobre la producción de frutos y legumbres de importancia nacional e internacional como el aguacate y los cítricos, entre otros. El abasto nacional e incluso la seguridad alimentaria de los mexicanos, además de la salud general de la población, puede verse seriamente amenazada si no abordamos científicamente este problema. (Castañeda-Vildózola, y otros, 1999)

## Capítulo 1. - Introducción

Aún con el gran impacto de este problema a nivel a mundial, no existen sistemas que integren la observación y registro de múltiples variables presentes en las colonias y apiarios enfocados totalmente a la investigación. La observación de variables individuales y aisladas, así como los sistemas enfocados a la industria, han sido el enfoque preponderante en el estudio de las abejas y en el desarrollo de sistemas para el monitoreo de las mismas.

Las soluciones industriales, además de poseer un enfoque distinto a los estudios científicos, son de costos muy elevados, generalmente de proveedores extranjeros. El gasto que implica adquirir y trabajar con estos sistemas, está muy por encima del presupuesto de la mayoría de las instituciones dedicadas a la investigación en países subdesarrollados, al igual que el desarrollo de sistemas a la medida a través de empresas consultoras privadas.

Las soluciones desarrolladas en centros de investigación, proveen un punto de vista muy limitado de los eventos en los apiarios, por el ya mencionado enfoque de estudio en variables aisladas o individuales. La observación se limita a alrededor de 3 variables en campo, comúnmente estas suelen ser temperatura interna de la colmena o ambiental, humedad igualmente interna o ambiental y/o peso, con recolección de datos periódica (en la ubicación donde se encuentren los instalados los sistemas), lo que involucra incursiones en campo y vacíados posteriores de la información (paso intermedio que podría omitirse haciendo uso de tecnologías de comunicación a distancia), estas soluciones (desde el punto de vista computacional) son muy poco escalables e imprácticas, además de ser poco automatizadas y amigables (punto importante ya que los usuarios de estos sistemas cuentan con conocimientos muy básicos de computación).

El presente proyecto propone la creación de una red de sensores que permita la observación de variables tanto dentro como fuera de la colmena (ver subsección de objetivos específicos 1.3.2), así como un conteo automatizado de la población en general en los apiarios. De la misma forma, se propone un diseño integrado para el estudio de las abejas, que permita estudiar diversas variables sin necesidad de adquirir varios sistemas individuales, comenzando por la observación individual, la

## Capítulo 1. - Introducción

observación ambiental dentro de las colmenas y la observación de factores ambientales que envuelven al apiario en general. El sistema tendrá ventajas en el uso en proyectos de investigación por su bajo costo y amplia gama de variables que permite observar, aunque es también viable para sistemas industriales en producción.

### 1.3. Objetivos

#### 1.3.1. Objetivo General

Desarrollar una red de sensores de bajo costo para el monitoreo de polinizadores, usando el paradigma internet de las cosas, sistemas embebidos para la observación de variables internas y ambientales. Así como conteo poblacional utilizando procesamiento de imágenes mediante aprendizaje de máquinas y microsensores.

#### 1.3.2. Objetivos Específicos

- Caracterización de sensores y asignación de rangos normales de operación.
- Desarrollo de algoritmos para obtener datos ambientales utilizando sensores tales como: pluviómetro (cantidad de lluvia en el área), anemómetro (velocidad del viento), termómetro ambiental (temperatura del área del apiario), fotómetro (cantidad de luz ambiental) e higrómetro (humedad del área del apiario).
- Desarrollo de algoritmos para obtener datos internos de la colmena, utilizando los siguientes sensores: Sonómetro, termómetro, vibrómetro, acelerómetro, balanza.
- Configuración de sistemas embebidos (Intel Edison y raspberry Pi) para trabajar con el gestor de bases de datos PostgreSQL.
- Desarrollo de bases de datos para almacenar información local.
- Implementación de mecanismos de transmisión de información utilizando una red GSM.
- Configuración de servidores para transferencia de información utilizando el protocolo MQTT mediante el servicio Mosquitto.

## Capítulo 1. - Introducción

- Desarrollo de algoritmos “suscriptores” para recepción y posterior almacenamiento de la información.
- Desarrollo de mecanismos para garantizar la integridad de la información y notificar de fallas en agentes o sensores.
- Entrenamiento de modelos para detección de abejas utilizando aprendizaje de máquinas.
- Desarrollo de algoritmos para detección de abejas en imágenes y conteo poblacional de individuos mediante entrenamiento de modelos de redes neuronales para aprendizaje de máquinas.

### 1.4. Hipótesis

Los métodos electromecánicos se han utilizado ampliamente en el conteo poblacional apícola, sin embargo, el procesamiento de imágenes por computadora mediante redes neuronales podría alcanzar un enfoque poblacional más extenso y preciso. El almacenamiento de información, generalmente llevado a cabo con “data loggers”, almacenan datos que deben ser recolectados en incursiones periódicas a campo. El internet de las cosas puede dar la ventaja de la observación y envío de datos en tiempo real, y así reducir las incursiones en campo, ya que permite trabajar con dispositivos de pocos recursos, en lugares apartados y frecuencias de envío en minutos, proveyendo mecanismos para el envío de datos desde sistemas embebidos a grandes plataformas, de una manera eficiente y a bajo costo. La red GSM podría ser una tecnología totalmente viable para el envío de lecturas de sensores mediante internet de las cosas.

### 1.5. Contribuciones

El enfoque predominante de los sistemas para la observación apícola, ha sido el estudio de variables individuales. Esta tesis presenta una mejora a la observación de variables aisladas, integrando variables desde un nivel individual dentro de las colmenas, hasta un nivel generalizado en el apiario y adicionalmente variables ambientales. También, se presenta una mejora en el tratamiento y transferencia de

## Capítulo 1. - Introducción

la información, ya que el sistema es capaz de comunicarse con los centros de procesamiento de las instituciones para envío de datos, eliminando la recolección en campo de los mismos, lo cual es posible, gracias a la incursión del paradigma de Internet de la Cosas, pues el sistema se basa en arquitectura IoT, enfocada a la observación biológica de las abejas y en tecnología optimizada para este paradigma.

El presente proyecto propone una arquitectura compuesta de tres secciones (ver Arquitectura Propuesta), la primera y más importante denominada “Campo” con el fin del monitoreo variables en tres niveles, individual, colmena y apíario; la siguiente sección denominada “Comunicación Remota” permite que cada dispositivo en campo envíe la información recolectada a los servidores designados para el proyecto y por último la subsección “Centro de Datos” que recibe la información gracias a la subsección precedente, la interpreta y la almacena en base de datos.

# Capítulo 2. Estado del arte

La observación apícola, tecnológica y tradicional, ha sido un área ampliamente estudiada y de gran preocupación en los últimos años debido al reciente descenso de individuos en las poblaciones de abejas. En este capítulo se hará una revisión general de las teorías más aceptadas acerca del creciente índice de mortalidad en las poblaciones de abejas, los trabajos más relevantes de distintos enfoques de este estudio, y trabajos importantes, realizados estos últimos años, con el uso de tecnologías inteligentes.

## 2.1. Introducción a la observación apícola

Los grandes servicios que proveen las abejas como polinizadores y la gran preocupación científica que ha provocado el descenso general dentro de sus poblaciones, ha requerido, ya desde el siglo pasado, soluciones tecnológicas capaces de monitorear a esta especie dentro de su propio hábitat y observar los comportamientos comunes de las mismas más allá de los escenarios controlados en laboratorios, para esto se ha ido diseñando y perfeccionando soluciones y adaptaciones tecnológicas, que permitiesen observar variables directamente en colmenas, que pudiesen llevar registros históricos del monitoreo de variables y ayudasen a los investigadores del campo a comprobar o refutar teorías sobre posibles variables o comportamientos que estuviesen influyendo en el descenso de las poblaciones a nivel mundial (Bortolotti, y otros, 2009).

### 2.1.1. Variables que afectan a las colonias

A través del tiempo han ido surgiendo teorías acerca de este problema que ha afectado a las abejas alrededor del mundo, y con éstas, los diferentes enfoques hacia donde se dirigen los objetivos de los proyectos y los sistemas en que se apoyan para la observación. Un enfoque reconocido de esta área es el de las señales y su influencia en las abejas, que argumenta que ciertas señales pueden influenciar en sistemas internos de las abejas e incluso en su comportamiento, así como dificultar su retorno a la colmena una vez que los individuos salen en busca

## Capítulo 2. - Estado del arte

de alimento, este enfoque sostiene que los órganos de las abejas, al ser más sensibles que los de muchos animales y seres humanos tienden a ser afectados por señales de comunicación como las señal celular (Favre, 2011) (Harst, Kuhn, & Stever, 2006) (Stever, y otros, 2007).

Otro enfoque importante es el estudio del desgaste de los suelos y el uso de fertilizantes y el impacto que estos pueden tener sobre especie pequeñas como las abejas (Bortolotti, y otros, 2009) (Commonwealth Scientific and Industrial Research Organisation (CSIRO), 2015) el cual estudia variables propiamente de las propiedades en distintos suelos y como el desgaste de estos afecta al ecosistema y principalmente a las abejas, a su vez el en fertilizantes se concentra en el estudio químico y efectos que tienen estos sobre los suelos y directamente sobre las abejas (ver Figura 2.1), recientemente se ha prohibido el uso de ciertos fertilizantes, pesticidas y sustancias químicas en países de Europa, debido al efecto nocivo que provocan sobre los ecosistemas, de igual manera se han estudiado riesgos naturales externos con un enfoque hacia depredadores naturales, animales como mapaches u osos hasta microorganismos, virus (ver Figura 2.2), parásitos y bacterias que afectan a las poblaciones de abejas (Fung, 2017) (Diana L., y otros, 2007) (W., 2007).

Por último, uno de los enfoques más reconocidos mundialmente, el estudio de variables ambientales que envuelven a las colonias de polinizadores, este enfoque se basa en la observación de variables presentes en las colmenas y en el ecosistema donde se encuentran, generalmente estudiando variables muy específicas mediante sistemas, sean mecánicos, electrónicos y más recientemente de computo; los proyectos pertenecientes a este enfoque basan sus teorías en los efectos que tienen las condiciones ambientales sobre las poblaciones estudiadas, específicamente se estudia un conjunto de variables limitadas y generalmente aisladas, es este enfoque al que está dirigido el presente proyecto de investigación, es un enfoque de estudio sobre las variables ambientales en las colmenas.

## Pesticidas neonicotinoides y las abejas

Son los insecticidas más utilizados en el mundo (1/3 de todas las ventas) Usados desde los años 90

### Neuroactivos

químicamente similares a la **nicotina**  
Los de mayor preocupación incluyen:

#### Imidacloprid

$C_9H_{10}ClN_5O_2$



#### Bayer

- Gaucho
- Confidor



#### Thiametoxam

$C_8H_{10}ClN_5O_3S$



#### Syngenta

- Cruiser
- Actara

#### Clotianidina

$C_6H_8ClN_5O_3S$



#### Bayer Sumitomo Chemical Takeda

- Poncho
- Cheyenne
- Dantop
- Santana

- Acusados de atacar el sistema nervioso de las abejas, paralizándolas y causando su muerte.

- También de reducir la fertilidad de las abejas y su resistencia a las enfermedades

Persisten en las semillas, las hojas, el agua, el suelo, el polen y el néctar



Legislación de la UE

Prohibición temporal y parcial de *imidacloprid*, *tiametoxam* y *clotianidina* desde 2013 en cultivos que atraen abejas (excepto cereales de invierno)

Posición de los miembros de la UE

8 estados se opusieron a la prohibición de 2013 pero deben respetarla: Austria, República Checa, Italia, Hungría, Portugal, Rumanía, Eslovaquia y Reino Unido

© AFP

Fuentes: Ministerio de Agricultura francés, ANSES, Ineris, Greenpeace, Universidad de Bern

Figura 2.1 Información sobre algunos pesticidas y efecto en las abejas (Agence France-Presse, 2018).



Figura 2.2 Virus varroa en abejas. Fuente: (Bulacio, 2018)

## 2.2. Observación poblacional y de variables ambientales

### 2.2.1. Monitoreo de variables específicas

El interés el estudio y observación de las abejas por largos períodos de tiempo no es un campo nuevo, en (Gates, 1914), se recolectaron datos referentes a la

## Capítulo 2. - Estado del arte

temperatura en las colmenas en intervalos de 1 a 3 horas por poco más de un año, sin embargo, los sistemas tecnológicos ayudaron a hacer de este trabajo una tarea más automatizada y auto-realizable, permitiendo a investigadores y apicultores llevar a cabo un monitoreo más eficiente y con el mínimo de esfuerzo humano (Meikle, Rector, Mercadier, & Holst, 2008). La primer gran combinación de sensores electrónicos y monitoreo de abejas fue un analizador de frecuencias llamado Apidictor (Estados Unidos Patente nº 2,806,082, 1957) el cual era un filtro de frecuencia de banda, (ver Figura 2.3) para detectar el cambio en el sonido que ocurre en las colonias de dos a tres semanas antes de la enjambrazón, que es el proceso natural por medio del cual, una colonia puede dividirse y dar varias colonias hijas, propagando así la especie (Francisco Padilla Álvarez, 2011); en recientes años ha resurgido en la idea del Apidictor en el Reino Unido, pensado con la funcionalidad de conexión a la red pública. Además, el proyecto aún sigue en desarrollo.



Figura 2.3 Apidictor original (Estados Unidos Patente nº 2,806,082, 1957)

La idea original del Apidictor fue seguida por métodos de etiquetado y rastreo de abejas, así como un dispositivo acústico para separar las frecuencias de cada abeja en el espectro de sonido (Estados Unidos Patente nº 4,876,721, 1989), dos invenciones originales del Oak Ridge National Laboratory en USA (Oak Ridge National Laboratory , 1989), estos sistemas mejoraron el Apidictor puesto que permitían un seguimiento de un individuo de la especie, una comprensión a un nivel más específico de las colonias, e incluso la técnica de etiquetado individual se fue

## Capítulo 2. - Estado del arte

mejorando con el tiempo hasta llegar en fechas recientes al uso de microsensores y sistemas electrónicos para la detección de abejas.

### 2.2.2. Monitoreo de variables en conjunto

Mejorando la accesibilidad de los sistemas para monitoreo de abejas, en 1990 se comenzaron a distribuir los primeros “Bee Counters”, sensores electromecánicos, colocados en las entradas de las colmenas para detectar el paso de abejas y poder registrar actividad en las mismas (Struye, Mortier, Arnold, Miniggio, & Borneck, 1994), entonces por primera vez en 1995 se dio a conocer al público un sistema conocido como “SmartHives”, que presentaba la primer integración de sensores electrónicos para monitoreo de colmenas (ver Figura 2.4), incluía balanza, contador de abejas, sensores de temperatura y humedad y comunicación vía celular mediante módulos GSM así como monitoreo a través de internet (Bromenshenk, y otros, 1996) (Estados Unidos Patente nº 6,910,941, 2004) una mejora notable en la observación y estudio de las colmenas, que hasta entonces se habían enfocado mayormente en el estudio de variables aisladas, en lugar del comportamiento de la población y las variables que la envuelven.



Figura 2.4 SmartHive web reporting Fuente: (Bromenshenk, y otros, 1996)

### 2.2.3. Métodos de conteo básicos

#### 2.2.3.1. Soluciones electromecánicas

## Capítulo 2. - Estado del arte

Uno de los problemas más estudiados, es aquel relacionado con el conteo de individuos en una colmena, esta variable, es uno de los índices más representativos en las colonias de polinizadores. Además, el comportamiento, y el estado de la colmena, puede inferirse a partir de un conteo de individuos. El problema como tal ha sido abordado ampliamente, la solución más usual a este problema fueron los sensores infrarrojos y las soluciones electro-mecánicas (Octavio, Márquez, Ángela, Sebastian, & Giovanni, 2012), en este tipo de proyectos, se instalaron dispositivos que envían entre si una señal continua y se ubican generalmente en ambos lados de las entradas y cuando un individuo interrumpe la señal se registra como un avistamiento (ver Figura 2.5), ya que muchos individuos pueden pasar por las amplias entradas de las colmenas en apiarios, se restringe la entrada para que existan varias entradas pequeñas con sus sensores cada una, esto favorece el conteo individual. Como una solución temprana, puede debió refinarse mucho más la idea para hacer un conteo más eficiente.

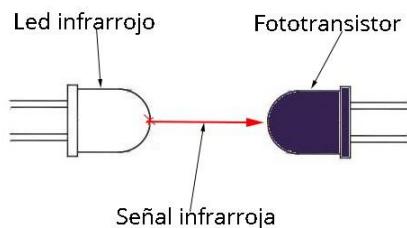


Figura 2.5 Diagrama de sensor led. Fuente (Octavio, Márquez, Ángela, Sebastian, & Giovanni, 2012)

### 2.2.3.2. Microsensores y chips

Se han realizado diversas contribuciones en esta área de investigación y se han refinado los conceptos del conteo de individuos a través del uso de tecnologías de microsensores para la identificación por radio frecuencia RFID (Commonwealth Scientific and Industrial Research Organisation (CSIRO), 2015), entre otros del mismo concepto, la idea tras esta innovación a los métodos electro-mecánicos fue el etiquetar a las abejas con microsensores (ver Figura 2.6) detectables al paso de estas por la entrada de la colmena (piquera), de esta manera no es necesario hacer un gran cambio al entorno en que ellas habitan, con este enfoque, se podrá

## Capítulo 2. - Estado del arte

identificar a individuos específicos y se puede registrar la hora exacta del evento, una importante limitante de este sistema, es la muestra reducida de individuos que se pueden identificar, además del esfuerzo humano que requiere la actividad del etiquetado. Tan solo para encontrar un nuevo lugar donde establecer una población, la decisión grupal debe tomada por más de 5000 abejas (Thomas D. Seeley, 2006), siendo el etiquetado comúnmente menor de 300 abejas (ver Figura 2.7) y teniendo estás un lapso de vida corto, la muestra es muy pequeña y la necesidad de renovar la muestra es alta, además, ya que no se puede predecir dónde las abejas terminaran su vida, la reutilización de los microsensores es imposible.



Figura 2.6 Abeja y comparación con chips (Commonwealth Scientific and Industrial Research Organisation (CSIRO), 2015)



Figura 2.7 Se etiqueta solo una muestra de la población

## 2.2.4. Métodos para detección en imágenes

### 2.2.4.1. Introducción al tratamiento de imágenes en ambientes biológicos

La incursión de soluciones tecnológicas en el área de las ciencias biológicas ha sido muy importante, debido a que ha permitido diversas mejoras en la fase de procesamiento y análisis de datos. Un campo que ha dado grandes contribuciones en los últimos años en las ciencias biológicas es el procesamiento de imágenes, la constante observación del entorno y las condiciones deseadas, realizadas por computadora gracias a los algoritmos que, a través de distintos métodos, pueden reconocer características, patrones, formas y/o colores, sin la mínima intervención humana. Este campo ha tenido diversos enfoques. Entre los más importantes se pueden mencionar los usos agrícolas, que haciendo uso de procesamiento de imágenes, proveen a los agricultores parámetros como el dosel arbóreo, la cantidad de producción, la existencia de plagas y la calidad de los productos, que son

## Capítulo 2. - Estado del arte

parámetros importantes desde el punto de vista de la cosecha (Fabiana S. Santana, 2014), la posibilidad de una observación inteligente y la disponibilidad de una red de comunicación permite adquirir asesoramiento en un tiempo aceptable y a un costo accesible. En la misma área se han implementado imágenes térmicas para la observación de los campos de cultivo, la mayoría se centran en la detección del agua, que afecta las propiedades térmicas de las plantas cuyas hojas contienen ciertas cantidades de agua, esto se usa como un parámetro importante en el proceso de pre-cosecha, en construcciones y estudios de viveros, irrigación, planeación, ataque de termitas, entre otros (Fabiana S. Santana, 2014).

### 2.2.4.2. Enfoques en detección de imágenes

#### 2.2.4.2.1. Catalogación

El procesamiento imágenes y el aprendizaje de máquinas, así como en otras áreas biológicas, ha hecho aportes a la apicultura con distintos enfoques. Podemos observar en los primeros años del milenio un trabajo cuyo enfoque fue la clasificación de distintas especies de abejas mediante imágenes (Tom, Stefan, Volker, & Dieter, 2001), el objetivo de este proyecto fue catalogar distintas especies de abejas mediante imágenes, específicamente mediante el reconocimiento de sus alas (ver Figura 2.8), el argumento principal de este proyecto fue el monitoreo, la protección y el tratamiento de las mismas, pero, de acuerdo con el argumento presentado, una tarea crucial y preliminar era el reconocimiento las mismas especies, sin embargo, su clasificación debía ser llevada a cabo por taxónomos expertos en el campo los cuales eran muy escasos, por esta razón se decidió llevar a cabo el desarrollo de una herramienta que, basada en datos estadísticos y reconocimiento de imágenes pudiera llevar a cabo la clasificación con el mínimo capital humano experto. El proceso se basó en un muestreo de las alas de abejas. Primero debían capturarse muestras y congelarse, una vez hecho esto, se procedía a recolectar imágenes mediante un microscopio para después, generar modelos estadísticos de las venas de las alas y poder reconocer los mismos patrones en otras muestras.

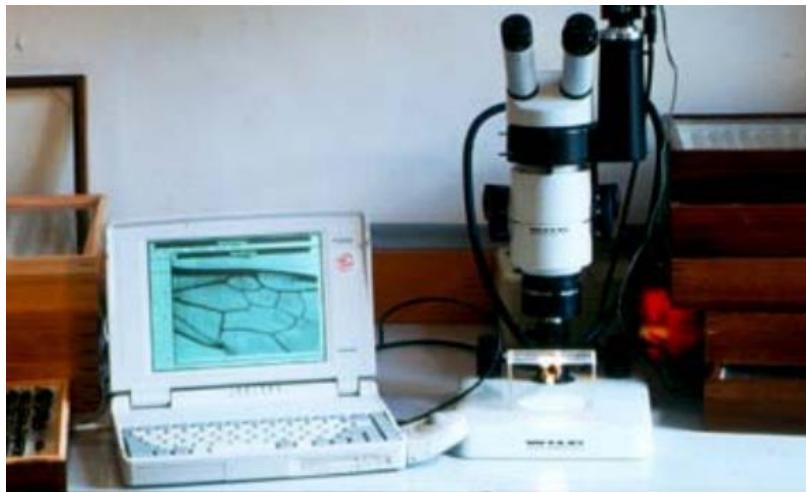


Figura 2.8 Microscopio utilizado para clasificación de patrones en alas. Fuente: (Tom, Stefan, Volker, & Dieter, 2001)

#### 2.2.4.2.2. Reacciones a variables externas

Otro enfoque de monitoreo mediante imágenes, estudia la actividad de las colmenas en presencia de pesticidas y químicos (Bethany S. Teeters, 2012), este proyecto en específico utilizó una solución industrial llamada EthoVisionXT (Noldus, 2018) que se basa en el seguimiento de distintos tipos de animales mediante procesamiento de imágenes, una herramienta ampliamente utilizada en proyectos de investigación de pero de costos poco accesibles para instituciones públicas de educación.

#### 2.2.4.2.3. Patrones de reconocimiento simple

La primera mejora implementada al método de las etiquetas, no fue muy distante al enfoque de los RFID, al igual que en este, la nueva mejora consistía en etiquetar un conjunto de abejas para el estudio, pero en este caso, se utilizaban etiquetas que contuviesen patrones fáciles de reconocer. En el proyecto (Chiu, En-Cheng, Joe-Air, & Ta-Te, 2012) como patrones se utilizaron letras del alfabeto en pequeñas etiquetas en la misma localización en el cuerpo de las abejas, esto resultó una mejora en los costos de operación de los sistemas, ya que no debía invertirse en la fabricación de micro-sensores ni en los dispositivos de reconocimiento de las etiquetas por radio frecuencia, ahora debían imprimirse etiquetas de papel o plásticos muy ligeros, y utilizar cámaras para obtener imágenes de las entradas donde aparecieran las abejas etiquetadas (ver Figura 2.9), este proyecto en

## Capítulo 2. - Estado del arte

particular ofreció una pauta importante para en análisis en las entradas, colocando pequeñas divisiones en las entradas, controlando mejor los accesos a la colmena y asegurando el paso un solo individuo, gracias a esta mejora ahora se pueden tener accesos parcialmente individuales y fluidos, a diferencia del caso más común donde las abejas se concentran en la entrada siendo más difícil así identificarlas mediante el uso de imágenes. Desde el punto de vista del estudio general de una población de abejas, aún era muy costoso etiquetar individuos uno a uno en campo, sin embargo, se obtuvo mayor economía en los materiales para la implementación. De un punto de vista del estudio de comportamiento sobre algunos individuos, esta mejora resultó un buen avance, ahora se podían hacer los estudios más precisos sobre algunos individuos de interés y la renovación de etiquetas no requiere una gran inversión.

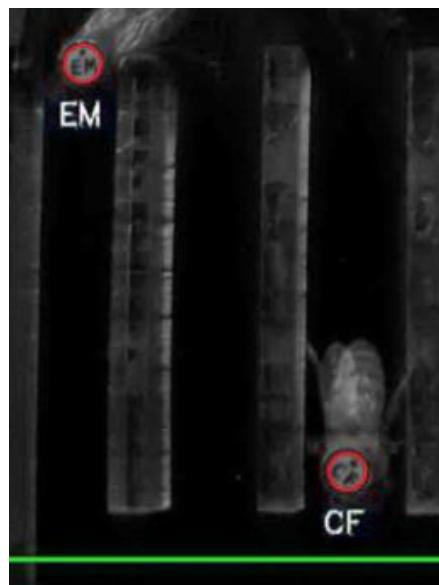


Figura 2.9 Abejas detectadas en tiempo real mediante etiquetas con letras. Fuente: (Chiu, En-Cheng, Joe-Air, & Ta-Te, 2012)

### 2.2.4.2.4. Detección de movimiento con asimilación a figuras simples

El procesamiento de imágenes se utiliza también ampliamente (en soluciones industriales y científicas) la visión por computadora en dispositivos de tamaño

## Capítulo 2. - Estado del arte

reducido, la más comúnmente utilizada es la librería OpenCV para Python o para lenguaje C++ (OpenCV, 2018) que cuenta con herramientas para segmentación y tratamiento de imágenes, ofrece la posibilidad de diseñar y desarrollar sistemas a bajo costo y funcionamiento constante, como podemos ver en proyectos de análisis automatizado (Gang Jun Tu, 2016) , en este proyecto se utilizó OpenCV en sistemas embebidos de bajo costo, se utilizó uno de los métodos más comunes para este tipo de aplicaciones, conocido como “background subtraction” o substracción de fondo, que es una técnica basada en el cambio de los pixeles de una imagen, a partir del cual, se decide, cuales forman parte del conjunto de objetos estudiados y cuales pertenecen solo al fondo (ver Figura 2.10), en adición a este método se utilizó una zona delimitada de vigilancia para poder contar entradas y salidas basándose en una imagen actual y una imagen anterior, de este modo, cuando un ente pasa por la zona de monitoreo, y el algoritmo detecta movimiento, se almacena ese cuadro y se compara con el siguiente, para observar el progreso del movimiento y la dirección que tomaba, pudiendo deducir así entradas y salidas a la colmena.

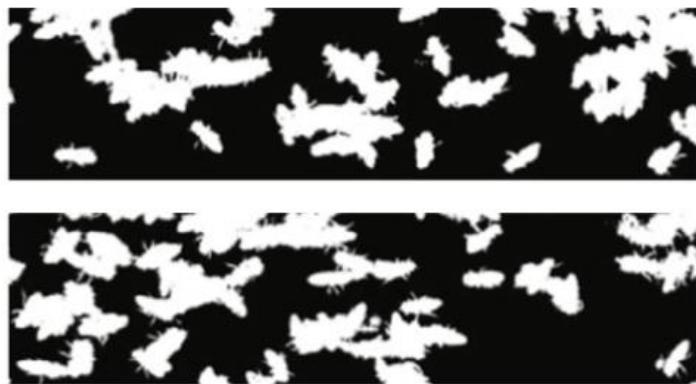


Figura 2.10 Píxeles blancos, movimiento; píxeles negros, fondo. Fuente: (Gang Jun Tu, 2016)

Este proyecto se basa en el modelaje de las abejas individuales como elipses, si se detectaba movimiento, se determinan rasgos muy pequeños hasta dejar en modelo en forma de elipse, se calcula el área dentro del objeto en movimiento en cual era uno de los principales indicadores de detección y se calculaba los centroides de las áreas en movimiento para realizar un conteo de centroides en la imagen (ver Figura 2.11Figura 2.11), para los casos donde se detectaban abejas muy cerca de otras y

## Capítulo 2. - Estado del arte

las áreas no podían modelarse a elipses totalmente, se un pequeño adiestramiento del sistema, para saber cuál es el área normal que ocupan una abeja y poder hacer un estimado de cuantas se localizan en área donde se confunden los límites de cada una en la substracción de fondo.

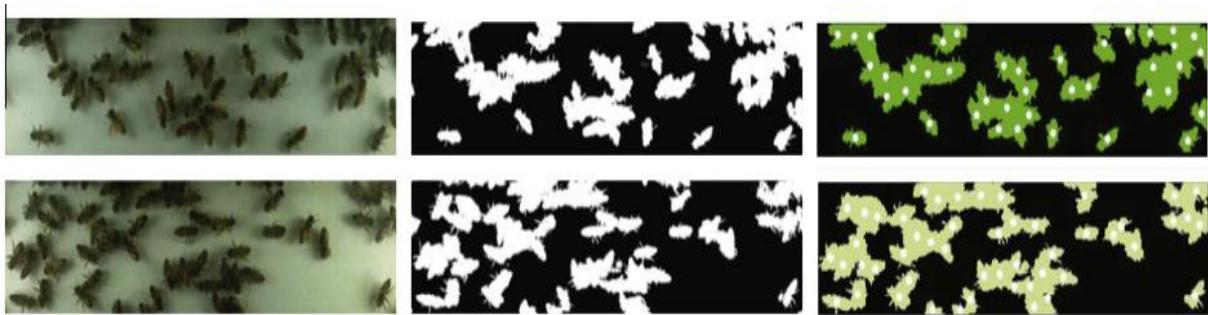


Figura 2.11 Asimilación a figuras sencillas y marcas en centroides. Fuente: (Gang Jun Tu, 2016)

Una ventaja importante de este tratamiento de imágenes con respecto al trabajo anterior, es que se elimina por completo el etiquetado, la población es analizada desde el punto de vista general, aunque esto tambien puede interpretarse como una desventaja dentro de los proyectos que buscan estudiar individuos muy particulares de la sociedad, otra ventaja este tipo de algoritmos es que son facilmente adaptables en equipos pequeños, de pocas capacidades, lo cual permite su instalación en colmenas reales en campo, con el mínimo de invasión, esta solución presenta características bastante escalables y mantenibles, aunque no se tiene contemplado el hecho de que las abejas vuelen cerca de la camara en el área de captura imágenes.

Uno de los principales problemas al intentar detectar y contar abejas en una colmena es la acumulación de las mismas en la entrada de esta. La entrada de una colemana, como lugar estrategico de observación, permitirá saber el número de entradas y salidas que se tienen en periodos específicos de tiempo, sin embargo, la acumulación y el constante movimiento de estas en la entrada, ha dificultado la observación y conteo en las entradas, en el proyecto (Campbell, Mummert, & Sukthankar, 2008), uno de los primeros que utilizaron el tratamiento de imágenes en la observación en apiarios, se utilizó substracción de fondo, pero de una manera

## Capítulo 2. - Estado del arte

distinta, manualmente, se hizo una estimación de los distintos valores que podía tomar el fondo y se creó un pequeño modelo del fondo de las imágenes, para facilitar el desarrollo y también mejorar el rendimiento, instalaron en los apiarios pequeñas plataformas de color blanco en donde las abejas podían llegar antes de entrar a la colmena (ver Figura 2.12).

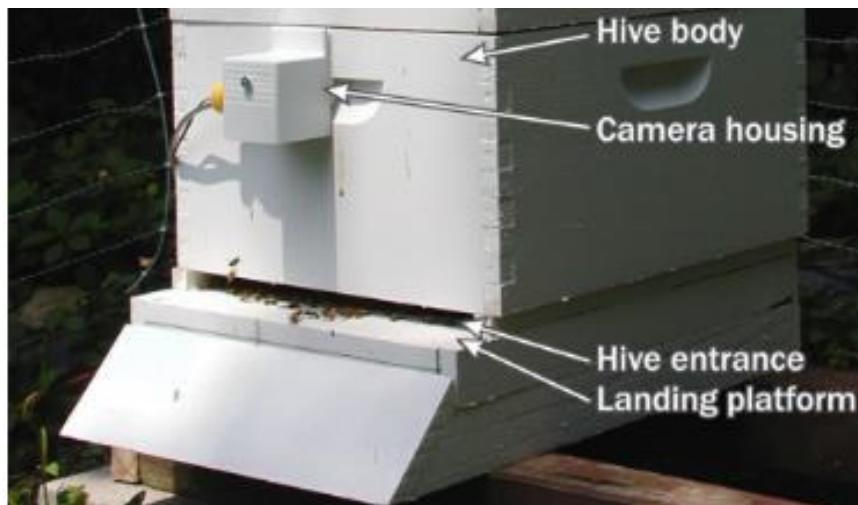


Figura 2.12 Posicionamiento en entorno real. Fuente: (Campbell, Mummert, & Sukthankar, 2008)

Para atacar el problema del movimiento constante en la plataforma, se desarrollaron manualmente, algoritmos que intentaban interpretar los movimientos en 4 modelos distintos loitering, crawling, flying out and flying in, que en español corresponderían a merodeando o movimiento amplio en la plataforma, arrastre o movimiento mínimo en la plataforma, vuelo hacia afuera, para abandonar la colmena y vuelo hacia adentro, o de vuelta a la colmena, los dos primeros, relacionados con el movimiento en la plataforma se diferenciaban del umbral de distancia recorrida entre cuadros del video y permitían identificar aquellas que se concentraban en la entrada sin intenciones mayores de ingresar o salir, los modelos de movimiento en el aire, permitían saber cuándo generalmente había una entrada o una salida, de acuerdo con los autores, el movimiento de salida es el más rápido de todos y el vuelo de entrada se caracteriza por ir

## Capítulo 2. - Estado del arte

alentando el vuelo para alcanzar la plataforma (ver Figura 2.13).



Figura 2.13 Entrada de una colmena real. Fuente: (Campbell, Mummert, & Sukthankar, 2008)

Del proyecto (Campbell, Mummert, & Sukthankar, 2008), si las abejas volaban muy cerca de la cámara, podían darse varias falsas detecciones, situación que se observó también en el proyecto anterior, ya que los dos realizan las detecciones asemejando las formas de los objetos en movimiento detectados a elipses, sin embargo, a pesar de contar aún con herramientas rústicas para tratamiento de imágenes y pocas librerías y algoritmos implementados para el reconocimiento, este proyecto estableció muchas pautas que aún se siguen en la detección de abejas mediante imágenes con algoritmos y librerías especializadas como en el caso anterior, fue de los primeros con una implementación de substracción de fondo que es una tendencia amplia, y de los pocos con modelos de movimiento especiales.

### 2.2.4.2.5. Clasificación

El siguiente paso importante en la detección automatizada de especies (insectos, plantas, animales) fue la clasificación por medio de imágenes, este método se implementó como una iniciativa con un enfoque diferente, las imágenes debían ser estáticas, pero en vez de reconocer los objetos basados en las características físicas de los mismos en el momento, se hacía la comparativa contra cientos de imágenes con características identificadas por los desarrolladores como se puede ver en el proyecto (Daoust, 2017), el objeto de este proyecto fue la clasificación de ciertas especies de plantas teniendo un set de imágenes identificadas de las

## Capítulo 2. - Estado del arte

especies deseadas, el algoritmo recibe una imagen a clasificar y, tomando las características en el set de imágenes, se clasifica la imagen de entrada. Más allá de clasificar solamente el objeto en una especie generalizada se puede diferenciar entre subespecies, ya sean plantas, animales o insectos, como se puede ver en el proyecto (Dagher, 2016) enfocado a la clasificación de polinizadores, el autor presenta una solución para clasificar abejas y abejorros mediante un dataset de características identificadas en imágenes de 200X200 (preferible imágenes pequeñas para menor consumo de recursos de procesamiento), en la Figura 2.14 pueden apreciarse las clasificaciones y distinciones que se lograron mediante los sets de datos y la clasificación.



Figura 2.14 Resultados de la clasificación, distinción entre abejas y abejorros Fuente: (Dagher, 2016)

Aunque fue un avance en el campo del cómputo inteligente, los clasificadores no fueron tan ampliamente usados debido al uso tan poco dinámico al que estaban dirigidos, además, el uso de un clasificador en video continuo pierde exactitud y consume mucho procesamiento, por lo que para la solución de en un video continuo se buscaron alternativas como se verá en la siguiente sección.

### 2.2.4.2.6. Aprendizaje de máquinas

Los avances más recientes en la detección de abejas en imágenes, están tomando dos tendencias tecnológicas básicas teniendo como factor común el aprendizaje de máquinas, el primero retoma el enfoque en etiquetas de los proyectos pasados (Wario, Wild, Couvillon, Rojas, & Landgraf, 2015), en este proyecto conocido como

## Capítulo 2. - Estado del arte

BeesBook, se diseñaron etiquetas en blanco y negro, circulares, las cuales contenían un ID binario para decodificar en la detección (ver Figura 2.15Figura 2.15), para esto se desarrolló un software de procesamiento al que llamaron “pipeline”, el cual se organizó en 5 capas:

**Preprocessor (preprocesamiento):** la cual procesa la imagen recibida del dispositivo, a fin de solucionar problemas de brillo, a fin de normalizar las condiciones de iluminación en la siguiente capa, aquí mismo se detecta las celdas del panal, que bajo ciertas condiciones podrían arrojan falsos positivos.

**Tag Localization (localización de etiquetas):** cuya tarea principal es detectar las zonas con las curvas más definidas.

**Ellipse Fitting (Detección de elipses):** esta capa detecta las figuras elípticas en concreto.

**Grid Fitting (obtención del grid):** cada elipse detectado contiene un grid para ser analizado y descompuesto.

**Decoding (Decodificación):** donde se interpretan las posiciones que componen el grid y se obtienen los identificadores de las abejas (ver Figura 2.16).



Figura 2.15 Patrón identificador, abejas etiquetadas en ambientes reales. Fuente: (Wario, Wild, Couvillon, Rojas, & Landgraf, 2015)



Figura 2.16 Detección de un patrón en tiempo real. Fuente: (Wario, Wild, Couvillon, Rojas, & Landgraf, 2015)

Los datos resultantes son almacenados en una base de datos PostgreSQL, y las consultas posteriores se realizan por medio esta, cada detección se almacena como una fila en una tabla.

La precisión que ofrece la detección con este tipo de etiquetas ha ayudado mayormente en proyectos enfocados a la comunicación de las abejas mediante patrones de movimiento, los cuales son llamados ‘danzas’ o ‘bailes’, sin embargo se siguen presentando los mismos inconvenientes que con el modelo de etiquetas anterior, el segundo enfoque empieza aún a tomar fuerza, gracias a la optimización de algoritmos de aprendizaje de máquinas y a capacidades de procesamiento mayores en dispositivos de tamaño muy reducido, se basa en el entrenamiento de modelos con imágenes del mundo real para realizar detección de objetos en fotogramas de un flujo de video constante. Este nuevo enfoque deja de lado el reconocimiento de formas y los métodos de filtrado de imágenes para centrarse en métodos y algoritmos de análisis de imágenes que tienen como base la detección de características plasmadas en redes neuronales mediante un proceso de entrenamiento, en este proyecto se toma como base el tratamiento de las imágenes, pero la misma lógica y conceptos podrían aplicarse para el estudio y detección de distintos comportamientos teniendo como base un modelo entrenado con datos sólidos y no necesariamente imágenes.

## Capítulo 2. - Estado del arte

Hasta el momento el enfoque predominante con este tipo de modelos es la detección, la aplicación del modelo previamente entrenado, para encontrar las características de las que tiene conocimiento, este mismo enfoque se sigue desde los clasificadores, que dependiendo de la imagen que se sometiera al clasificador, podía dar varias clasificaciones de objeto a las que podría pertenecer el objeto en la imagen, esto acorde a su conocimiento.

Uno de los paradigmas más ampliamente tratados en la literatura de la inteligencia artificial son las redes neuronales. Las redes neuronales artificiales (RNA) intenta imitar la naturaleza de las neuronas del cerebro humano (García, 2013). Lo que básicamente ocurre en una neurona biológica es lo siguiente: la neurona es estimulada o excitada a través de sus entradas (inputs) y cuando se alcanza un cierto umbral, la neurona se dispara o activa, pasando una señal hacia el axón. Posteriores investigaciones condujeron al descubrimiento de que estos procesos son el resultado de eventos electroquímicos (Matich, 2001).

Como ya se sabe, el pensamiento tiene lugar en el cerebro, que consta de billones de neuronas interconectadas. Así, el secreto de la “inteligencia” -sin importar como se defina- se sitúa dentro de estas neuronas interconectadas y de su interacción. (Matich, 2001). El modelo básico, para representar una neurona puede observarse en la Figura 2.17.

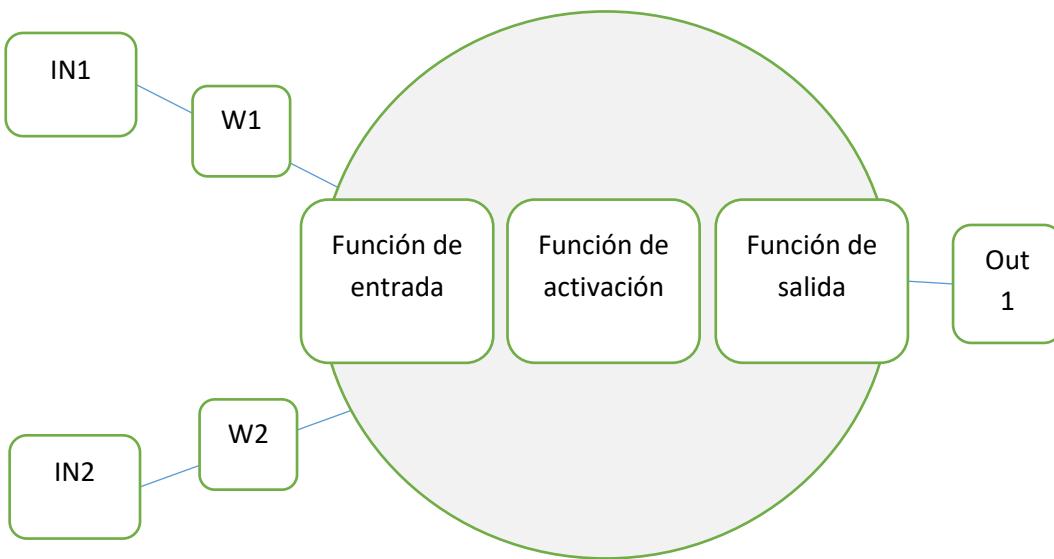


Figura 2.17 Elementos de una neurona básica. Fuente: (Matich, 2001)

Algunas de las funciones de entrada más comúnmente utilizadas y conocidas son:

- 1) Sumatoria de las entradas pesadas: es la suma de todos los valores de entrada a la neurona, multiplicados por sus correspondientes pesos. (Matich, 2001)

$$\sum_j (n_{ij} w_{ij}), \text{ con } j = 1, 2, \dots, n$$

- 2) Producto de las entradas pesadas: es el producto de todos los valores de entrada a la neurona, multiplicados por sus correspondientes pesos. (Matich, 2001)

$$\prod_j (n_{ij} w_{ij}), \text{ con } j = 1, 2, \dots, n$$

- 3) Máximo de las entradas pesadas: solamente toma en consideración el valor de entrada más fuerte, previamente multiplicado por su peso correspondiente. (Matich, 2001)

$$\underset{j}{\operatorname{Max}}(n_{ij} w_{ij}) \text{ con } j = 1, 2, \dots, n$$

Una neurona biológica puede estar activa (excitada) o inactiva (no excitada); es decir, que tiene un “estado de activación”. Las neuronas artificiales también tienen diferentes estados de activación; algunas de ellas solamente dos, al igual que las biológicas, pero otras pueden tomar cualquier valor dentro de un conjunto determinado.

La función activación calcula el estado de actividad de una neurona; transformando la entrada global (menos el umbral,  $\Theta_i$ ) en un valor (estado) de activación, cuyo rango normalmente va de (0 a 1) o de (-1 a 1). Esto es así, porque una neurona puede estar totalmente inactiva (0 o -1) o activa (1). (Matich, 2001)

La función de salida proporciona el valor de salida de la neurona, en base al estado de activación de la neurona (Barrera), por ende, la función de salida determina qué valor se transfiere a las neuronas vinculadas. Si la función de activación está por debajo de un umbral determinado, ninguna salida se pasa a la neurona subsiguiente (Matich, 2001).

La distribución de neuronas dentro de la red se realiza formando niveles o capas, con un número determinado de dichas neuronas en cada una de ellas. A partir de su situación dentro de la red, se pueden distinguir tres tipos de capas: - De entrada: es la capa que recibe directamente la información proveniente de las fuentes externas de la red. - Ocultas: son internas a la red y no tienen contacto directo con el entorno exterior. El número de niveles ocultos puede estar entre cero y un número elevado. Las neuronas de las capas ocultas pueden estar interconectadas de distintas maneras, lo que determina, junto con su número, las distintas topologías de redes neuronales. - De salidas: transfieren información de la red hacia el exterior. (Matich, 2001) Una red neuronal simple puede representarse tal como se puede ver en la Figura 2.18.

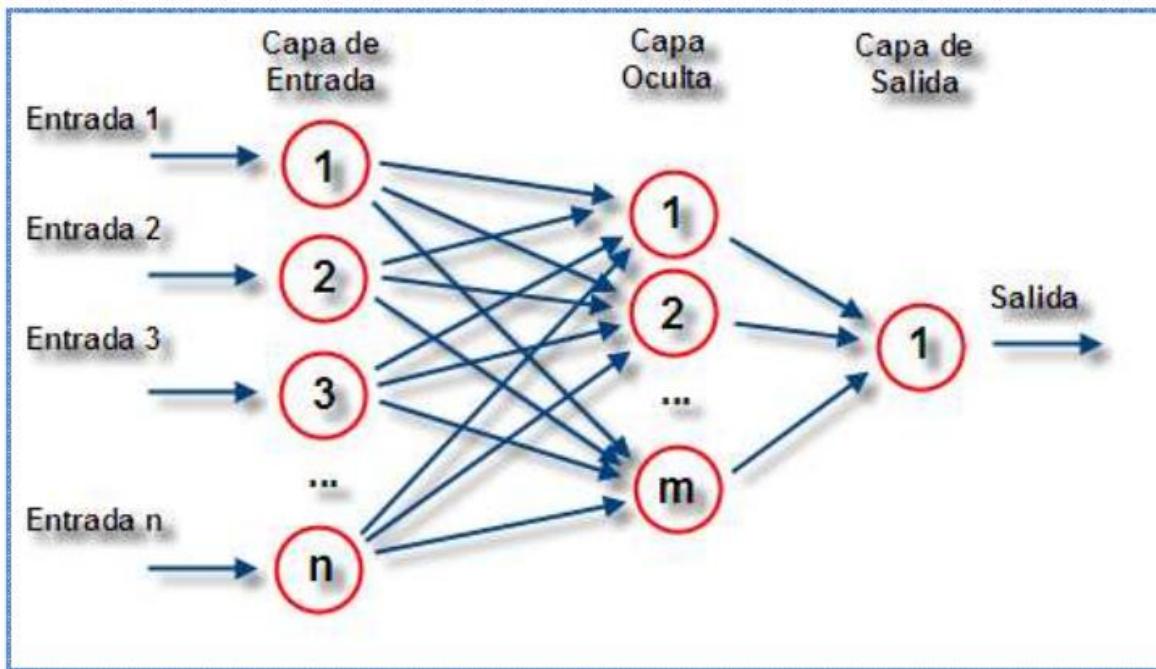


Figura 2.18 Red neuronal simple con capas. Fuente: (García, 2013)

Una red neuronal debe aprender a calcular la salida correcta para cada constelación (arreglo o vector) de entrada en el conjunto de ejemplos. Este proceso de aprendizaje se denomina: proceso de entrenamiento o acondicionamiento. (Matich, 2001) En este proceso se usa un conjunto de datos o patrones de entrenamiento para determinar los pesos (parámetros) que definen el modelo de red neuronal. Se calculan de manera iterativa, de acuerdo con los valores de entrenamiento, con el objeto de minimizar el error cometido entre la salida obtenida por la red neuronal y la salida deseada (Introducción a las redes neuronales aplicadas).

Hay dos métodos de aprendizaje importantes que pueden distinguirse:

- a- Aprendizaje supervisado.
- b- Aprendizaje no supervisado.

El aprendizaje supervisado se caracteriza porque el proceso de aprendizaje se realiza mediante un entrenamiento controlado por un agente externo (supervisor, maestro) que determina la respuesta que debería generar la red a partir de una entrada determinada (Matich, 2001).

## Capítulo 2. - Estado del arte

Las redes con aprendizaje no supervisado (también conocido como auto supervisado) no requieren influencia externa para ajustar los pesos de las conexiones entre sus neuronas. La red no recibe ninguna información por parte del entorno que le indique si la salida generada en respuesta a una determinada entrada es o no correcta. Estas redes deben encontrar las características, regularidades, correlaciones o categorías que se puedan establecer entre los datos que se presenten en su entrada. Existen varias posibilidades en cuanto a la interpretación de la salida de estas redes, que dependen de su estructura y del algoritmo de aprendizaje empleado (Matich, 2001).

La detección de objetos en imágenes mediante redes neuronales ha sido el siguiente paso en los sistemas observación biológica, el proyecto (Weiguang & Graham, 2016), se habla del uso de redes neuronales como medio de detección de plagas en imágenes, su proceso de aprendizaje supervisado se dividió en tres fases para las que se utilizaron un total de 177 imágenes, de las cuales, 110 utilizadas en la primer fase o entrenamiento, 27 en la fase de validación y 40 para realizar pruebas al modelo obtenido. Al igual que algunos proyectos de tratamiento de imágenes anteriores, este proyecto utilizó un ‘Pipeline’ para la detección, es decir, el proceso mediante ciertos pasos secuenciales, mediante el enfoque de ‘ventana deslizante’ donde se aplica un clasificador entrenado, en este caso el ‘Convolutional neural networks’ con ConvNet (LeCun, Bottou, Bengio, & Haffner, 1998) a áreas de las imágenes definidas por una pirámide de escalas, estas áreas de detección están regular y densamente distribuidas sobre la imagen, la salida de estas es un escalar  $p \in [0, 1]$ , que representa la posibilidad de que esa área en particular contenga un objeto buscado, en este caso la plaga analizada era la polilla de manzana.

La primera etapa del ‘pipeline’ en (LeCun, Bottou, Bengio, & Haffner, 1998) son las capas convolucionales, que aplican un banco de filtros lineares y elementos de conocimiento no lineares a sus ‘mapas de características’ (feature maps) de entrada, transformándolos en mapas de características diferentes. Aplicando capas convolucionales varias veces, podemos extraer Representaciones cada vez más representativas de alto nivel de la entrada, al mismo tiempo que preservan su

## Capítulo 2. - Estado del arte

relación espacial. En la primera capa, los mapas de funciones de entrada son simplemente los canales de la entrada. En capas posteriores, estas representan transformaciones más abstractas de la imagen. Una capa convolucional es un caso especial de una capa completamente conectada, donde las conexiones locales tienen valores distintos de cero, y los pesos están vinculados en todas las ubicaciones.

Cada capa convolucional va seguida de una capa de acumulación máxima (max-pooling layer). Esta capa aplica operaciones de agrupamiento local a sus mapas de características de entrada, preservando solo el valor máximo dentro de un campo receptivo local y descartando todos los demás valores. Es similar a una capa convolucional en el sentido de que ambos operan localmente. Aplicando capas de máximo aprovechamiento tiene 2 beneficios principales:

- 1) reducir el número de parámetros libres.
- 2) introducir una pequeña cantidad de invariancia traslacional en la red

Las dos últimas capas en la ConvNet están completamente conectadas (fully connected layer). Este es el tipo de capas que se encuentran en las redes neuronales ‘feedforward’ estándar. La primera capa aplana (vectoriza) todos los mapas de características después de la última capa de acumulación máxima (max-pooling layer), tratando este vector unidimensional como una representación característica de la imagen completa. La segunda capa completamente conectada se parametriza como un clasificador lineal.

Completados estos pasos, se obtienen las probabilidades asociadas a cada cuadro, pero si simplemente se mostrara la detección en cada punto, se tendrían múltiples marcas sobreuestas unas con otras. Este problema es comúnmente resuelto usando Supresión de no máximos (non-maximum suppression, NMS) que retiene aquellas probabilidades de detección locales más altas, este proyecto en particular, llevó a cabo esta técnica de la siguiente manera: primeramente se hizo un ordenamiento de las detecciones basado en las probabilidades de presencia que

## Capítulo 2. - Estado del arte

mostraban, entonces de mayor a menor probabilidades, se detectan las área con mayor acumulación de posibilidades y se escogen dichas áreas (ver Figura 2.19).

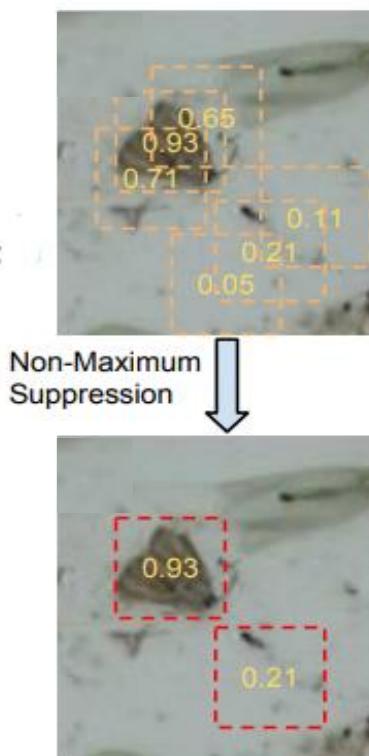


Figura 2.19 Supresión No Máxima de áreas con probabilidad de detección. Fuente: (Weiguang & Graham, 2016)

Entre los trabajos más recientes de aprendizaje de máquina y más aun de aprendizaje profundo se encuentra el proyecto (Astha, 2018) que presenta una alternativa para la detección de abejas en un flujo de video real. A diferencia de los trabajos analizados con anterioridad, no se enfoca completamente al conteo masivo, sino al análisis de tráfico en forrajeras como medio de monitoreo de monitoreo de los panales.

La alternativa presente se enfoca en el entrenamiento extensivo de redes neuronales complejas con varias capas, los datos de entrada son tomados de imágenes previamente recolectadas en varios panales que se han seleccionado para el proyecto. Los datos que el autor proporciona mencionan que el total de imágenes utilizadas para el entrenamiento fue de 50000, con 90% para entrenamiento y 10% para pruebas. Los experimentos realizados con las imágenes

## Capítulo 2. - Estado del arte

se hicieron tomando 100 “epochs” para el entrenamiento. Se evaluó y comparó con los métodos MLP y ConvNet,

MLP, uno de los métodos más tradicionales, es una red simple de neuronas llamadas “perceptrons”, los cuales computan una salida simple de múltiples entradas con valores reales formando una combinación lineal de acuerdo a sus pesos de entrada y dando una salida a través de una función no lineal de activación. Se compone de las tres capas expuestas anteriormente y cada elemento de la capa anterior está conectado a cada elemento de la capa siguiente.

ConvNet, Convolutional Neural Network, deriva del estándar de redes neuronales MLP, optimizado para problemas de reconocimiento de patrones bidimensionales. A diferencia de las redes MLP, estas se caracterizan por la estructura especial de sus redes, la cual se basa en convoluciones y sub-muestreo de capas. En estas redes las neuronas pueden detectar características complejas en objetos reales y llevar a cabo la clasificación.

En (Astha, 2018) se expone una mejora a la clasificación, permitiendo en tiempo real, obtener los reconocimientos gracias al entrenamiento de redes neuronales, pero así también de algoritmos de tratamiento de imágenes simples como es la detección de movimiento. El flujo de información del algoritmo se centra en dos puntos secuenciales de importancia:

- Detección de movimiento en flujo de video. Al detectar movimiento en los cuadros dentro del flujo de video, el área determinada se corta y redimensiona (de ser necesario) en imágenes de tensor de tercera orden 32X32 y tres canales de color (tamaño que mejora el rendimiento del software y la optimización de los recursos disponibles), estas imágenes son enviadas al segundo punto.
- La imagen tomada del área donde se detectó movimiento es enviada al clasificador, en este caso la red neuronal entrenada para clasificarla en dos categorías: “Bee” y “NoBee”.

## Capítulo 2. - Estado del arte

Como se ha podido observar, la computación y los sistemas inteligentes, han sido usados ampliamente en la observación biológica. Los métodos tradicionales de monitoreo mediante sensores y más recientemente la visión por computadora ha ayudado a automatizar tareas simples de manera estructurada y con grandes opciones de escalabilidad. La Tabla 1, crea un punto de comparación visual entre varios trabajos destacados de monitoreo mediante varios sensores integrados en un mismo sistema. Puede observarse como variables como la temperatura, la humedad y el peso, son datos importantes para la comprensión y observación de las colonias, sin embargo, un caso común de observación han sido las variables aisladas, pequeños conjuntos de sensores incluidos en el mismo sistema, dependiendo del enfoque que los investigadores quieran aplicar. No es común encontrar un sistema que incluya la observación desde el nivel individual, pasando por el nivel colmena hasta llegar al nivel apiario. El siguiente capítulo, abordará la explicación acerca de cómo el presente proyecto, intenta cubrir esos tres niveles, en un sistema integrado y haciendo uso de los paradigmas más recientes en el campo de la computación como son el internet de las cosas y el aprendizaje de máquinas.

## Capítulo 2. - Estado del arte

| Referencia  | Temp. Ext. | Temp. Int. | Luz | Sonido | Vibr. | Accel. | Puvl. | Anem. | Peso | Hum. | Detección por imágenes | Conteo | Admón. | GSM | B.D. local | B.D. centralizada | Proyecto libre | RFID |
|---|------------|------------|-----|--------|-------|--------|-------|-------|------|------|------------------------|--------|--------|-----|------------|-------------------|----------------|------|
| LanaseBeeMonitor (Propuesta)  |            |            |     |        |       |        |       |       |      |      |                        |        |        |     |            |                   |                |      |
| (Jurijs, Aleksejs, & Egils, 2009)   |            |            |     |        |       |        |       |       |      |      |                        |        |        |     |            |                   |                |      |
| (Estados Unidos Patente nº 2,806,082, 1957)   |            |            |     |        |       |        |       |       |      |      |                        |        |        |     |            |                   |                |      |
| (Tenczar, Lutz, Bikyath, Goldenfeld, & Robinson, 2014)  |            |            |     |        |       |        |       |       |      |      |                        |        |        |     |            |                   |                |      |
| (Meikle, Holst, Mercadier, Derouané, & James, 2006)   |            |            |     |        |       |        |       |       |      |      |                        |        |        |     |            |                   |                |      |
| (Human, Nicolson, & Dietemann, Do honeybees, <i>Apis mellifera scutellata</i> , regulate humidity in their nest?, 2006) |            |            |     |        |       |        |       |       |      |      |                        |        |        |     |            |                   |                |      |
| (Bencsik, y otros, 2011)  |            |            |     |        |       |        |       |       |      |      |                        |        |        |     |            |                   |                |      |
| (Ferrari, Silva, Guarino, & Berckmans, 2008)  |            |            |     |        |       |        |       |       |      |      |                        |        |        |     |            |                   |                |      |
| (Schneider, Tautz, Grünwald, & Fuchs, 2012)   |            |            |     |        |       |        |       |       |      |      |                        |        |        |     |            |                   |                |      |

## Capítulo 2. - Estado del arte

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|
| (Chiu, En-Cheng, Joe-Air, & Ta-Te, 2012)                 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| (Octavio, Márquez, Ángela, Sebastian, & Giovanni, 2012)  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| (Jason, Lily, & Rahul, 2017)                             |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| (Gonzalez & Beck, 2017)                                  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| (O'Keefe Electronics Inc., 2016)                         |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| (Arnia, 2017)  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| (BeeWatch - Biene & Natur GmbH - Beekeeping Needs, 2014) |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| (Gang Jun Tu, 2016)                                      |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| (Estados Unidos Patente nº 6,910,941, 2004)              |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

Tabla 1 Comparativa entre trabajos relevantes

# Capítulo 3. Arquitectura Propuesta

El capítulo presente se enfocará en mostrar las variables estudiadas y el flujo de información, dentro de los protocolos y esquemas en los que se desarrolló el proyecto, así como los detalles de la implementación del mismo, se dará un amplio panorama de la funcionalidad del proyecto, mención de la tecnología utilizada, ventajas y desventajas de la misma, el lector podrá entender a profundidad la implementación del sistema y el funcionamiento interno que se lleva a cabo en los distintos módulos y capas de la arquitectura propuesta por el autor. Se comienza el capítulo con una revisión general de la arquitectura, las partes que la componen y los esquemas en que se basa. Después una revisión del flujo de información, comportamiento e implementación de cada una de las capas que conforman la implementación.

La tecnología de internet de las cosas (internet of things, IoT), ha revolucionado la industria en los últimos años, añadiendo valor a los productos tradicionales mediante mejoras informáticas que permiten, en su mayoría, el monitoreo y manipulación de sistemas informáticos remotos. Este paradigma, más que centrarse en el internet, se centra en las cosas, y en cómo estas pueden comunicarse, entre ellas y con los humanos.

De acuerdo con (PTC University, 2017), los proyectos de IoT, pueden dividirse en tres componentes, físicos, circuitos, carcasa, etc; componentes inteligentes, sensores, algoritmos de análisis, búsquedas, etc; y componentes de conectividad, bluetooth, Wifi, etc. Los cuales se discutirán para el presente proyecto en este capítulo.

Los proyectos de IoT, según (McKinsey Global Institute, 2017), en los últimos años se han focalizado en 9 áreas de desarrollo importante, entre las que desatacan, fábricas, oficinas y ciudades. El presente proyecto se encuentra en la categoría “sitio de trabajo” al ser un proyecto desarrollado a la medida para un trabajo en sitio específico fuera de la industria y enfocado a la investigación.

## Capítulo 3. - Arquitectura Propuesta

La arquitectura de recolección de datos que atañe al presente trabajo se basa en los 4 primeros niveles de la arquitectura de IoT presentada por (Khan, Khan, Zaheer, & Khan, 2012), cabe destacar que el sistema en general cuenta con módulos extra de la última y la penúltima capa, pero ya que no son el objeto del presente trabajo, solo se mencionarán dentro de la explicación de la capas de esta primera arquitectura general (ver Figura 3.1).

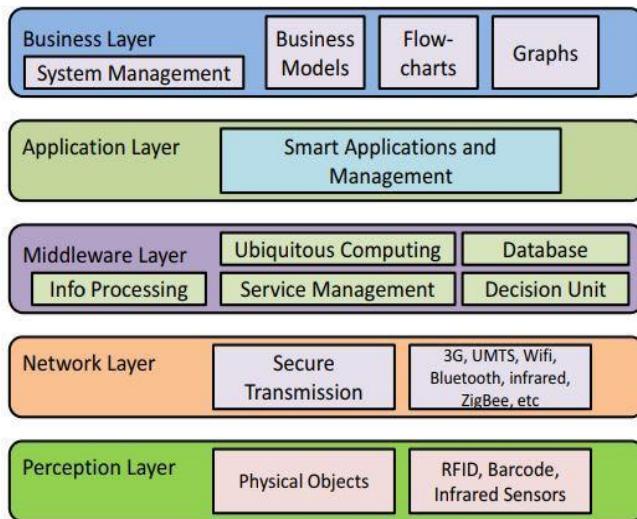


Figura 3.1 Arquitectura de IoT, Fuente: (Khan, Khan, Zaheer, & Khan, 2012)

- **The Perception Layer (Capa de percepción):** es como la piel facial y los cinco órganos de los sentidos de IoT (Miao, Ting-lie, Fei-Yang, Jing, & Hui-Ying, 2010), en el caso de este proyecto los diferentes sensores presentados en la subsección 1 “Campo”.
- **Network Layer (Capa de redes y comunicaciones):** Esta capa transfiere de forma segura la información de los dispositivos sensores al sistema de procesamiento (Miao, Ting-lie, Fei-Yang, Jing, & Hui-Ying, 2010). En la arquitectura del presente proyecto se relaciona con “Comunicación Remota” en la subsección 2.
- **Middleware Layer (Capa de programa intermedio):** Esta capa es responsable de la gestión del servicio y del enlace a la base de datos. Recibe

## Capítulo 3. - Arquitectura Propuesta

la información de la capa de red y la almacena en la base de datos (Miao, Ting-lie, Fei-Yang, Jing, & Hui-Ying, 2010). Análogamente en el presente proyecto se presenta la subsección 3 “Centro de datos”, que se desarrolla más adelante.

- **Application Layer (Capa de aplicación):** Esta capa proporciona una administración global de la aplicación basada en la información de los objetos procesados en la capa de Middleware (Khan, Khan, Zaheer, & Khan, 2012). El presente sistema cuenta con un sistema de administración de Base de Datos, sin embargo, como no es propósito de este proyecto no se desarrollará más a fondo, también se cuenta con un demonio para auto- configuración de bases de datos remotas y configuración de demonios para garantizar el funcionamiento continuo, estos temas se tocarán más a fondo en el siguiente capítulo.
- **Business Layer (Capa de visualización y negocios):** Construye gráficos, diagramas de flujo etc. basado en los datos recibidos. El sistema también cuenta con un sistema de gráficas por sensores, sin embargo, ya que no es propósito del presente trabajo, no se tocará más a fondo este tema.

La arquitectura propuesta para el proyecto abarca la observación de variables en los 3 grupos de información según (Human, y otros, 2013) que son:

- Nivel apíario (apiary-level)
- Nivel colonia (colony-level)
- Individual (individual bee-related parameters)

(ver Figura 3.2). Esta información se almacena de manera local en la base de datos en los agentes y también es transmitida a través de la red GSM/3G al centro de datos (Subsección 3), donde se interpreta y se almacena debidamente, siguiendo los lineamientos de la base de datos en el servidor, para esto, anteriormente se dieron de alta agentes (dispositivos en campo) en la base de datos, con sus debidos

## Capítulo 3. - Arquitectura Propuesta

datos de localización y tipos de sensores que tienen en funcionamiento, estos temas se tocarán más a fondo en el siguiente capítulo dedicado a la implementación del sistema.

La arquitectura se esquematiza en tres partes:

- Campo (Componentes físicos, inteligentes y de conectividad)
- Comunicación Remota (Componentes de conectividad)
- Centro de Datos (Componentes inteligentes)

Dando un preámbulo, en la primera parte se toman los datos de campo, en esta parte se encuentran los sensores y los agentes en su totalidad, para salir a la red pública y encontrar el servidor se tienen la segunda parte que es la comunicación GSM y el enlace con el centro de datos, una vez encontrado el servidor en la internet este recibe la información desde los apíarios y la almacena en la base de datos centralizada, se describirá a fondo los elementos de cada parte en las siguientes sub-secciones (ver Figura 3.2).

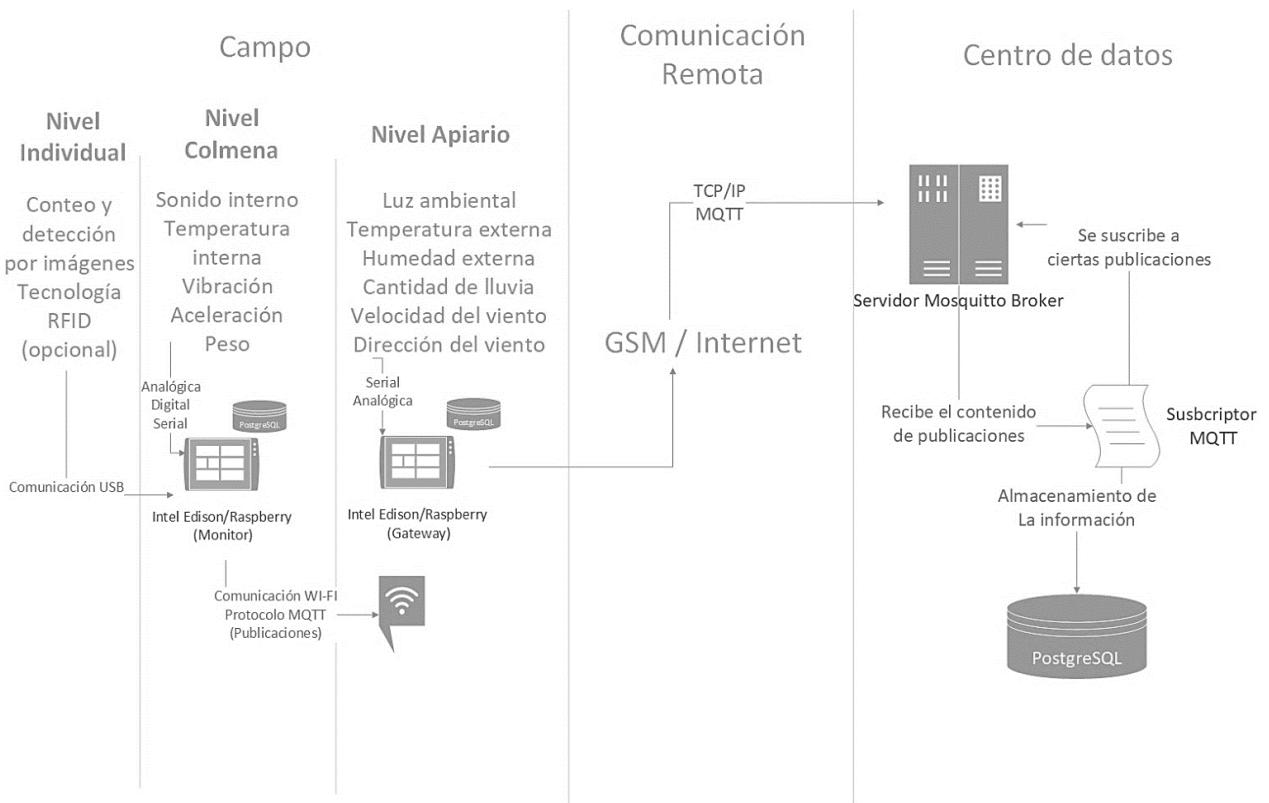


Figura 3.2 Arquitectura propuesta y flujo de la información.

### Capítulo 3. - Arquitectura Propuesta

En general, en el lago de datos de este proyecto, tal como se puede ver en Figura 3.3, se conformará de cuatro componentes principales:

**Datos climáticos:** los cuales son generales para todo el apiario, estos se analizarán a fondo en la subsección

Campo, infraestructura en los apiarios.

**Datos del agente:** datos técnicos específicos del sistema en campo, su localización por coordenadas y por país y ciudad, datos de autodiagnóstico que determinen problemas físicos en los sensores a la hora de intentar obtener información de los mismos, listado de los sensores que se encuentran trabajando en ese agente y las características de cada sensor p. j., observaciones especiales, modelos, puerto en el agente, unidades de medición, etc; por último los chips RFID registrados para cada agente (si cuenta con ellos).

**Datos internos de la colmena:** sensores que se encuentran dentro de las colmenas para la medición de variables interiores, se analizaran más a fondo en la subsección

Campo, infraestructura en los apiarios.

**Datos poblacionales:** un modelo de datos para registrar avistamientos, entradas o salidas del apiario, los métodos y descripción para obtener esta información se tocarán más a fondo en la subsección

Campo, infraestructura en los apiarios.

## Capítulo 3. - Arquitectura Propuesta

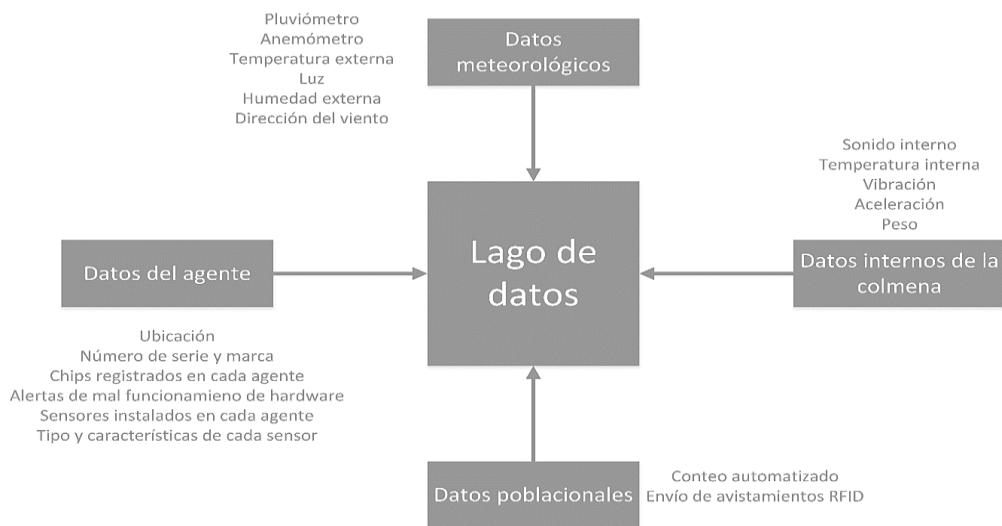


Figura 3.3 Detalles del lago de datos y flujo de información

### 3.1. Campo, infraestructura en los apiarios

La primer y más importante subsección es “campo” (ver Figura 3.21), en esta encontramos todos los equipos e infraestructura que funcionan y recolectan datos desde el apiario, sea en panales (e. g.: sensores) o al aire libre (e. g. estación meteorológica, ver Figura 3.4). Según la clasificación general de los componentes de IoT de (PTC University, 2017), en esta sección se pueden encontrar los tres tipos de componentes empezando por físicos tales como piqueras, cables y circuitos, componentes inteligentes como lo es el conjunto de sensores y componentes de conectividad, wifi y red interna del apiario para conexión con el dispositivo puente. Esta sección se basa en la recolección de datos de los tres niveles en que se divide el ecosistema del apiario según (Human, y otros, 2013), permitiendo integrar en un solo sistema variables que regularmente en otros sistemas u otros proyectos se estudian aisladamente, permitiendo la observación de diferentes tipos de datos tomados en el mismo momento.

1. **Nivel apíario:** el estudio de las variables del apiario, debe incluir monitoreo de variables que afecten por igual a todas las colonias del grupo estudiado, por lo tanto, este consiste en el monitoreo ambiental de la zona mediante estaciones meteorológicas que incluyen: pluviómetro (cantidad de lluvia), anemómetro (velocidad del viento), higrómetro

### Capítulo 3. - Arquitectura Propuesta

(cantidad de humedad relativa), fotómetro (cantidad de luz medida en unidades “lux”) y termómetro externo (temperatura ambiental). El nivel apíario permite tomar lecturas que afectan a todas las poblaciones estudiadas en ese momento en el mismo sitio. Permite establecer un punto de referencia del ambiente en el que se desenvuelve día a día la colonia, e incluso comparar a simple vista que eventos ambientales disparan consecuencias negativas en las poblaciones, como por ejemplo algún evento ambiental prolongado que este menguando la producción de alimento de las colonias. La estación meteorológica de este módulo toma las medidas antes mencionadas a excepción de la cantidad de luz, la humedad relativa y la temperatura externa que es son sensores individuales. Al igual que todos los módulos de esta implementación, este está codificado en lenguaje Python (van Rossum, 2009) por su portabilidad y sencilla implementación. Los dispositivos utilizados para la recolección de datos (como se verá más adelante) son de poca capacidad de procesamiento y memoria, por esto se necesitó un lenguaje interpretado de amplias capacidades como Python, además este cuenta con implementaciones para IoT que en otros lenguajes no están disponibles o son más complicadas de implementar.

La comunicación con las estaciones meteorológicas (ver Figura 3.4Figura 3.4Figura 3.4) se lleva a cabo de manera serial directa. La información recibida de esta se decodifica y se interpreta para obtener los valores deseados.

### Capítulo 3. - Arquitectura Propuesta



Figura 3.4 Estación meteorológica (pluviómetro y anemómetro)

La obtención de lecturas desde el sensor de luz se lleva a cabo de manera analógica, gracias a los puertos de comunicación analógica que poseen los dispositivos y las mediciones de humedad y temperatura externas, se llevan a cabo mediante los puertos digitales con los que se cuenta. La lectura de los datos, en estos dos casos se realizan mediante librerías que se encuentran en repositorios públicos, que vinculan la obtención de señales de bajo nivel con la interpretación y obtención de los datos a alto nivel.

La información recolectada de estos sensores se almacena en una base de datos local en el servidor PostgreSQL (PostgreSQL, 2005), cuya ventaja es ser un SGBD robusto que garantiza la disponibilidad del servidor en mayor medida que otros sistemas libres, además la portabilidad de este y la compatibilidad con los sistemas utilizados en el proyecto y el lenguaje Python lo hicieron una elección de primera mano.

2. **Nivel de la colmena:** este nivel nos permite observar la situación dentro de las poblaciones. Para el monitoreo de variables micro-ambientales se

## Capítulo 3. - Arquitectura Propuesta

utilizan los siguientes sensores dentro de las colmenas: sonómetro (cantidad de sonido en el panal en decibeles), medidor de vibración (unidades de vibración), termómetro interno (temperatura interna en grados centígrados), acelerómetro (movimiento del panal en ejes x, y, y z) y balanza (sensor de peso). En la observación de las poblaciones de abejas, puede decirse, que esta es la sección más importante de estudio, la comprensión de la población individual en general. Aquí podemos encontrar variables de vital importancia para la producción y sobrevivencia de las abejas como son la temperatura interna y el peso (índice de producción más relevante). Mediante una observación conjunta podemos observar relaciones a simple vista entre las variables del primer nivel y las de este nivel.

Las lecturas de todas las variables a excepción de la aceleración y la balanza se llevan a cabo de manera analógica, en los dispositivos de entrada que poseen los sistemas embebidos. Los datos de la balanza, al igual que los de la estación meteorológica, se obtienen de manera serial, mediante códigos previamente establecidos; estos a su vez también deben de ser interpretados para su posterior tratamiento. Los datos obtenidos de este nivel y los otros dos niveles (apiario e individual) se almacenan en una base de datos dentro del servidor PostgreSQL, en particular, las lecturas de este nivel y las del nivel apiario, se almacenan bajo la definición de “observaciones”, y cada observación posee un identificador único del dispositivo en el que está trabajando, un identificador único del sensor del que proviene, la observación en sí y los datos de la fecha y hora en que se tomó la observación. Como se verá más adelante, estos campos se comparten y son de vital importancia para el tercer nivel de la arquitectura (ver Figura 3.23).

3. **Nivel individual:** este se centra en el estudio de cada individuo dentro de la población, o de los individuos de una muestra representativa de la

### Capítulo 3. - Arquitectura Propuesta

misma. Busca la observación y monitoreo del comportamiento individual más allá del estudio de variables ambientales. El estudio de esta capa puede darnos una idea de los comportamientos más comunes de los individuos de la población, para el estudio de variables individuales se utiliza: reconocimiento de imágenes para el conteo masivo de individuos, lo cual se enfoca a un censado automatizado en la entrada de los panales, y microsensores para el monitoreo de muestras de abejas representativas, de las cuales se busca un entendimiento más profundo de sus horarios de recolección, longevidad, avistamiento en entrada, etc.

El monitoreo con micro-sensores se lleva a cabo mediante tecnología RFID (Schneider, Tautz, Grünewald, & Fuchs, 2012), que tiene como objetivo la identificación individual de algunas abejas que generalmente provienen de muestras seleccionadas. Los micro-sensores son colocados en la parte trasera de las abejas, siendo lo menos invasivos posibles, para esto ya se han dado de alta los identificadores de los micro-sensores y relacionado con ciertas abejas a las que pertenecen en la base de datos, al pasar estas por las entradas o lugares estratégicos donde se coloquen los sensores RFID (generalmente “piqueras” extensiones en las entradas con el fin de colocar los dispositivos RFID de manera que las abejas pasen encima de ellos, ver Figura 3.5), estos detectan los micro-sensores y se pueden obtener todos los datos que se tengan dados de alta en la base de datos hasta el momento de la abeja avistada.

### Capítulo 3. - Arquitectura Propuesta



Figura 3.5 Piquera adaptada en entrada de un panal. Fuente: (Engelke, Marendy, Nguyen, De Souza Junior, & Williams, 2014)

El conteo generalizado, se llevó a cabo mediante el framework Tensorflow (Abadi, y otros, 2015), a través el método de aprendizaje supervisado. El entrenamiento se llevó a cabo en un equipo con un GPU integrado, siendo esta la tecnología recomendada para entrenamientos de modelos de aprendizaje de máquina como veremos más adelante, el procesamiento en CPU también es un recurso adecuado para este tipo de entrenamientos, pero es más tardado y puede afectar la precisión del modelo.

Primeramente, el entrenamiento necesita todo un entorno donde poder llevarse a cabo, desde librerías nativas y API's en Python hasta la misma información de la que aprenderá el modelo. Tensorflow cuenta con un API que permite comunicar el código Python con toda la implementación nativa en lenguaje C. Ya instalada esta librería se pueden instalar los paquetes que ofrece el proyecto de tensorflow para el entrenamiento asistido y que se encuentran en sus repositorios.

La idea general del presente proyecto, más allá de un clasificador o de una implementación que requiera etiquetas o identificar el movimiento mediante el cambio de pixeles, es detectar abejas en un flujo de video constante mediante un grafo de inferencia, utilizando aprendizaje de máquinas para identificar las áreas de la imagen donde se encuentran las abejas, el uso del aprendizaje automático permite no solo la

### Capítulo 3. - Arquitectura Propuesta

detección, sino también saber que lo que se está detectando sea lo adecuado. Para llegar a un modelo aceptable que permita este reconocimiento, es necesario contar con un conjunto de datos que, según el propio aprendizaje humano a través del tiempo, son correctos. En este caso, la identificación se hizo dentro de cuadros de video, por lo que el conjunto de datos debe ser proveniente de imágenes, de las cuales el modelo aprenderá con ayuda humana, extrayendo características que van desde pixeles hasta formas definidas y creará su propia idea del objeto en cuestión, en este caso, abejas.

La etapa del etiquetado de las imágenes (en el entrenamiento supervisado) puede ser una de las más largas, ya que debe transmitirse el conocimiento humano a la máquina de manera que se lleve a cabo una comunicación fácil para el humano y comprensible para la máquina. En esta etapa, quien entrena, debe conseguir grandes cantidades de información (en este caso imágenes) e identificar las características o patrones de los que se desea la máquina aprenda. En el presente proyecto se etiquetaron un total de 11000 imágenes dando como resultado un total de alrededor de 14200 registros en el archivo de conjunto de datos. Para compilar estos registros se debe etiquetar la zona donde se localiza el objeto en cuestión, obteniendo xmin, ymin, xmax, ymax, a fin de cuadrar el objeto en la imagen, y escoger la categoría a la que pertenece dicho objeto, en el caso presente solo una categoría denominada “bee”. Este etiquetado se recomienda mayormente en imágenes donde el objeto a detectar no es el centro de atención en la imagen. Las abejas al ser entidades pequeñas (en relación al ambiente) y que generalmente se encuentran en agrupadas en la colmena, pueden no pasar como una característica importante de la imagen en la automatización del método, por lo tanto, es necesario aplicar la técnica denominada “fine-tuning” en el momento de escoger la información que entrará a los registros para comenzar el entrenamiento. Teniendo estos datos se generan archivos propios del framework

### Capítulo 3. - Arquitectura Propuesta

llamados tfrecords, que contienen toda la información del conjunto de datos en términos comprensibles para el entrenamiento.

Generado el conjunto de datos para el aprendizaje, fue llevada a cabo la configuración del entrenamiento (hiperparámetros), donde se establece el número de clases a identificar, los ángulos y resoluciones de las imágenes, en este punto se define uno de los parámetros más importantes, el “batch size”, que es el número de muestras que se manejan al mismo tiempo y que deben revisarse antes de hacer cambios en el modelo internamente, esto influye en la precisión y tiempo de entrenamiento del modelos, el presente trabajo utilizó un “batch size” de 10 muestras, entrenando con un total de 42000 epochs. Un batch size muy bajo puede llevar a la imprecisión del modelo debido a los frecuentes reajustes en los pesos de las primeras capas, y un batch size demasiado grande puede incrementar la demanda de procesamiento y memoria. Un batch size de 10 – 20 es uno de los más recomendados por la comunidad desarrolladora.

El proceso de entrenamiento tiene como fin generar una Red Neuronal Convolutacional (Convolutacional Neural Network), que como se vio en el capítulo anterior, puede tener varias capas ya que es una aplicación de aprendizaje profundo. La estructura de la red neuronal se caracteriza en tres capas: capa de entrada (input layer), capa oculta (hidden layer) y capa de salida (output layer) (ver Figura 2.18). La red neuronal convolutacional de igual forma se divide en estas capas. Con el propósito de entender mejor el proceso de entrenamiento, a continuación, se explica el funcionamiento de la capa oculta, ya que en esta se lleva a cabo como tal el aprendizaje.

Tal como se muestra en la Figura 2.18, las subdivisiones de la capa oculta son el núcleo del funcionamiento de una red neuronal convolutacional, CNN a partir de ahora. Al momento de iniciar el proceso, la primera y más importante capa se denomina convolución puesto que

## Capítulo 3. - Arquitectura Propuesta

esta etapa lleva a cabo esta operación. La convolución se puede ver como una forma de multiplicación de matrices, cuyo fundamento matemático se desarrolla en vectores mediante la expresión en Ecuación 3.1, esta expresión sirve como fundamento para la forma matricial de la expresión que se puede apreciar en la Ecuación 3.2. Las matrices que entran en estas operaciones de convolución son, la imagen de entrada (la imagen con las características que queremos aprenda el modelo) y los kernels o filtros, que son matrices que al principio de la ejecución se inicializan con valores aleatorios, en estos reside en si el aprendizaje como se verá a continuación.

$$y[n] = \sum_{k=-\infty}^{\infty} x[k] \cdot h[n-k]$$

Ecuación 3.1 Expresión matemática para convolución en vectores (IBM Corporation, 2016)

$$I' = \sum_{u,v} I(x-u, y-v)g(u, v)$$

Ecuación 3.2 Expresión de la convolución matricial (IBM Corporation, 2016)

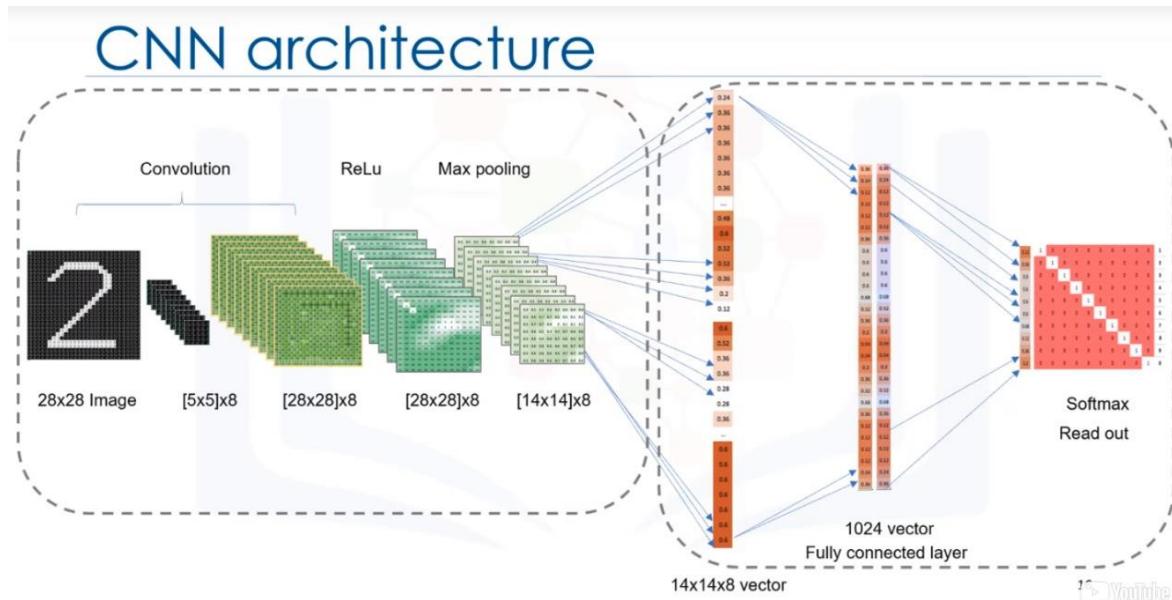


Figura 3.6 Capas comunes en una red neuronal convolucional (IBM Corporation, 2016)

## Capítulo 3. - Arquitectura Propuesta

Como ya se dijo, un kernel es una matriz que entrará en convolución con la matriz que representa la imagen de entrada. Estos kernels pueden verse también como filtros, pueden modificar la imagen según se requiera. Como se puede observar en la Figura 3.6, se toma una imagen de resolución 28x28 blanco y negro, se le aplican 8 kernels de resolución 5x5 que dan como resultado 8 imágenes de 28x28. El objetivo de la primera parte de la arquitectura es reconstruir la información tan fidedignamente como sea posible, en una arquitectura real, la primera parte se suele repetir varias veces las capas tanto de convolución, ReLu y max-pooling para lograr su objetivo, una aproximación más real a una arquitectura puede apreciarse en la Figura 3.7, cuando se analiza el contenido de un batch, se actualizan los pesos de los kernels (valores de las matrices) de acuerdo a los valores que se sabe son reales, esto a través de cada iteración con el fin de generar pesos que puedan llevar a cabo predicciones cada vez más acertadas. El proceso de mejora se lleva a cabo en los kernels, las capas ReLu y Max-pooling se utilizan para hacer más eficiente y menos demandante computacionalmente todo el proceso.

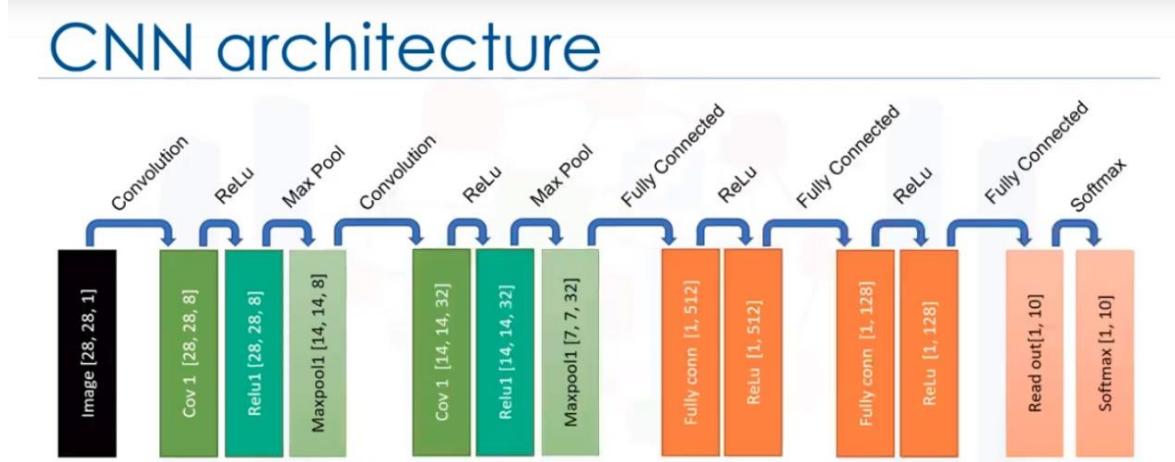


Figura 3.7 Aproximación a una arquitectura CNN real donde se repiten las capas. (IBM Corporation, 2016)

La tecnología más adecuada para este tipo de entrenamientos es el GPU. Como acabamos de explicar, la convolución (y de igual forma todos los procesos de una red neuronal) requieren operaciones matriciales como producto punto, que pueden descomponerse en multiplicaciones y sumas simples entre las filas y columnas del kernel que se esté aplicando y un subconjunto de puntos de la imagen

### Capítulo 3. - Arquitectura Propuesta

de entrada. En este punto la complejidad de las operaciones no es un inconveniente, el problema se debe a la gran cantidad de operaciones simples que se llevan a cabo para generar la salida de cada capa, por ejemplo, una red neuronal con 16 capas, posee alrededor de 140 millones de valores matriciales que se utilizan en el proceso recursivo del aprendizaje en la red neuronal, para cada muestra de entrada. Un CPU toma una operación matricial y realiza las operaciones de manera serial, una tras otra y dependiendo del número de núcleos con los que cuente, puede paralelizar el proceso, el inconveniente es que los CPU generalmente tienen pocos núcleos. Un GPU puede realizar paralelización en estas operaciones ya que generalmente cuentan con cientos o miles de núcleos, un nivel de paralelización como el de un GPU reduce drásticamente el tiempo de entrenamiento de un modelo generado mediante aprendizaje de máquinas, yendo de días o semanas en un CPU a horas en un GPU, en la Figura 3.89 se puede apreciar el manejo de una pequeña cantidad de información (datos generados con un rango) contenida en dos matrices, este fue un ejemplo comparativo del tiempo que se toma la operación producto punto con un CPU y un GPU, con matrices de tamaño considerable (la altura de los primeros valores es debido a la comunicación del GPU con el sistema, la cual debe realizarse mediante el CPU). Otra característica importante del GPU es que está diseñado para manejar grandes volúmenes de datos, así puede manipular de una manera más eficiente los grandes bancos de datos con los que se entrena estos modelos.

### Capítulo 3. - Arquitectura Propuesta

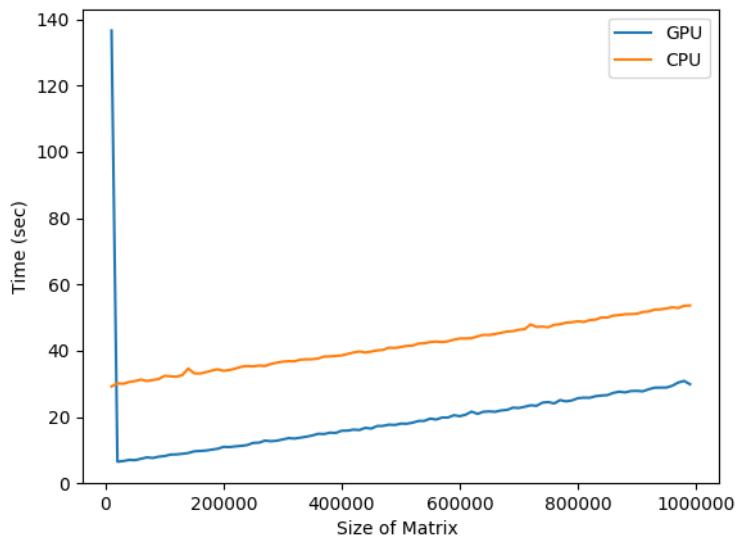


Figura 3.8 Comparativa en milisegundos entre GPU y CPU

En medida del tiempo transcurrido, los pesos se van ajustando de acuerdo a los valores de entrada y los resultados de salida, así como la perdida de información se va haciendo cada vez más pequeña: La comunidad desarrolladora, aconseja, generalmente, que el entrenamiento se dé por terminado al alcanzar una exactitud por encima de los 95%, así como una pérdida de información menor al 2%. Con ayuda de la herramienta “Tensorboard” dentro de la librería Tensorflow, se pueden determinar los niveles de exactitud y perdida de información al alcanzar el número determinado de epochs. De igual manera, se aconseja el uso de tecnología GPU para acelerar el proceso de entrenamiento. Para la obtención del modelo del presente trabajo, se realizaron entrenamientos con ambos métodos, dejando el procesamiento en GPU muy por detrás del procesamiento en CPU, alcanzando el entrenamiento por CPU las 3 semanas a diferencia del entrenamiento en GPU que apenas superó las 4 horas.

Una vez alcanzados los parámetros óptimos, el entrenamiento se dio por terminado, porque lo que se debe “congelar el grafo” en el punto donde terminó, esto para obtener un binario interpretable como modelo de conocimiento por la arquitectura de la librería Tensorflow. Este es el modelado de las características con las que se

### Capítulo 3. - Arquitectura Propuesta

entrenó el modelo representadas mediante CNNs que es una de las estructuras más recomendadas para detección y ubicación de objetos dentro de imágenes.

Tensorboard, da la oportunidad de generar una representación gráfica del grafo generado en el entrenamiento, por razones de complejidad del gráfico completo, no es incluirá en este apartado, sin embargo, se explicarán los elementos más importantes del mismo a continuación.

En primera instancia se inicia el proceso de inicialización, como medida estándar, las matrices de pesos, tendencias y kernels deben ser creadas e inicializarse con números aleatorios como se aprecia en la Figura 3.90, los valores que asigne esta inicialización serán actualizados a través de todo el proceso de aprendizaje, en la primer fase de “forward” de la información en el grafo, se aplicaran tal como se asignaron aleatoriamente, posteriormente en la fase “backward” se actualizarán de acuerdo al error obtenido en la primera fase y comenzará así el proceso de entrenamiento iterativo.

## Capítulo 3. - Arquitectura Propuesta

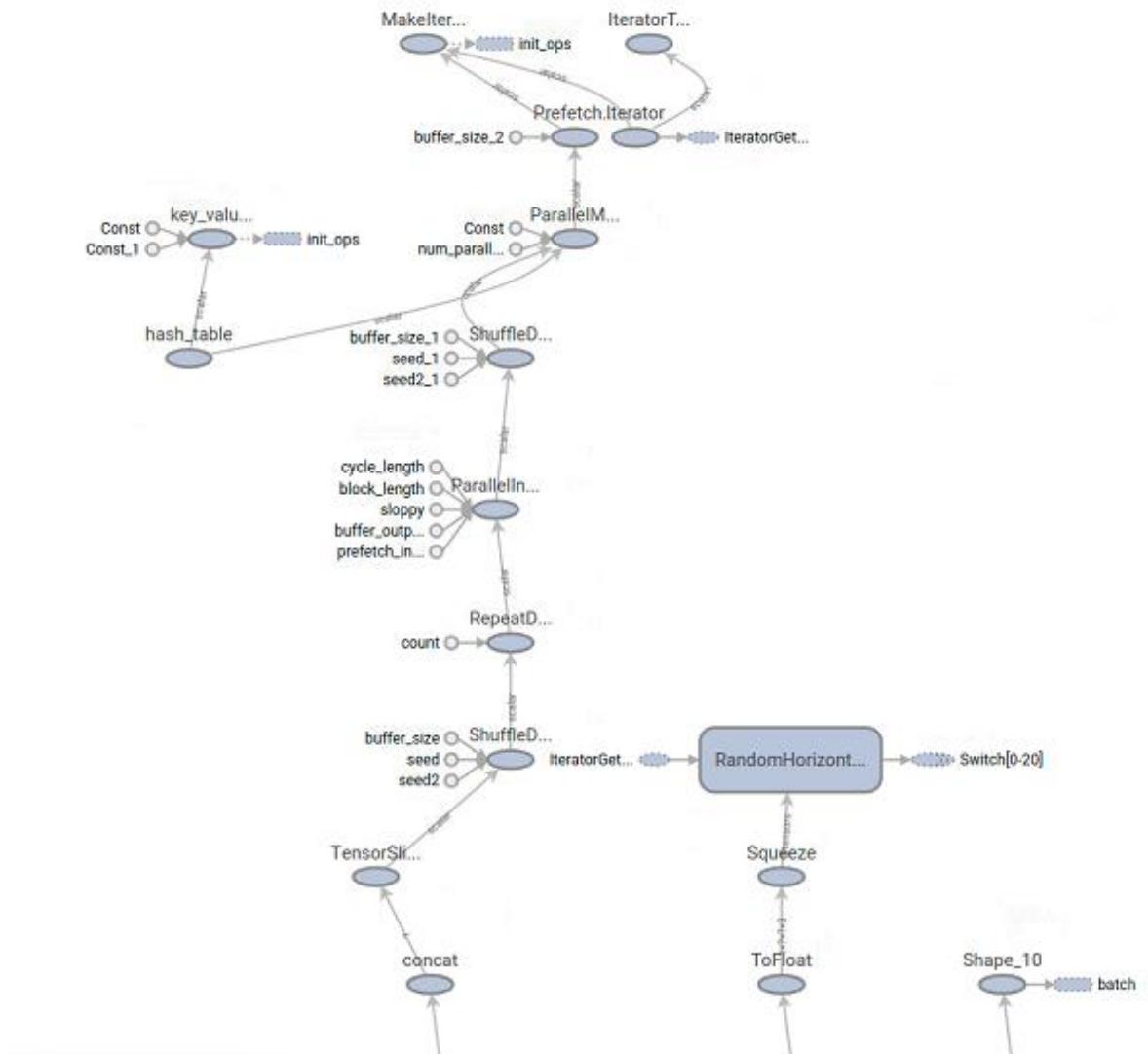


Figura 3.9 Diagrama de inicialización de matrices

Siguiendo con el proceso para aplicar los kernel, los tensores donde se almacenan las imágenes se cortan de acuerdo a las dimensiones preestablecidas de los Kernel, esto mediante funciones condicionales especiales que se encargan de empatar las matrices en segmentos del tamaño del kernel a aplicar, iterando sobre el tensor de la imagen hasta completar la tarea, puede apreciarse el flujo de la información en la Figura 3.101

## Capítulo 3. - Arquitectura Propuesta

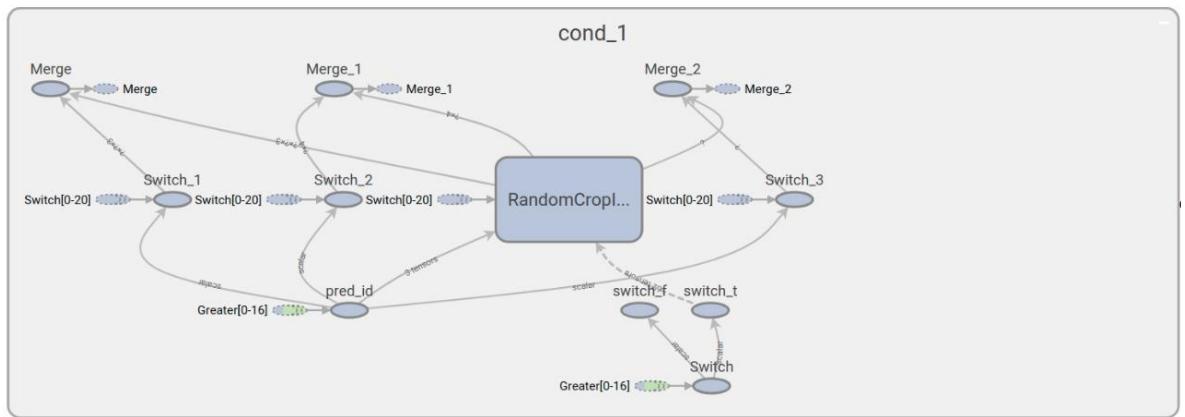


Figura 3.10 Flujo de información para la segmentación de tensores de imágenes

La primera parte del entrenamiento consiste en operaciones que, como esta, contribuyen a la preparación de los datos de entrada para mejor comprensión y optimización de los algoritmos internos. Generada esta información para el “batch” a analizar, comienza en el entrenamiento. La base para conocer, tanto el error como el porcentaje de mejora en el aprendizaje es el módulo de “Learning Rate”, este módulo arroja el porcentaje de aprendizaje basado en la correcta reconstrucción de los datos de entrada al final de la fase forward, tal como se aprecia en la Figura 3.112, se almacena el porcentaje de aprendizaje, pero también tiene salida a un módulo llamada RMSProp, el cual es un optimizador, esta clase de módulos, reducen el número de datos optimizando el espacio y hace más eficiente el aprendizaje.

## Capítulo 3. - Arquitectura Propuesta

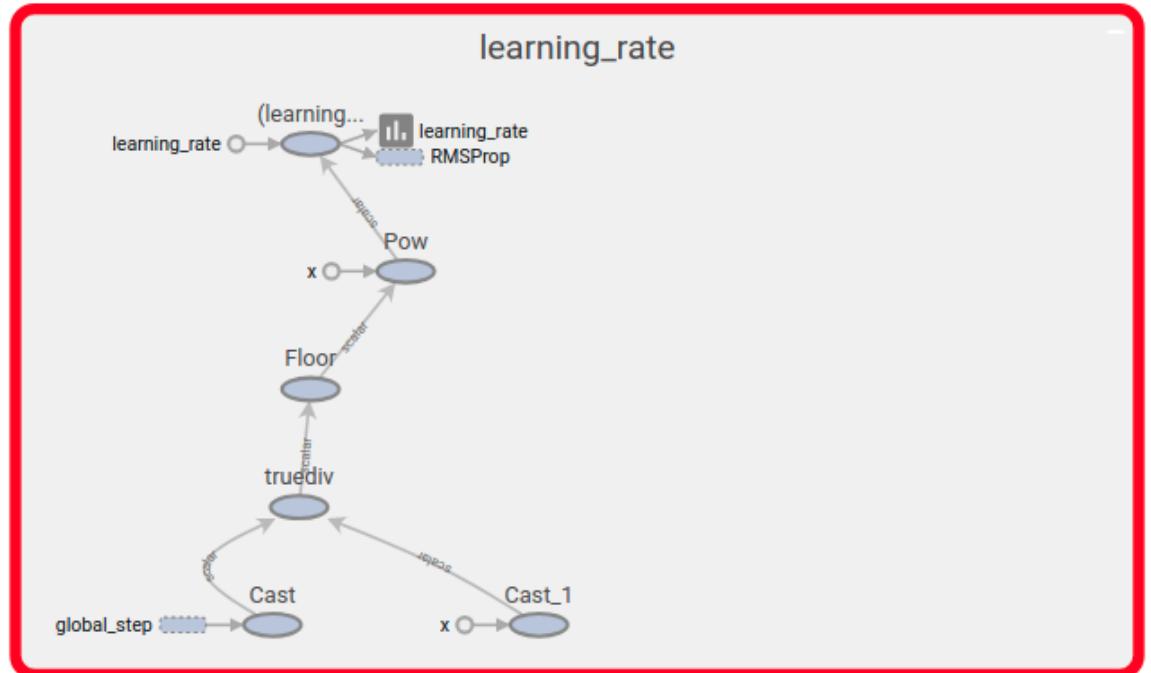


Figura 3.11 Flujo de datos principal del módulo "Learning Rate"

El módulo de optimización cuenta con varios submódulos, en los cuales se realiza la actualización de los datos sensibles del aprendizaje después del proceso de optimización de la última iteración, tal como se aprecia en la Figura 3.12.

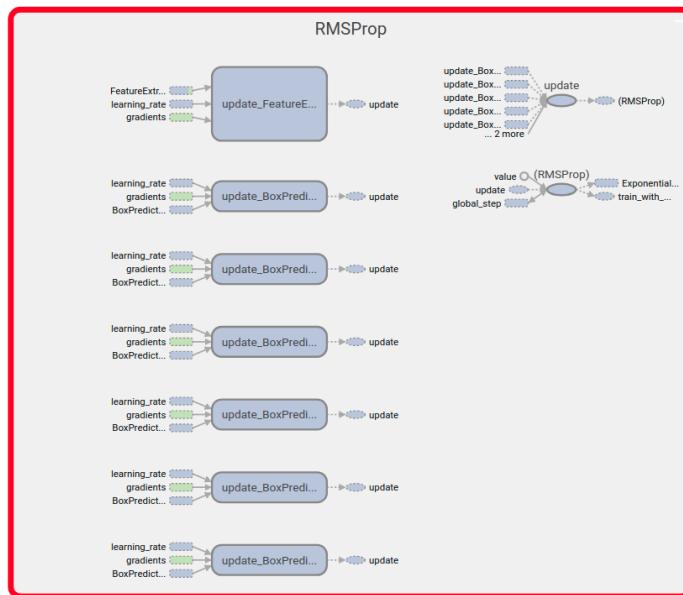


Figura 3.12 Entradas y salidas de los submódulos de optimizador

## Capítulo 3. - Arquitectura Propuesta

Ya en el proceso de aprendizaje, existen varios modelos que pueden aplicarse para la estructura de una red neuronal convolucional. En este proyecto se utilizó el modelo MobilNet (Howard, y otros, 2017) que se enfoca en aplicaciones móviles y embebidas de visión por computadora.

El proceso MobilNet se enfoca en convoluciones separadas por profundidad (depthwise separable convolution), que se puede descomponer a grandes rasgos en 2 tipos de capas:

Profundidad (Depthwise): se encarga de aplicar un filtro por cada canal de entrada de los datos.

Puntual (Pointwise): se utiliza para crear una combinación lineal de la salida de las capas de profundidad.

Las salidas de las funciones preliminares, de lectura y estos dos tipos de capas se utilizan para llevar a cabo el aprendizaje en módulo de “FeatureExtractor” del modelo (ver Figura 3.13), en este módulo se llevan a cabo las operaciones convolucionales de modelo tal como se puede ver en la Figura 3.14, al igual que contribuye con los distintos módulos que almacenan el nivel de aprendizaje y el aprendizaje como tal.

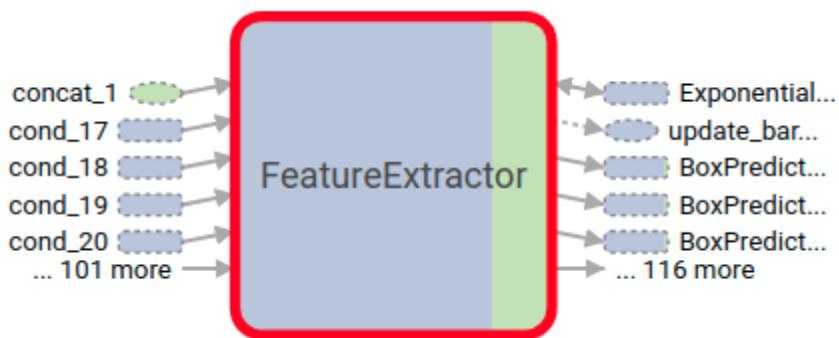


Figura 3.13 Entradas y salidas generales del módulo FeatureExtractor

## Capítulo 3. - Arquitectura Propuesta

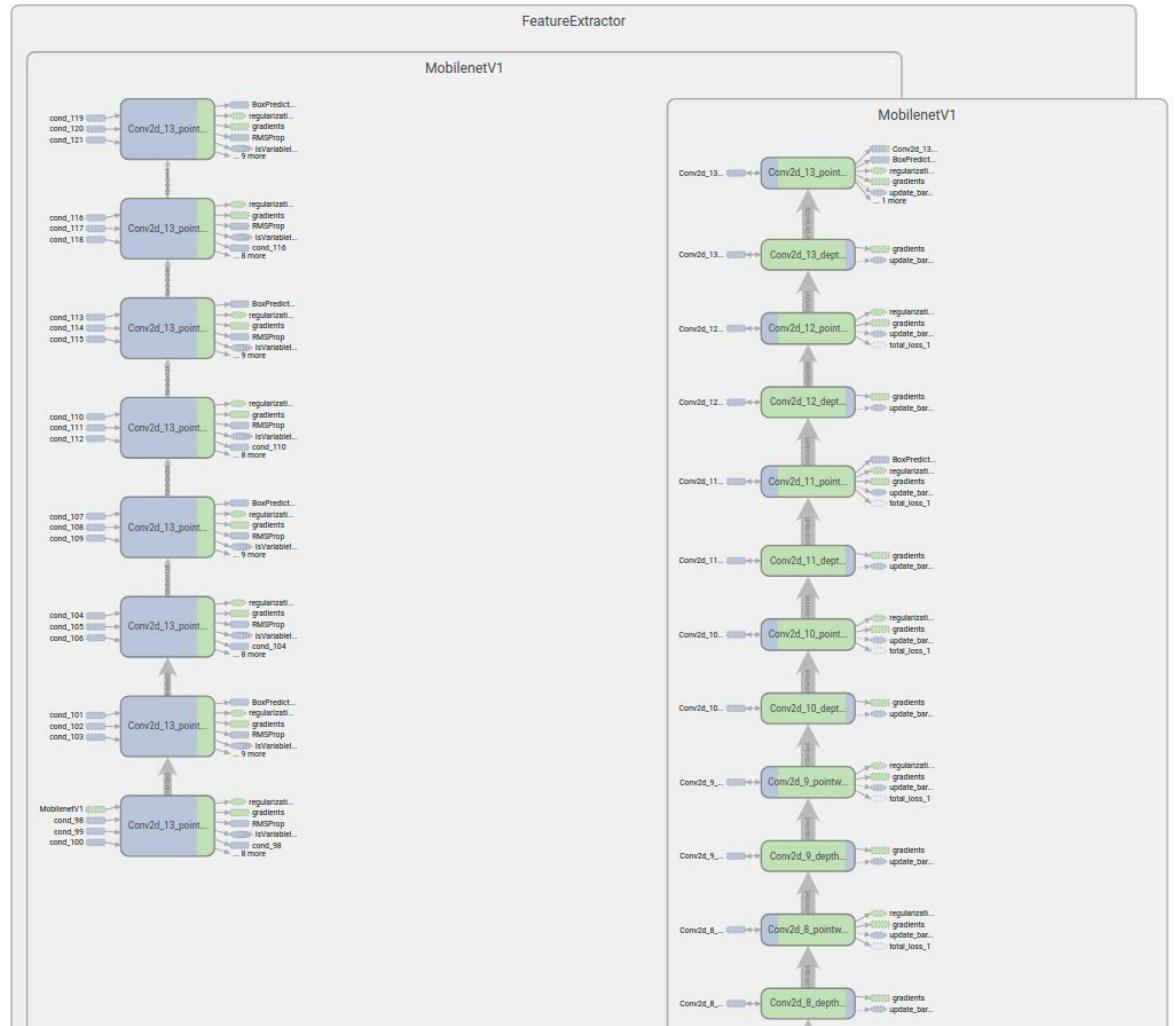


Figura 3.14 Fragmento del módulo desglosado "FeatureExtractor".

Hasta ahora, el modelo ha aplicado todos los kernel que se planearon al inicio a una muestra del tamaño del batch, sin embargo, el siguiente y último paso, determinará que tan bien ha interpretado y desglosado la información y basándose en ese resultado, se ajustarán las matrices kernel para adecuarse a los valores esperados.

El módulo classPredictor enfrenta, los valores arrojados por el módulo "FeatureExtractor" con los valores originales en la iteración en un proceso de convolución de dos dimensiones, los resultados se añaden a la tendencia, este proceso también ayuda a saber la cantidad de perdida de información (ver Figura 3.15). Sin embargo, el propósito fundamental de esta clase es asignar una clase de objeto a las características encontradas en el módulo "FeatureExtractor", y de esta

## Capítulo 3. - Arquitectura Propuesta

manera, al hacer posteriormente el proceso de inferencia identificar los objetos con nombres en lenguaje humano.

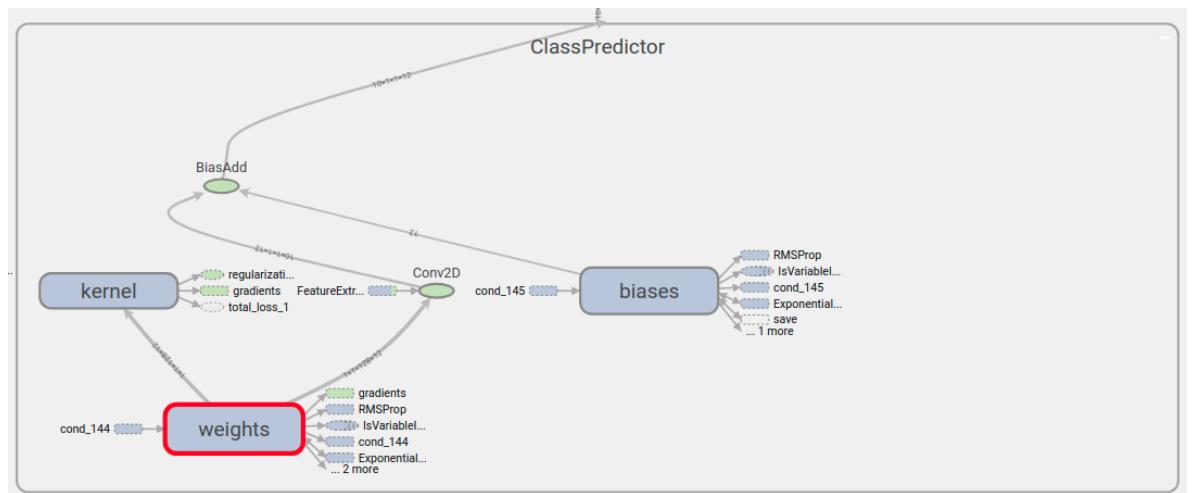


Figura 3.15 Desglose del módulo "ClassPredictor"

"Feature Extractor" pertenece a una clase superior que se denomina "BoxPredictor", la cual en general posee 2 subclases (ver Figura 3.16), la expuesta anteriormente "FeatureExtractor" y una adicional llamada "BoxEncodingPredictor", está más relacionada con la presentación de los datos al usuario, permite obtener dos pares de coordenadas a fin de identificar el objeto detectado dentro de los lados de un cuadrado.

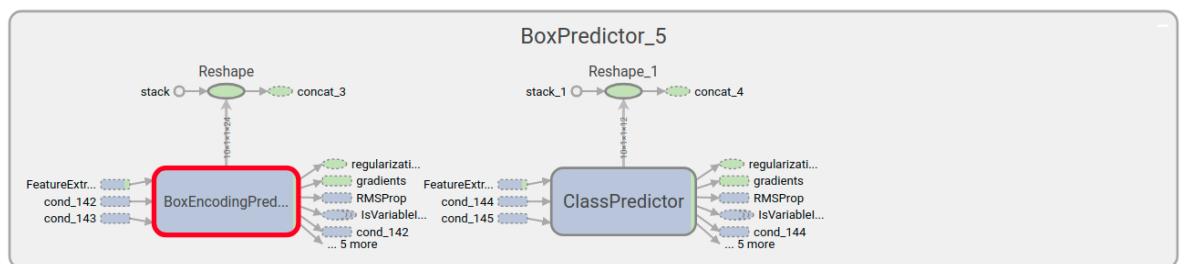


Figura 3.16 Desglose del módulo "BoxPredictor" y flujo de información

Como medida para la reducción de procesamiento en sistemas embebidos, se redujo el tamaño de las imágenes en cuestión de resolución, de esta manera los filtros aplicados por el modelo generado analizaran un número menor de pixeles, y de igual manera las capas

### Capítulo 3. - Arquitectura Propuesta

empleadas para la reducción dimensional ocuparan menor procesamiento.

La universalidad del lenguaje seleccionado para el trabajo (Python) permitió realizar una instalación y configuración del ambiente similar a la que se llevó a cabo en los servidores de entrenamiento, con excepción de algunas librerías específicas para los tipos de sistemas embebidos utilizados.

El tráfico de acceso a la colmena fue un aspecto importante, un tráfico descontrolado al sistema puede dar como resultado conteos inexactos. Las entradas y salidas, como se puede apreciar en el proyecto (Chiu, En-Cheng, Joe-Air, & Ta-Te, 2012), pueden controlarse de manera que sean mejor comprensibles y asimilables desde el punto de vista de la programación. El presente trabajo planteó la solución de utilizar esa técnica en conjunto con el aprendizaje de máquinas, desarrollando un “piquera”, que integrara tanto rieles de acceso para los polinizadores (ver Figura 3.17) (como medida para mejorar el registro) así como periféricos de entrada de imágenes (cámaras de tamaño muy reducido) que no invadieran el comportamiento natural de los individuos (ver Figura 3.18). De esta manera, las imágenes pueden ser capturadas y procesadas en un mismo flujo lineal, y la información sobre los avistamientos enviada a los laboratorios como se verá en la subsección 3.3 Centro de datos. El prototipo final de piquera se plantea como se puede ver en la Figura 3.19, incluyendo un soporte pequeño para vidrio, puesto que las abejas tienden a llenar los espacios con polen y el prototipo es un poco alto para que la cámara alcance a observar todos los rieles.

## Capítulo 3. - Arquitectura Propuesta

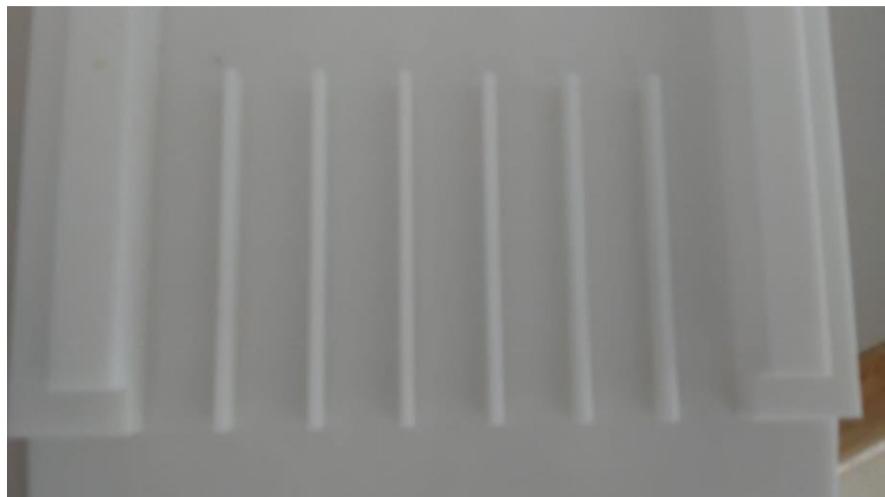


Figura 3.17 Rieles para el paso de polinizadores



Figura 3.18 Cámara colocada en la tapa de la piquera

### Capítulo 3. - Arquitectura Propuesta



Figura 3.19 Vista trasera general del prototipo de piquera

Una vez obtenidas las detecciones, puede llevarse a cabo en tiempo real, la toma de decisión sobre si fue un avistamiento de entrada o de salida. Esto con base en por lo menos 2 frames continuos de video, fue llevado a cabo mediante la búsqueda de centros de abejas más cercanos. Por ejemplo, si se detecta una abeja en ciertas coordenadas de pixeles y al siguiente segundo se detecta de nuevo en un radio menor a 20 pixeles, dependiendo si las últimas coordenadas son negativas o positivas con respecto a las primeras se puede determinar si fue una entrada o una salida. La utilización de rieles hace más preciso y menos demandante de procesamiento este trabajo, ya que, debido a la orientación de los rieles, puede buscarse solamente un radio de avistamiento en el eje Y pudiendo dejar de lado el eje X. Siguiendo los conceptos más básicos de los elementos principales en una unidad LSTM como se muestran en la Figura 3.20 (Long Short-Term Memory), que es un tipo especial de red

neuronal recursiva (RNN) enfocada para la predicción en datos secuenciales, los avistamientos se almacenan en un matriz o celda de información, en esta se almacenan las coordenadas y el centro del avistamiento para futuras consultas, se cuenta con puentes (gates) para lectura (output), escritura (update), olvido (forget), el puente de lectura permite el acceso a la matriz de información; el puente de escritura, permite modificar o añadir nuevos datos a la celda de información y el puente de olvido decide la edad y expiración de cada avistamiento, con esto se elimina la necesidad almacenar grandes volúmenes de coordenadas innecesarias y se puede conservar la información que es útil hasta el momento de la consulta.

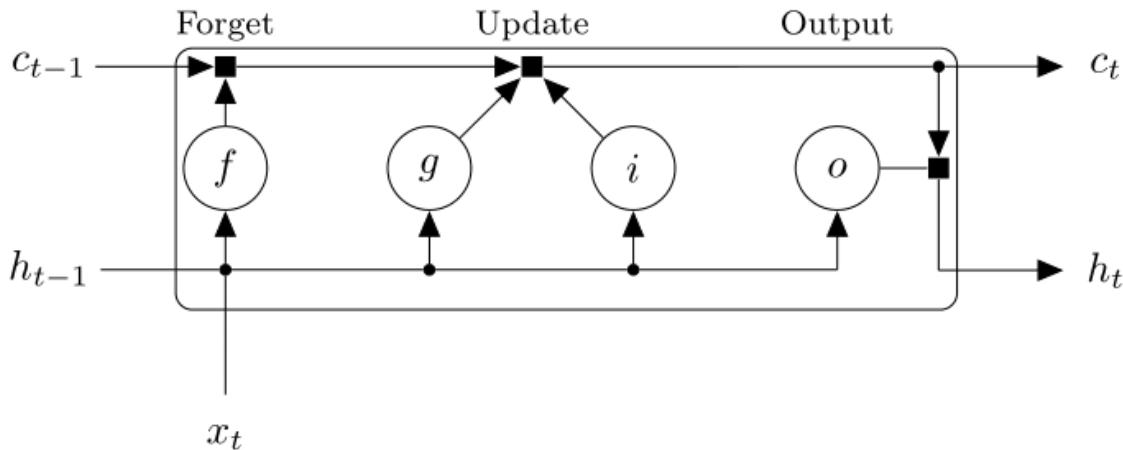


Figura 3.20 Apreciación gráfica de los elementos principales de un LSTM (Mathworks, 2018)

Las variables estudiadas en esta sección de la arquitectura cubren los tres niveles de estudio y compilación de datos. Las lecturas son tomadas desde dispositivos llamados agentes y se dividen en 2 clasificaciones:

1. **Monitor:** Este es el agente más común, consiste en un dispositivo de pocas capacidades, suficientes para albergar sistemas operativos de reducida complejidad (pensado para estos dispositivos), que corre procesos para la toma de datos desde los sensores que recompilan datos del segundo nivel (nivel colmena) y de la cámara y sensor RFID que obtienen los datos del primer nivel (nivel individual). Estos dispositivos se

## Capítulo 3. - Arquitectura Propuesta

encuentran bajo los panales, poseen mecanismos y herramientas para almacenamiento local de información y se comunican con el servidor mediante el agente de tipo “Gateway”. Los dispositivos de este tipo cuentan con entradas analógicas, digitales, seriales, USB, conector CSI para la cámara del dispositivo y conexión wifi para la red local.

2. **Gateway:** Este dispositivo es el encargado de tomar lecturas de la estación meteorológica, que corresponden a la información del tercer nivel (nivel apiario), almacenarla localmente y enviarla al servidor en el centro de datos (Subsección 3), aunado a esto, actúan como puerta de enlace entre la red pública y el apiario. Este dispositivo abre la red wifi a la que se conectan todos los “Monitores” en el apiario, y realiza el “Forwarding” entre la red local y la red GSM, mediante el módulo PPP (incluido como módulo del Kernel) para posteriormente salir a internet desde los sistemas del proveedor de red GSM. Los dispositivos de esta categoría idealmente deben contar con varios puertos para comunicación serial ya que tanto la estación meteorológica como el módulo GSM hacen uso de este protocolo, también cuentan con puertos analógicos (para la toma de lecturas desde el sensor de luz) y digitales (toma de lecturas de temperatura y humedad). Ha sido importante que este tipo de dispositivos no se encarguen de numerosos sensores como el tipo monitor, ya que son el puente entre todo el apiario y la red pública, una falla relacionada con los sensores o con los dispositivos de detección individual podría poner fuera de servicio a este e incomunicar todo el sistema en el apiario.

Nótese la diferencia entre el adjetivo “agente” utilizado con más frecuencia en este documento para referirse a dispositivos en general y las clasificaciones “Monitor” y “Gateway” en la arquitectura propuesta y utilizadas con menos frecuencia en el resto del documento.

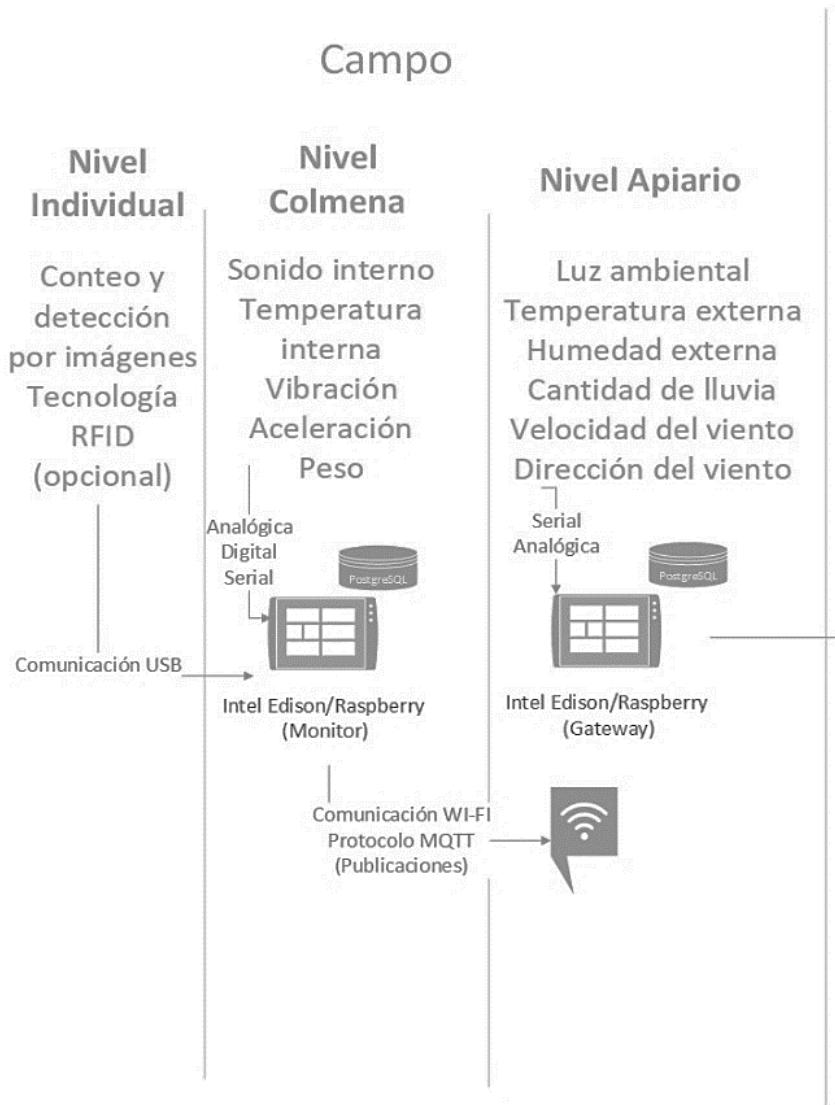


Figura 3.21 Subsección "Campo", Detalles de tecnologías, sensores y dispositivos utilizados

### 3.2. Comunicación Remota (salida a internet)

La siguiente subsección llamada “Comunicación Remota” (ver Figura 3.22) es la entrada a la red GSM del proveedor y sus sistemas, para posteriormente salir a internet y encontrar el servidor del centro de datos (Subsección 3). De acuerdo a los componentes generales según (PTC University, 2017), en esta capa se encuentran los componentes principales de conectividad que permiten la comunicación remota y el paso de información. Esta capa es el puente de comunicación entre el sistema en el apiario (dispositivos monitores y gateways y todos los sensores que tengan en

### Capítulo 3. - Arquitectura Propuesta

funcionamiento) y el almacén masivo de información. La comunicación inicia desde el módulo GSM, el cual se configura para establecer la conexión con los dispositivos “Gateway”, estos a su vez deben tener habilitados los módulos del protocolo PPP del kernel para interacción con la tarjeta GSM. Para el sistema propuesto en este trabajo, como usuario de la red GSM de su proveedor, le son transparente los procesos que el proveedor implementa para la salida a internet. Mediante la internet y la IP pública del servidor se localiza el centro de datos y se establece la conexión con el puerto correspondiente para la recepción de la información. Los aspectos más importantes en esta parte de la arquitectura serían:

La configuración del módulo GSM, generalmente esta configuración se realiza mediante comandos que permiten observar el estado de la conexión a bajo nivel, el proveedor, frecuencias, protocolos que se utilizaran en la red del proveedor (como la conexión a internet o en otros casos el envío de mensajes y el marcado). Algunos proveedores piden de igual manera la verificación mediante PIN para la entrada a su red y este debe configurarse también directamente en el módulo GSM.

La investigación de la cobertura de los proveedores: debe verificarse que efectivamente el sistema está dentro del alcance de los sistemas de difusión del proveedor, de lo contrario, investigar proveedores alternos, las localizaciones de los apíarios, generalmente alejadas de las zonas más transitadas o pobladas, es lo que da importancia a este aspecto, la disponibilidad de red es un aspecto crítico de las zonas donde podría ser instalado el sistema.

Configuración del sistema embebido a utilizar: una vez habilitado el protocolo PPP en el kernel, deben instalarse librerías propias del sistema operativo para la operación e interacción con los módulos del kernel, esto permitirá al dispositivo reconocer el dispositivo serial como un dispositivo de red, reconocible para las comunicaciones y protocolos en uso. Aunado a esto, deben establecerse permisos para el paso de paquetes a través del dispositivo en cuestión, de esta manera se le otorga el permiso al dispositivo para transferir paquetes desde la red local (el apíario) a la red del proveedor (la internet), si este último paso la transferencia de información y el paso de paquetes no sería posible.

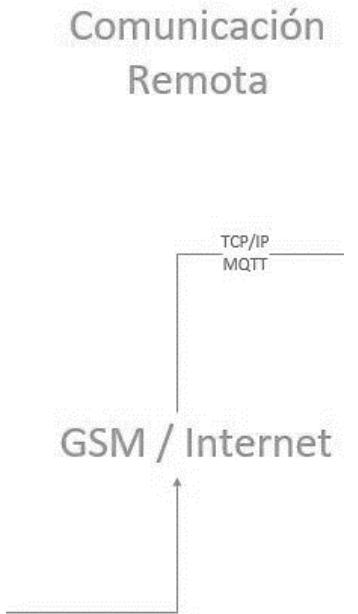


Figura 3.22 Subsección "Comunicación Remota", Tecnologías y protocolos

### 3.3. Centro de datos (almacén central de información)

La última subsección de la arquitectura llamada “Centro de datos” (ver Figura 3.23) corresponde al centro de almacenamiento masivo de la información, que de acuerdo con la clasificación general de los componentes de (PTC University, 2017), se conforma por componentes inteligentes, ya que se compone de una base central que contiene lecturas de todos los agentes instalados, esta parte se compone de tres elementos indispensables para su propósito:

1. **Demonio “broker”**: La arquitectura interna de este demonio es del tipo publicación-suscripción. Por lo tanto, a este se conectan todos los dispositivos “Monitor” y “Gateway” para realizar sus publicaciones desde ubicaciones remotas. Los mensajes se envían a modo de publicaciones al servidor y dentro de este, deben existir scripts que se suscriban a tales publicaciones, para poder recibir la información que estás contienen, estos scripts se detallan en el siguiente punto.
2. **Suscriptor**: dentro del servidor existe otro proceso el cual se suscribe a ciertas publicaciones en el “broker” para recibir la información, de

esta manera interactúan el demonio principal y el demonio que interpretará y almacenará la información en la base de datos. Cada tipo de sensor en campo, cuenta con una publicación diferente, de modo que tenemos 10 tipos de sensores diferentes más la información de los avistamientos, en el servidor se reciben 11 tipos de publicaciones diferentes y el “Suscriptor” debe suscribirse a esas publicaciones para recibir la información. De esta manera, si los agentes envían publicaciones con el nombre “/site\_1/sensors/light” (siguiendo el formato estandarizado para las publicaciones) el suscriptor debe estar suscrito a la publicación “/site\_1/sensors/light” (este formato sigue una abstracción del tipo orientado a objetos separado por diagonales y aunque no es obligatorio es un estándar para este tipo de identificadores en el “broker”), para recibir la información del sensor de luz del sitio número 1.

El siguiente paso, después de recibir la información es interpretar la cadena de datos que se envía desde campo, ya que esta contiene los identificadores únicos del agente y sensor de procedencia dentro del sistema, los cuales están registrados en la base de datos central y los pertenecientes a cada agente en su base de datos local; después de los identificadores el siguiente dato la lectura tomada, un campo de contenido numérico donde viajan los datos obtenidos en campo del sensor que publica, y por último la fecha y hora de la lectura.

3. **Base de datos:** los últimos elementos de importancia son el demonio de base de datos en el servidor y la base de datos como tal. Esta base de datos relacional contiene los identificadores únicos de cada sensor y de cada agente. Los identificadores de los agentes están relacionados a la ubicación (latitud, longitud y ubicación por ciudades) de los sitios donde se encuentran instalados, la información propia del mismo dispositivo (siendo el campo más relevante el numero serial del dispositivo, ya que este permite inicializar la base de datos local

### Capítulo 3. - Arquitectura Propuesta

en los agentes) e información de los sensores que tienen en funcionamiento. Los identificadores de los sensores se relacionan con una tabla de lecturas, donde se almacenan las mediciones hechas en campo y la hora en que se realizó, con información propia de cada sensor y esta a su vez con la tabla de unidades que nos permite saber cuál es la medida que se utiliza para cada medición en la tabla de lecturas (por ejemplo: un sensor de temperatura debe estar asociado con la unidad “grados centígrados” y un sensor de peso debe asociarse con la unidad “kilogramos”).

De esta manera, todos los agentes en campo (dispositivos “Monitor” y “Gateway”) publican la información de los 3 niveles en el “bróker” del servidor y el proceso que recibe la información se suscribe a esas publicaciones y toma la información para interpretarla y almacenarla en la base de datos central. Con este esquema, aplicamos los principios de las bases de datos semi-distribuidas, sincronizando las bases de datos en campo con la base de datos central y compartiendo los identificadores únicos que sirven como referencia de los elementos en todo el sistema.

De esta manera también se aplican los principios generales del campo de bases de datos relaciones, así como los protocolos desarrollados y optimizados para aplicaciones de Internet de las cosas, implementables en dispositivos poco invasivos y de muy pocas capacidades de procesamiento y memoria.

## Capítulo 3. - Arquitectura Propuesta

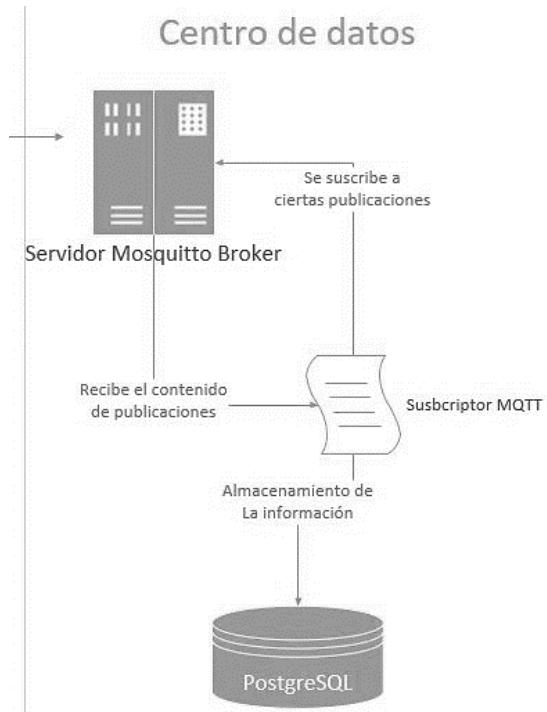


Figura 3.23 Subsección "Centro de Datos", Funcionamiento y flujo de la información en el servidor

Mediante la implementación explicada en el presente capítulo, el autor cubre los niveles más importantes en la observación apícola, haciendo uso de tecnología flexible y pensada para la eficiente transmisión de datos. Un sistema integrado permite varios puntos de vista de algún suceso en cuestión y la observación en tiempo real permite la rápida acción en respuesta a algún suceso fuera de lo común. El paradigma IoT y el aprendizaje de máquinas, más allá de su uso común en industrias o ciudades, ahora son llevados al medio natural en la observación apícola y el siguiente capítulo expondrá los resultados de esa iniciativa.

# Capítulo 4. Resultados

La presente sección se propone revisar los resultados obtenidos de la implementación de la tecnología en campo, revisar los logros que se obtuvieron en el proceso de optimización del flujo de información, mediante internet de las cosas, desde los apiarios a los laboratorios y su presentación a los usuarios, de igual forma, presentar los resultados del aprendizaje de máquinas aplicado a la apicultura implementando detección y conteo automatizado en las colmenas.

El capítulo anterior mostró a detalle, los aspectos de la implementación y el flujo de información a través de las subsecciones. Se revisaron los métodos mediante los cuales, la información llega desde el apiario hasta los servidores a través del paradigma y las facilidades del internet de las cosas y se presentaron los detalles de la implementación de aprendizaje de máquinas para el reconocimiento y conteo de polinizadores transfiriendo aprendizaje humano a un modelo computacional a través de una arquitectura compuesta de redes neuronales convolucionales.

En este capítulo se presentarán los logros organizados en las 2 secciones principales de la arquitectura, la implementación de la red de sensores mediante IoT y el aprendizaje de máquina para reconocimientos de objetos.

## 4.1. Red de sensores mediante IoT

La implementación de la red de sensores permitió observar la evolución de las variables analizadas a través del tiempo, y comparándolos a simple vista generar patrones y relaciones entre las variables en contexto.

Esta sección presenta los resultados de cada objetivo específico que va encaminado a la implementación de la red de sensores.

## Capítulo 4. - Resultados

### 4.1.1. Caracterización de sensores y asignación de rangos normales de operación.

La caracterización de sensores permitió seleccionar los datos más relevantes a recolectar en campo y así seleccionar, tanto los tipos de sensores como la subsección a donde debían pertenecer. La asignación de rangos normales de operación, realizada mediante el rango promedio de las mediciones, permite deducir con mayor facilidad cuando un evento poco común se está detectando en un sensor determinado y así lanzar una alerta o notificación como se verá en los siguientes objetivos específicos.

### 4.1.2. Recolección de lecturas, envío y almacenamiento

El proceso constante de comunicación del apíario con los laboratorios, permitió observar los cambios constantes en tiempo real de las variables estudiadas y hacer relaciones a simple vista de los datos mediante gráficas. En esta subsección se expondrán las mejoras aportadas a través IoT a 2 métodos tradicionales que son los dataloggers y los envíos por FTP.

#### 4.1.2.1. Comparativa con dataloggers tradicionales

Uno de los objetivos principales del internet de las cosas es la automatización de procesos, para que, eventualmente se reduzca al mínimo la necesidad de intervención humana. En la industria, por mencionar un ejemplo, esto ha llevado a suplantar la mayoría de los numerosos puestos de monitoreo de las distintas maquinarias, de los cuales una gran parte además de ser repetitivos podían ser peligrosos, por soluciones de monitoreo automatizado utilizando internet de las cosas. Acorde a (Cisco Networking Academy, 2018), los usos y las cosas que pueden monitorearse mediante este nuevo protocolo solo están limitadas por la imaginación.

Tanto IoT como dataloggers proveen información estructurada, en la comparativa presentada los resultados arrojados por el datalogger fueron guardados en archivos CSV, y los resultados arrojados por el presente proyecto fueron almacenados en

## Capítulo 4. - Resultados

bases de datos. Ambos llevan a cabo un proceso automatizado de toma de datos en campos mediante sensores, el dataloggers utiliza un sensor RFID para detectar el paso de los polinizadores etiquetados con microchips (este tema se abordará más a fondo en la subsección 4.2), el presente proyecto lleva un proceso automatizado de toma de datos de sensores ambientales dentro y fuera de la colmena.

La principal mejora entre el método tradicional y el método propuesto radica en la disponibilidad de la información en los laboratorios y la organización de la misma. En primera instancia se expondrá la organización de la información.

### 4.1.2.1.1. Organización de la información

- Para análisis posteriores, los CSV requirieron una compilación previa de todos los archivos individuales conseguidos, lo cual puede llevarse a cabo realizando un programa para generar el nuevo archivo y leerlo posteriormente o guardar todos los registros de los archivos en memoria y utilizar el compendio temporal para los análisis. La organización en registros de una base de datos, que es la solución propuesta de proyecto presente, no solo realiza ya el compendio de la información como su forma natural de almacenamiento (ver Figura 4.1), también el mismo lenguaje SQL, que sirve para almacenar la información, puede llevar a cabo parte de los análisis de manera eficiente, ya que su motor de explotación de datos fue diseñado para trabajar con grandes cantidades de información. Esto demuestra que la solución propuesta es más escalable en términos computacionales que la solución tradicional.

| pk_id_observation | observation_value | observation_date    | fk_id_agent | fk_id_sensor |
|-------------------|-------------------|---------------------|-------------|--------------|
| 33441             | 0.11474609375     | 2018-05-01 20:07:31 | 1           | 2            |
| 33951             | 299.944444444     | 2018-05-15 07:06:00 | 1           | 3            |
| 34589             | 23.79             | 2018-05-21 13:36:47 | 1           | 7            |
| 35099             | 22.40766666667    | 2018-05-22 05:37:06 | 1           | 7            |
| 33187             | 0.2197265625      | 2018-04-25 14:04:11 | 1           | 2            |

Figura 4.1 Los datos se guardan como un compendio y cada tupla se relaciona con su lugar de procedencia y tipo de información

- Un sistema general (sea distribuido o centralizado) que almacena la información aumenta la coherencia de los resultados finales.

## Capítulo 4. - Resultados

- Existe una mayor eficacia en la captura, validación e ingreso de datos al sistema mediante una base de datos. Por ejemplo, como se ve en la Figura 4.1, un sensor y un agente deben estar previamente dados de alta en la base de datos para poder almacenar observaciones, en dado caso que no estén registrados se produce un error debido a las relaciones. Los identificadores tanto de sensor como de agente, se relacionan en tablas externas donde se almacena otro tipo de información acerca de cada dispositivo.
- Una base de datos ofrece una mayor disponibilidad y capacidad de transferencia en comparación con un archivo de texto plano como un CSV. Como puede observarse en la Figura 4.2 la base de datos puede ser consultada a cualquier hora y contar con los datos más recientes.

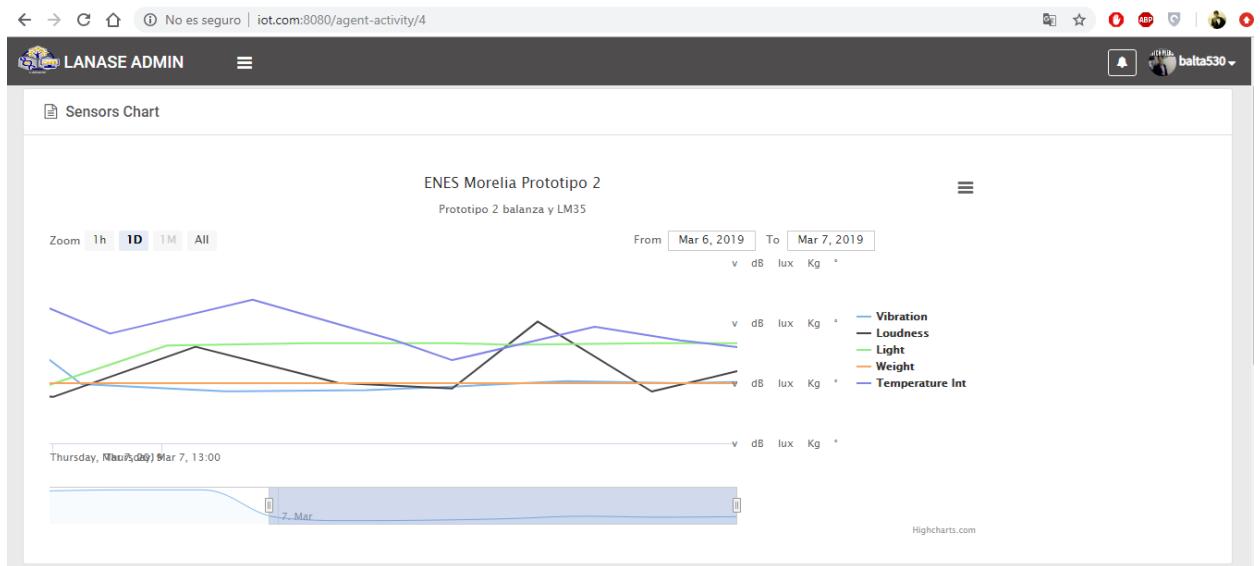


Figura 4.2 Los datos están en constante actualización y son accesibles en cualquier momento

- El proceso de recuperación de la información está más optimizado incluso al nivel de un solo registro en un sistema de base de datos que en un fichero compuesto por texto plano.

### 4.1.2.1.2. Disponibilidad de la información

- La recuperación de los datos de campo es la mejora principal del presente proyecto en comparación con los datalogger. Los últimos requieren de

## Capítulo 4. - Resultados

intervención humana para el transporte de la información desde campo a los laboratorios, estas se realizan cada determinado tiempo a las locaciones donde se encuentre cada sistema funcionando para descargar los datos llevarlos al laboratorio para posteriores análisis. Esto puede llevar incluso días completos ya que los estándares existentes para la colocación de apíarios ordenan cierta distancia de lejanía entre las instalaciones y las zonas pobladas, sobre todo ciudades. Esto ha llevado a que las locaciones se encuentren lejanas a los laboratorios o en lugares de difícil acceso.

- Al ser los datos recolectados en lapsos de semanas o meses, no se pueden apreciar los datos y la evolución de las variables más que en los lapsos de tiempo de las recolecciones. El sistema propuesto permite el envío de datos incluso en tiempo de minutos o segundos (según se configure). Los tiempos de envío en las pruebas en campo fueron 3 y 5 minutos (ver Figura 4.3) sin ninguna intervención humana además de la instalación del sistema, lo cual quiere decir, que cada 3 y 5 minutos se estuvieron recibiendo datos en los laboratorios directamente desde campo, los cuales permitió observar la evolución de las variables observadas prácticamente en tiempo real, quedando comprobada la parte de la hipótesis que discute las ventajas del uso de IoT para la automatización y el uso de la red 2G, como medio eficiente para la trasmisión de información desde campo.

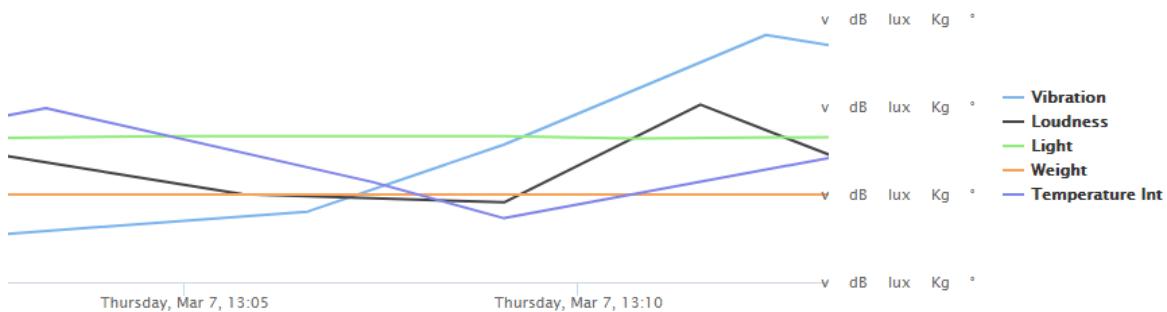


Figura 4.3 Los tiempos pueden ajustarse incluso en minutos

#### 4.1.2.2. Comparativa con envíos por FTP

FTP (File Transfer Protocol) es el protocolo más popular para la transferencia de documentos a través de la red. Permite la modificación de archivos remotos, y el almacenamiento en la nube, el protocolo FTP ha sido utilizado en proyectos industriales como (BeeWatch - Biene & Natur GmbH - Beekeeping Needs, 2014), para permitir el paso de información remota a servidores, sean propios de la empresa o de propietarios de las balanzas.

El principal impedimento que muestra FTP, es ser orientado a paso de archivos, esto limita el paso de información en tiempo real, ya que para cada dato debería enviarse todo un archivo. Las soluciones industriales, tratando de hacer más eficiente el paso de documentos completos, almacenan cantidades de información de un cierto rango de tiempo para enviarla y repetir el proceso, proyecto industrial mencionado envía un archivo con información cada 15 minutos, sin embargo, existen sistemas que tardan horas completas o el envío se realiza diariamente.

El internet de las cosas, ofrece una mayor flexibilidad para el envío de datos, enfocándose más en datos individuales que, en el envío de archivos completos, como se puede ver en la Figura 4.4, el envío que se muestra se realizó mediante el protocolo MQTT, desarrollado para IoT, el paso de mensajes (como se explica en el capítulo anterior) se lleva a cabo mediante publicaciones, que se conforman por dos datos importantes, el nombre de la publicación y la información a enviar, lo cual aumenta la sencillez de la implementación y economiza los gastos de operación a la hora de depender de un proveedor de red GSM. Una comunicación orientada a datos, nos permite la trasferencia de información en tiempo real.

```
('/id_site_1/temperature_int', '7 70 26.6 2019-05-02 17:01:43')
('/id_site_1/humidity', '7 71 36 2019-05-02 17:01:43')
('/id_site_1/Brood', '7 85 0.0 2019-05-02 17:01:44')
('/id_site_1/Pluviometer', '7 86 0.0 2019-05-02 17:01:44')
('/id_site_1/Anemometer', '7 87 0.0 2019-05-02 17:01:44')
```

Figura 4.4 Las publicaciones consisten en dos cadenas sencillas

## Capítulo 4. - Resultados

El tiempo de envío de una publicación es comparativamente menor al de un envío mediante protocolo FTP como puede apreciarse en la Figura 4.5 y la Figura 4.6, ambas pruebas se corrieron en el mismo intervalo de tiempo, 6 horas, enviando información por FTP cada hora y mediante protocolo MQTT cada 10 minutos. Debido al contenido de los archivos FTP pudo enviar la información de 1 hora en aproximadamente .16 segundos en promedio, sin embargo, el envío mediante MQTT tomó aproximadamente de 0.0015 a 0.0030 segundos por publicación. En cada hora aproximadamente se reunían 70 datos en bruto, FTP envío los 70 datos en un solo archivo cada hora, MQTT envío una publicación por cada dato, o sea 70 publicaciones con un tiempo aproximado por publicación de 0.0022 segundos, tomando el tiempo acumulado de las 70 publicaciones en ese intervalo de tiempo se obtiene un monto de tiempo aproximado al tiempo que ocupó FTP para el envío de los datos cada hora, sin embargo, MQTT permitió observar la evolución de los datos dentro de la hora y no al finalizar la misma.

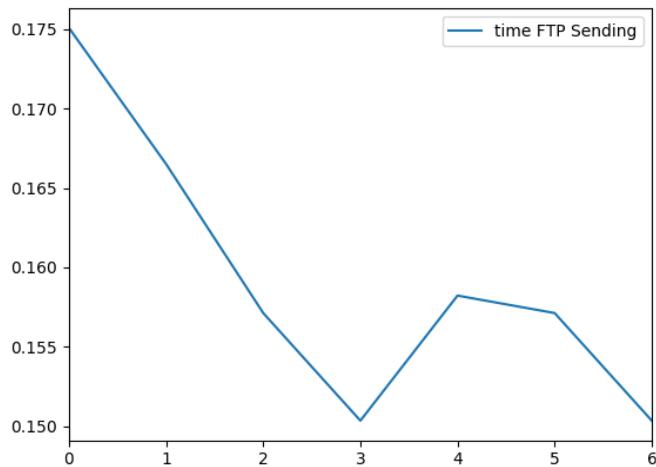


Figura 4.5 Tiempos representativos de envíos mediante FTP

## Capítulo 4. - Resultados

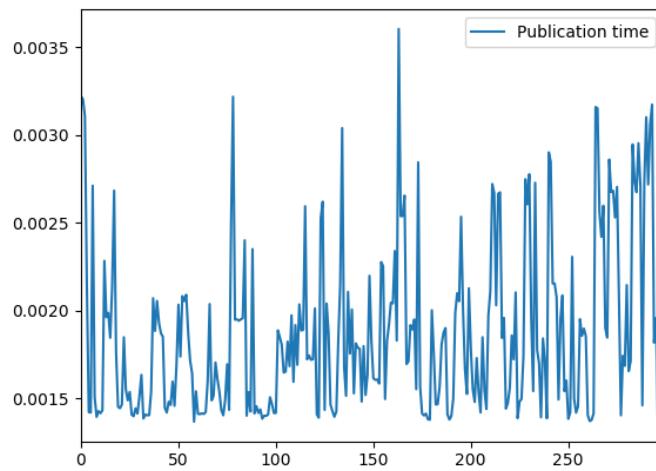


Figura 4.6 Tiempos representativos de publicaciones mediante protocolo MQTT

La cuestión del peso de los paquetes es otro aspecto notable al trabajar con tecnología a la cual se le ancho de banda restringido. Aproximadamente el paquete de envío por FTP de los datos de toda una hora fue de 53 kb, según el monitoreo de la interfaz de salida del agente de pruebas, por otra parte, el cúmulo de publicaciones cada 10 minutos ascendía a 3 kb lo que en toda la hora acumulativamente fueron 18 kb, esto gracias al menor número de cabeceras del protocolo MQTT y la sencillez de las mismas.

### 4.1.3. Desarrollo de mecanismos para garantizar la integridad de la información y notificar de fallas en agentes o sensores.

La obtención del promedio de un número determinado de lecturas antes de enviar a servidor, garantizó que los datos recolectados fueran verídicos y significativos en un periodo de tiempo en minutos. El sistema de alertas posibilita el lanzamiento de notificaciones al usuario cuando alguna variable sale del rango promedio drásticamente, sea en aumento o en declive, como se puede ver en la Figura 4.7, el sistema cuenta con un tipo especial de publicación en caso de alerta, el cual almacena información relevante en la tabla de observaciones y guarda establece la relación entre la tupla insertada y la tabla de alertas, de igual forma permite saber

## Capítulo 4. - Resultados

cuándo ha habido fallas de hardware, ya que los valores obtenidos de ese tipo de fallas son establecidos como -2 y ese tipo de valor siempre se detecta como una alerta. Los sensores que generalmente están más al aire libre o sus circuitos han quedado cerca de mucha de actividad tienden a sufrir deterioro con el tiempo, el cual se refleja en cambios erráticos en las lecturas o fallas que arrojan excepciones.

```
if float(value) < 0 or float(value) > 150:  
    countWeightAlert += 1  
    #Incluir llamada a inicializador del sistema en la balanza  
  
    if countWeightAlert > triggerAlert:  
        #Insert alert into observation table  
        string_observation = insert_observation.insert(value, topic)  
  
        #Insert into alert table  
        string_observation = string_observation + " " + insert_alert.insert(value)  
  
        #Publish alert  
        mosquitto_publisher.publish("alert/" + topic, string_observation)  
  
    countWeightAlert = 0
```

Figura 4.7 Ejemplo simple de alerta cuando el valor obtenido sale del rango normal de operación

### 4.2. Detección de abejas utilizando aprendizaje de máquinas.

Esta sección expondrá los resultados del entrenamiento utilizando un servidor especial adecuado para aplicaciones de procesamiento intenso, el cual cuenta con 16 sockets de Intel Xeon con capacidad de 2.1GHz y 8 núcleos por socket además de un GPU NVIDIA Quadro 5000 con capacidad de 352 núcleos cuda, 62Gb de RAM usando 31Gb para swap.

#### 4.2.1. Comparativa entre métodos

Las pruebas comparativas se llevaron a cabo en una computadora con Intel i7-6700 con 8 sockets y 4 núcleos por socket con capacidad de 3.4GHz, 15Gb de RAM usando 2Gb para swap.

##### 4.2.1.1. COCO Model

COCO dataset (Tsung-Yi, y otros, 2015) es uno de los principales modelos para inferencia de objetos en imágenes. El significado de sus siglas es relativo a los usos para los que se planteó. Desglosado COCO quiere decir Common Objects in

## Capítulo 4. - Resultados

Context. Entrando con 330000 mil imágenes, 1.5 millones de instancias de objetos y capaz de reconocer 91 objetos de la vida cotidiana, COCO dataset se ha convertido alcanzado gran popularidad. Sin embargo, en esa subsección se analizará la desventaja que tiene un datase tan numeroso frente a un trabajo tan específico de reconocimiento como el que compete a este trabajo.

COCO dataset es una buena herramienta para trabajos de reconocimiento de objetos generales y cotidianos. Su amplia gama de objetos comunes lo hacen una herramienta esencial en aplicaciones del día a día. Sin embargo, el uso de este y otros datasets similares no son una buena herramienta para el reconocimiento y distinción de objetos específicos, ya que el mismo modelo, por la cantidad de objetos que puede reconocer y la variedad de clasificaciones que existen dentro de la gama de un solo objeto, llega a caer en confusión y más a allá de arrojar falsos positivos del objeto en cuestión arrojar clasificaciones erróneas relacionadas con otros tipos de objetos como se puede apreciar en la Figura 4.8 y Figura 4.9.

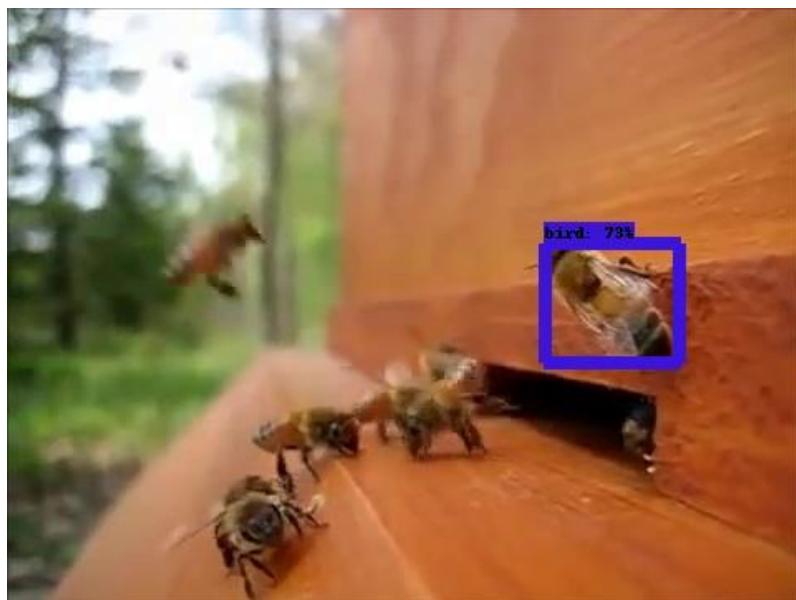


Figura 4.8 Confusión del modelo entre objetos

## Capítulo 4. - Resultados

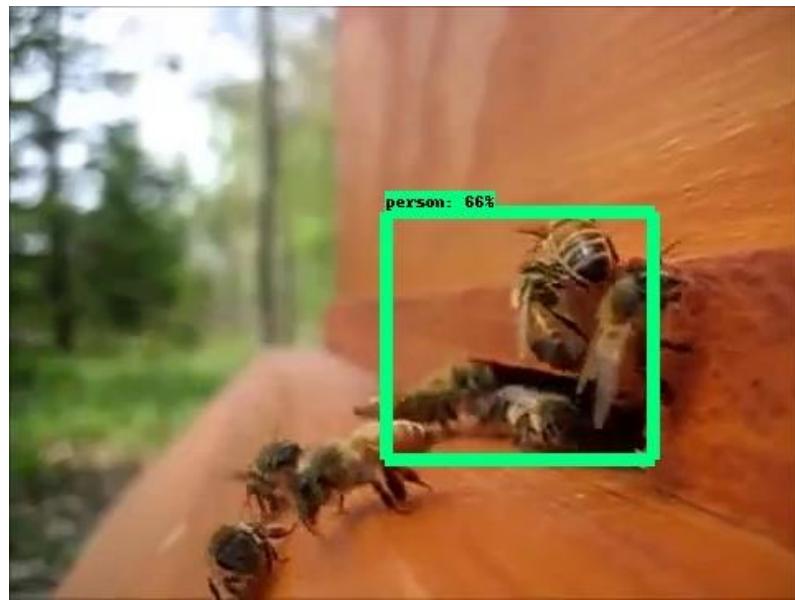


Figura 4.9 Detección de una persona en un área de agrupación de abejas

Un modelo ad hoc permite centrarse en las diferentes características del objeto en cuestión, así como en todas las formas que este puede adoptar, esto último de gran importancia ya que en el medio natural los insectos, en este caso abejas pueden presentar una gran variedad de formas y aspectos dentro de una imagen, y mejorar la calidad de las detecciones, como se puede ver la Figura 4.10 y Figura 4.11.

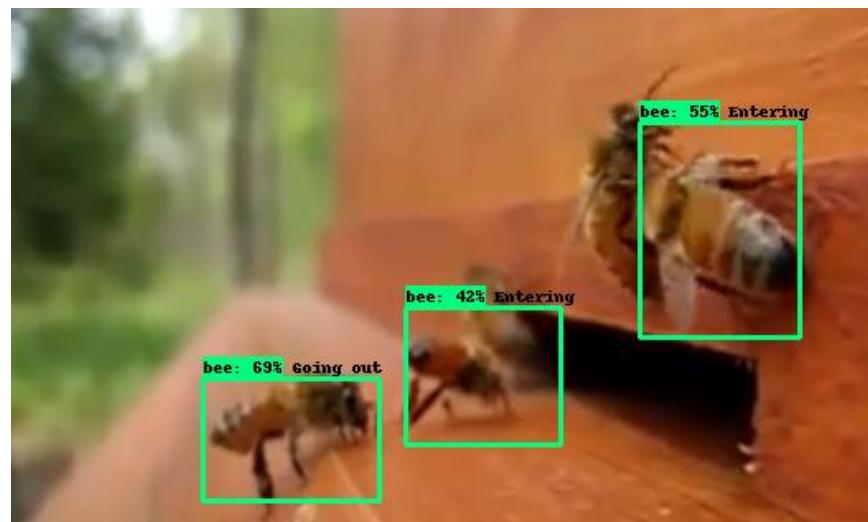


Figura 4.10 Detección de abejas con modelo a la medida

## Capítulo 4. - Resultados

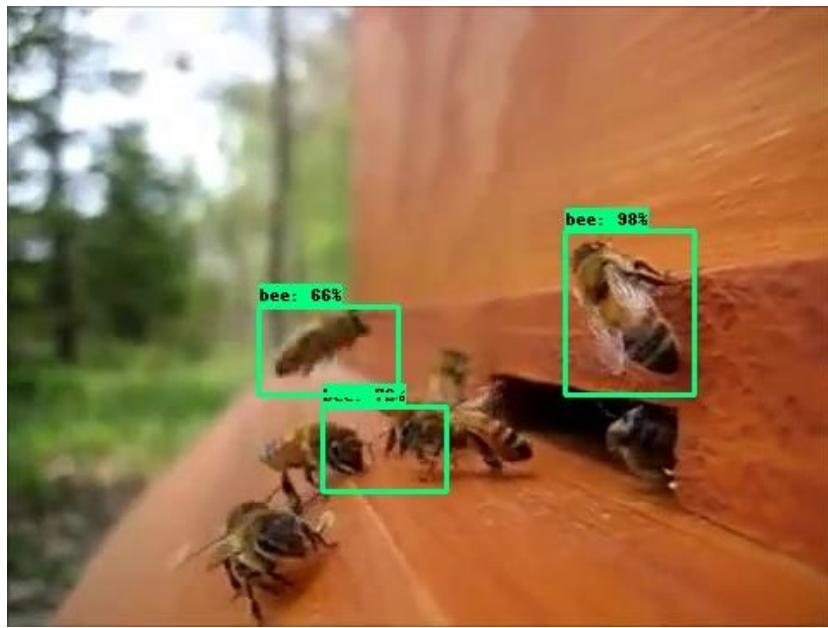


Figura 4.11 Detección de modelo ad hoc

### 4.2.1.2. iNaturalist Model

Una de las aplicaciones orientadas a la naturaleza más populares del mundo, iNaturalist ayuda a científicos alrededor del mundo a identificar plantas y animales de su entorno. Esto a través de una red mundial para compartir las observaciones individuales de sus miembros. El proyecto iNaturalist es una iniciativa conjunta de California Academy of Sciences y National Geographic Society (iNaturalist, 2019).

La plataforma iNaturalist, permite compartir observaciones de animales o insectos de distintos países, comúnmente las observaciones son fotografías tomadas del ser vivo en cuestión que en el momento son identificadas por quien las toma y validadas posteriormente por la comunidad. La gran cantidad de imágenes y el hecho de contar con una identificación previa, ha facilitado la realización de un modelo de inferencia, que permite la identificación automatizada de un conjunto de especímenes con un número suficiente de imágenes para realizar el entrenamiento. Apis Melífera se cuenta entre los especímenes identificables. La última versión del entrenamiento cuenta con 2854 especies identificables.

El modelo de iNaturalist, ha sido entrenado con millones de muestras provenientes de los 5 continentes, ciertamente un modelo bastante extenso. En materia de

## Capítulo 4. - Resultados

clasificación general, es bastante útil el uso de modelos tan grandes, sin embargo, para una aplicación en tiempo real y de objetos específicos como es el objeto del autor, un modelo de tales capacidades puede ser una gran carga para el sistema.

A pesar de contar con la información para la clasificación de abejas (*Apis Melífera*), el modelo arrojó detecciones poco satisfactorias, tanto en abejas obtenidas en campo como en fotografías de alta resolución, la Figura 4.12, fue un resultado representativo de las pruebas con este modelo, que al parecer, confunde *Apis Melífera* con una especie llamada flogofora meticulosa, la Figura 4.13 muestra el mismo caso sometido al modelo implementado para el proyecto, bajo las mismas condiciones de hardware y software.



Figura 4.12 Imagen de abeja. Detección y porcentaje arrojado por iNaturalist model

## Capítulo 4. - Resultados

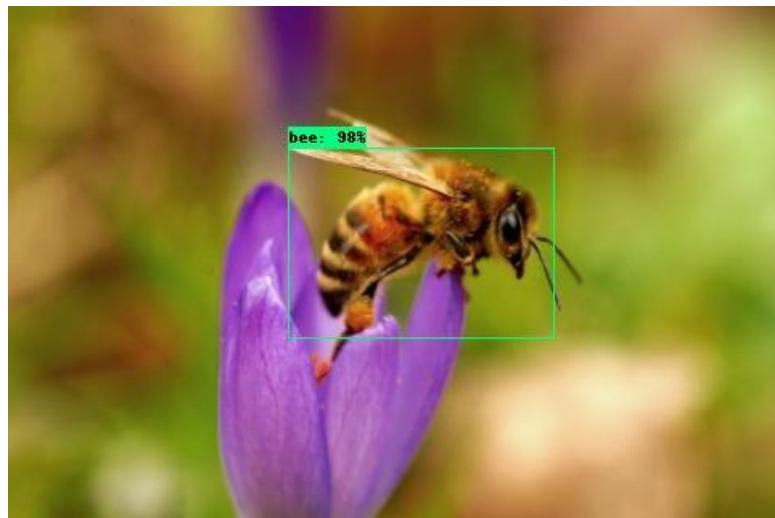


Figura 4.13 Detección de abeja y porcentaje de similitud arrojado por el modelo del proyecto

La desventaja principal del uso de modelos tan extensos para aplicaciones en tiempo real, es el consumo de recursos y la duración de cada iteración. Las pruebas realizadas se efectuaron en imágenes estáticas, sin embargo, el caso idóneo es la detección en video, no obstante, los tiempos del modelo, no dieron cabida para pruebas en un flujo continuo. La Figura 4.14 muestra las diferencias de tiempo entre los dos modelos, bajo las mismas condiciones de hardware y software, los tiempos por iteración del modelo iNaturalist en minutos, imposibilitan el modelo para aplicaciones en tiempo real, sin embargo, para trabajos científicos de clasificación general de especies sigue siendo una herramienta importante.

Cada iteración toma alrededor de un minuto mediante el extenso modelo de iNaturalist. Las pruebas con el modelo entrenado para el proyecto presente tomaron alrededor de medio segundo. Un modelo ad-hoc, con objetos bien establecidos como objetivo de la detección, optimiza el tamaño del modelo y su ejecución, lo cual es un factor importante en dispositivos de pocas capacidades.

## Capítulo 4. - Resultados

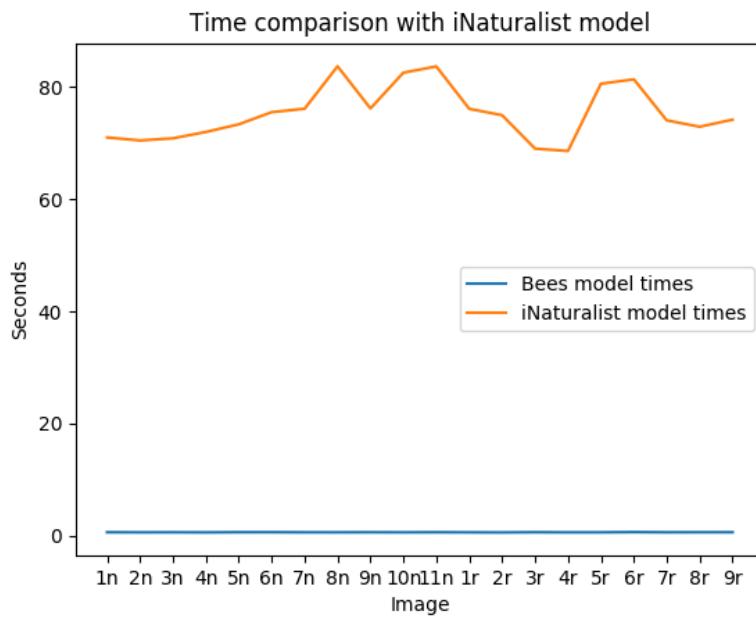


Figura 4.14 Comparativa de tiempo por iteración

### 4.2.1.3. Reconocimiento de color

La técnica de reconocimiento de color tiene muy marcadas ventajas y desventajas. Una de sus principales ventajas es ser sencilla de implementar y de bajo costo de procesamiento, una ventaja esencial para los dispositivos con un procesamiento muy limitado o que comparten recursos con otros procesos más demandantes, sin embargo, una de sus principales desventajas es que los colores, desde el punto de vista computacional, pueden estar sujetos a cambios de luz, una gran desventaja si se habla del funcionamiento en un medio natural poco controlado. La gama de amarillos según los sistemas RGB o HSL que se usan comúnmente en estos algoritmos, puede pasar por alto muchos amarillos naturales que se vean distorsionados por la cantidad de luz o incluso por el reflejo de la luz de otros objetos en el entorno, así como también puede dar falsos negativos con objetos amarillos que se presenten en el radio de observación del periférico utilizado para la captura de imágenes.

En las figuras Figura 4.15 y Figura 4.16 se puede observar la detección simple de amarillos en imágenes de abejas tomadas en alta resolución, a pesar de que un color característico de las abejas es el amarillo, este amarillo en el entorno natural

## Capítulo 4. - Resultados

puede llegar a ser muy polifacético y pasarse por alto desde la definición computacional (RGB) de la gama de amarillos.

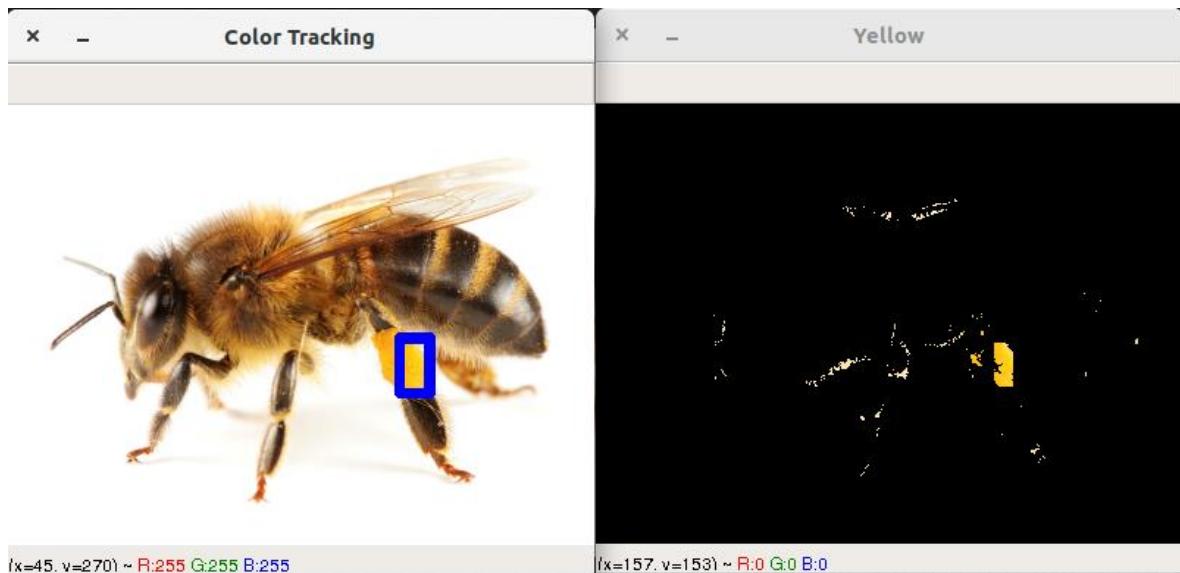


Figura 4.15 El amarillo detectado en la abeja es el mínimo

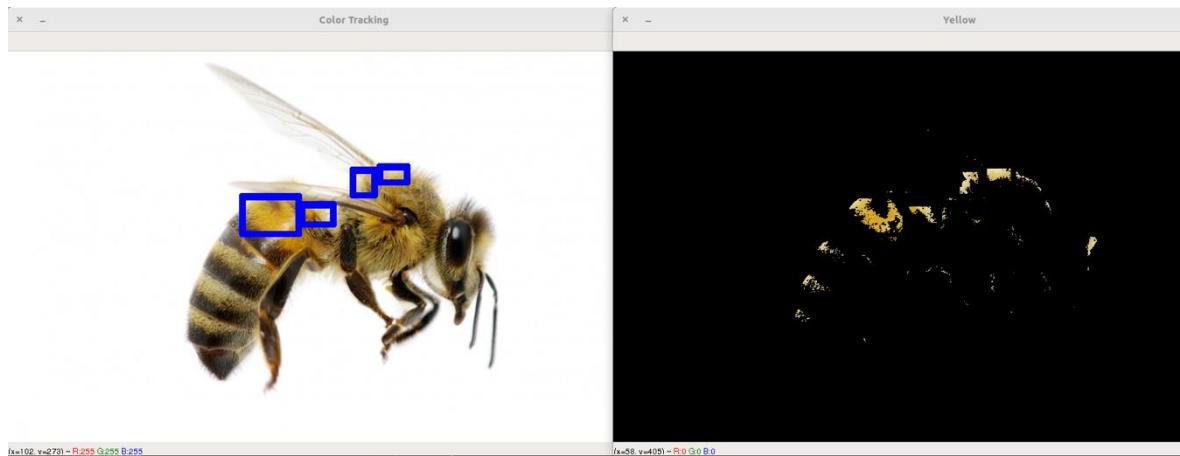


Figura 4.16 Debido a la menor cantidad de luz se aprecia una mejor reconstrucción de la abeja

Otro inconveniente de este método es el entorno, en el cual pueden haber objetos del mismo color a detectar y, ya que el algoritmo carece de la capacidad de discernir entre los objetos amarillos encontrados, reconstruir también los objetos ajenos encontrados, tal como se en la Figura 4.17, la madera café mediante una adecuada cantidad de luz puede pasar como amarillo y el amarillo natural de una abeja verse afectado por la misma razón

## Capítulo 4. - Resultados



Figura 4.17 El entorno también se clasifica

Uno de los principales objetivos de la clasificación y más tarde de la detección de objetos fue, precisamente evitar este tipo de inconvenientes de exposición a la luz y en cierta medida del ángulo en que estuviera el objeto al momento de la captura de imagen.

Tal como se puede apreciar en la Figura 4.18, Figura 4.19 y Figura 4.20 la luz, la posición del objeto y en cierta medida su entorno, no deberían ser un problema si el objeto en cuestión se encuentra en la imagen.

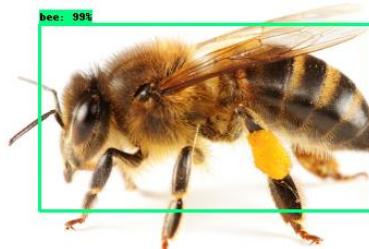


Figura 4.18 Abeja localizada mediante aprendizaje de máquinas

## Capítulo 4. - Resultados

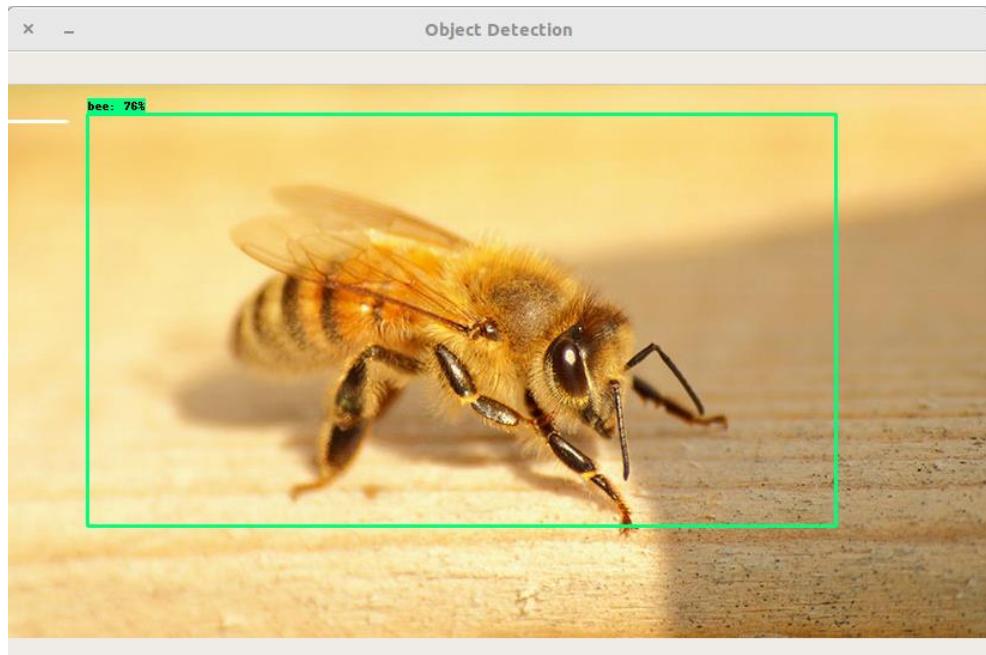


Figura 4.19 Discernimiento entre objetos y entorno

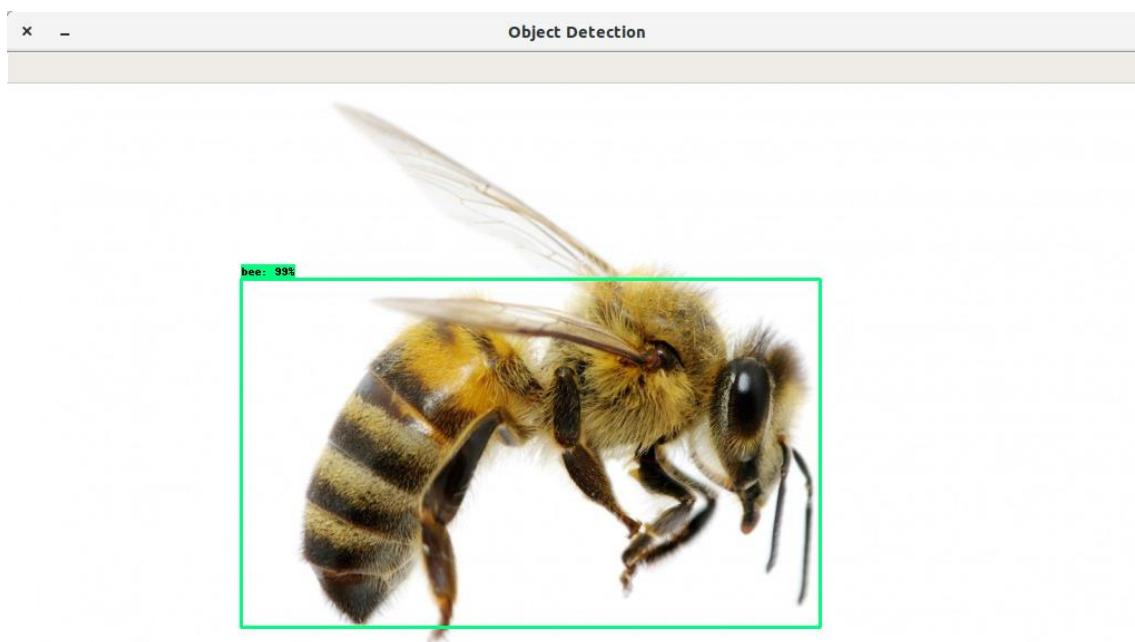


Figura 4.20 A pesar de los cambios de luz sigue detectado ya que se basa en las características

Ahora, pasando ya a un entorno totalmente natural, la detección de colores es un procedimiento más difícil de llevar a cabo. Hasta ahora se mostraron imágenes en alta resolución que permiten ubicar a simple vista los diferentes colores de las abejas, no obstante, esas imágenes aun pertenecen a un entorno controlado de

## Capítulo 4. - Resultados

captura y tener una cámara de alta definición en cada panel, que enfoque a ese nivel a cada abeja sería muy costoso. Las siguientes imágenes muestran un entorno natural poco controlado, fueron tomadas en las incursiones a campo realizadas en el presente proyecto. La Figura 4.21, Figura 4.22 y Figura 4.23 muestran los resultados de la detección de color en un ambiente no controlado, con luz ambiental normal (de 10000 a 20000 lux) y en una resolución de imagen de 480x540.

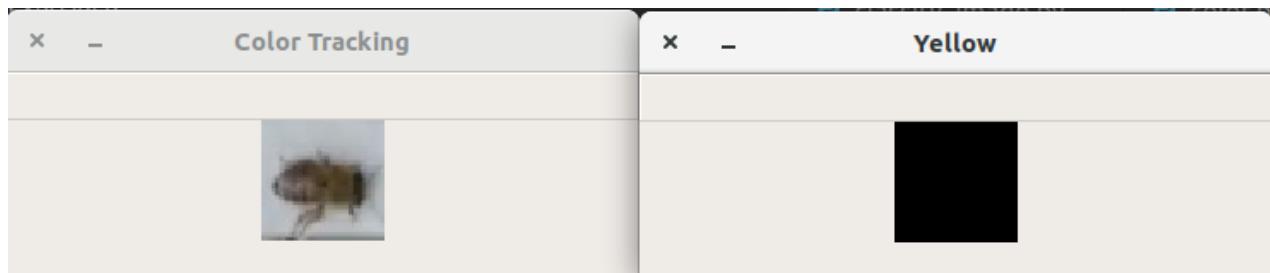


Figura 4.21 Imagen 1 de detección de color en ambiente no controlado

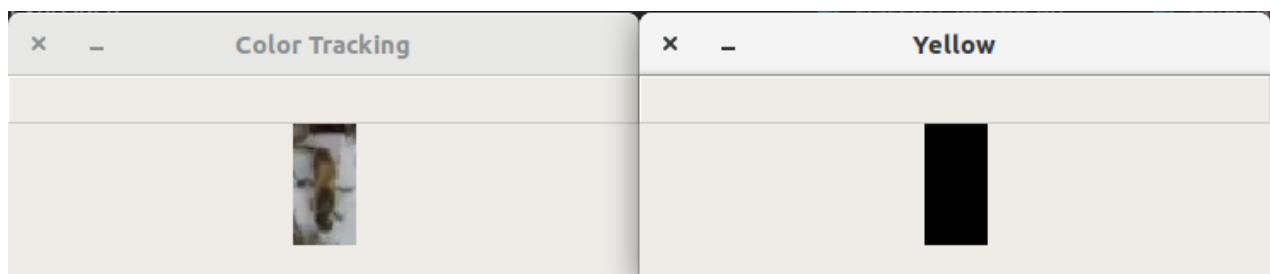


Figura 4.22 Imagen 2 de detección de color en ambiente no controlado

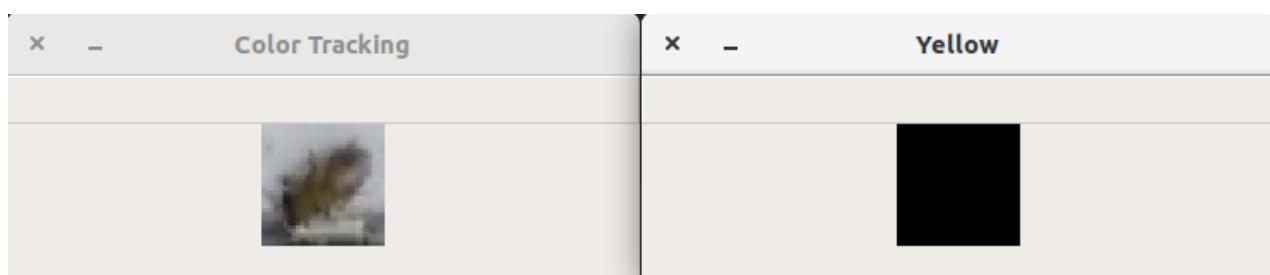


Figura 4.23 Imagen 3 de detección de color en ambiente no controlado

La Figura 4.24, Figura 4.25 y Figura 4.26 muestran la detección mediante aprendizaje de máquinas, bajo la misma configuración y ambiente que las 3 imágenes que se acaban de analizar.

## Capítulo 4. - Resultados



Figura 4.24 Detección de varias abejas en ambiente no controlado

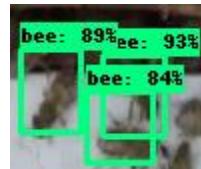


Figura 4.25 Detección de abejas en ambiente no controlado



Figura 4.26 Detección individual en ambiente no controlado

Las imagen original de la detección puede apreciarse en la Figura 4.27.



Figura 4.27 Imagen base de los ejemplos anteriores

### 4.2.1.4. Clasificación ImageNet

ImageNet Large Scale Visual Recognition Challenge es un reto para la comunidad programadora, que tiene como la realización de un modelo capaz de detectar miles de objetos, entrenado con millones disponibles en red (Olga Russakovsky, 2014).

## Capítulo 4. - Resultados

Viene organizado por el proyecto Image Net que es un dataset global de imágenes, dedicado a ofrecer grandes volúmenes de imágenes realizando búsquedas sencillas (Stanford Vision Lab, Stanford University, Princeton University , 2016). Nace de la creciente necesidad global de grandes datasets para proyectos de visión por computadora. El clasificador resultante del reto lanzado en 2014, es el aplicado como clasificador en la sección presente y en la siguiente.

La clasificación, como método para detectar y diferenciar objetos en imágenes, ha sido ampliamente usado como se pudo observar en el capítulo 2. Un método eficiente incluso para diferenciar entre varias sub-ramas del objeto en cuestión. No obstante, es también un método demandante de recursos de procesamiento y memoria lo cual se ve reflejado en el tiempo que toma en realizarse la operación por completo. Los clasificadores, fueron y siguen siendo un método reconocido y ampliamente utilizado en aplicaciones científicas. Sin embargo, su mayor limitación es el uso de imágenes estáticas cuyo enfoque principal sea el objeto a encontrar, aunado a esto, un clasificador no puede definir la localización del objeto en la imagen, contar los objetos ni arrojar un resultado final acerca de varios objetos si estos no son el enfoque principal de la imagen.

Como se puede ver en la Figura 4.28 y Figura 4.29, un clasificador no es tan dependiente de su entorno o los niveles de luminosidad, sin embargo, arroja el resultado deseado como una interpretación de la imagen sin incluir la localización de la detección .

## Capítulo 4. - Resultados

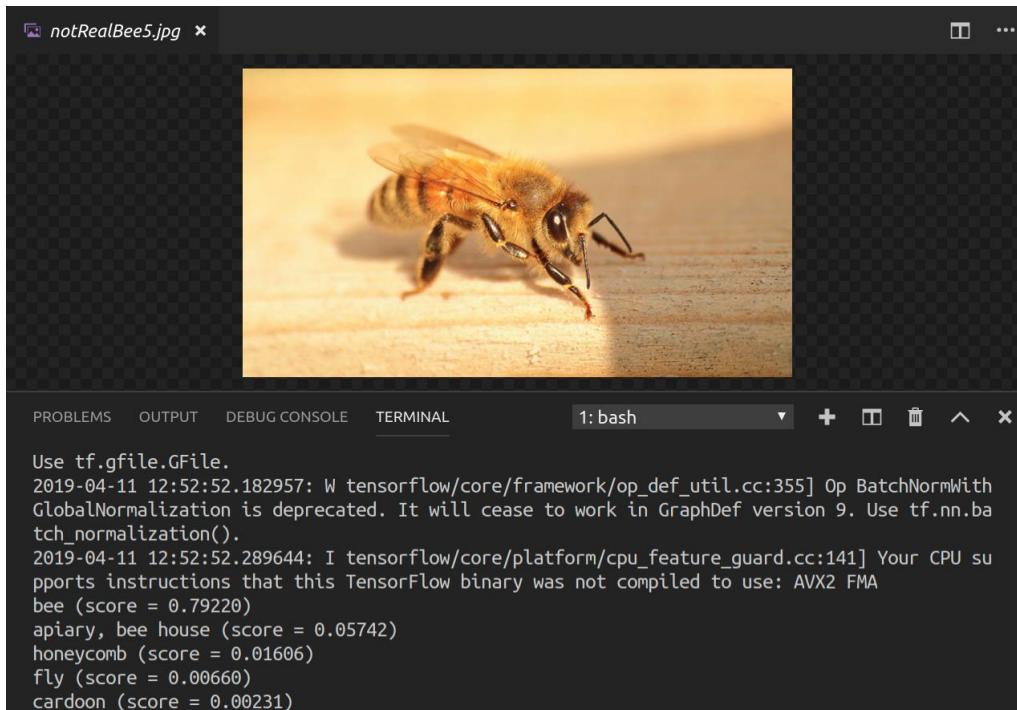


Figura 4.28 Clasificaciones arrojadas y porcentaje de similitud

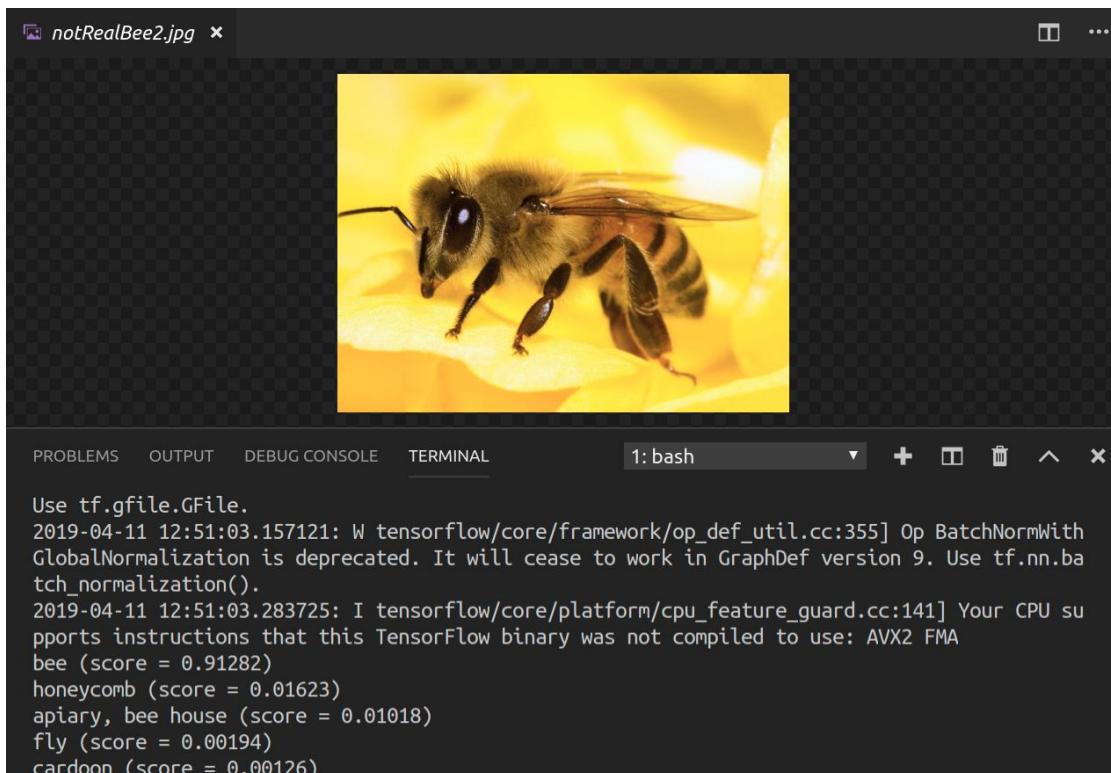


Figura 4.29 El entorno no confunde la clasificación siempre que el objeto ocupe un plano principal

## Capítulo 4. - Resultados

Pueden obtenerse resultados similares mediante la detección de objetos, el cual, para el presente caso, ofrece la ventaja de la localización de la detección (ver Figura 4.30) y la capacidad de interpretar múltiples detecciones, aun cuando los objetos no se encuentran en el plano principal de la imagen (ver Figura 4.31)



Figura 4.30 El algoritmo delimita la zona de detección

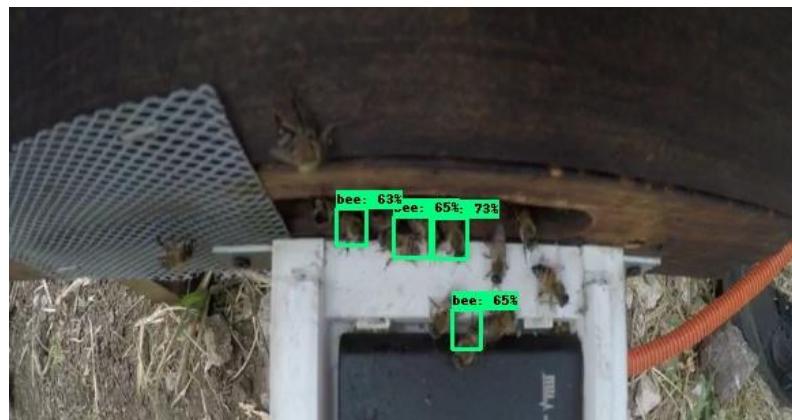


Figura 4.31 Múltiples detecciones con su porcentaje de similitud cada una

Como preámbulo a la siguiente sección y en comparativa con la Figura 4.32, para aplicar el clasificador en la figura mencionada, deberían establecerse los individuos como objeto central de una nueva imagen y someterse al análisis de algoritmo. Sin embargo, puede haber inconvenientes. La toma y manejo de imágenes de gran calidad desde dispositivos de pocas capacidades puede ser un proceso riguroso para los dispositivos y el hecho de contar con una calidad baja en las imágenes, puede

## Capítulo 4. - Resultados

confundir al clasificador. En la Figura 4.33 y Figura 4.34 se puede apreciar el resultado del aislamiento (manual) y clasificación de dos abejas que aparecen en la Figura 4.32.

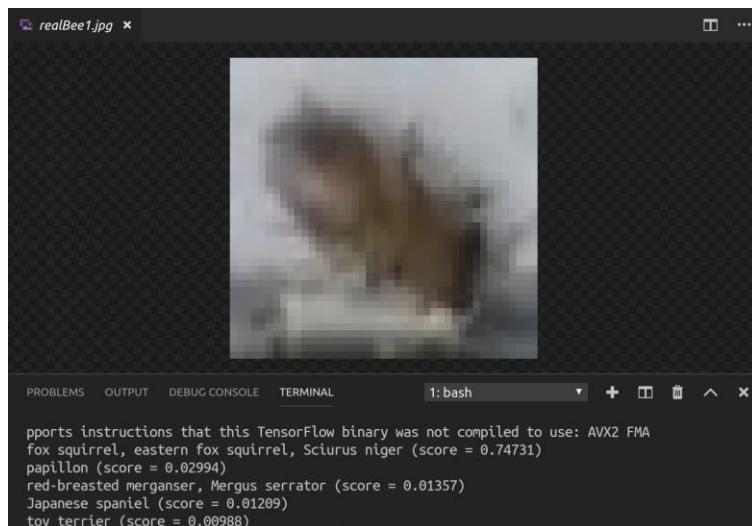


Figura 4.32 Abeja aislada y clasificada

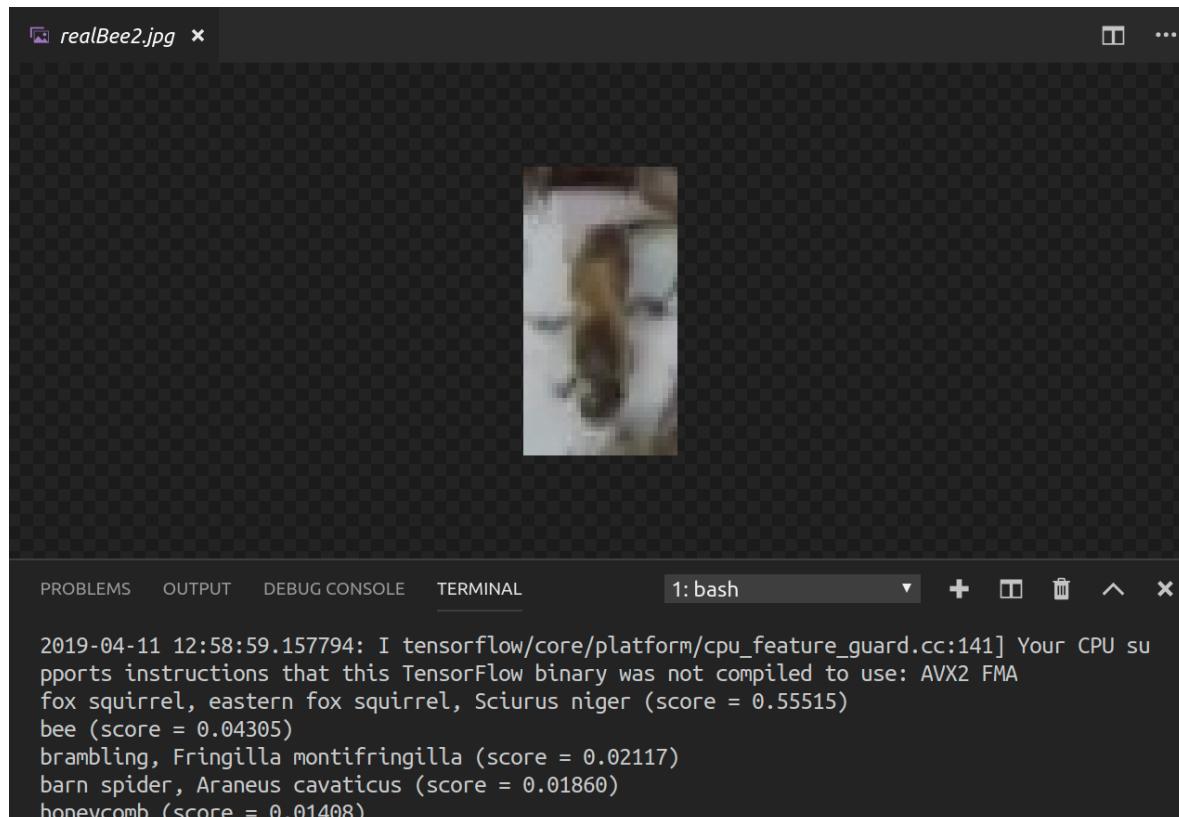


Figura 4.33 Mejor resultado obtenido con clasificador

#### 1.1.1.1. Detección de movimiento y clasificación

Anteriormente se explicó el funcionamiento, ventajas, desventajas y comparativa con el trabajo presente de un clasificador, ahora se expondrá una comparativa de un intento de uso de un clasificador en tiempo real.

Visto el potencial de la clasificación, un intento común fue la realización de algoritmos que apoyaran el desempeño de esta técnica para asimilar a un algoritmo en tiempo real. Uno de estos algoritmos fue la detección de movimiento. Como se explicó en el capítulo 2, la idea general es sencilla, se detecta movimiento mediante algún algoritmo como background subtraction, se aisla la zona de la detección, se generaba un archivo temporal de imagen y este se mandaba al proceso de clasificación, simple y eficiente a simple vista, no obstante, la práctica ha demostró la necesidad de algoritmos más eficientes.

La siguiente comparativa se llevó a cabo con imágenes de prueba, donde los objetos a localizar (abejas) se distinguen mejor que en las imágenes de campo. Se utilizó substracción de fondo para detectar movimiento, por ser un algoritmo simple pero eficaz. El algoritmo sigue los pasos ya descritos de ese tipo de trabajos. Rodea dentro de un círculo el área donde se concentra el cambio de pixeles que se considera como movimiento tal como se aprecia en la Figura 4.34 y Figura 4.35.

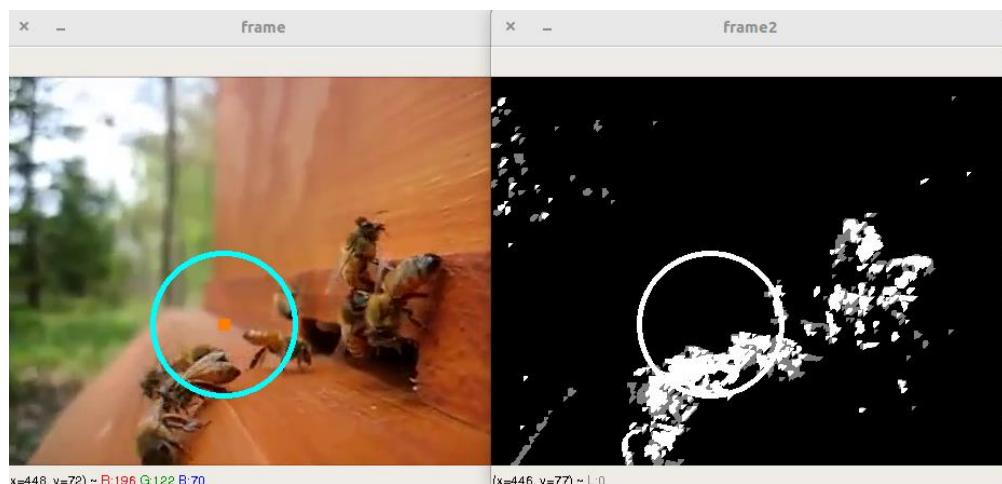


Figura 4.34 Aislamiento de movimiento en un círculo (pocas abejas)

## Capítulo 4. - Resultados

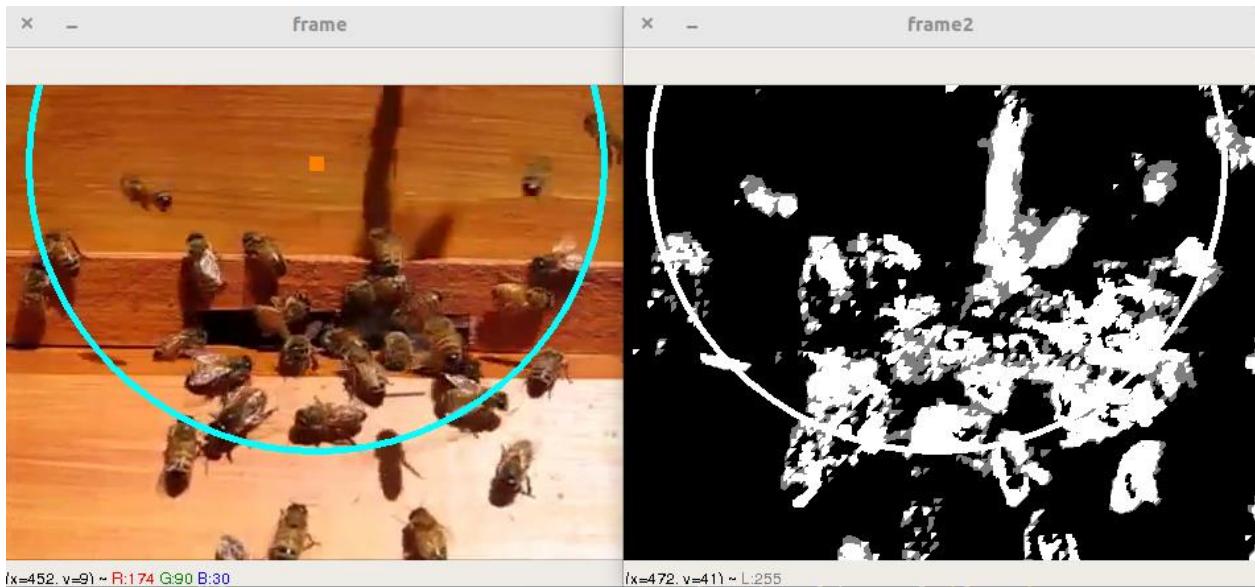


Figura 4.35 Circunferencia rodeando área grande de detección de movimiento

A continuación se aísla, el área detectada en un archivo temporal como se puede observar en la Figura 4.36, esta imagen cuenta ya con las características apropiadas para ser enviada al clasificador, y este a su vez produzca un resultado medianamente favorable, como se puede ver en la Figura 4.37.



Figura 4.36 Se aísla el movimiento detectado

## Capítulo 4. - Resultados

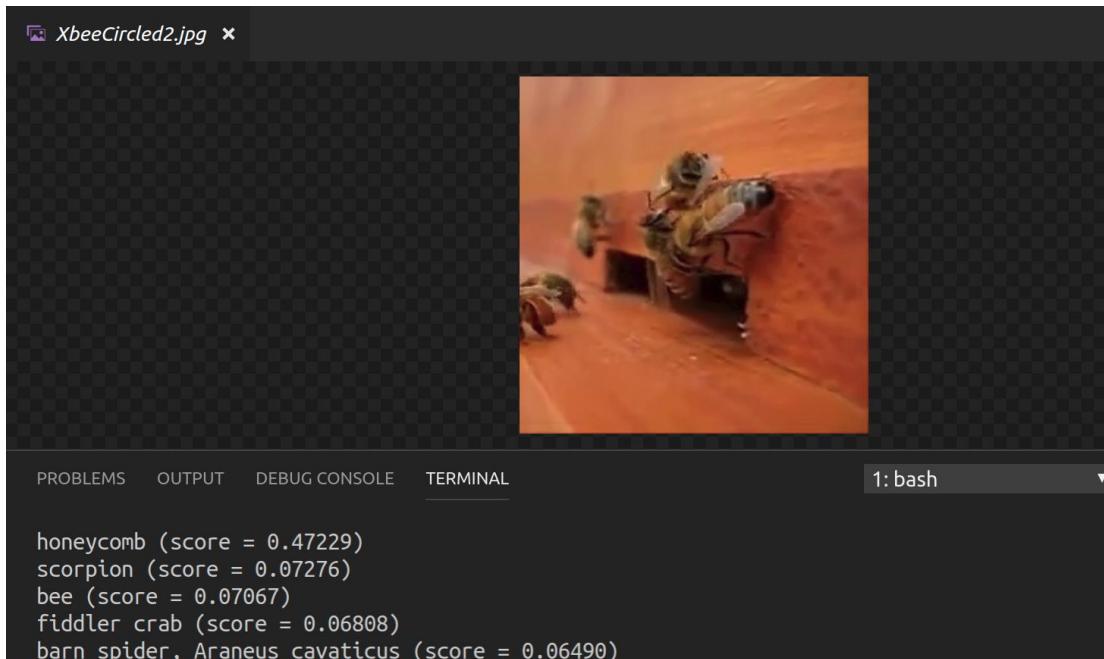
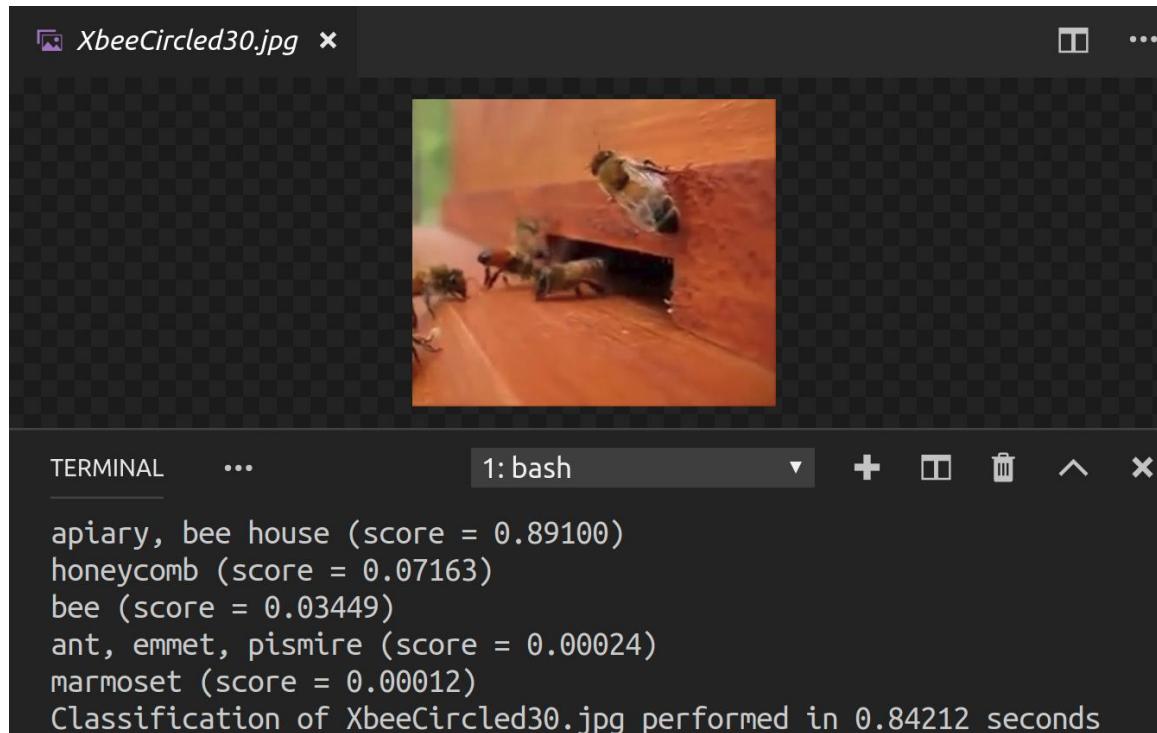


Figura 4.37 Clasificación del aislamiento de movimiento

El principal problema radica en la eficiencia del método afectada principalmente por el número de veces que se tiene que realizar la clasificación y el tiempo que esta toma realizarse. Como puede observarse en la Figura 4.38, Figura 4.39 y Figura 4.40, el tiempo que toma el proceso de clasificación oscila entre entre 0.8 segundos y 1.2 segundos, efectuado este tiempo por cada una de las detecciones de movimiento a analizar, las detecciones de varias abejas individuales tomarían bastante tiempo y el conteo individual de abejas agrupadas sería más difícil aun. Este algoritmo puede proveer un acercamiento a la localización del objeto ya detectado, sin embargo, como se vio en figuras anteriores es una aproximación con un considerable margen de error.

El proceso de la convolución mejora los tiempos presentados en la clasificación como se puede apreciar en la Figura 4.41, los cuales oscilan desde 0.72 segundos a 0.87 segundos, pero tomando en cuenta que el proceso se lleva a cabo una vez por imagen, y devuelve varias detecciones, los cuales son puntos cruciales para el manejo de la gran actividad de los apiarios, el incremento de la eficiencia es notable.

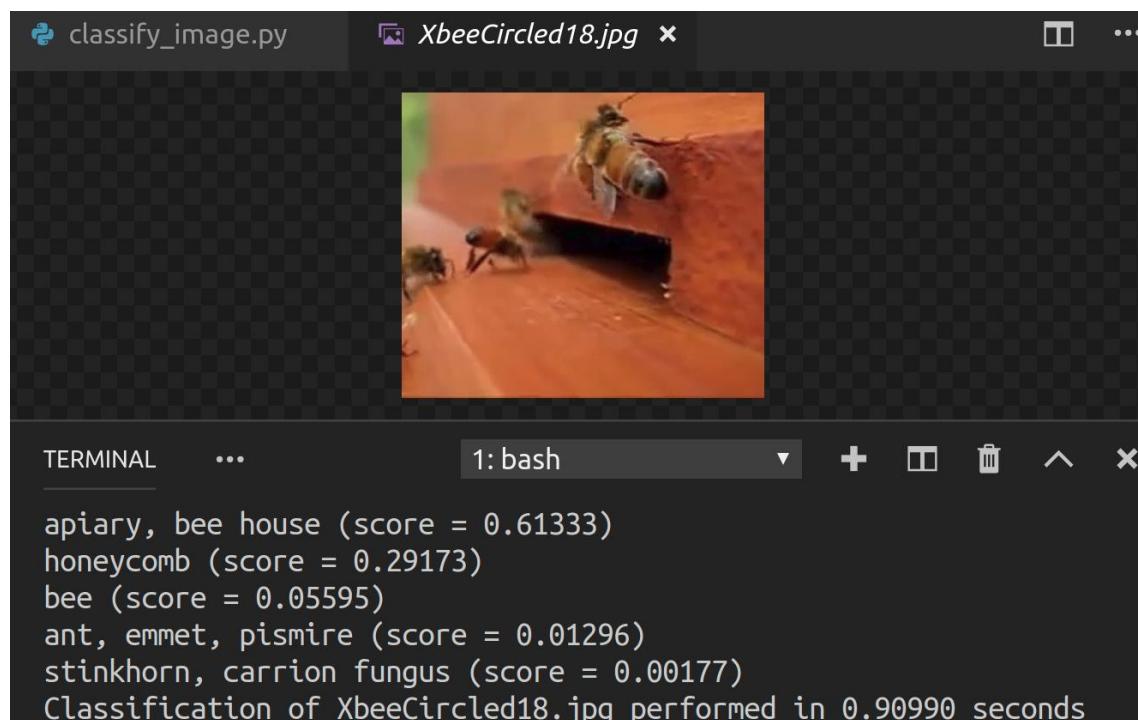
## Capítulo 4. - Resultados



The screenshot shows a terminal window with the following interface elements:

- Top bar: A file icon, the filename "XbeeCircled30.jpg", a close button, and three vertical dots.
- Image preview: A small thumbnail of the image showing several bees on a wooden surface.
- Terminal tab: "1: bash" with a dropdown arrow.
- Icons: A plus sign, a square, a trash can, a double up arrow, a double down arrow, and an X.
- Output area:
  - Text: "TERMINAL ..."
  - Text: "1: bash"
  - Classification results:
    - apiary, bee house (score = 0.89100)
    - honeycomb (score = 0.07163)
    - bee (score = 0.03449)
    - ant, emmet, pismire (score = 0.00024)
    - marmoset (score = 0.00012)
  - Text: "Classification of XbeeCircled30.jpg performed in 0.84212 seconds"

Figura 4.38 Mejor tiempo de clasificación

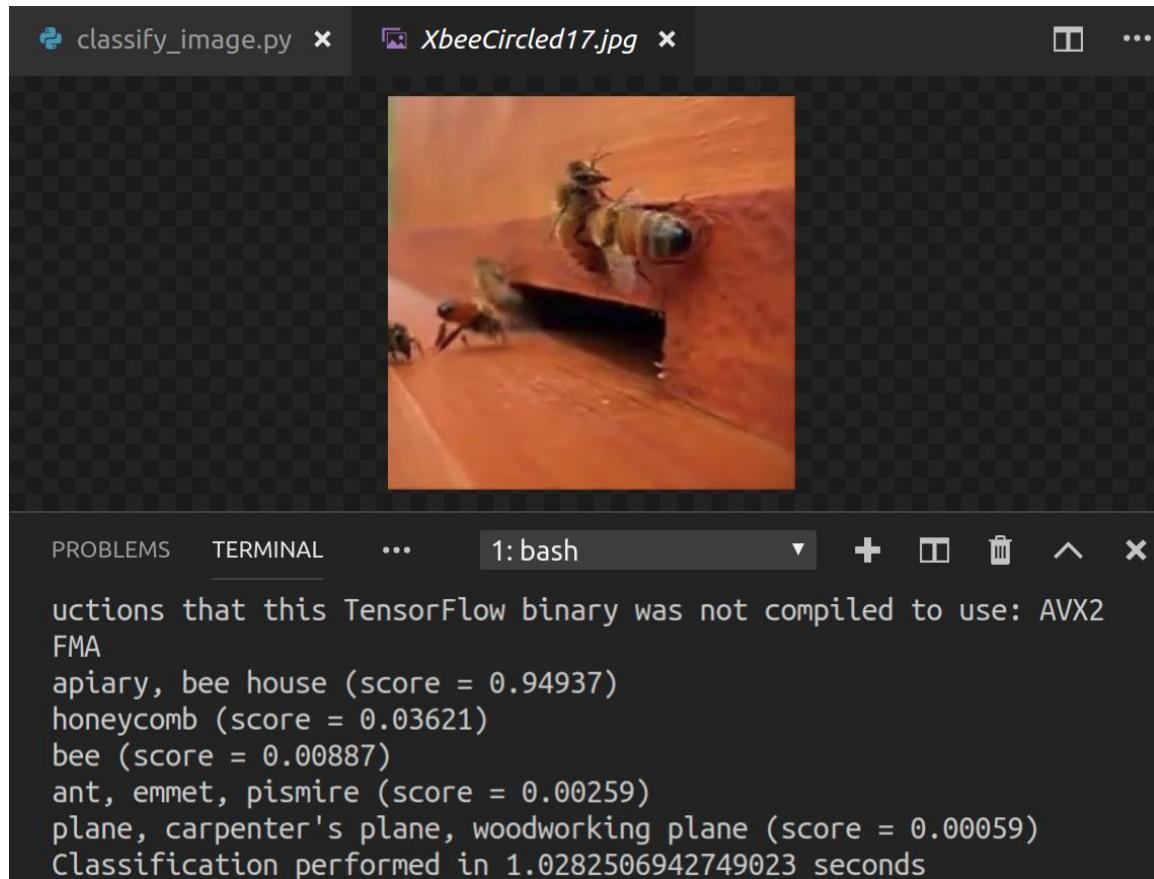


The screenshot shows a terminal window with the following interface elements:

- Top bar: A file icon, the filename "XbeeCircled18.jpg", a close button, and three vertical dots.
- Image preview: A small thumbnail of the image showing several bees on a wooden surface.
- Terminal tab: "1: bash" with a dropdown arrow.
- Icons: A plus sign, a square, a trash can, a double up arrow, a double down arrow, and an X.
- Output area:
  - Text: "classify\_image.py ..."
  - Text: "1: bash"
  - Classification results:
    - apiary, bee house (score = 0.61333)
    - honeycomb (score = 0.29173)
    - bee (score = 0.05595)
    - ant, emmet, pismire (score = 0.01296)
    - stinkhorn, carrion fungus (score = 0.00177)
  - Text: "Classification of XbeeCircled18.jpg performed in 0.90990 seconds"

Figura 4.39 Tiempo de detección más común

## Capítulo 4. - Resultados



The screenshot shows a terminal window with two tabs: "classify\_image.py" and "XbeeCircled17.jpg". The "XbeeCircled17.jpg" tab displays a photograph of a bee on a wooden surface. Below the terminal tabs, there are icons for PROBLEMS, TERMINAL, and a menu. The main terminal area contains the following text:

```
PROBLEMS TERMINAL ... 1: bash + □ ⚡ ^ ×
uctions that this TensorFlow binary was not compiled to use: AVX2
FMA
apiary, bee house (score = 0.94937)
honeycomb (score = 0.03621)
bee (score = 0.00887)
ant, emmet, pismire (score = 0.00259)
plane, carpenter's plane, woodworking plane (score = 0.00059)
Classification performed in 1.0282506942749023 seconds
```

Figura 4.40 Peor tiempo de detección

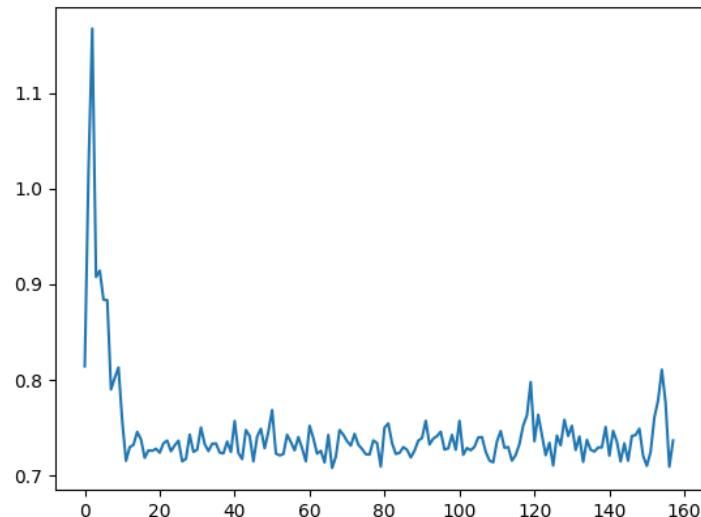


Figura 4.41 Gráfica de tiempos efectuados en 160 procesos de convolución completos

#### 4.2.2. Complementación RFID-Aprendizaje de máquinas

Desde el punto de vista poblacional, una detección automatizada por imágenes puede alcanzar cifras más cercanas al extenso número poblacional con el que cuentan la mayoría de panales en buen estado y aplicando algoritmos de pre-processing puede enviarse al servidor un número de detecciones significativo y limpio desde campo, aprovechando las características de procesamiento de los sistemas embebidos, sin embargo, el uso de la tecnología de reconocimiento individual, como en el presente caso es el RFID, puede dar un observación de una pequeña parte de la población, con mayor profundidad, ya que el método de detección de imágenes no permite diferenciar abejas entre sí. Teniendo esto en cuenta el nivel individual de observación, podría dividirse en 2 subniveles, observación social y observación particular.

La observación social, puede mostrar el comportamiento general de la población, por ejemplo, horarios de más actividad, recuento de entradas y salidas, épocas de mayor proliferación, asenso o descenso poblacional, el objetivo de esta subsección es ayudar a la comprensión de la situación generalizada de una colmena.

La observación particular, se enfoca solo en una pequeña muestra de la población, la cual es identificable y diferenciable tanto al interior de la muestra como dentro de toda la población. Ayudando así, por ejemplo, a la comprensión a profundidad del estilo de vida de los individuos, que tan frecuentemente sale a recolección una forrajera determinada, cuánto dura la recolección, etc. El objeto primordial de este apartado es la comprensión del estilo de vida, y la identificación y reconocimiento de un individuo en cuestión.

Así, ambas tecnologías, pueden complementarse cubriendo las debilidades una de la otra y contribuir a la mejor comprensión de la colmena. El método de detección individual, lejos de quedar obsoleto u opacado por un método de detección masivo, es complementado por este, y a su vez el método de detección masiva viene complementado por la identificación individual. Cada uno cubriendo las debilidades del otro con sus propias fortalezas.

# Capítulo 5. Conclusiones

El presente capítulo hace una síntesis del trabajo en general, discutiendo aportes y avances del mismo, así como marcando una ruta para los aportes futuros y el constante uso y actualización de la tecnología involucrada.

## 5.1. Resumen

Las bases de la investigación y una breve explicación de la situación crítica de las abejas a nivel mundial se discutieron en el capítulo 1, se expuso el objetivo principal del trabajo y la hipótesis a probar, así como una descripción sencilla de la solución propuesta. El capítulo 2, mostró las soluciones más relevantes a través del tiempo y el desarrollo de la tecnología que se han aplicado a la observación apícola y a algunas otras áreas biológicas. Se discutieron soluciones tradicionales para el monitoreo de variables en el apiario, tal como temperatura, humedad, velocidad del viento entre otras, y se dio una visión general de como el tratamiento de imágenes por computadora ha evolucionado y mejorado con el tiempo, y como este compite con los métodos electromecánicos más comunes. El capítulo 3, permitió abarcar el proceso de implementación de una manera profunda, explicar cada una de las partes de la arquitectura basada en IoT propuesta para el proyecto y comprender tanto el proceso interno de aprendizaje de máquinas mediante redes neuronales convolucionales como su papel dentro de la arquitectura integrada con IoT y su uso en el tratamiento de imágenes. Las comparativas con otros distintos métodos de tratamiento de imágenes fueron plasmadas en el capítulo 4, donde pudo verse el gran problema que representa el entorno para los métodos de tratamiento de imágenes tradicionales y como puede mejorarse la eficiencia de un sistema haciendo uso del aprendizaje de características. Se discutió también las ventajas que ofrece IoT frente al envío de datos tradicional por FTP y la flexibilidad que ofrece la transferencia de información mediante internet de las cosas.

## 5.2. Conclusiones Finales

Las conclusiones se exponen en desde dos puntos generales, la implementación de IoT y el procesamiento de imágenes.

### 5.2.1. Implementación IoT

El presente proyecto demuestra que la tecnología IoT, puede ser utilizada para observación biológica, el potencial del paradigma aporta elementos y facilidades de tiempo, distancias y automatización únicas.

La tecnología IoT, protocolos y las propiedades únicas del paradigma, permiten una mayor eficiencia y flexibilidad en el paso de mensajes. Permiten un empaquetado de la información mucho más ligero, concentrándose más en el envío de mensajes importantes que en el envío de grandes archivos lo cual optimiza la comunicación y es un factor decisivo al contar con recursos limitados de comunicación.

La tecnología 2G, aun en locaciones con poca recepción, probó ser un medio adecuado para el paso de mensajes mediante protocolos de IoT. Ni la integridad de los datos ni los tiempos de constante envío se vieron afectados.

### 5.2.2. Procesamiento de imágenes

El aprendizaje de máquinas, como herramienta para la automatización de conteo, requiere un importante porcentaje de las capacidades de procesamiento de los sistemas embebidos para el análisis por cuadro, sin embargo, el flujo de imágenes y su tratamiento probó ser una herramienta factible para la optimización del conteo y observación poblacional.

Puede optimizarse un modelo realizando entrenamientos ad-hoc, con imágenes y características deseadas en la detección, de esta manera se reducen la complejidad del modelo y puede centrarse el entrenamiento en objetos más específicos.

Un punto de partida importante en el desarrollo de soluciones que trabajen con imágenes es, analizar el problema y decidir que método se centra en las características que se desean monitorear, tanto el aprendizaje de máquinas como

## Capítulo 5. - Conclusiones

el tratamiento de imágenes mediante filtros pueden ofrecer características decisivas en la toma de decisión para el curso de un proyecto.

Las redes neuronales y la flexibilidad de lenguajes interpretados como Python, han permitido llevar métodos de inteligencia artificial (en el caso presente aprendizaje de máquinas) a dispositivos de capacidades muy limitadas, haciendo capaces a estos últimos de tomar decisiones y enfrentarse a distintos problemas de la vida real de una manera más optimizada, al contar con un panorama más amplio y herramientas más capaces a la hora de discernir en el mundo real. La monitorización bien puede ser el comienzo de este proceso de aprendizaje.

### 5.2.3. Trabajo futuro

- Comparar eficiencia del sistema con el uso único de sensores digitales.
- Aplicar más técnicas de “fog computing” en los agentes, principalmente sobre los datos que se obtienen de los sensores, este método permitirá realizar parte de proceso de caracterización en los agentes, y la parte que requiere más procesamiento en el servidor.
- Añadir el control remoto de los agentes a la plataforma existente para la visualización de los datos.
- Desarrollar una aplicación de realidad aumentada destinada a trabajar en los apíarios.
- Desarrollar una aplicación móvil para visualización de la información.
- Utilizar aprendizaje de máquinas y redes neuronales para el entrenamiento de modelos capaces de caracterizar la información obtenida de los sensores, un modelo entrenado de aprendizaje, que pueda discernir sobre condiciones benéficas o contradictorias en las colmenas, por ejemplo, futuro crecimiento desmedido o condiciones que demuestren debilidad o enfermada en la misma.

## Capítulo 5. - Conclusiones

- Desarrollo de redes neuronales recursivas que aprendan de los datos secuenciales arrojados por los sensores y envase a modelo de conocimiento que se rigen y al entorno que presenta la información predecir el funcionamiento normal y alertar cuando la diferencia sea considerable.
- Desarrollo de sistema que permita el aprendizaje no supervisado, que basado en las frecuencias de las muestras dadas permita clasificar de manera sencilla los patrones presentados en la información, esto puede aplicarse para caracterizar el comportamiento por zonas geográficas, por tipos de suelos donde se encuentren los apiarios e incluso el impacto de las condiciones climáticas presentadas con más frecuencia en la zona del apiario.

# Capítulo 6. Bibliografía

- Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., . . . Jia, Y. (9 de Noviembre de 2015). *tensorflow.org*. Obtenido de download:  
<http://download.tensorflow.org/paper/whitepaper2015.pdf>
- Aleksejs, Z., Valters, B., Juris, M., & Egils, S. B. (30 de Diciembre de 2014). Challenges in the development of Precision Beekeeping. *ScienceDirect*, págs. 60-71.
- Astha, T. (agosto de 2018). *digitalcommons.usu.edu*. Obtenido de  
<https://digitalcommons.usu.edu/cgi/viewcontent.cgi?article=8185&context=etd#page=31&zoom=100,0,129>
- Barrera, J. A.-T. (s.f.). *cucei.udg.mx*. Recuperado el 11 de septiembre de 2016, de  
[http://www.cucei.udg.mx/sites/default/files/pdf/toral\\_barrera\\_jamie\\_areli.pdf](http://www.cucei.udg.mx/sites/default/files/pdf/toral_barrera_jamie_areli.pdf)
- BETHANY S. TEETERS, R. M. (2012). USING VIDEO-TRACKING TO ASSESS SUBLETHAL EFFECTS OF PESTICIDES ON HONEY BEES (APIS MELLIFERA L.). *Environmental Toxicology and Chemistry*, 1349-1354.
- Bonilla, M. A. (2012). Capítulo I: abejas. Universidad Nacional de Colombia. En *La polinización como servicio ecosistémico*. En: *Iniciativa colombiana de polinizadores (ICPA)* (págs. 1-103). Bogotá.
- Bortolotti, L., Porrini, C., Mutinelli, F., Daniele, P., Enzo, M., Carlotta, B., . . . Anna Gloria, S. (2009). Salute delle api: analisi dei fattori di rischio. Il progetto Apenet. En L. Bortolotti, *APOidea Vol. 6* (págs. 3-22).
- Bromenshenk, J., Seccomb, R., Rice, S., & Etter, R. (2004). *Estados Unidos Patente nº 6,910,941*.
- Bromenshenk, J., Smith, G., King, B., Seccomb, R., Alnasser, A., Henderson, C., . . . Wrobel, C. (1996). *C.L. New and improved methods for monitoring air quality and the terrestrial environment*. Springfield, VA.
- Bulacio, N. V. (16 de Marzo de 2018). *redlac-af.org*. Obtenido de <https://www.redlac-af.org/single-post/2018/03/16/%C2%A1Alerta-Monitoreo-de-Varroa>
- Campbell, J., Mummet, L., & Sukthankar, R. (2008). Video monitoring of honey bee colonies at the hive entrance. *Visual Observation and Analysis of Animal and Insect Behavior*, 1-4.
- Castañeda-Vildózola, A., Equihua-Martínez, A., Valdés-Carrasco, J., Barrientos-Priego, A. F., Ish-Am, G., & Gazit, S. (1999). INSECTOS POLINIZADORES DEL AGUACATERO EN LOS ESTADOS DE MÉXICO Y MICHOACÁN. *Revista Chapino Serie Horticultura*, 129-136.

## Capítulo 6. - Bibliografía

- Chautá-Mellizo, A., Campbel, S., Bonilla, M., Thaler, J., & Poveda, K. (2012). Effects of natural and artificial pollination on fruit and offspring quality. En A. Chautá-Mellizo, S. Campbel, M. Bonilla, J. Thaler, & K. Poveda, *Basic and Applied Ecology* (págs. 524-532).
- Chiu, C., En-Cheng, Y., Joe-Air, J., & Ta-Te, L. (2012). An imaging system for monitoring the in-and-out activity of honey bees. *Computers and Electronics in Agriculture*, 100-109.
- Cisco Networking Academy. (2018). Introduction to IoT.
- Commonwealth Scientific and Industrial Research Organisation (CSIRO). (25 de Agosto de 2015). CSIRO calls on researchers worldwide to join forces to save honey bees. Australia.
- Dagher, P. (19 de Enero de 2016). Classifying Bees With Google TensorFlow.
- Daoust, M. (2017 de Noviembre de 2017). *premade\_estimator*. Obtenido de [github.com/tensorflow/models/blob/master/samples/core/get\\_started/premade\\_estimator.py](https://github.com/tensorflow/models/blob/master/samples/core/get_started/premade_estimator.py)
- Diana L., C.-F., Sean, C., Edward C., H., Gustavo, P., Jay D., E., & Nancy A., M. (2007). A Metagenomic Survey of Microbes in Honey Bee robes in Honey Bee . *Science*, 283-287.
- Engelke, U., Marendy, P., Nguyen, T., De Souza Junior, P., & Williams, R. (2014). *research.csiro.au*. Recuperado el 9 de Noviembre de 2018, de <https://research.csiro.au/ci/projects/vizzbees/>
- Fabiana S. Santana, A. H. (2014). A reference process for automating bee species identification based on wing images and digital image processing. *Ecological Informatics*, 248-260.
- Favre, D. (2011). Mobile phone-induced honeybee worker piping. *Apidologie*, 270-279.
- Francisco Padilla Álvarez, J. M. (2011). La embrazón. *El colmenar*, 24-31.
- Fung, E. (2017). *digital.library.adelaide.edu.au*. Obtenido de <https://digital.library.adelaide.edu.au/dspace/handle/2440/111954>
- Gang Jun Tu, M. K. (2016). Automatic behaviour analysis system for honeybees using computer vision. *Computers and Electronics in Agriculture*, 10-18.
- García, P. P. (2013). *eprints.ucm.es*. Obtenido de <https://eprints.ucm.es/23444/1/ProyectoFinMasterPedroPablo.pdf>
- Gates, B. N. (1914). *The temperature of the bee colony*. United States Department of Agriculture.
- Groh, C., Tautz, J., & Rossler., W. (2004). Synaptic organization in the adult honey bee brain is influenced by brood–temperature control during pupal development. *Proceedings of the National Academy of Sciences of USA*.

## Capítulo 6. - Bibliografía

- halweb.uc3m.e. (s.f.). Recuperado el 11 de septiembre de 2018, de  
[halweb.uc3m.es/esp/Personal/personas/jmmarin/esp/DM/tema3dm.pdf](http://halweb.uc3m.es/esp/Personal/personas/jmmarin/esp/DM/tema3dm.pdf)
- Harst, W., Kuhn, J., & Stever, H. (2006). *agbi.uni-landau.d*. Obtenido de [http://agbi.uni-landau.de/material\\_download/IAAS\\_2006.pdf](http://agbi.uni-landau.de/material_download/IAAS_2006.pdf)
- Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., . . . Adam, H. (2017 de Abril de 2017). MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. Obtenido de arXiv.
- Human, H., Brodschneider, R., Dietemann, V., Dively, G., Ellis, J. D., & Forsgren, E. (2013). Miscellaneous standard methods for *Apis mellifera* research. *Apicultural Research*, 1-55.
- IBM Corporation. (22 de Noviembre de 2016). Deep Learning with tensorflow.
- Jurijs, M., Aleksejs, H., & Egils, S. (2009). AUTOMATIC MICROCLIMATE CONTROLLED BEEHIVE OBSERVATION SYSTEM. *Jelgava*, 265-271.
- Kerr, H., Buchanan, M., & Valentine, K. (1989). *Estados Unidos Patente nº 4,876,721*.
- Khan, R., Khan, S., Zaheer, R., & Khan, S. (2012). Future Internet: The Internet of Things Architecture, Possible Applications and Key Challenges. *2012 10th International Conference on Frontiers of Information Technology*, (págs. 257-260).
- Kulincevic, J. M., Rothenbuhler, W. C., & Rinderer, T. E. (1982). Disappearing disease. 1. Effects of certain protein sources given to honeybee colonies in Florida. *American Bee Journal*, págs. 91-189.
- LANASE. (2015). *Área de Investigación: Declive de polinizadores*. Obtenido de LANASE ENES UNAM: [www.lanase.unam.mx/es/proyectos\\_investigacion.php?ra=1](http://www.lanase.unam.mx/es/proyectos_investigacion.php?ra=1)
- LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE* 86, 2278–2324.
- Mathworks. (2018). *mathworks.com*. Recuperado el 13 de Marzo de 2019, de Long Short-Term Memory Networks: <https://www.mathworks.com/help/deeplearning/ug/long-short-term-memory-networks.html;jsessionid=90f36f62f88858c40a9e64807927>
- Matich, D. J. (Marzo de 2001). [www.frro.utn.edu.ar](http://www.frro.utn.edu.ar/repositorio/catedras/quimica/5_anio/orientadora1/monografias/matich-redesneuronales.pdf). Obtenido de [https://www.frro.utn.edu.ar/repositorio/catedras/quimica/5\\_anio/orientadora1/monografias/matich-redesneuronales.pdf](http://www.frro.utn.edu.ar/repositorio/catedras/quimica/5_anio/orientadora1/monografias/matich-redesneuronales.pdf)
- McKinsey Global Institute. (2017). *cecpport.com*. Obtenido de <http://img.cecpport.com/mediaCms/pdf/201507/15141040mcg6.pdf>
- Meikle, W., Rector, B., Mercadier, G., & Holst, N. (2008). Within day variation in continuous hive weight data as a measure of honey bee colony activity. *Apidologie*, 694-707.

## Capítulo 6. - Bibliografía

- Miao, W., Ting-lie, L., Fei-Yang, L., Jing, S., & Hui-Ying, D. (2010). Research on the architecture of Internet of things . *International Conference on Advanced Computer Theory and Engineering(ICACTE)* .
- Nates-Parra, G. (2005). Abejas silvestres y polinización. *Manejo Integrado de Plagas y Agroecología*, 7-20.
- Noldus. (2018). [www.noldus.com](http://www.noldus.com). Obtenido de <https://www.noldus.com/animal-behavior-research/products/ethovision-xt>
- Oak Ridge National Laboratory . (1989). [www.nytimes.com/](http://www.nytimes.com/1988/10/25/science/chip-to-track-killer-bees-is-invented.html). Obtenido de <http://www.nytimes.com/1988/10/25/science/chip-to-track-killer-bees-is-invented.html>
- Octavio, A., Márquez, R., Ángela, A. M., Sebastian, E. G., & Giovanni, B. R. (2012). Beekeeping Monitoring Module. *Beekeeping Monitoring Module*. Bogotá D.C., Bogotá D.C., Colombia.
- Olga Russakovsky, J. D.-F. (1 de Septiembre de 2014). *arXiv*. Obtenido de <https://arxiv.org/abs/1409.0575>
- OpenCV. (2018). *About*. Obtenido de [opencv.org](http://opencv.org): [opencv.org/about.html](http://opencv.org/about.html)
- PostgreSQL. (4 de Agosto de 2005). *postgresql.org*. Obtenido de <https://www.postgresql.org/about/>
- PTC University. (2017). *Unpacking the Internet of Things*. Obtenido de <https://www.ptcu.com/course/course-view/st/4/15/192>
- Quesada Avedaño, M. (2010). *INFORME FINAL DEL PROYECTO “EVALUACIÓN DE LOS IMPACTOS DEL CAMBIO CLIMÁTICO EN POLINIZADORES Y SUS CONSECUENCIAS POTENCIALES EN EL SECTOR AGRÍCOLA EN MÉXICO”*.
- Schneider, C., Tautz, J., Grünwald, B., & Fuchs, S. (11 de Enero de 2012). *journals.plos.org*. Obtenido de RFID Tracking of sublethal effects of two neonicotinoid insecticides on the foraging behavior of *Apis mellifera*.: [10.1371/journal.pone.0030023](https://doi.org/10.1371/journal.pone.0030023)
- Stanford Vision Lab, Stanford University, Princeton University . (2016). *image-net.org*. Obtenido de <http://image-net.org/about-overview>
- Stankus, T. (11 de Octubre de 2008). A Review and Bibliography of the Literature of Honey Bee Colony Collapse Disorder: A Poorly Understood Epidemic that Clearly Threatens the Successful Pollination of Billions of Dollars of Crops in America. *Journal of Agricultural & Food Information*, págs. 115-143.
- Steuer, H., Harst, W., Kimmel, S., Kuhn, J., Otten, C., & B, W. (2007). *agbi.uni-landau.de*. Obtenido de [http://agbi.uni-landau.de/material\\_download/elmagexp\\_bienen\\_06.pdf](http://agbi.uni-landau.de/material_download/elmagexp_bienen_06.pdf)

## Capítulo 6. - Bibliografía

- Struye, M., Mortier, H., Arnold, G., Miniggio, C., & Borneck, R. (1994). Microprocessor-controlled monitoring of honeybee flight activity at the hive entrance. *Apidologie*, 384-395.
- Thomas D. Seeley, P. K. (2006). Group desicion making in honey bee swarms. *American Scientist*, 220-229.
- Tom, A., Stefan, S., Volker, S., & Dieter, W. (2001). Biodiversity Informatics in Action: Identification and Monitoring of Bee Species using ABIS. *EnviroInfo 2001: Sustainability in the Information Society*, 425-430.
- Tsung-Yi, L., Michael, M., Serge, B., Lubomir, B., Ross, G., James, H., . . . Piotr, D. (2015). Microsoft COCO: Common Objects in Context. *arxiv*.
- United States Department of Agriculture. (2017). The Importance of Pollinators. Recuperado el 30 de Enero de 2019, de  
[https://www.nrcs.usda.gov/wps/portal/nrcs/detail/pa/plantsanimals/?cid=nrcs142p2\\_018171](https://www.nrcs.usda.gov/wps/portal/nrcs/detail/pa/plantsanimals/?cid=nrcs142p2_018171)
- van Rossum, G. (2009 de Enero de 2009). *python-history.blogspot.com*. Obtenido de  
<http://python-history.blogspot.com/2009/01/brief-timeline-of-python.html>
- VerdeGen. (6 de Abril de 2016). *generacionverde.com*. Obtenido de  
[generacionverde.com/blog/ambiental/importancia-de-la-polinizacion/](http://generacionverde.com/blog/ambiental/importancia-de-la-polinizacion/)
- W., R. (2007). UCSF sleuths identify suspects in mystery of vanishing honeybees. *UCSF Today*.
- Wario, F., Wild, B., Couvillon, M. J., Rojas, R., & Landgraf, T. (2015). Automatic methods for long-term tracking and the detection and decoding of communication dances in honeybees. *Frontiers in Ecology and Evolution*.
- Weiguang, D., & Graham, T. (24 de Febrero de 2016). *arxiv.org*. Obtenido de  
<https://arxiv.org/pdf/1602.07383.pdf>
- Woods, E. (1957). *Estados Unidos Patente nº 2,806,082*.

# Capítulo 7. Apéndices

A continuación, se presentará un listado de las principales herramientas utilizadas en el desarrollo del presente proyecto, comenzando por la implementación de la red de sensores y terminando con la implementación del modelo de aprendizaje.

## 7.1. Red de sensores

### 7.1.1. Componentes de software

**Python:** lenguaje de programación interpretado, lenguaje de implementación del servidor y el software en los agentes.

**Paho-mqtt:** comunicación mediante protocolo MQTT de internet de las cosas, mediante esta herramienta se publican valores en el bróker y se suscribe el software a distintos temas.

**Raspbian:** sistema operativo basado en Linux, sistema operativo de los dispositivos Raspberry pi, utilizado como sistema base para los agentes en campo.

**Linux:** sistema operativo utilizado en los dispositivos Intel edison, sistema base para los agentes en campo.

**Pyupm:** librería de alto nivel para obtención de datos desde sensores analógicos.

**PostgreSQL:** SGBD, en el cual se almacena información local dentro de los agentes e información centralizada en el servidor.

### 7.1.2. Componentes de hardware

**Raspberry Pi:** sistema basado en Linux, utilizado en el presente proyecto como una de las plataformas base para los agentes en campo.

**Intel Edison:** sistema basado en Linux y arduino, utilizado en el presente proyecto como una de las plataformas base para los agentes en campo.

**Fotómetro, sensor de luz.**

**Sonómetro, sensor de sonido.**

**Vibrómetro, sensor de vibraciones.**

**Acelerómetro.**

**Termómetro ambiental, sensor de temperatura externa.**

**Termómetro, sensor de temperatura interna.**

**Higrómetro, sensor de humedad**

**Estación meteorológica (anemómetro y pluviómetro)**

**Balanza, sensor de peso para las colmenas.**

## 7.2. Aprendizaje de máquinas

### 7.2.1. Componentes de software

**Tensorflow:** plataforma libre para aprendizaje de máquina. Incluye el diseño de algoritmos y estructuras (por ejemplo, la estructura de una neurona) utilizadas para el desarrollo y entrenamiento de modelos de inferencia. Mediante esta plataforma se desarrolló la CNN (red neuronal convolucional) que se usó en el presente proyecto, desglosada más a detalle en la sección 3.1 Campo, infraestructura en los apiarios.

**OpenCV:** librería que compila numerosos algoritmos de visión por computadora, aunado a tensorflow, permite una fácil visualización de los resultados de las detecciones mediante el modelo entrenado.

**Python:** lenguaje de programación interpretado, lenguaje de la interfaz utilizada para el manejo de tensorflow y opencv, además de los dispositivos de hardware se expondrán a continuación.

**Matplotlib:** biblioteca de generación de gráficos, utilizada en la comparación de los resultados de los distintos métodos en la sección Capítulo 4.- Resultados

## Capítulo 7. - Apéndices

**Blender:** diseño de gráficos en 3D, utilizado para el modelo de la piquera.

### 7.2.2. Componentes de hardware

**Cámara:** utilizada como medio de recolección de imágenes. Incrustada en la parte superior interior de la piquera.