

딥러닝을 활용한 말벌퇴치 스마트 양봉 시스템

김다현, 김중현, 김영탁
영남대학교 기계 IT대학 정보통신공학과
(우) 38541 경북 경산시 대학로 280

kdhyun8011@naver.com, kjhn9@naver.com, ytkim@yu.ac.kr

요 약

양봉농가에서 꿀 생산량을 감소시키는 가장 큰 요인은 말벌의 공격이다. 기존에 존재하는 달콤한 용액으로 유인하는 상용화된 말벌 포획용품이나 모션감지 센서로 모션이 감지되면 카메라의 영상을 분석, 크기로 말벌임을 판단하여 살상하는 특허 장치로는 실질적으로 말벌을 직접 선별하여 퇴치하지 못해 효과적으로 말벌에 대처할 수 없는 것으로 분석되었다. 본 논문에서는 머신러닝 기술을 활용하여 실시간 카메라 영상을 분석해 입구로 들어오는 개체가 꿀벌인지 말벌인지 구분하는 말벌 퇴치 시스템을 제안한다. 제안된 시스템에서는 입구로 들어오는 개체가 말벌로 판별되면 말벌 덫을 작동시켜 말벌을 덫에 가두고 전기충격을 주어 말벌을 죽인다. 또한 양봉업자가 자리를 비운 사이 말벌의 침입에 대비할 수 있도록 사용자 스마트폰 앱에서 실시간 영상을 확인할 수 있으며 말벌이 나타나면 버튼을 눌러 말벌을 퇴치할 수 있다. 제안된 시스템은 실제 “말벌”만을 구분해냄으로써 직접 말벌임을 구별할 수 없는 이전의 방식에 비해 훨씬 효과적으로 말벌에 대처할 수 있다. 본 연구는 오랫동안 양봉농가에서 해결하지 못한 말벌의 공격에 대해 최신 IT 기술인 딥러닝을 활용하여 대처함으로써 비교적 IT기술과 많이 결합되지 못한 농업기술에 최신 IT기술을 접목시켜 자동화 시키는 것에 중요한 의미를 두고있다.

1. 서 론

말벌의 공격에 의해 발생하는 양봉농가의 피해는 연간 1750억원에 달한다[1]. 양봉농가에서는 이러한 말벌을 퇴치하기 위해 낮 시간 내내 벌통 근처에서 사람이 대기를 하며 말벌을 잡아 낸다. 특정한 상황에서 벌통 근처에 대기하고 있지 못할 경우에는 말벌의 공격에 전혀 대비를 할 수 없다.

이미 양봉농가에서는 말벌 퇴치를 위한 많은 방법들을 고안해 냈다. 덫에 달콤한 용액을 넣어 말벌을 유인하는 방식[2]과 꿀벌만 지나갈 수 있는 망으로 입구를 막는 등의 물리적인 차단방법부터 모션감지 센서를 이용해 움직임이 탐지되고, 움직임이 탐지되면 탐지된 영역의 영상에서 객체 크기를 분석하여 말벌인지 구분하여 그 부분에 레이저를 쏘아 말벌을 죽이는 방식까지 존재한다[3]. 하지만 이러한 방식들은 많은 한계점을 가지고 있다. 우선 덫에 달콤한 용액을 넣어 말벌을 유인하는 방식은 말벌이 시기별로 선호하는 먹이가 달라 완전히 말벌을 퇴치할 수 있

는 방식이 아닌 것으로 분석되었으며[4], 단순 꿀벌만 지나갈 수 있는 망으로 입구를 막는 물리적인 차단방식은 꿀벌에게 큰 스트레스를 주어 꿀 생산량의 감소로 이어지는 것으로 분석되었다. 또한 꿀벌통의 꿀을 노리는 일반 벌과 다르게 꿀벌 그 자체를 사냥하는 등검은 말벌에는 대응이 될 수 없다. 또한 모션감지 센서를 이용해 움직임이 탐지되면 영상을 분석하는 방식은 실제로 말벌만을 구별해내는 방식이 아니므로 오작동의 확률이 높다고 판단된다.

본 논문에서는 이러한 한계점들을 극복하고 실질적으로 말벌을 구별해내 죽이는 시스템을 고안하였다. 실시간으로 영상정보를 받아 이를 deep-learning input에 넣게 되면, 실시간 영상이 말벌인지 아닌지를 판별하게 되고, 단순 영상정보 객체의 크기정보로 판단하는 것이 아닌 인공지능을 활용해 말벌을 탐지하므로 기존 연구의 오작동률을 낮출 수 있을 것으로 보인다. 또한 말벌을 효과적으로 죽일 수 있는 전기충격을 사용함으로써 기존의 방법에 비해 훨씬 더 확실하게 말벌의 폐사를 이끌어낼 수 있다. 전기충격에 의한 꿀벌의 피해를 최소화 하기 위해 꿀벌의 높이보다는 높고, 말벌의 높이보다는 낮은 꿀벌 덫을 설계하였다. 말벌이 검출되어 덫이 움직여 말벌을 가둘 때 꿀벌과 같이 갇히더라도, 크기가 작은 꿀벌은 전기충격을 받지 않고 크기가 큰 말벌만 전기충격을 받도록 덫을 고안하였다.

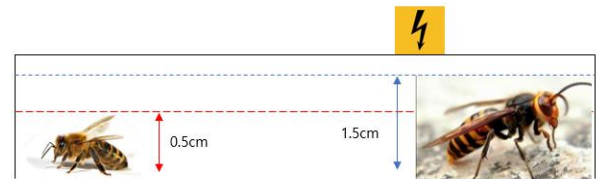


그림 1. 말벌만 영향을 받는 덫

또한 단순 말벌의 공격에만 대비하는 기존 연구와는 달리 양봉업자가 벌통을 확인하고 싶을 경우 안드로이드 어플리케이션을 활용해 실시간 영상정보를 받아볼 수 있다. 이를 통해 양봉업자는 양봉 농가를 계속해서 지키지 못할 상황이 왔을 때 외부에서 수시로 앱의 영상정보를 확인이 가능하다. 말벌이 검출되면 인공지능기능과는 별도로 사용자가 직접 앱의 버튼을 눌러 말벌을 퇴치하는 기능 또한 제공된다. 그리고 온도에 예민한 꿀벌들의 벌통 내부 온도를 안드로이드 앱으로 전송해 줌으로써 양봉업자는 실시간으로 온도정보를 확인할 수 있게 된다.

이 시스템을 사용하게 될 경우, 효과적으로 사람이

직접 벌통 근처에 대기하지 않고 말벌을 퇴치함으로써 양봉 농가의 수고를 덜어주고, 사람이 항상 벌통 근처에 대기할 수 없다는 한계점을 극복할 수 있다. 이로 인해 말벌에 의한 양봉농가의 피해를 최소화할 수 있고 이는 꿀 생산량의 증대로 이어질 것으로 보인다.

이 논문은 다음과 같이 구성되어 있다. 2절에서는 물리적인 말벌 차단방법과 단순 영상처리를 통한 관련 연구를 간략하게 소개하며, 이들 기존 방식의 문제점과 한계점에 대하여 설명한다. 3절에서는 본 논문에서 제안하는 딥러닝을 통한 말벌 인식 및 퇴치, 실시간 영상/온도 전송, App을 통한 Servo Motor Control에 대하여 상세하게 설명한다. 4절에서는 본 논문에서 제안하는 기술/기능의 기능 구현에 대하여 설명하며, 5절에서는 제안된 기술/기능의 성능을 측정하고 측정 결과를 분석한다. 6절에서 결론을 맺으며 향후 연구개발 계획에 대하여 간략하게 기술한다.

2. 관련 연구

2.1 꿀벌통을 이용한 말벌 포획기

기존의 말벌퇴치 시스템은 단순히 벌통의 입구를 막거나 외부에 말벌 유인 덫을 설치하는 식이며 이는 벌통 외부에서부터 꿀벌을 공격하기 시작하는 말벌의 특성상 퇴치에 큰 효과를 주지 못하는 것으로 분석된다[2].

2.2 벌통에 고정시켜 사용하는 말벌 살상장치

벌통의 하우스징 내부에서 물체 감지 센서에 움직임이 감지 되면 실시간 카메라를 이용해 물체의 크기를 분석 후 특정 크기 이상이면 말벌로 판단하여 레이저 장치를 이용해 살상하는 시스템은 특허가 존재한다. 지만 이는 물체감지 센서와 크기 분석을 통해 말벌인지를 판단하는 시스템으로써 완벽히 말벌인지를 구별하지 못하며, 다른 큰 물체가 들어올 경우에도 말벌로 인식할 수 있으며, 크기가 큰 여왕벌이 하우스징 안으로 들어갈 경우 여왕벌조차 말벌로 인식하여 공격할 우려가 있다[3].

3. 실시간 영상 말벌 검출 양봉 시스템

3.1 실시간 영상 말벌 검출 양봉 시스템의 기능 구조

그림2는 이 시스템의 기능블록도를 나타낸다. 위 기능블록도에서는 이 시스템 전체의 기능을 보여준다. 우선 라즈베리파이에서 실시간 영상정보를 Caffe 딥러닝 프레임워크로 구성된 Google Mobilenet V1 CNN의 input에 넣어 말벌인지 꿀벌인지 구별한다. 말벌로 판단되면 서보 모터를 구동시켜 말벌을 가두고 전기충격을 가해 죽인다. 그리고 실시간 영상정

보와 온도정보 라즈베리파이 V2 카메라와 DS18B20 센서를 통해 읽어와 게이트웨이 PC 서버로 전송하고, 게이트웨이 서버에서는 그를 사용자 Android App으로 전송해준다. 또 Android App에서 Control 버튼을 누르면 그 정보가 게이트웨이 서버로 전송된 후 게이트웨이 서버는 그 정보를 라즈베리파이로 전송해 서보 모터 및 전류 회로를 동작시킨다.

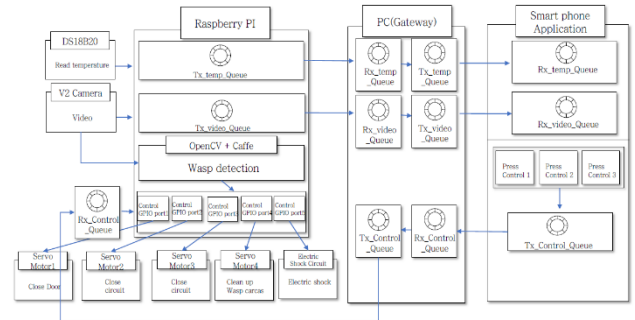


그림 2. 실시간 영상 말벌 검출 시스템 기능블록도

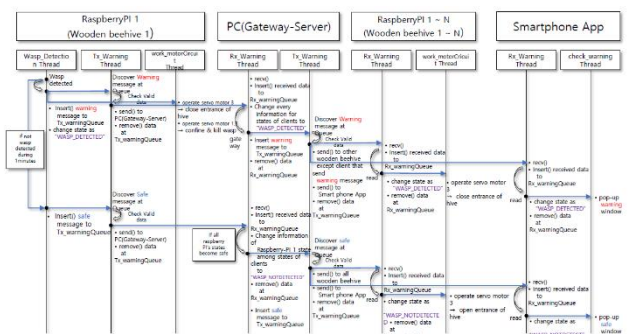


그림 3. 실시간 영상 말벌 검출 시스템 기능흐름도

그림 3은 이 시스템의 기능 흐름도를 나타낸다. 위 기능 흐름도에서는 이 시스템 전체의 시간에 따른 기능의 흐름을 나타낸다. 왼쪽부터 순서대로 라즈베리파이, PC (Gateway Server), 다른 라즈베리파이, User Application 순서이다. 우선 라즈베리파이에서 말벌이 감지되면 말벌을 죽인 뒤 20분 동안 wasp_warning 상태로 대기한다. 말벌이 죽었더라도 페로몬을 통해 동료들을 불러올 수 있기 때문에 빠른 공격을 위해 Wasp Warning 상태에서는 Servo Motor를 90도 각도로 세워 말벌 덫을 세워 둔다. 20분 이내로 또다른 말벌이 감지될 경우 타이머를 초기화하여 다시 20분동안 말벌 덫을 90도 각도로 세워둔다. 만약 20분 동안 말벌이 감지 되지 않는다면 서보모터를 다시 0도로 설정해 꿀벌의 동행에 방해가 되지 않도록 한다. 그리고 라즈베리파이에서 지속적으로 온도정보를 읽고, 게이트웨이 서버로 전송하며, 실시간 영상 또한 게이트웨이 서버로 전송한다. 게이트웨이 서버는 이들 정보를 User의 Android Application으로 전송해준다. User의 Android Application에서는 Button을 누르는 이벤트가 발생

하면 그 정보를 Gateway Server로 전송해주고, 그 정보는 라즈베리파이로 전송된다. 그 정보가 라즈베리파이에 도착하면 말벌 뗏을 180도로 움직여 말벌을 잡고, 이후 20분간 말벌 뗏을 세운 채 대기한다.

3.2 실시간 말벌 검출을 위한 영상검색 및 정보 전송의 의사코드

```

1  queue <cv::Mat> classification;
2  queue<cv::Mat> sendVideo;
3  void Real-timeVideo()
4  {
5      Time old_time;
6      Time new_time;
7      while(true)
8      {
9          cv::Mat realtimevideo;
10         realtimevideo ← getvideofromRaspberrypi();
11         if((new_time - old_time) == 3)
12         {
13             classification.push(realtimevideo);
14             sendVideo.push(realtimevideo);
15         }
16     }
17 }
18 end Real-timeVideo()

```

그림 4. 실시간 영상 → Classification & Send video Pseudo Code

그림 4는 실시간 영상을 Open CV Mat 구조체로 받아 그를 라즈베리 화면상에 띄워줌과 동시에 그 구조체를 Send Queue와 Classification Queue로 넣어주는 Pseudo code 이다. 3초에 한번씩 해당 영상 프레임을 각 큐에 enqueue 해준다.

```

1  extern queue <cv::Mat> classification;
2  void classification()
3  {
4      load model();
5      extern queue <cv::Mat> classification;
6      int count = 0;
7      while(true)
8      {
9          {
10             model.input() ← classification.front()
11             classification.remove();
12             string classifiedname ← getclassifiednamefromoutputlayer();
13             float probability ← Getprobabilityfromoutputlayer();
14             if(classifiedname=="Wasp"&&probability >=0.95)
15             {
16                 count++;
17                 if(count == 3)
18                 {
19                     killwasp();
20                 }
21             }
22         }
23     }
24 }
25 end classification

```

그림 5. Wasp / Bee Classification Pseudo Code

그림 5는 실시간영상으로부터 받은 Mat 구조체를 통해 해당 영상 프레임이 말벌인지 꿀벌인지 구별하는 코드이다. 만약 Classification Queue가 비어있지 않다면, 현재 큐에 판단해야 할 영상 프레임이 존재하는 것이므로 그 프레임을 미리 5000회 학습된 Google Mobilenet V1의 입력 레이어로 놓고 출력레이어로부터 Classification Name과 확률을 받는다. 만약 Classification Name이 “Wasp” 이고 그 확률이 0.95를 넘는다면, count를 올리고, 그 count가 3회 이상이라면 말벌을 죽인다. 3회를 판단하는 이유는 특정한 이유에서 단 한번 말벌일 확률이 95% 이상 나올 수 있기 때문이다. 실제 말벌이 들어 온 경우 세 영상 프레임 모두 95% 확률일 것이므로, 95% 확률이 세번 이상 나온 경우 말벌 뗏을 작동시킨다.

```

1  extern queue <cv::Mat> sendVideo;
2  void sendVideofunction()
3  {
4      char videoPacket[1020];
5      char videoframe[51000];
6      while(true)
7      {
8          if(sendVideo queue is not empty)
9          {
10             cv::Mat realtimevideo = sendVideo.front()
11             sendVideo.pop();
12             videoframe ← realtimevideo.RGB();
13             for(int i = 0 ; i < 50; i++)
14             {
15                 for(int j = 0 ; j < 1020; j++)
16                 {
17                     videoPacket[j] = videoframe[i*1020 + j];
18                 }
19                 send(videoPacket);
20             }
21         }
22     }
23 }
24 end sendvideofunction()

```

그림 6. SendVideo Pseudo Code

그림 6은 실시간 영상으로부터 받은 Mat 구조체를 RGB 170*100 크기로 변환 후 각 픽셀의 RGB 값을 받아 1차원 char 배열에 넣는다. 그 배열을 1020Byte씩 Segmentation해 Gateway Server로 전송하게 된다.

3.3 스마트 양봉 벌통 구조 및 기능

우선 말벌이 벌통을 공격할 때 입구의 꿀벌들로 인해 곧바로 진입할 수 없기 때문에 벌통 입구 근처에 착륙하여 그 근처의 꿀벌부터 공격하기 시작한다. 이 특성을 활용해 벌통에 말벌이 침입하는 시간을 지연시키기 위해 벌통의 입구를 다음과 같이 늘린다. 만약 말벌이 검출되면 벌통 내부와 연결된 부분의 입구를 서보모터(SG-90)를 작동시켜 꿀벌만 지나갈 수 있는 망으로 막아버리게 되고 말벌은 벌통 내부로 침입할 수 없게 된다.

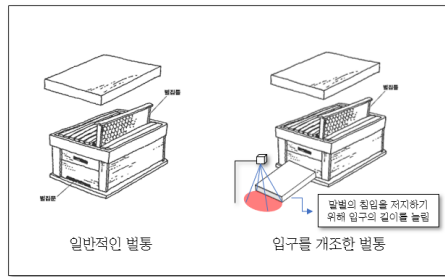


그림 7. 양봉 벌통의 개조

이렇게 늘어난 입구로 인해 말벌이 벌통 내부로 곧바로 침입할 수 없도록 한다. 말벌이 착륙하고 난 이후 실시간 카메라에서는 지속적으로 현재 영상을 미리 학습된 딥러닝 input layer에 넣게 되고, 그 영상에 해당하는 정보가 꿀벌인지 말벌인지를 판단하게 된다. 만약 말벌로 판단되게 되면 벌통입구를 꿀벌만 지나다닐 수 있는 망으로 차단하게 되고, 다음과 같이 생긴 말벌 덫을 이용하여 말벌을 가둔 후 전기충격을 가해 말벌을 살상한다.

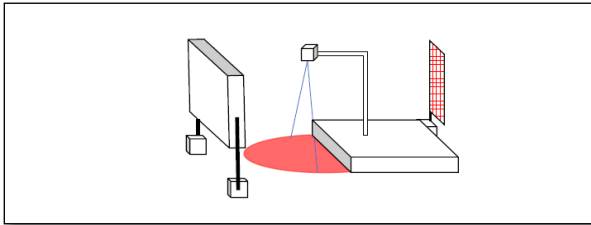


그림 8. 말벌 퇴치를 위한 덫

또한 양봉업자가 외출을 하는 등 양봉 농가 근처에 있지 않을 경우에 시스템이 오작동할 확률이 존재하므로, 실시간 영상정보를 Gateway Server를 통하여 사용자 Android App으로 전송하게 된다. 이 App에서는 벌통의 온도 또한 전송받아 현재 벌통이 안전한 온도 상태에 있는지 또한 확인할 수 있다. 사용자가 영상을 보고 만약 말벌이 침입했음을 인지하게 되면, App의 말벌퇴치 버튼을 눌러 수동으로 말벌 덫을 작동시켜 말벌 퇴치를 할 수 있다.

4. 기능구현

4.1 하드웨어 및 소프트웨어 플랫폼

표 1. 플랫폼 정보

Platform	주요내용
Hardware Platform	<ul style="list-style-type: none"> Raspberry Pi Model 3B (CPU : 1.2GHz ARM Cortex-A53 MP4) Memory : 1GB LPDDR2) Temperature Sensor : DS18B20 Servo Motor : SG-90
Software Platform	<ul style="list-style-type: none"> OS : Raspbian
Application	<ul style="list-style-type: none"> OpenCV 3.4.4

Program	<ul style="list-style-type: none"> Caffe deep-learning framework
Platform	<ul style="list-style-type: none"> Google Mobilenet V1

이 논문에서 사용된 임베디드 환경은 라즈베리파이 Model 3B이다. 운영체제로는 Raspbian을 사용하고, 말벌/ 꿀벌 인식기능을 위해 딥러닝 프레임워크로는 Caffe를 사용하였다. 또한 실시간 영상을 받아 오기 위해 OpenCv 3.4.4 버전을 사용하였으며 CNN(Convolution Neural Network)는 Mobilenet V1을 사용하였다.

표 2. 함수 정보

Function block	Detailed Function Block	main function
Raspberry pi (Client)	Real-time video	실시간 영상을 화면에 띄우고 실시간 영상을 send_Queue와 classification Queue에 넣음.
	Classification	Classification queue가 empty 상태가 아니면 그 정보를 학습 되어있는 CNN의 input으로 넣어 말벌이 검출되는지 실시간으로 확인 연속 3frame 이상 말벌 일 확률이 0.9를 넘을 경우 ifwaspsdetected 함수 호출
	Send-video	Video 정보를 170*100 RGB 값으로 읽어 char[] 배열에 1020씩 끊어 Packet을 만들어 전송
	Receive-control	App으로부터 control 정보를 받음.
	ifwaspsdetected	만약 말벌이 검출되거나 App으로부터 Control 정보를 받으면 말벌 덫을 작동시킴
	Send-temperature	온도정보를 Gateway Server로 전송
Gateway PC Server (Server)	Receivetemp	Raspberry pi로부터 온도 정보를 받음
	Receivevideo	Raspberry Pi로부터 video 정보를 1020크기의 byte 배열로 받음
	Receivecontrol	App으로 control 정보를 받음
	Sendvideo	받은 video 정보를 app으로 전송
	Sendtemp	받은 온도 정보를 app으로 전송

	sendcontrol	받은 control 정보를 Raspberry Pi로 전송
Android Application (Client)	Receivetemp	온도 정보를 받음
	Receivevideo	영상 정보를 받음
	Sendcontrol	Control 정보를 전송
	displayVideo	받은 영상 정보를 총 50회 합쳐(1020*50 = 51000 = 170*100*3) 영상정보를App 화면에 띄워줌
	Displaytemp	온도정보를 App화면에 띄워줌

4.2 말벌 인식을 위한 실시간 영상 Deep Learning Classification/Detection 기능[5] [6]

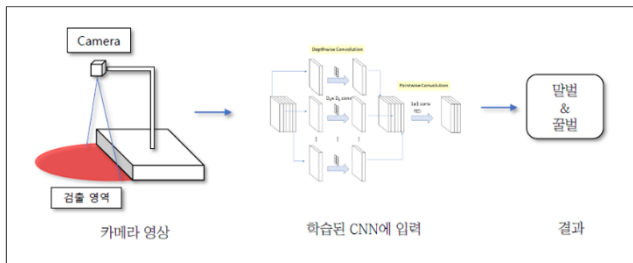


그림 9. 실시간 영상 말벌 탐색

실시간 영상을 탐지하기 위해 사용된 카메라는 Raspberry pi v2 camera이며, 사용된 딥러닝 프레임워크는 Caffe이다. 실질적인 분류 기능을 위해 사용된 CNN(Convolution Neural Network)은 Google Mobilenet v1이다. Raspberry pi v2 카메라에서 실시간으로 영상정보를 3초당 1frame씩 학습된 Google Mobilenet v1의 입력 레이어에 넣는다. 그리고 출력으로 Classification Name과 확률을 받는다. 만약 Classification 결과가 연속해서 3번 이상 Wasp(말벌)이고 그 확률이 0.95 이상이면 Servo Motor를 동작시켜 말벌을 잡는다. 학습에 사용된 데이터는 말벌 700장 꿀벌 500장이며 5000회를 학습시켰다. 데이터셋을 확장시키기 위해 Caffe에 Augmentation 기능을 추가한 Caffe-augmentation Framework를 사용하여 학습시켰다. Caffe-augmentation Framework의 Augmentation 기능을 활용하여 1200장의 데이터셋을 회전시키거나 rescale, histogram equalization등을 적용하여 17만장 이상의 데이터를 학습시킨 것과 비슷한 수준의 모델을 얻을 수 있었다. 또한 학습 횟수를 줄이기 위하여 이미 pretrained 된 Google Mobilenet v1 model로 Finetuning(전이학습)으로 학습을 진행하였다.

4.3 말벌 덫을 움직이기 위한 Servo Motor Control / Electronic shock circuit control 기능

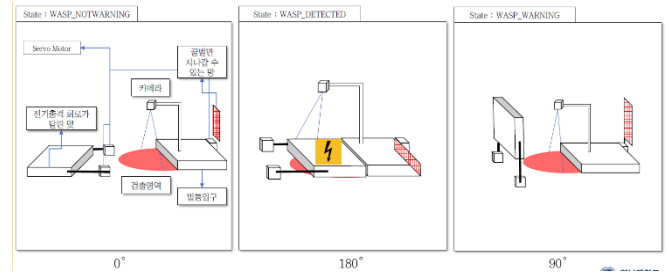


그림 10. 말벌 퇴치를 위한 덫의 동작

말벌이 검출되지 않은 경우 Servo Motor를 0°로 유지하며, 말벌이 검출되면 Servo Motor를 180°로 이동시켜 말벌을 덫에 가둔다. 그리고 벌통내부와 이어져있는 입구를 꿀벌만 지나갈 수 있는 망으로 닫는다. 이후 말벌의 2차 공격 시 더 빠르게 말벌을 가두기 위해 20분동안 Servo Motor를 90° 각도로 유지시킨다. 벌통 내부 입구 또한 꿀벌만 지나갈 수 있는 망으로 20분동안 막아둔다. 만약 20분이 지나고 난 이후에는 더 이상의 말벌의 침입이 없다고 판단하고 Servo Motor를 0°로 재위치시키고, 망으로 막힌 벌통내부와 연결된 입구를 개방시킨다. 여기서 사용된 서보모터는 SG-90 모델이다.

4.4 영상 전송 및 온도정보 전송 기능 / App으로 말벌 덫 Control 기능[7] [8]

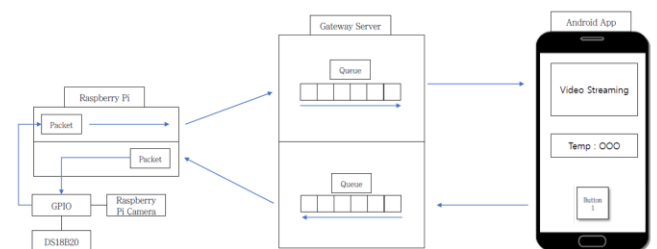


그림 11. 실시간 영상/온도 전송/Control

온도정보를 읽어오는데 사용된 모듈은 DS18B20이다. 이 모듈은 온도정보를 읽어 디지털화시켜 라즈베리 파이에서 읽을 수 있도록 해 준다. 이정보를 GPIO로 읽어 온도정보를 TCP/IP 프로토콜을 통해 Gateway Server로 전송한다. Gateway Server에서는 그 온도정보를 받아 그대로 Android App으로 전송해주게 된다. 또한 영상정보 전송 시 실시간 영상을 170*100 size로 변환 뒤 OpenCv Mat 구조체에 담아[2] 이를 170*100*3의 RGB 데이터로 읽어 char 배열로 정보를 담는다.(51000Bytes) 그리고 이 정보를 1020Bytes 씩 잘라 전송한다. 1020Bytes로 잘라 전송하는 이유는 전송할 데이터가 너무 작거나 큰 경우 전송속도가 매우 느려졌기 때문이다. 이후 Android App에서는 Gateway Server로부터 이 정보를 받아 영상/온도를 화면에 띄우게

된다. 마지막으로 Android Application에서 말벌 퇴치 버튼을 눌러 말벌을 직접 퇴치할 수 있는 기능도 제공된다.

5. 성능측정 및 분석

5.1 Google Mobilenet V1 전이학습(파인튜닝)에 대한 성능 측정 및 분석

그림 12는 말벌데이터 700장 꿀벌데이터 500장을 가지고 다른 이미지로 이미 가중치가 학습되어 있는 Google Mobilenet V1 CNN을 전이학습(Finetuning)한 결과이다. Test Set의 시작 Loss율은 2.5에서 시작하여 5000 iteration 이후 0에 근접하였고, Validation Loss율 또한 1.5에서 시작하여 5000 iteration 이후에는 0에 근접하였다. 또한 Validation Set에 대한 Accuracy는 5000회 학습 이후 0.9 이상으로 출력되었다.

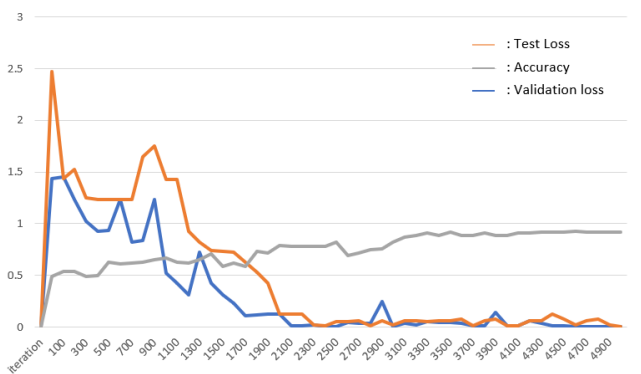


그림 12. Deep-learning 학습 Graph

5.2 실시간 영상 말벌 탐지 성능측정 및 분석

그림 13과 그림 14는 말벌이 탐지되지 않는 일반 상태에서의 측정값이다. 이 경우 말벌/꿀벌 확률이 50% ~ 70%범위 내에 존재하며 서보모터는 동작하지 않는다

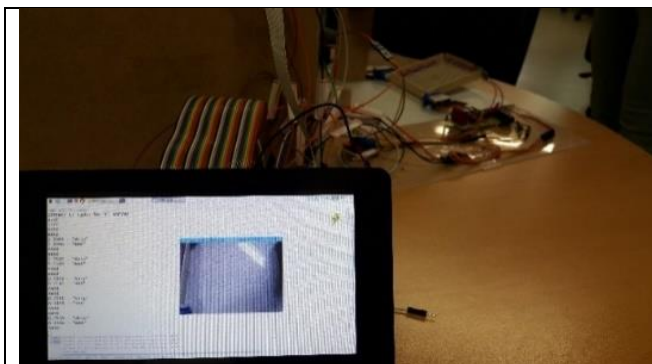


그림 13. 기본상태 - Wasp Trap : 0°

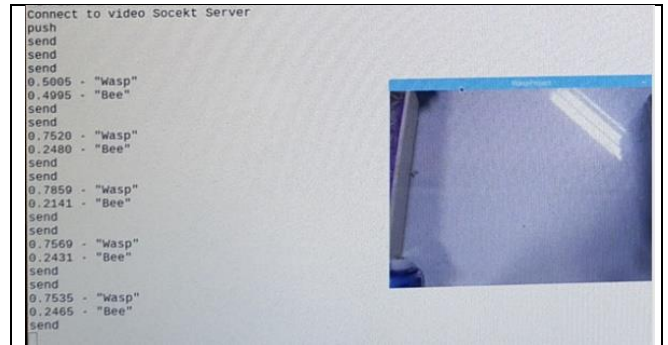


그림 14. 기본상태 - 아무것도 없을 때 말벌일 확률이 0.5 ~ 0.7로 나오는 모습

그림 15와 그림 16은 말벌이 검출된 상태이며 말벌일 확률이 95%이상인 상태가 3번이상 유지된 것을 확인할 수 있다. 서보모터가 동작하여 말벌을 가렸으며 전기충격을 주었다.

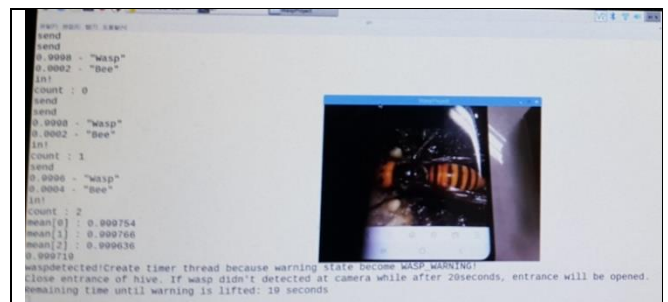


그림 15. 말벌 검출된 상태 - 말벌일 확률이 0.95 이상

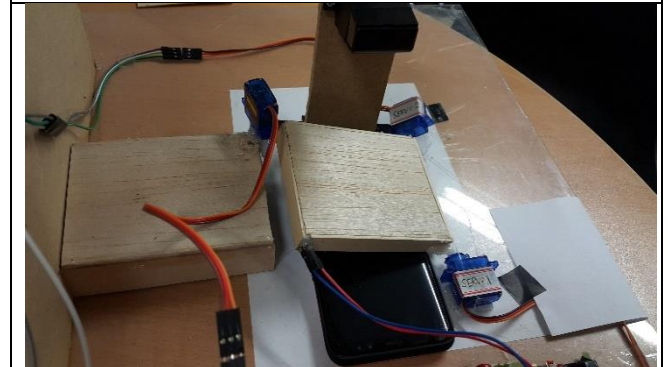


그림 16. 말벌이 검출된 상태 - Wasp Trap: 180°

그림 17은 말벌이 검출된 이후 20분간 말벌 덮이 90도로 세워져 있는 모습이다.

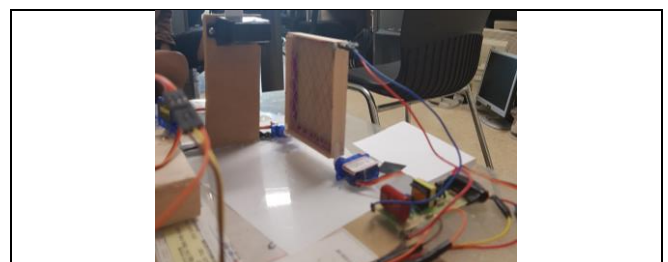


그림 17. 말벌이 검출된 이후 상태 - Wasp Trap: 90°

5.3 실시간 영상/온도정보 성능측정 및 분석

그림 18은 실시간 영상/온도정보가 전송되는 모습을 보인 것이다. 실시간 온도정보는 5초에 1번 User의 Android App으로 전송되며, 실시간 영상은 3초당 1Frame으로 나타난다. TCP/IP Protocol로 전송하여 전송속도에 한계가 존재한다.



그림 18. 실시간 영상 전송 및 온도정보 전송

5.4 결과 분석

말벌사진을 넣을 시 95%의 확률로 말벌임을 감지하며, 꿀벌 사진을 넣을 시 90% 이상의 확률로 꿀벌임을 감지한다. 일반적인 배경 또는 다른 물체의 경우 말벌 또는 꿀벌일 확률이 0.5 ~ 0.7 정도가 나오는데, 이는 초기 Deep learning 학습 시 Class를 Bee / Wasp 두 종류로만 학습시켰기 때문으로 보인다. 학습 시 Class를 Bee / Wasp / Unknown으로 학습시킨다면 말벌 또는 꿀벌이 아닐 경우에 Unknown으로 구분시킬 수 있을 것으로 예상된다.

영상 전송의 경우 TCP/IP 프로토콜의 전송의 한계로 5초당 1Frame이 전송 가능했다. UDP 프로토콜 사용 시 훨씬 좋은 성능의 실시간 영상 전송기능을 구현할 수 있을 것으로 보인다. 또한 온도 정보가 App 화면에 올바른 값으로 띄워지는 것을 확인할 수 있었다. 그리고 App의 말벌퇴치 버튼을 눌러 Raspberry Pi의 말벌 퇴치를 제어할 수 있음을 확인하였다.

6. 결론

양봉농가에서 꿀 생산량이 급감하는 큰 이유 중 하나로 말벌의 공격을 꼽을 수 있다. 본 시스템에서는 딥러닝을 활용해 미리 학습된 모델로 실시간 영상을 분석하여 말벌을 잡아낸다. 또한 양봉업자가 스마트폰 어플리케이션으로 직접 영상을 보며 벌통을 감시할 수 있다. 이 시스템은 실제 말벌만을 딥러닝으로 구분하므로 단순 유인액을 활용하는 말벌 포획기나 움직임이 감지될 경우 그 영상의 크기만을 분석하여 말벌임을 감지하는 시스템에 비해 훨씬 효과적으로 말벌을 퇴치할 수 있을 것으로 보인다. 본 논문에서는 실시간으로 돌아가는 카메라에 말벌 사진을 비추어 테스트를 진행하였다. 말벌이 검출되면 퇴치를 작

동시키는 것을 확인할 수 있었으며, 말벌의 검출 성능은 약 95% 이상임을 확인할 수 있었다. 영상 전송의 경우 3초에 1frame이 전송되는 것을 확인할 수 있었고 온도 전송률 또한 100%로 확인되었다. 이 시스템을 사용함으로써 양봉농가는 꿀벌의 피해를 최소화하고 그로 인해 꿀 생산의 증대를 기대할 수 있다. 또한 양봉업자가 말벌 걱정 없이 마음 놓고 자리를 비울 수 있게 되며, 실시간으로 영상정보를 App으로 받아 원거리에서도 말벌 침입을 감시할 수 있게 된다. 다음의 연구에서는 분류 Class를 Wasp, Bee, Unknown으로 분류하여 배경 및 다른 개체들을 Unknown으로 구분할 수 있도록 구성해볼 예정이다.

참고문헌

- [1] 남상인 기자, “환경부, 외래종 등검은 말벌 피해액 연간 1750억원”, 서울신문 [Online] <https://www.seoul.co.kr/news/newsView.php?id=20181226500008>
- [2] 이준길, 이진질, “꿀벌통을 이용한 말벌 포획기”, 대한민국 특허청 [Online] <https://patents.google.com/patent/KR101822029B1/ko>
- [3] 김부국, “벌통에 고정시켜 사용하는 말벌 살상장치”, 대한민국 특허청 [Online] <https://patents.google.com/patent/KR101635317B1/ko>
- [4] 한국양봉협회 관리자, “‘꿀벌 킬러’ 등검은 말벌 확산으로 양봉농가 초토화, 대책마련 시급”, 한국 양봉협회 [Online] https://www.korapis.or.kr/jsp/sub6-4_01.jsp?no=5631
- [5] 오일석, 기계 학습, 한빛 아카데미, 2017
- [6] 바라스 람순다르, 레자 자데, 한권으로 끝내는 딥러닝 텐서플로, 한빛미디어, 2018
- [7] 정석용, 정석용의 TCP/IP 소켓 프로그래밍, 프리렉, 2009
- [8] 김동근, C++ API OpenCV 프로그래밍, 가메출판사, 2015