# Take a picture

# Function

This section teaches you how to take a picture by using the monocular camera on the head of the Pepper robot.

## **Steps**

Use the QiSDK API to implement the following functions:

### 1. Take a picture

If you want to get the picture information, first perform the TakePicture behaviour and return an instance of TimestampedImageHandle as follows:

```
// Create the behaviour
TakePicture takePicture = TakePictureBuilder.with(qiContext).build();
// Execute the behaviour asynchronously
```

#### TimestampedImageHandle result = takePicture.run();

### 2. Timestamped

The returned TimestampedImageHandle instance gets the timestamp of the picture via the getTime() method as follows:

```
Log.i(TAG, 'Picture taken at: ' + result.getTime());
```

#### 3. Picture data

Since Pepper uses the camera on top of its head to take pictures, the pictures taken are first saved in Pepper's head, if you want to get the picture data on the tablet, you need to copy the picture data taken in Pepper's head to the tablet. Depending on the size of the image, it will take some time, so the previous run() method can't return the copied data directly, but we can get it by proxy. We need to call the getImage() method of the TimestampedImageHandle instance to get the EncodedImageHandle instance, and get the data information of the image through its getValue() method, as follows:

```
// 1. Getting the data through the proxy
EncodedImageHandle encodedImageHandle = result.getImage(); ``java // 1. Get the data through the pro// 2. Copy the remote data via the getValue() method.
EncodedImage encodedImage = encodedImageHandle.getValue(); // 3. Copy the remote data using the Byte// 3. Use a ByteBuffer to receive the returned data.
ByteBuffer data = encodedImage.getData();
```

#### 4. Show image

After getting the image ByteBuffer data, it needs to be converted into Bitmap and can be displayed on Pepper's tablet.

```
// Recalculate the position of the buffer.
data.rewind();
final int pictureBufferSize = data.remaining();
final byte[] pictureArray = new byte[pictureBufferSize];
data.get(pictureArray);
Bitmap pictureBitmap = BitmapFactory.decodeByteArray(pictureArray, 0, pictureBufferSize);
// Update the ImageView in the main UI thread.
runOnUiThread(() -> {
    mImageView.setImageBitmap(pictureBitmap);
});
```

### Reference

To learn more about using the TakePicture class, please refer to:

- TakePicture's javadoc
- javadoc for TimestampedImageHandle. TimestampedImageHandle.html) \* How to use and understand the [PepperPicture.html].
- How to use and understand Pepper's take-picture function