

Human perception

Human-like Pepper robot

In the chapter on Pepper's Autonomous Activity, we learnt that the Pepper robot itself already has BasicAwareness, which can be used in scenarios where a user wants to interact with Pepper intentionally, and it will actively approach the user and look at the user who wants to interact with it.

The Pepper robot is a humanoid robot that can autonomously and intentionally interact with people and participate in the world like a human being. Typically, the steps in actively seeking to interact with a person are:

- When a human actively and intentionally approaches Pepper, Pepper will actively approach the human, ready to interact with the human. 2.
- Pepper follows certain rules to identify a person to interact with. 3.
- After determining the person to interact with, Pepper can interact with the person through voice, body movement and other means.

Environment Sensing

Pepper is a robot that mainly interacts with human beings, and it can sense the environment and actively contact and communicate with human beings. In this section, we introduce Pepper's perception of people in the environment. Firstly, it needs to recognise a person, then actively approach the person, and further interact with the person. Following the above steps of perception, we will introduce Human perception, ApproachHuman and EngageHuman interactions one at a time.

1. Sensing the Human

When Pepper senses its surroundings, it focuses on the humans in its environment and obtains information about their characteristics.

Pepper perceives humans in two ways, either as all people in the environment or as a single person.

human perception

Call the getHumansAround() method to get the people around you.

```
HumanAwareness humanAwareness = qiContext.getHumanAwareness();
List<Human> humansAround = humanAwareness.getHumansAround();
```

Single Person Awareness

Pepper selects people who are interested in interacting with it, calling the getEngagedHuman() method to get the individuals who want to participate in the interaction.

```
HumanAwareness humanAwareness = qiContext.getHumanAwareness();
Human engagedHuman = humanAwareness.getEngagedHuman();
```

2. ApproachHuman

When Pepper senses that someone wants to interact with it, instead of waiting for the person to approach it, it moves directly to the person of its own accord.

Who will Pepper approach?

Pepper will actively try to approach people who are interested in it, which is not very easy for Pepper to do. In the QiSDK API, the getRecommendedHumanToApproach method in HumanAwareness can give a recommendation on which person is interested in it.

```
HumanAwareness humanAwareness = qiContext.getHumanAwareness();
Human recommendedHuman = humanAwareness.getRecommendedHumanToApproach();
```

Note: The getRecommendedHumanToApproach method will return null if there is no human to recommend.

Pepper provides this method based on people with intent to interact and is related to the proximity of the person to the robot.

How does approach?

When Pepper finds a person it needs to interact with, it can create an ApproachHuman action to perform the approach action:

```
ApproachHuman approachHuman = ApproachHumanBuilder.with(qiContext)
    .withHuman(recommendedHuman)
    .build();

approachHuman.async().run();
```

Note: Always remember to close the power switch cover when running the ApproachHuman action.

3. Preparing to manage interactions

When the Pepper robot is ready to interact with a human, it needs its eyes to focus on a particular person. As mentioned in the previous section on Pepper's autonomy, EngageHuman itself is part of BasicAwareness and does not require us to implement it.

However, in the following scenarios, we do need to implement EngageHuman ourselves, for example:

- Manually controlling the Pepper's interaction based on the interaction policy of your own programme.
- Pepper's interactions may be lost or shifted, and these messages can be captured in your own line of sight.
- Multiple interaction functionality (switching interactions between different people) can be implemented.

Interaction Lifecycle

The interaction lifecycle of the Pepper robot is divided into three phases:

- Preparing for interaction phase: Pepper actively attracts people or waits for people to approach it.
- Interaction phase: Pepper has focused on the person it is interacting with.
- Stop Interaction Phase: Pepper can realise that the person it is interacting with has left and stops the interaction.

Preparing for interaction phase

Similarly, Pepper first knows the person it needs to interact with, it can get the person it wants to interact with through the getRecommendedToEngage method of HumanAwareness:

```
HumanAwareness humanAwareness = qiContext.getHumanAwareness();
Human recommendedHuman = humanAwareness.getRecommendedHumanToEngage();
```

Interaction Stage

When the person you want to interact with is found, an EngageHuman action is created to perform the interaction:

```
EngageHuman engageHuman = EngageHumanBuilder.with(qiContext)
    .withHuman(recommendedHuman)
    .build();

engageHuman.async().run();
```

Stopping the interaction phase

The OnHumanIsDisengagingListener listener allows to detect that the interacting human has left.

```
EngageHuman engageHuman = EngageHumanBuilder.with(qiContext)
    .withHuman(recommendedHuman)
    .build();

Say say = SayBuilder.with(qiContext)
    .withText("Goodbye!")
    .build();

engageHuman.addOnHumanIsDisengagingListener(() -> {
    say.run();
    engagement.requestCancellation();
});

engagement = engageHuman.async().run();
```

For more information, [refer here](#)