

# 프로젝트 보고서

## 애완동물 배변 알림이

과 목 명	임베디드 시스템
학 과	컴퓨터공학과
학 번	15012974
성 명	최태호
제 출 일	2021년 6월 12일
지도교수	박기호



**세종대학교**  
SEJONG UNIVERSITY

# 목차

## 가. 서론

1. 프로젝트 목적
2. 프로젝트 개요
3. 개발환경

## 나. 프로젝트 내용

1. 시나리오
2. 주요 코드

## 다. 프로젝트 결과

1. 결과물
2. 어려웠던 점
3. 유지보수

## 라. 결론

1. 느낀점

## 가. 서론

### 1. 프로젝트 목적

임베디드 시스템의 특징을 고려하여 Raspberry pi를 사용하여 다양한 optimization 수행과 개인 프로젝트 제작을 경험하기 위함.

### 2. 프로젝트 개요

코로나19로 인한 재택근무 및 온라인 강의 중 애완동물과 다른공간에 있을 때 애완동물의 배변활동을 체크할 수 없다.

배변활동을 제때에 처리해주지 않으면 집안이 더러워지는 등 불편사항이 발생한다.

이 프로젝트에서는 애완동물의 배변활동을 감지하여 알람으로 알려주고 알람을 들은 사용자가 배변상황을 인지하고 처리할 수 있도록 도와준다.

### 3. 개발환경

#### 1. PuTTY

PuTTY는 SSH, Telnet, rlogin, rawTCP를 위한 클라이언트로 동작하 자유 및 오픈소스 단말 에뮬레이터 응용 프로그램이다.

#### 2. OpenCV

OpenCV는 실시간 컴퓨터 비전을 목적으로한 프로그래밍 라이브러리이다.  
실시간 이미지 프로세싱에 중점을 둔 라이브러리이다.

#### 3. YOLO

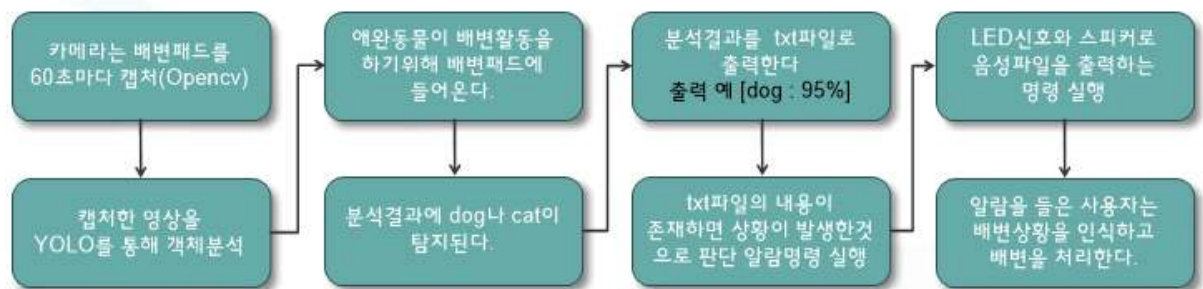
YOLO는 You Only Look Once의 약자로 객체 탐지 기법 모델 중 하나이다.

## 나. 프로젝트 내용

### 1. 시나리오

#### 시나리오

사용자와 애완동물은 다른 공간에 있는 상황이고  
따라서 사용자는 애완동물의 배변활동을 인지하지 못하고 제때에 처리하지 못하는 상황이다.



### 2. 주요 코드

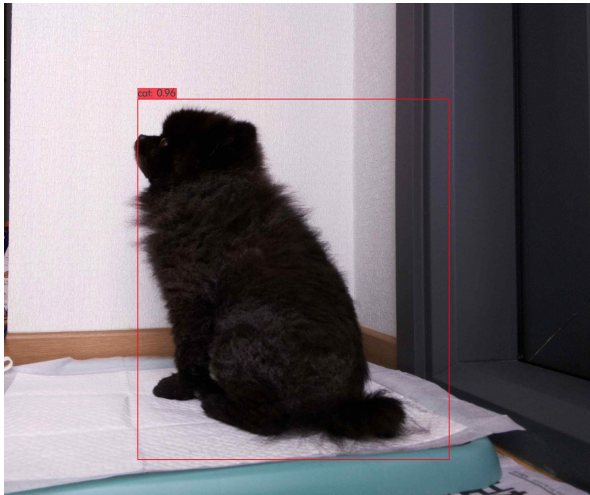
YOLO분석 후 분석결과를 텍스트파일로 출력하는 부분(INSET주석 사이 내용)

```
349 for (i = 0; i < selected_detections_num; ++i) {
350     const int best_class = selected_detections[i].best_class;
351     /*****INSERT*****/
352     if((strcmp(names[best_class],dog)==0)|| (strcmp(names[best_class],cat)==0))
353         flag=1;
354     /*****INSERT*****/
355
356     printf("%s: %.0f%%", names[best_class], selected_detections[i].det.prob[best_class] * 100);
357     if (ext_output)
358         printf("\t(left_x: %4.0f top_y: %4.0f width: %4.0f height: %4.0f)\n",
359             round((selected_detections[i].det.bbox.x - selected_detections[i].det.bbox.w / 2)*im.w),
360             round((selected_detections[i].det.bbox.y - selected_detections[i].det.bbox.h / 2)*im.h),
361             round(selected_detections[i].det.bbox.w*im.w), round(selected_detections[i].det.bbox.h*im.h));
362     else
363         printf("\n");
364     int j;
365     for (j = 0; j < classes; ++j) {
366         if (selected_detections[i].det.prob[j] > thresh && j != best_class) {
367             printf("%s: %.0f%%", names[j], selected_detections[i].det.prob[j] * 100);
368             if (ext_output)
369                 printf("\t(left_x: %4.0f top_y: %4.0f width: %4.0f height: %4.0f)\n",
370                     round((selected_detections[i].det.bbox.x - selected_detections[i].det.bbox.w / 2)*im.w),
371                     round((selected_detections[i].det.bbox.y - selected_detections[i].det.bbox.h / 2)*im.h),
372                     round(selected_detections[i].det.bbox.w*im.w), round(selected_detections[i].det.bbox.h*im.h));
373             else
374                 printf("\n");
375         }
376     }
377 }
378 /*****INSERT*****/
379 if(flag==1){
380     fp = fopen("/home/pi/darknet/index.txt","w");
381     fprintf(fp,"%s",Detected);
382     fclose(fp);
383 }
384 else{
385     fp = fopen("/home/pi/darknet/index.txt","w");
386     fprintf(fp,"%s",space);
387     fclose(fp);
388 }
389 /*****INSERT*****/
```

## 다. 프로젝트 결과

### 1. 결과물

(1). 배변패드 위에 올라온 애완동물 캡처 및 분석결과 이미지



(2). 분석결과가 저장되는 index.txt파일의 내용



(3). 배변상황 발생 시 알람출력



## 2. 어려웠던 점

YOLO분석 후 분석결과를 텍스트파일로 출력하는 부분이 어려웠다.

image\_opencv.cpp 소스파일의 draw\_detection\_cv\_v3 함수에서 분석결과와 파일입출력을 계속 시도했지만 번번히 실패하였다.

결국 YOLO에 있는 소스파일들을 하나하나 열어가며 분석했다.

결과적으로 image.c 소스파일의 draw\_detections\_v3 함수에서 분석결과와 파일입출력 정상작동에 성공했다.

계속 시도했던 부분은 분석결과 이미지의 라벨을 담당하는 부분이었다.

아래의 사진은 image\_opencv.cpp소스파일의 draw\_detection\_cv\_v3 함수이다.

```
884 // =====
885 // Draw Detection
886 // =====
887 extern "C" void draw_detections_cv_v3(mat_cv * mat, detection * dets, int num, float thresh, char** names, image * *alphabet, int classes, int ext_output)
888 {
889     try {
890         cv::Mat* show_img = (cv::Mat*)mat;
891         int i, j;
892         if (!show_img) return;
893         static int frame_id = 0;
894         frame_id++;
895         FILE *fp;
896         for (i = 0; i < num; ++i) {
897             char labelstr[4096] = { 0 };
898             int class_id = -1;
899             for (j = 0; j < classes; ++j) {
900                 int show = strncmp(names[j], "dont_show", 9);
901                 if (dets[i].prob[j] > thresh&& show) {
902                     if (class_id < 0) {
903                         strcat(labelstr, names[j]);
904                         class_id = j;
905                         char buff[20];
906                         if (dets[i].track_id) {
907                             sprintf(buff, "(id: %d)", dets[i].track_id);
908                             strcat(labelstr, buff);
909                         }
910                         sprintf(buff, " (%2.0f%%)", dets[i].prob[j] * 100);
911                         strcat(labelstr, buff);
912                         printf("%s: %.0f%% ", names[j], dets[i].prob[j] * 100);
913                     }
914                 }
915                 else {
916                     strcat(labelstr, ", ");
917                     strcat(labelstr, names[j]);
918                     printf(", %s: %.0f%% ", names[j], dets[i].prob[j] * 100);
919                 }
920             }
921         }
922     }
```

## 3. 유지보수

애완동물의 배변활동 관찰결과 배변활동 시 평균 3~40초가량 배변패드에서 머무는 것을 확인 하지만 YOLO분석속도는 약 65초가량의 시간이 필요하다.

따라서 캡처에 잡히지 않는 2~30초의 시간이 존재하기 때문에 분석속도의 개선이 필요함.

또한 분석속도가 개선된다면 실시간 분석등 다양한 유지보수가 가능할것으로 생각된다.

## 라. 결론

### 1. 느낀점

처음에는 Linux환경과 Raspberry pi가 너무 낯설어 프로젝트를 할 수 있을지 걱정이 앞섰습니다. Linux환경 실습과제조차 정말 많은시간이 걸려서 정말 막막했지만 간단한 프로젝트라도 완성시키자는 마음으로 한 개씩 해 나아가다 보니 어떻게든 완성은 시킨 것 같습니다. 비록 교수님께서 원하시는 optimization의 수행은 하지 못했지만 교수님께서 말씀하신 경험의 의미는 느낄 수 있었습니다.  
좋은 경험이었습니다. 감사합니다.