

# 2025-1 IoT 시스템

## 프로젝트 결과 최종 보고서

Smart PetFeeder : 스마트 반려견 자동 급식 시스템 (SmaPF)



과목	2025-1 IoT 시스템
팀 명	Team. 라즈베리베리
조원	32200584 김민중
	32202136 서범석
	32217072 김도익

# 목 차

I. 서론	3 page
II. 배경 및 개발 동기	4 page
2.1 기존 시스템의 한계점	
2.2 스마트 급식 시스템의 필요성	
2.3 본 프로젝트의 접근 방식 및 차별성	
III. 시스템 설계 및 구현	6 page
3.1 시스템 구성 및 주요 기능	
3.2 하드웨어 및 소프트웨어 구성	
3.3 센서 및 AI 기술 적용 방식	
IV. 개발 진행 및 완성도	9 page
4.1 구현 범위 및 주요 기능 요약	
4.2 실제 작동 흐름 예시	
V. 기술적 도전과 해결 방안	11 page
5.1 주요 기술적 어려움	
5.2 문제 해결 전략 및 창의적 접근	
VI. 프로젝트 수행에 대한 고찰 및 개선점	13 page
6.1 프로젝트를 통해 얻은 기술적 고찰	
6.2 프로젝트 완성도를 높이기 위한 개선점	
VII. 결론 및 향후 연구 방향	15 page
7.1 프로젝트 요약 및 의의	
7.2 향후 발전 방향 및 기대 효과	

## 1. 서론

현대 사회에서 반려동물은 단순한 애완의 개념을 넘어, 가족 구성원으로 인식되며 그 돌봄의 수준 또한 점차 고도화되고 있다. 특히 반려견의 경우 일정한 시간에 맞춘 식사 제공과 적절한 급여량 조절은 건강 관리의 핵심 요소이지만, 보호자의 부재 또는 불규칙한 생활 패턴으로 인해 식사 관리가 일관되게 이루어지지 않는 문제가 자주 발생한다.

기존의 자동 급식기 제품들은 주로 **정해진 시간에 동일한 사료량을 배급하는 기계적 기능에 국한**되어 있으며, 반려견이 실제로 식사를 했는지의 여부나 섭취량에 대한 정량적 피드백을 제공하지 못한다. 일부 제품은 보호자의 음성을 녹음하여 급식 시점에 재생하는 기능을 제공하지만, 이는 실제 반려견의 반응이나 건강 상태와는 무관하게 일률적으로 적용된다는 한계가 존재한다.

본 프로젝트는 이러한 현실적인 제약을 인식하고, 센서 및 AI 기반 기술을 활용한 스마트 사료 급여 시스템(Smart PetFeeder)을 설계하였다. 사용자는 모바일 앱을 통해 사전에 음성을 녹음하고, 타이머를 설정하여 원하는 시간에 자동으로 급식기가 작동하도록 설정할 수 있다. 급식 시점이 도달하면, 초음파 센서(HY-SRF05)를 통해 반려견의 접근 여부를 감지하고, 감지된 경우에는 카메라가 활성화되어 YOLOv8 기반 객체 인식 모델을 통해 반려견의 개체 식별이 이루어진다. 이후, 서보모터(SG-90)를 활용하여 사료가 공급되며, 로드셀(HX711)을 통해 설정된 목표 무게 도달 여부를 판단하여 자동 종료된다.

이 시스템은 기존 제품들이 제공하던 단순 자동 배급이나 보호자 음성 재생 기능을 넘어서, **실제 반려견의 존재 여부를 감지하고, 각 개체에 따라 설정된 맞춤 사료량을 배급하는 기능을 결합**하였다는 점에서 실용적 의의가 있다. 특히 다두 반려 환경에서 개별 맞춤 급식이 가능하다는 점은, 실생활 적용성을 높이는 차별적인 요소로 작용할 수 있다.

본 시스템은 단순 자동화 기기의 범주를 넘어, 사용자의 정서적 상호작용과 인공지능 기반 객체 인식, 그리고 다양한 센서의 연동을 결합한 **실생활 중심의 IoT 융합 프로젝트**이다. 본 보고서는 해당 시스템의 설계 및 구현 과정, 실험을 통한 검증, 기술적 고찰 및 향후 개선 가능성에 대해 서술한다.

## 2. 배경 및 개발 동기

### 2.1. 기존 시스템의 한계점

반려동물 양육 가구의 증가와 함께 자동 급식기의 수요도 꾸준히 늘고 있으나, 현재 시중에 유통 중인 대부분의 제품은 구조적으로 한계가 뚜렷하다. 가장 큰 문제는 자동 급식 시스템이 보호자의 설정에 따라 정해진 시간에 일정량의 사료를 일률적으로 제공하는 데 그친다는

## 반려동물이 하루 평균 집에 혼자 있는 시간

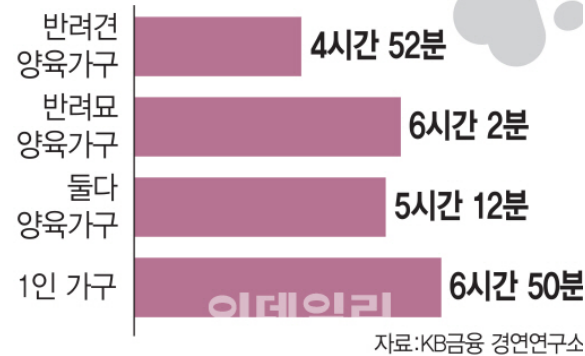


Fig 1. 반려동물이 하루 평균 집에 혼자 있는 시간

점이다. 이 방식은 보호자가 집을 자주 비우는 상황에서는 편리해 보일 수 있으나, 실질적인 식사 관리 측면에서는 여전히 여러 가지 맹점을 가진다.

실제로 국내 반려동물 보호자들의 약 80%는 **반려동물이 하루 평균 5시간 이상 혼자 있는 시간**이 발생한다고 보고되고 있으며, 이 수치는 1인 가구 또는 맞벌이 가정일수록 더욱 높아지는 경향을 보인다(Fig 1). 이러한 보호자 부재 환경이 일반화되는 상황은, 단순 편의성이 아닌 **실질적 돌봄 공백의 문제**로 이어질 수 있음을 시사한다.

그럼에도 불구하고, 현재 상용화된 자동 급식기는 반려동물의 실제 섭취 여부를 확인하거나, 개별 동물의 식사 습관에 맞춰 급식량을 조정하는 기능이 거의 전무하다. 급식기가 작동해서 사료가 배급되었다는 사실만 기록될 뿐, 반려견이 이를 실제로 섭취했는지, 혹은 전혀 먹지 않았는지에 대한 정보는 보호자가 알 수 없다. 이는 식욕 저하나 스트레스, 질병 등 반려동물의 건강 이상 신호를 조기에 인지할 기회를 놓치게 만들 수 있다.

또한 다두 반려 환경에서는 사료를 독점하거나 빠르게 먹는 개체가 존재할 수 있으며, 각 개체의 체격, 연령, 건강 상태 등에 맞는 식사량 조절이 필요한 상황에서도 대부분의 급식기는 획일적 배급 방식을 고수한다. 반려동물을 개체 단위로 구분할 수 있는 기능이 없다 보니, 결과적으로 정밀한 영양 관리나 맞춤 급식이 필요한 상황이다.

이 밖에도, 보호자의 음성을 재생하거나 카메라를 탑재한 일부 고급형 제품이 존재하지만, 이들 역시 실시간 인식 및 반응 기능이 부재하거나, 반려동물의 행동에 따라 동적으로 작동하지 못하는 구조를 가지고 있다. 특히 센서 기반 접근 감지, 실시간 객체 인식, 무게 기반 정량 피드백 등 다양한 정보를 융합한 종합 시스템은 대부분의 제품에서 구현되어 있지 않다.

이처럼 현재 자동 급식기의 기술 수준은 여전히 단순화된 자동화에 머무르고 있으며, 보

호자의 실제 필요를 반영하거나 반려동물의 섭취 행태를 능동적으로 분석·관리하는 구조로 발전하지 못하고 있다. 이러한 상황은 본 프로젝트가 지향하는 **개체 인식 기반 맞춤형 급식 시스템의 필요성과 의의를 분명히 보여주는 출발점**이 된다.

## 2.2 스마트 급식 시스템의 필요성

반려동물 돌봄 환경이 급속도로 정교화됨에 따라, 기존의 단순한 자동 급식기는 점점 실질적인 사용자 요구를 충족하지 못하는 한계에 부딪히고 있다. 보호자들이 외부에 있는 시간 동안에도 정확한 시간과 양을 개체별로 맞춤 제공할 수 있는 급식 솔루션에 대한 수요는 꾸준히 증가하고 있으며, 이에 대응하기 위한 새로운 방식의 시스템이 요구된다.

**SmaPF(Smart PetFeeder)**는 이러한 필요에 대응하기 위해 기획된 시스템으로, 단순히 정해진 시간에 일정량의 사료를 제공하는 자동화 장치를 넘어, **개체 인식, 센서 기반 상호작용, 녹음 음성 트리거** 등 복합적인 기술을 융합하여, 다두 반려 가구에서 개체별 급식이 가능하도록 설계되었다. 이는 특히 동일 공간에 여러 반려동물이 공존할 경우, **개별 건강 상태나 체중에 따라 식사량이 조절되어야 하는 상황에 대응**할 수 있다.

스마트한 급식 시스템이 필요한 이유는, 이제 ‘시간 맞춰 사료를 주는 기계’를 넘어서, 동물의 상태와 맥락을 인식하고 이에 반응하는 지능적 시스템이 요구되기 때문이다. 특히 SmaPF는 초음파 센서(HY-SRF05), 개체 인식 기반 카메라, 로드셀(HX711), 서보 모터(SG-90) 등 다양한 센서를 결합해, **단순 급식 → 대상 확인 → 사료 정량 배급 → 섭취 여부 확인**이라는 일련의 절차를 자동화한다. 이는 사용자의 개입 없이도 정확하고 반복 가능한 급식 환경을 구축할 수 있다는 점에서 큰 강점을 지닌다.

또한, 보호자가 미리 녹음한 음성 메시지를 녹음/타이머 기능을 통해 재생하는 심리적 유대감 기능도 포함하고 있으며, 이는 기술적 기능과 정서적 돌봄의 균형을 맞추려는 시도로 볼 수 있다.

한편, 1인 가구, 맞벌이 부부, 혹은 출퇴근 시간이 불규칙한 보호자에게 있어 **자동화 이상의 맞춤형 반려동물 관리 시스템**은 생활의 효율성을 높이는 동시에, 반려동물의 건강을 보다 정교하게 관리할 수 있는 중요한 도구가 될 수 있다. SmaPF는 이러한 생활 환경의 변화와 기술의 접점을 실현한 시스템으로서, 기존의 급식 방식에 비해 보다 실질적이고, 데이터를 기반으로 한 급식 및 관리 루틴을 가능케 한다.

결론적으로, SmaPF는 **기존 자동 급식기의 단점을 보완하고, 반려동물 중심의 돌봄 환경 구축을 위한 새로운 기술적 모델**로 기능할 수 있으며, 향후 건강 모니터링, 식습관 분석 등의 확장 기능과도 자연스럽게 연계될 수 있다.

## 2.3 본 프로젝트의 접근 방식 및 차별성

SmaPF(Smart PetFeeder)는 단순한 정시 사료 공급 기능을 넘어서, **반려동물 개체별 맞춤 급식**을 실현하기 위해 다양한 센서와 AI 기반 인식 기술을 통합적으로 활용하는 방식으로 개발되었다. 기존 시장에 존재하는 다수의 자동 급식기 시스템은 대부분 다음과 같은 제한을 가진다:

- 동일한 시간과 동일한 사료량을 일괄적으로 공급
- 수동 설정된 타이머 기반으로만 동작
- 반려동물 개체 구분 불가능
- 급식 여부를 센서로 확인할 수 없음

이에 비해 SmaPF는 다음과 같은 방식으로 기존 접근을 혁신하고자 하였다:

### 1. YOLO 기반 반려동물 객체 인식 도입

사용자의 녹음 음성이 재생된 후, 초음파 센서를 통해 거리 인식이 감지되면 카메라가 활성화되고, YOLO 모델을 통해 반려동물의 종류 및 개체를 분류한다. 이를 통해 어떤 반려동물이 접근했는지에 따라 배급되는 사료량을 사전 설정값에 따라 맞춤 조절할 수 있다.

### 2. 실시간 감지 기반 급식 제어 시스템

초음파 센서(HY-SRF05)는 반려동물이 일정 거리 내에 접근했는지를 감지하고, 로드셀(HX711)은 실제 사료가 배급되었는지를 중량 단위로 실시간 검증한다. 이를 통해 공급 오류나 미섭취 상황을 감지하고 급식 작동을 자동 중지할 수 있다.

### 3. 사용자 중심 인터페이스와 정서적 상호작용 기능

사용자는 Flutter 기반 모바일 앱을 통해 타이머를 설정하고 음성 녹음을 전송할 수 있으며, 급식기의 작동은 사전에 설정된 시간에 맞춰 자동으로 실행된다. 보호자의 음성이 출력되는 것은 단순한 기능을 넘어, 반려동물에게 정서적 안정감을 제공하기 위한 장치로 설계되었다.

### 4. 기기 간 협업 기반 IoT 시스템 구현

라즈베리파이, 초음파 센서, 카메라, 서보 모터, 로드셀, 스피커 등 다양한 하드웨어가 하나의 시나리오 흐름에 맞춰 유기적으로 동작하도록 구성되었다. 이러한 IoT 기반 연동 구조는 다양한 조건에서 유연한 행동을 수행할 수 있는 구조적 강점을 제공한다.

이와 같은 통합적이고 정교한 접근 방식은, SmaPF가 단순한 자동화 시스템이 아닌, **상황 인식과 맞춤 반응이 가능한 스마트 반려동물 관리 장치**로 진화할 수 있는 기반이 된다. 특

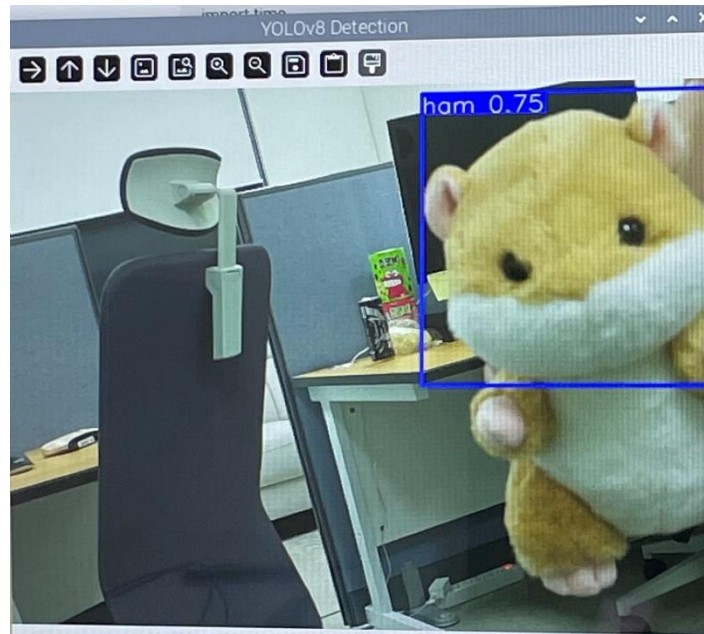


Fig 2. YOLO 기반 실시간 반려동물 인식 화면

이 개체 인식에 따른 차등 급식은 현재까지 상용 제품에서는 흔히 않은 기능으로, **데이터 기반의 건강 관리, 비만 방지, 다두 반려 환경 최적화**라는 측면에서 중요한 차별적 가치를 지닌다.

### 3. 시스템 설계 및 구현

#### 3.1 시스템 구성 및 주요 기능

본 프로젝트인 SmaPF (Smart PetFeeder)는 사용자의 음성 녹음, 타이머 설정, 거리 감지, 반려동물 인식, 급식량 조절 및 자동 중지까지 전 과정을 자동으로 수행하는 지능형 IoT 시스템이다. 하드웨어와 소프트웨어의 유기적인 통합을 통해 반려동물의 개별 행동에 최적화된 급식이 가능하도록 설계되었다. 전체 시스템은 다음과 같은 세부 모듈로 구성된다.

##### 1) 사용자 앱 (Flutter 기반)

- **음성 녹음 기능:** 보호자가 녹음을 직접 수행하며, 반려동물에게 들려줄 메시지를 설정할 수 있다.
- **타이머 기반 예약 설정:** 특정 시간에 녹음이 재생되고 급식기가 작동하도록 예약할 수 있다.

- **간단한 UI 구성:** 사용자 친화적인 Flutter UI를 적용하여 빠르고 직관적인 조작이 가능하다.

## 2) 라즈베리파이 중심 제어 시스템

- **Flask 기반 서버:** 앱에서 전달되는 사용자 설정(시간, 음성 파일, 반려동물 이름 등)을 수신하고, 이를 바탕으로 예약 스케줄을 구성한다. 시간 도래 시 자동으로 급식 제어 흐름을 시작한다.
- **녹음 재생 및 제어 흐름 시작:** 예약된 시간에 사용자의 음성이 재생되며, 이후 급식 시퀀스가 자동으로 시작된다.
- **초음파 센서 (HY-SRF05):** 반려동물이 급식기 앞에 접근했는지를 실시간으로 감지한다.
- **YOLO 기반 객체 인식 (카메라):** 접근한 반려동물이 어떤 종류 혹은 개체인지를 인식하며, 사전에 등록된 클래스별 사료량을 참조한다.
- **서보 모터 (SG-90):** 인식된 개체에 따라 사료 투입구를 일정 시간 개방한다.
- **로드셀 (HX711):** 일정 무게 이상의 사료가 투입되었는지를 감지하며, 목표 무게 도달 시 자동으로 투입을 중단한다.

이와 같이 SmaPF는 사용자 앱과 센서/모터 기반 제어 시스템, 그리고 중간 서버 역할을 수행하는 Flask 백엔드가 완전히 연결된 형태의 IoT 구조를 형성하고 있다. 각각의 센서와 모듈은 독립적으로 동작하지 않고, 상호 의존적으로 연결되어 **개별 반려동물 맞춤 급식**을 실현한다는 점에서 기존 급식기 시스템과 뚜렷한 차별점을 보인다.

## 3.2 하드웨어 및 소프트웨어 구성

본 프로젝트는 라즈베리파이 기반의 IoT 시스템과 Flutter 기반의 모바일 애플리케이션으로 구성되며, 이들을 연결하는 Flask 서버를 통해 전체 시스템이 유기적으로 작동한다. 시스템의 전체 구조는 다음과 같이 구분된다.

### 3.2.1. 하드웨어 구성

- **라즈베리파이 5**  
시스템의 중앙 제어 유닛으로, 센서 제어, 카메라 영상 처리, 모터 동작, 무게 측정 등의 모든 물리적 장치를 통합하여 제어한다.
- **초음파 거리 센서 (HY-SRF05)**  
반려동물이 급식기 근처에 접근했는지를 감지하는 역할을 수행한다. 일정 거리 이하



로 감지되면, 이후 카메라가 활성화된다.

- **카메라 (OpenCV 지원 USB 카메라)**

반려동물의 얼굴 및 형태를 인식하기 위한 장치로, YOLO 기반의 객체 탐지 모델과 함께 사용된다.

- **서보 모터 (SG-90)**

사료 투입구를 제어하는 모터로, 사료 배출 시간에 따라 개방 및 닫힘 동작을 수행한다.

- **무게 센서 (로드셀 + HX711 모듈)**

사료의 무게를 측정하여 설정된 기준 이상이 투입되었는지를 판단하고, 필요 시 모터를 자동으로 종료시킨다.

- **스피커 및 오디오 출력 장치**

사용자 앱에서 설정한 보호자 음성 녹음을 재생하는 장치로, 반려동물의 반응을 유도하기 위한 요소이다.

### 3.2.2. 소프트웨어 구성

#### 1) Flutter 기반 모바일 애플리케이션

- 음성 녹음 기능 및 타이머 설정 기능 제공
- UI 디자인은 간결한 구조로 구현되어 사용자 접근성과 직관성을 높임
- 설정된 정보는 Flask 서버를 통해 라즈베리파이에 전달됨

#### 2) Flask 기반 백엔드 서버

- 앱과 라즈베리파이 간 통신을 담당
- 사용자의 녹음, 타이머 요청 등을 수신하고 이를 라즈베리파이 내부 제어 흐름에 연결
- Flask 서버는 main.py를 통해 주기적으로 예약을 확인하고, 지정된 시간에 급식 시퀀스를 실행

#### 3) YOLO 기반 객체 인식

- 라즈베리파이에서 OpenCV와 함께 YOLO 모델을 활용하여 강아지 얼굴 인식
- 인식된 클래스에 따라 사료량 결정



Fig 3. 실제 앱의 메인 화면 및 프로필 관리 화면

#### 4) Python 기반 센서 제어 로직

- flask\_server.py, main.py에 각종 센서 제어 및 처리 흐름 구현
- 모듈 간 타이밍, 예외 처리, 센서 신호 정제 등이 포함됨

## 4. 개발 진행 및 완성도

### 4.1 구현 범위 및 주요 기능 요약

SmaPF는 반려동물의 행동과 개체에 따라 자동으로 맞춤형 급식을 제공하는 IoT 시스템이다. 전체 시스템은 모바일 애플리케이션과 라즈베리파이 기반 하드웨어가 연동되는 구조로 구성되었으며, 실제 환경에서 동작 가능한 수준으로 주요 기능을 완성하였다.

이번 프로젝트에서 구현된 주요 기능은 다음과 같다:

- Flutter 앱을 통해 사용자가 녹음한 음성 파일을 서버로 업로드하고, 급식 시작 시점을 타이머로 예약할 수 있는 기능을 구현하였다. 설정된 시간에 맞춰 라즈베리파이 측에서 음성 재생이 자동으로 이루어진다.

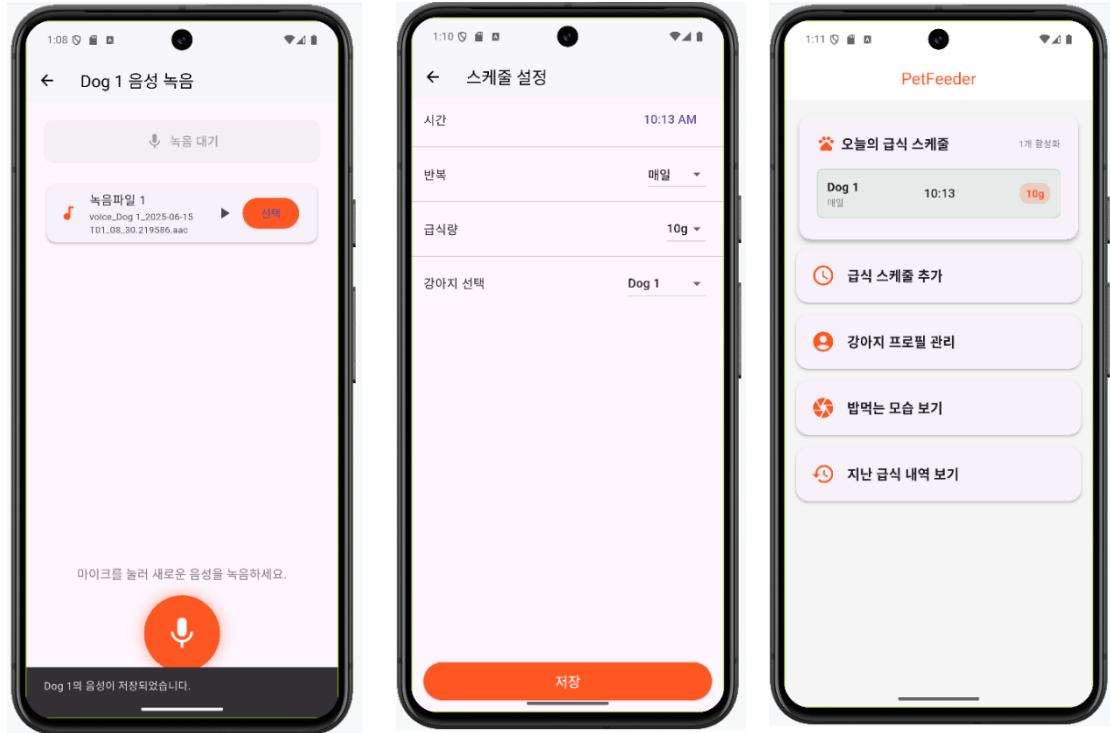


Fig 4. 실제 앱의 스케줄 설정 관련 화면

- 라즈베리파이는 초음파 센서(HY-SRF05)를 이용해 반려동물이 급식기 근처에 접근했는지를 감지한다. 감지 시 카메라가 활성화되며, YOLO 모델을 통해 개별 반려동물의 식별이 이루어진다.
- 인식된 개체에 따라 사전에 정의된 급식량을 기준으로 서보모터(SG-90)가 사료 투입구를 개방하며, 로드셀(HX711)을 이용해 목표 중량 도달 여부를 확인한 후 사료 투입을 자동 종료한다.
- 전체 제어 흐름과 데이터 전달은 Flask 기반 서버를 통해 수행되며, 앱과 라즈베리파이 간의 통신을 중계하는 역할을 한다.

핵심 기능들이 유기적으로 연결되어 작동하도록 구현되었으며, 실제 실험 환경에서도 정상적으로 동작하는 것이 확인되었다. 본 프로젝트는 제한된 시간과 자원 속에서도 IoT 기술과 간단한 AI 모델을 융합한 스마트 펫케어 시스템의 가능성을 입증하였다.

## 4.2 실제 작동 흐름 예시

SmaPF는 보호자가 미리 설정한 시간에 맞추어 자동으로 음성 메시지를 재생하고, 반려동물의 접근을 감지하여 AI 기반 급식을 수행하는 일련의 흐름을 갖는다. 전체 과정은 다음과

같은 순서로 진행된다.

1. 사용자가 앱에서 음성 녹음과 타이머 설정을 완료한다. 이 설정은 Flask 서버를 통해 라즈베리파이로 전송되며, 지정된 시간에 맞춰 동작이 시작된다.
2. 설정된 시간이 되면, 라즈베리파이는 녹음된 음성 파일을 재생한다. 이 음성은 반려동물의 관심을 끌기 위한 것으로, 급식이 곧 이루어짐을 알리는 역할을 한다.
3. 음성 재생과 동시에 초음파 센서(HY-SRF05)가 작동을 시작하여 일정 거리 이내에 반려동물이 접근했는지를 실시간으로 감지한다.
4. 감지가 이루어지면 카메라가 활성화되고, YOLO 모델을 통해 반려동물을 인식한다. 인식된 클래스(예: 강아지 A, 강아지 B 등)에 따라 사료의 투입량이 달라진다.
5. 인식 결과에 따라 라즈베리파이는 서보 모터(SG-90)를 작동시켜 사료 투입구를 개방하고, 사료가 투입되기 시작한다.
6. 동시에 로드셀(HX711) 센서가 사료의 무게를 실시간으로 측정하고, 사전에 설정된 목표 무게에 도달하면 서보 모터를 닫아 사료 투입을 자동으로 종료한다.

이와 같은 일련의 흐름을 통해 SmaPF는 보호자의 개입 없이도 반려동물의 행동과 상태에 따라 맞춤형 급식을 제공할 수 있는 자동화된 구조를 갖추고 있다. 시스템은 사용자의 타이머 설정을 시작으로, 각 센서와 AI 인식 결과를 기반으로 하드웨어 동작을 순차적으로 제어하는 형태로 설계되어 있으며, 이는 실제 테스트 환경에서도 안정적으로 작동하는 것을 확인하였다.

## 5. 기술적 도전과 해결 방안

### 5.1 주요 기술적 어려움

SmaPF의 개발 과정에서는 여러 IoT 센서와 인공지능 모델이 동시에 작동하는 구조를 구축해야 했기에, 다양한 기술적 어려움이 발생하였다. 특히 다음과 같은 세 가지 영역에서 도전 과제가 두드러졌다.

첫째, **센서 간 통합 및 하드웨어 제어의 복잡성**이다. 초음파 센서(HY-SRF05), 서보모터(SG-90), 로드셀(HX711), 카메라 모듈 등을 동시에 제어해야 하는 라즈베리파이 환경에서 각 장치의 전원 공급, 신호 타이밍, 통신 프로토콜 등이 상이하여 회로 구성과 납땜 과정에서 잦은 오류와 하드웨어 간 간섭이 발생하였다. 특히 납땜 불량이나 접촉 불량으로 인해 센서가 비정상적으로 작동하거나 오작동을 반복하는 문제가 있었다.

둘째, **YOLO 객체 탐지 모델의 근거리 식별 한계**다. 프로젝트에 적용된 YOLO 모델은 주

로 일반 객체 탐지를 위해 설계된 알고리즘이기 때문에, 급식기와 매우 가까운 거리에서 촬영된 반려동물의 이미지에서는 정확한 객체 구분이 어려웠다. 얼굴이나 형태가 지나치게 확대되어 인식이 불안정해지는 현상이 나타났으며, 이에 따라 라벨링 기반 커스텀 데이터셋을 수집하고 학습시키는 과정이 추가로 필요했다.

셋째, **라즈베리파이와 애플리케이션 간의 통신 안정성 확보**도 도전 과제였다. Flask 서버를 통해 라즈베리파이와 앱 간 HTTP 요청을 처리해야 했는데, 일정 간격으로 보내는 타이머 기반 요청이나 파일 업로드/다운로드 과정에서 간헐적인 지연 및 타이밍 오류가 발생하였다. 특히 모바일 앱과 서버 간 오디오 파일을 공유하고 예약 시간에 맞춰 정확하게 재생되도록 동기화하는 작업은 많은 테스트와 수정이 필요했다.

이 외에도 사료통 구조적 문제로 인해 사료 입구가 막히는 물리적 장애도 발생하였다. 특히 습식 사료 혹은 조각이 큰 사료를 사용하는 경우 입구가 막히는 일이 잦았고, 이에 따라 사료 배출 구조를 수정하거나 서보모터 동작 시간을 조절하는 시도도 병행되었다.

## 5.2 문제 해결 전략 및 창의적 접근

이러한 문제를 해결하기 위해 다음과 같은 접근 방식을 활용하였다.

- **하드웨어 보완 및 반복 테스트:** 초음파 센서와 로드셀의 신호 품질을 높이기 위해 별도의 회로 보강 및 납땜 수정이 이루어졌으며, 센서 간 간섭을 방지하기 위해 각 센서의 작동 순서를 논리적으로 구분하였다.
- **YOLO 모델 커스터마이징:** 근거리 촬영 조건에 맞는 반려견 이미지셋을 직접 수집하고, 이에 대해 추가 라벨링을 진행하여 YOLO 모델을 재학습시킴으로써 인식률을 개선했다.
- **통신 안정성 확보:** Flask 서버의 라우팅 구조를 단순화하고, 응답 대기 시간과 요청 처리 시점을 조정하여 앱과의 연동 안정성을 높였다. 이를 통해 타이머 기반 녹음 재생 및 센서 시퀀스 동기화 과정이 원활하게 작동하도록 조정하였다.
- **기구적 설계 개선:** 사료 투입구의 경사와 입구 지름을 조정하고, 사료 무게에 따라 모터의 개방 시간을 변경하는 등 하드웨어 구조 자체를 개선하여 입구 막힘 현상을 줄였다.

결과적으로 이러한 기술적 도전과 문제 해결 과정을 통해, 본 프로젝트는 제한된 환경에서도 IoT 기반 스마트 급식의 전 과정을 구현할 수 있었고, 하드웨어와 소프트웨어의 통합적 제어에 대한 실질적인 경험을 얻을 수 있었다

## 6. 프로젝트 수행에 대한 고찰 및 개선점

### 6.1 프로젝트를 통해 얻은 기술적 고찰

SmaPF 프로젝트를 수행하며 팀은 단순한 기능 구현을 넘어서 **IoT 시스템의 통합적 설계와 실제 동작 간의 간극**을 체감할 수 있었다. 특히 센서 제어, 모터 작동, AI 인식 결과의 연동을 하나의 시나리오로 연결하는 과정에서 **하드웨어와 소프트웨어의 조율**이 얼마나 정교하게 이루어져야 하는지에 대한 현실적인 인사이트를 얻었다.

또한, AI 기반 시스템에서의 **데이터셋의 중요성**을 직접 체험하였다. YOLO 모델의 커스터마이징을 위해 강아지의 근거리 이미지를 직접 수집하고 라벨링하는 과정은, 데이터 품질이 모델 성능에 직결된다는 사실을 명확히 보여주었다. 이 과정에서 데이터 편향 및 학습 오류 등의 문제도 발생하였으며, AI 시스템 개발에 있어 데이터 전처리 및 학습 설계의 중요성을 실감할 수 있었다.

아울러, 라즈베리파이를 활용한 센서 제어 및 Flask 기반 서버 구현은 팀원들에게 **임베디드 시스템의 실무적 경험**을 제공하였다. 특히, 웹 서버를 통한 실시간 제어 및 모바일 앱과의 통신 연동 등은 단순한 로컬 제어가 아닌 **클라우드 연계 IoT 구조의 기반 설계 경험**으로 이어졌다.

### 6.2 프로젝트 완성도를 높이기 위한 개선점

프로젝트는 기본적인 작동 흐름과 기능을 갖춘 상태로 마무리되었지만, 완성도를 높이기 위해 다음과 같은 측면에서 보완이 필요하다.

#### 1. 회원 시스템 및 로그 기록 기능

사용자 기반으로 급식기 작동 이력을 기록하거나, 반려동물의 급식 데이터를 저장·분석하는 기능은 아직 구현되지 않았다. 향후 백엔드 서버와 데이터베이스를 연동하여 이러한 기능을 도입한다면 시스템의 실효성과 사용자 경험이 크게 향상될 것이다.

#### 2. 사료 투입 구조의 기계적 안정성

프로젝트 중 사료 입구가 막히는 문제는 반복적으로 발생하였으며, 이는 단순한 코드나 센서 보정으로 해결하기 어려운 하드웨어 설계의 한계로 나타났다. 향후에는 사료 구조에 맞는 기구 설계를 병행하거나, 복수 개의 센서로 피드백 제어를 구현하는 방식이 필요하다.

#### 3. YOLO 탐지의 정확도 및 신뢰성 확보

근거리 이미지에서 객체 인식 정확도가 낮았던 문제는 커스텀 학습으로 일정 부분 보완되었으나, 여전히 조명, 거리, 자세 변화 등에 따라 인식이 불안정한 문제가 있다. 향후 모델 경량화와 동시에 후처리 방식(예: 인식 반복 횟수 기반 보정 등)을 도

입하면 개선 가능성이 있다.

#### 4. 센서 신호 정합과 에러 처리 구조 강화

실험 중 간헐적으로 센서 오작동 또는 신호 누락이 발생하였다. 보다 안정적인 동작을 위해 센서 간 인터럽트 우선순위 조정, 예외 상황 발생 시 대체 루틴 삽입, 타임아웃 처리 로직 등 정교한 예외 제어 구조가 추가되어야 한다.

종합적으로 본 프로젝트는 IoT 시스템의 구성 요소가 실제로 동작하기 위해 요구되는 **하드웨어 설계, 데이터 처리, 통신 안정성** 등의 측면에서 다각적인 경험을 제공하였으며, 이러한 경험은 향후 보다 복잡하고 정교한 스마트 시스템 개발에 중요한 기반이 될 것이다.

## 7. 결론 및 향후 연구 방향

### 7.1 프로젝트 요약 및 의의

본 프로젝트 SmaPF(Smart PetFeeder)는 반려동물의 개별 특성에 따라 맞춤형 급식을 제공하는 지능형 IoT 시스템을 목표로 설계·구현되었다. 사용자가 설정한 시간에 보호자의 음성을 재생하고, 초음파 센서를 통해 반려동물의 접근 여부를 감지한 뒤, YOLO 기반 객체 인식을 통해 해당 반려견을 식별한다. 이후, 사전에 설정된 급식량에 따라 서보 모터가 작동하여 사료를 배급하며, 로드셀 센서를 통해 실시간으로 사료 투입량을 감지하고 목표 중량 도달 시 자동으로 중단되는 구조이다.

해당 시스템은 단순한 타이머 기반 자동 급식기에서 한 단계 진화한 형태로, 반려동물의 개별 식습관을 반영할 수 있는 구조를 갖추었다. 특히, YOLO 기반 실시간 객체 인식과 다중 센서 제어의 연계는 기존 시제품 자동 급식기와는 차별화된 정밀성과 지능화를 지향한 실질적인 구현 사례라 할 수 있다.

SmaPF 프로젝트는 **IoT 설계의 이론적 이해를 실제 구현으로 전환하는 실습형 경험**으로써, 하드웨어, AI, 모바일 개발이 복합적으로 통합된 환경에서의 협업 역량을 배양하는 데 큰 의의를 가진다.

### 7.2 향후 발전 방향 및 기대 효과

현재 시스템은 기본적인 급식 시퀀스를 정상적으로 수행할 수 있는 수준까지 구현되었으며, 다음과 같은 방향으로의 고도화가 가능하다.

#### 1. 사용자 기반 로그 저장 및 분석 기능 도입

급식 이력, 인식 통계, 반려동물의 접근 시간 등을 자동으로 저장하고 시각화하는 기능을 통해 보호자가 반려동물의 식습관을 정량적으로 관리할 수 있도록 개선할 수 있다. 이를 위해 서버-DB 연동 구조와 백엔드 구축이 향후 필수적이다.

## 2. 실시간 스트리밍 및 원격 제어 기능 강화

보호자가 외부에서도 실시간으로 급식 상황을 확인하고 직접 개입할 수 있도록 영상 스트리밍 및 제어 기능을 앱에 추가하는 방향으로 기능 확장이 가능하다.

## 3. 하드웨어 구조 개선 및 상용화 고려

현재 구현된 시스템은 실험 환경에 기반한 프로토타입 수준이며, 장기 운용을 위한 기계적 구조 개선, 센서 내구성 강화, 전원 안정화 설계가 추가되어야 한다. 나아가 상용 제품과 경쟁하기 위한 외관 디자인 및 사용자 친화적 설치 방법도 고려될 수 있다.

## 4. AI 인식 성능 고도화 및 다양화

YOLO 모델 외에도 경량화된 MobileNet, EfficientDet 등 다양한 모델을 적용하고, 단일 객체 인식 외에도 행동 인식까지 확대할 수 있다면, 반려동물의 상태에 맞는 급식 제어가 가능할 것이다.

본 프로젝트를 통해 확인된 기술적 가능성과 개선 여지는 향후 더 정교한 반려동물 관리 시스템으로 확장될 수 있으며, 단순한 자동화 기기를 넘어, **실제 생활에 적용 가능한 스마트 IoT 서비스의 기초 토대**가 될 수 있을 것이다.

## Appendix

### GitHub Repository :

본 프로젝트의 전체 소스코드로, 아래 GitHub 저장소에서 확인할 수 있습니다.

<https://github.com/DoIkk/AutoFeeder>