

The background of the slide is a solid light blue. It is decorated with various white and teal icons: gears, clouds, a large white cloud with a black Wi-Fi signal icon above it, a smartphone with a Wi-Fi icon, a tablet with a Wi-Fi icon, and a desktop monitor with a Wi-Fi icon. A semi-transparent light blue rectangle is positioned behind the text.

# WiBi

TEAM MOBICOM



# MOBICOM

- 팀장: 이경재

- 팀원: 양재영

- 팀원: 박수찬

- 팀원: 김태기



# 목차

- 프로젝트 소개

- 진행 과정

- 시연 동영상

- 기대 효과



# 프로젝트 소개

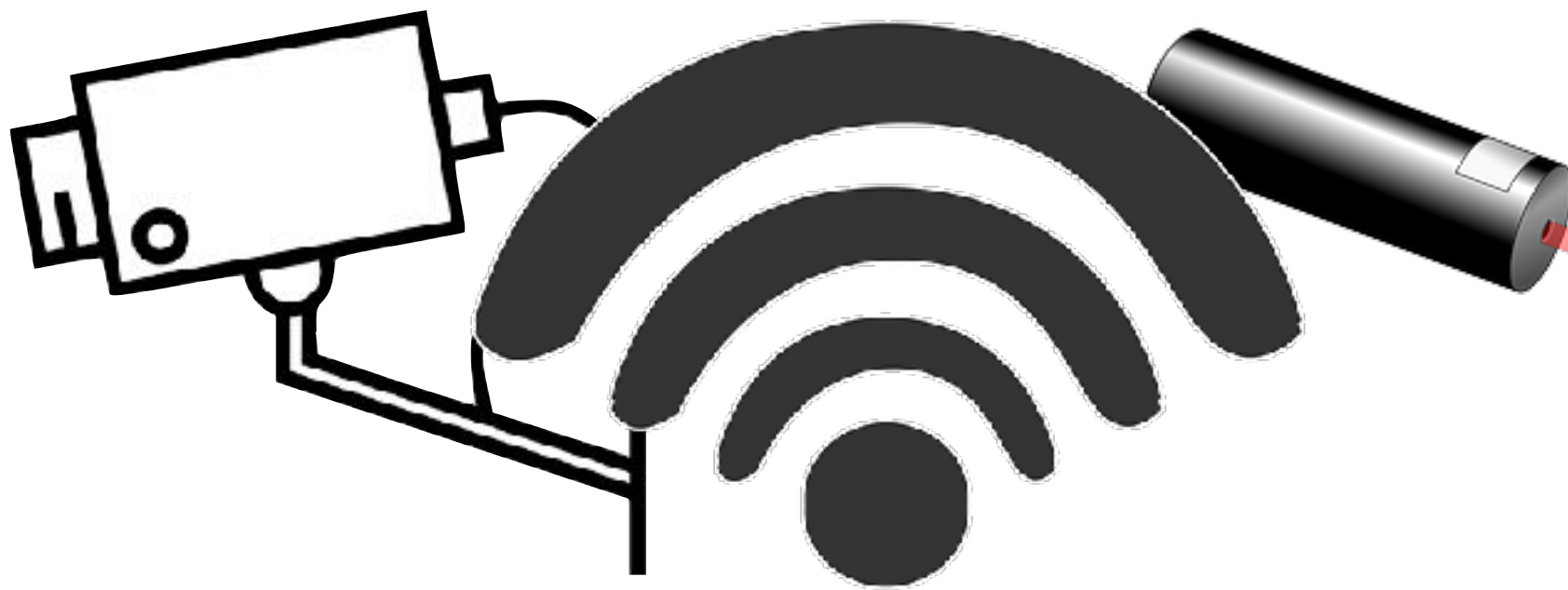


## 1. 개발 배경

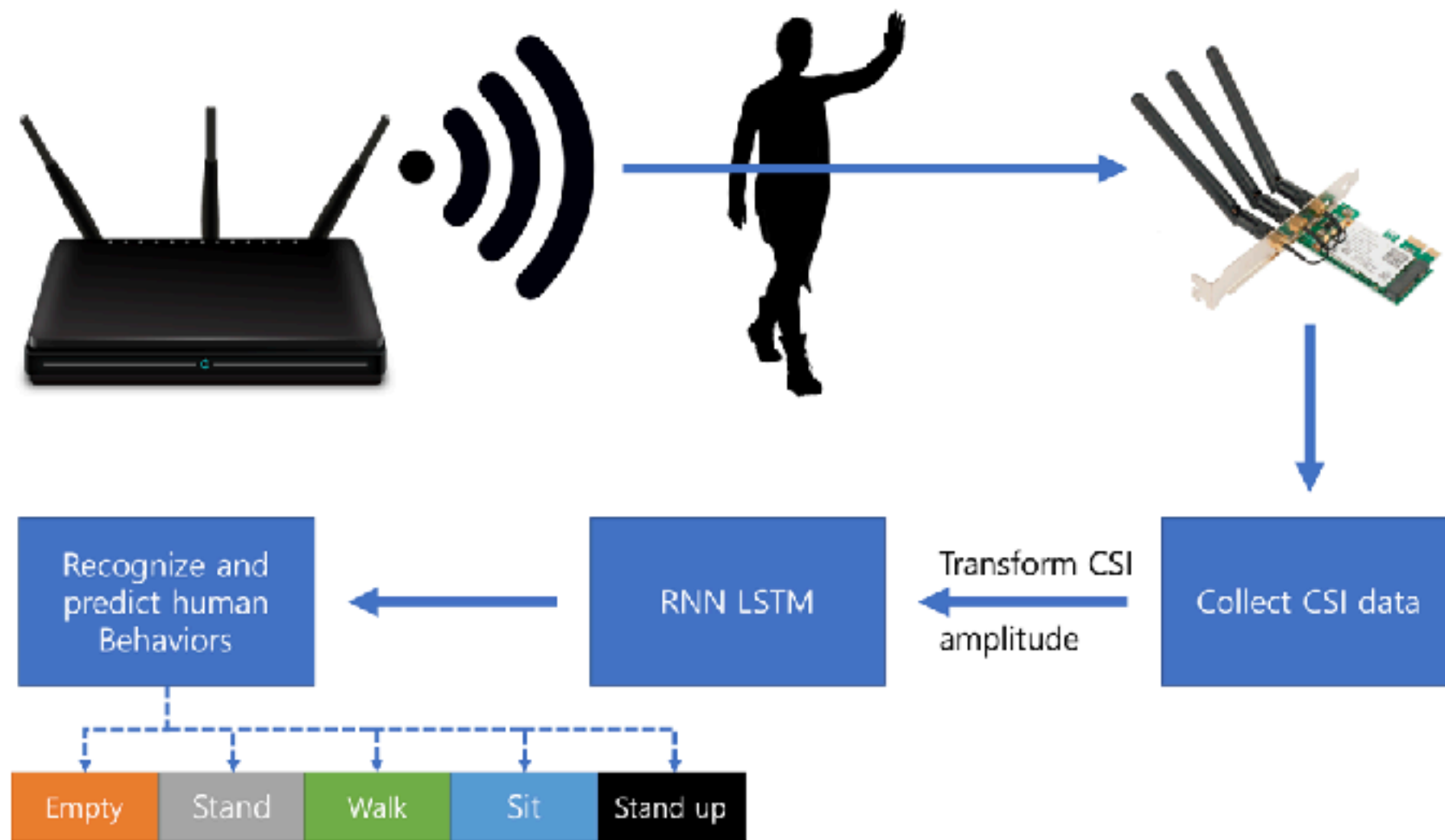


## 2. 흐름도

## (1) 개발 배경



## (2) 흐름도




send packet

human  
behavior

receive  
packet

convert to  
csv file

to be learned  
and predict



# 진행 과정



1. 패킷 전송으로 사람의 행동 데이터 셋 만들기



2. 데이터 셋을 이용한 학습 단계



3. 학습된 데이터를 이용한 예측 단계



# (1) 데이터 셋 생성

```
krul@kali:~/linux-00211-supplementary/injection$  
..... 155k  
..... 156k  
..... 157k  
..... 158k  
..... 159k  
..... 160k  
..... 161k  
..... 162k  
..... 163k  
..... 164k  
..... 165k  
..... 166k  
..... 167k  
..... 168k  
..... 169k  
..... 170k  
..... 171k  
..... 172k  
..... 173k  
..... 174k  
..... 175k  
..... 176k  
..... 177k  
..... 178k  
..... 179k  
..... 180k  
..... 181k  
..... 182k  
..... 183k  
..... 184k  
..... 185k  
..... 186k  
..... 187k  
..... 188k  
..... 189k  
..... 190k  
..... 191k  
..... 192k  
..... 193k  
..... 194k  
..... 195k  
..... 196k  
..... 197k  
..... 198k  
..... 199k  
..... 200k
```



```
krul@kali:~/linux-00211-supplementary/netlink$  
00605 received 213 bytes: fd: 26 val: 1 seq: 0 clens: 213  
00606 received 213 bytes: fd: 26 val: 1 seq: 0 clens: 213  
00607 received 213 bytes: fd: 26 val: 1 seq: 0 clens: 213  
00608 received 213 bytes: fd: 26 val: 1 seq: 0 clens: 213  
00609 received 213 bytes: fd: 26 val: 1 seq: 0 clens: 213  
00610 received 213 bytes: fd: 26 val: 1 seq: 0 clens: 213  
00611 received 213 bytes: fd: 26 val: 1 seq: 0 clens: 213  
00612 received 213 bytes: fd: 26 val: 1 seq: 0 clens: 213  
00613 received 213 bytes: fd: 26 val: 1 seq: 0 clens: 213  
00614 received 213 bytes: fd: 26 val: 1 seq: 0 clens: 213  
00615 received 213 bytes: fd: 26 val: 1 seq: 0 clens: 213  
00616
```

send packet

human behavior

Receive packet



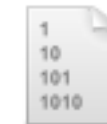
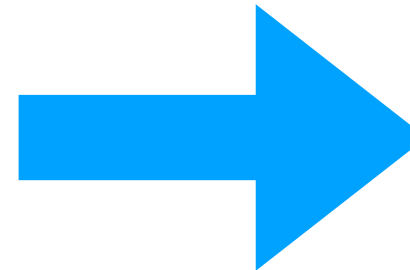
## (2) 학습 단계

```
# Keep training until reach max iterations
while step < training_iters:
    batch_x, batch_y = wifi_train.next_batch(batch_size) #wifi_train에 저장된 x와 y를 배
    x_vali = wifi_validation.images[:]
    y_vali = wifi_validation.labels[:]
    # Reshape data to get 28 seq of 28 elements
    batch_x = batch_x.reshape((batch_size, n_steps, n_input)) #batch_x를 500행 90열 ba
    x_vali = x_vali.reshape((-1, n_steps, n_input)) #x_vali를 500행, 90열로 만들고 남은걸
    # Run optimization op (backprop)
    sess.run(optimizer, feed_dict={x: batch_x, y: batch_y})

    # Calculate batch accuracy
    acc = sess.run(accuracy, feed_dict={x: batch_x, y: batch_y})
    #x에 batch_x, y에 batch_y를 입력하여 accuracy 실행하여 결과값을 acc에 저장
    acc_vali = sess.run(accuracy, feed_dict={x: x_vali, y: y_vali})
    # Calculate batch loss
    loss = sess.run(cost, feed_dict={x: batch_x, y: batch_y})
    loss_vali = sess.run(cost, feed_dict={x: x_vali, y: y_vali})

    # Store the accuracy and loss
    train_acc.append(acc)
    train_loss.append(loss)
    validation_acc.append(acc_vali)
    validation_loss.append(loss_vali)

    if step % display_step == 0:
        print("Iter " + str(step) + ", Minibatch Training Loss= " + \
              "{:.5f}".format(loss) + ", Training Accuracy= " + \
              "{:.5f}".format(acc) + ", Minibatch Validation Loss= " + \
              "{:.5f}".format(loss_vali) + ", Validation Accuracy= " + \
              "{:.5f}".format(acc_vali) )
    step += 1
```



model.ckpt.data-  
00000-of-00001



model.ckpt.index



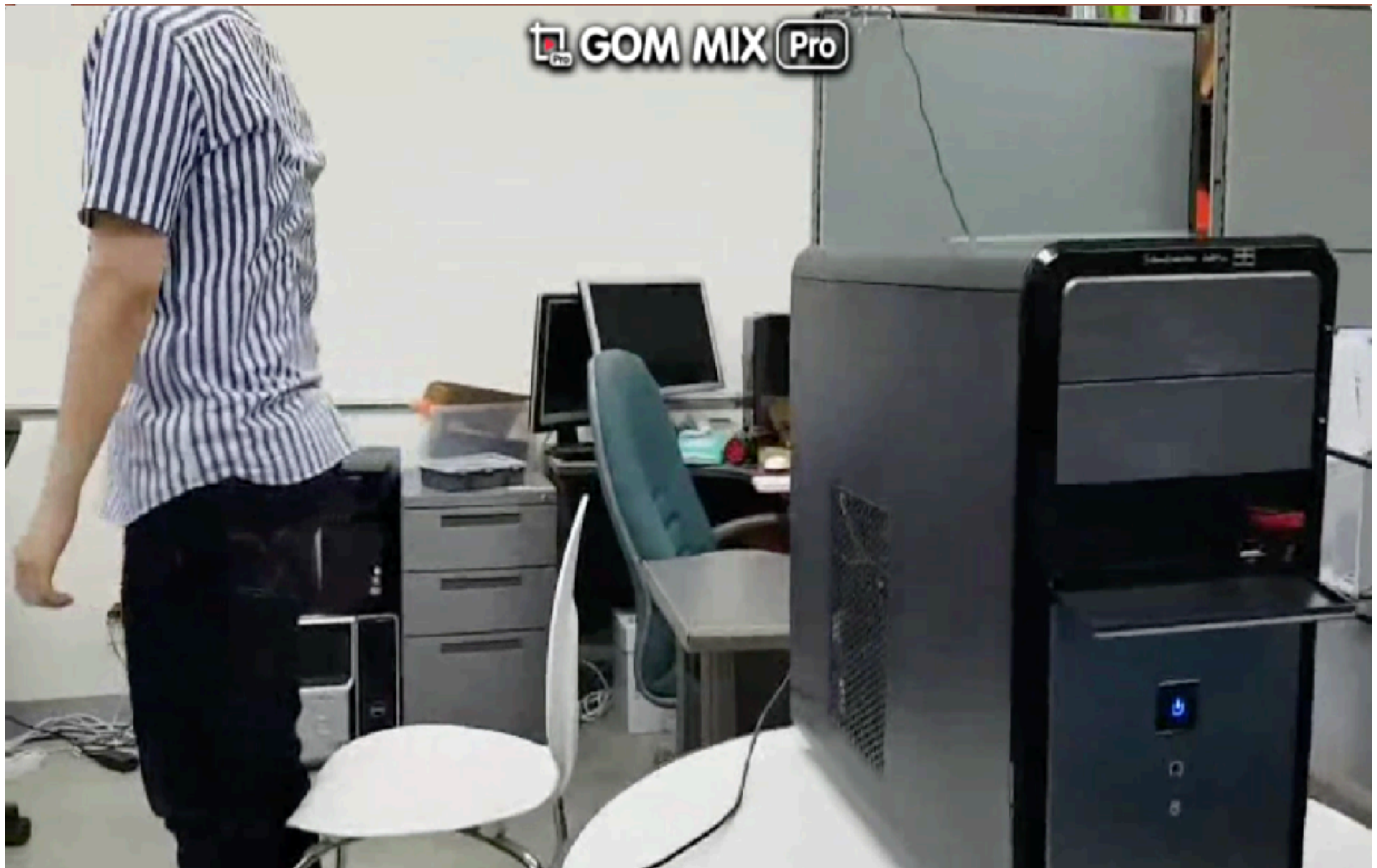
model.ckpt.meta

### (3) 예측 단계

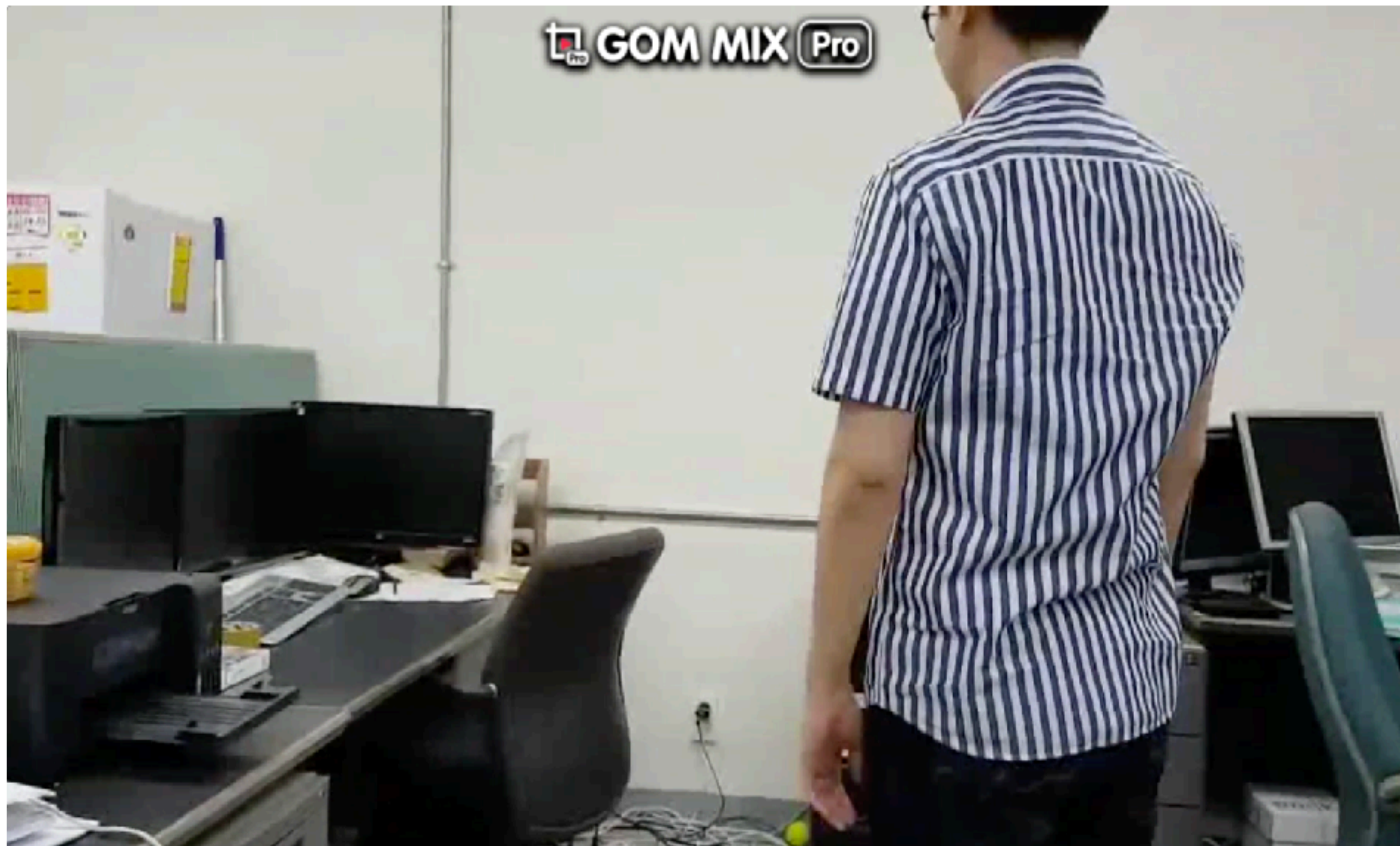




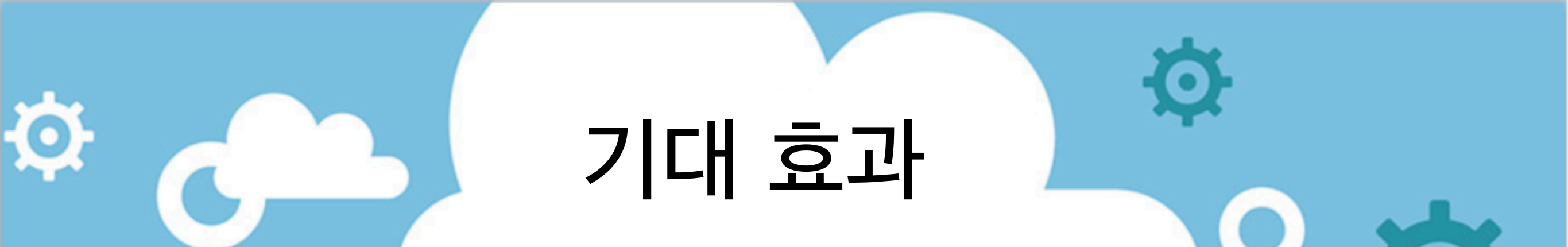
# 시연 동영상



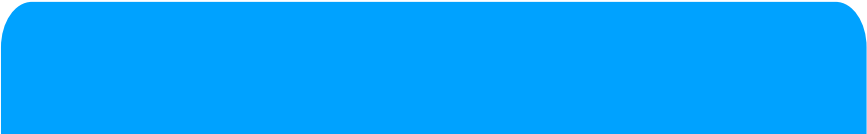
# 시연 동영상







# 기대 효과



- 무인 자동차



- Home IoT



- 감시 체계