

camera2 调节焦距

在使用 Android 的 Camera2 API 时，可以通过调节 `CaptureRequest.CONTROL_AF_MODE` 和 `CaptureRequest.LENS_FOCUS_DISTANCE` 来控制相机的焦距（即对焦距离）。下面是如何在 Kotlin 中实现的示例：

1. **设置自动对焦模式**：在开始手动调节焦距之前，确保将自动对焦模式设置为 `CONTROL_AF_MODE_OFF`。
2. **手动调节焦距**：通过设置 `LENS_FOCUS_DISTANCE` 来手动调节焦距。焦距的值是以无穷远为 0，最近对焦距离为 1 的比例。

```
import android.hardware.camera2.CameraCaptureSession
import android.hardware.camera2.CameraDevice
import android.hardware.camera2.CameraManager
import android.hardware.camera2.CaptureRequest
import android.hardware.camera2.CaptureResult
import android.hardware.camera2.TotalCaptureResult
import android.os.Handler
import android.util.Log

// 假设你已经通过 CameraManager 获取了 CameraDevice 和 CameraCaptureSession
fun adjustFocusDistance(cameraDevice: CameraDevice, cameraCaptureSession:
    CameraCaptureSession, focusDistance: Float) {
    val captureRequestBuilder =
        cameraDevice.createCaptureRequest(CameraDevice.TEMPLATE_PREVIEW).apply {
            // 设置自动对焦模式为关闭
            set(CaptureRequest.CONTROL_AF_MODE,
                CaptureRequest.CONTROL_AF_MODE_OFF)

            // 调节焦距
            set(CaptureRequest.LENS_FOCUS_DISTANCE, focusDistance)
        }

    cameraCaptureSession.setRepeatingRequest(captureRequestBuilder.build(),
        object : CameraCaptureSession.CaptureCallback() {
            override fun onCaptureCompleted(session: CameraCaptureSession,
                request: CaptureRequest, result: TotalCaptureResult) {
                super.onCaptureCompleted(session, request, result)
                Log.d("CameraFocus", "Focus distance adjusted: $focusDistance")
            }
        }, Handler())
}

// 使用示例
val focusDistance: Float = 0.5f // 设置所需的焦距 (0.0f 到 1.0f)
adjustFocusDistance(cameraDevice, cameraCaptureSession, focusDistance)
```

重要说明：

- LENS_FOCUS_DISTANCE 的值范围从 0.0f（无穷远）到 1.0f（最近对焦距离），但具体值要根据相机的焦距范围进行调整。
- 并非所有设备都支持手动调焦，因此在使用之前应检查设备是否支持 LENS_FOCUS_DISTANCE。

你可以通过查询相机的特性来检查设备是否支持这些功能。

要判断手机是否支持手动调节焦距，你可以使用 CameraCharacteristics 类中的一些属性进行检查。具体来说，你需要检查设备是否支持 LENS_FOCUS_DISTANCE 控件。以下是在 Kotlin 中如何实现这个检查的示例：

```
import android.content.Context
import android.hardware.camera2.CameraCharacteristics
import android.hardware.camera2.CameraManager
import android.hardware.camera2.CameraMetadata
import android.util.Log

fun isManualFocusSupported(context: Context): Boolean {
    // 获取 CameraManager 实例
    val cameraManager = context.getSystemService(Context.CAMERA_SERVICE) as CameraManager

    try {
        // 遍历所有的摄像头
        for (cameraId in cameraManager.cameraIdList) {
            // 获取摄像头的特性
            val characteristics = cameraManager.getCameraCharacteristics(cameraId)

            // 获取镜头特性信息
            val lensFocusCapabilities = characteristics.get(CameraCharacteristics.CONTROL_AF_AVAILABLE_MODES)

            // 检查是否支持 OFF 模式（即手动对焦模式）
            if (lensFocusCapabilities != null && lensFocusCapabilities.contains(CameraMetadata.CONTROL_AF_MODE_OFF)) {

                // 检查是否存在 LENS_FOCUS_DISTANCE 特性
                val focusRange = characteristics.get(CameraCharacteristics.LENS_INFO_MINIMUM_FOCUS_DISTANCE)
```

```

        if (focusRange != null && focusRange > 0) {
            // 如果有最小对焦距离信息，表示支持手动调焦
            return true
        }
    }
} catch (e: Exception) {
    Log.e("CameraCheck", "Error checking manual focus support", e)
}

// 如果没有找到任何支持的摄像头，则返回 false
return false
}

```

解释：

1. **检查自动对焦模式：**代码首先获取 `CONTROL_AF_AVAILABLE_MODES`，查看是否支持 `CONTROL_AF_MODE_OFF`，因为手动调焦需要关闭自动对焦。
2. **检查最小对焦距离：**接着检查 `LENS_INFO_MINIMUM_FOCUS_DISTANCE` 是否存在和大于 0。该值存在并且大于 0 表明设备支持手动调节焦距。

使用示例：

```

val isSupported = isManualFocusSupported(context)
if (isSupported) {
    Log.d("FocusCheck", "This device supports manual focus adjustment.")
} else {
    Log.d("FocusCheck", "Manual focus adjustment is not supported on this device.")
}

```

通过这段代码，你可以判断当前设备是否支持手动调焦功能。