

camera2 散景

实现散景效果 (Bokeh effect) 在 Camera2 API 中是一个复杂的过程，因为散景通常需要通过硬件特性或后处理来实现。散景效果指的是在照片中前景清晰、背景模糊的效果，常用于人像摄影。

在 Camera2 API 中，你可以通过以下几个步骤尝试实现散景效果：

1. 使用大光圈或长焦镜头

如果设备支持多摄像头配置或具有大光圈镜头，你可以选择适合的摄像头。使用大光圈时，可以减少景深，从而使背景模糊。

2. 手动控制对焦

通过手动调节焦距，使得前景物体清晰对焦，而背景物体模糊。这可以通过 Camera2 的 `LENS_FOCUS_DISTANCE` 来实现。

```
fun adjustFocusDistanceForBokeh(cameraDevice: CameraDevice,
cameraCaptureSession: CameraCaptureSession, focusDistance: Float) {
    val captureRequestBuilder =
cameraDevice.createCaptureRequest(CameraDevice.TEMPLATE_PREVIEW).apply {
        set(CaptureRequest.CONTROL_AF_MODE,
CaptureRequest.CONTROL_AF_MODE_OFF)
        set(CaptureRequest.LENS_FOCUS_DISTANCE, focusDistance)
    }

    cameraCaptureSession.setRepeatingRequest(captureRequestBuilder.build(),
null, null)
}

// 调用示例
val focusDistance: Float = 0.2f // 将焦距设置为靠近相机的物体
adjustFocusDistanceForBokeh(cameraDevice, cameraCaptureSession,
focusDistance)
```

3. 启用散景模式 (如果支持)

某些设备提供硬件级别的散景效果。在 Camera2 API 中，这可能会作为一个特殊模式或场景模式暴露出来，如 `CaptureRequest.CONTROL_EFFECT_MODE` 或 `CaptureRequest.CONTROL_SCENE_MODE`。

```

fun enableBokehMode(cameraDevice: CameraDevice, cameraCaptureSession:
CameraCaptureSession) {
    val captureRequestBuilder =
cameraDevice.createCaptureRequest(CameraDevice.TEMPLATE_STILL_CAPTURE).appl
y {
        set(CaptureRequest.CONTROL_MODE,
CameraMetadata.CONTROL_MODE_USE_SCENE_MODE)
        set(CaptureRequest.CONTROL_SCENE_MODE,
CameraMetadata.CONTROL_SCENE_MODE_PORTRAIT) // 检查设备是否支持这个模式
    }

    cameraCaptureSession.capture(captureRequestBuilder.build(), null, null)
}

```

4. 使用图像后处理

如果设备不支持硬件级别的散景效果，你可以通过软件进行后处理。例如，使用 OpenCV 或其他图像处理库对捕获的图像进行模糊处理，仅保留前景清晰。

5. 组合双摄像头数据

对于支持双摄像头的设备，可以结合来自两个镜头的数据进行深度感知，以便在软件层面模拟散景效果。这种实现通常更复杂，需要处理深度图像和前景分离。

6. 使用第三方库

一些第三方库可以帮助你在图像处理中实现散景效果。如果你需要在应用中显示图像，可以考虑在拍摄后应用这些库进行处理。

总结：

Camera2 API 直接支持的散景效果取决于设备硬件的支持。对于没有硬件支持的设备，你可以尝试手动对焦结合图像后处理来模拟散景效果。

要判断手机是否支持散景效果 (Bokeh effect)，可以通过以下几种方式进行检测。需要注意的是，散景效果的支持通常取决于手机硬件（如多摄像头配置）和相机厂商提供的特定模式。

1. 检查 `CONTROL_AVAILABLE_EFFECTS` 列表

有些设备可能在 `CONTROL_AVAILABLE_EFFECTS` 中支持散景模式，通常这个模式可能被标记为 `CameraMetadata.CONTROL_EFFECT_MODE_BOKEH` 或其他特定的模式名称。

```

import android.content.Context
import android.hardware.camera2.CameraCharacteristics
import android.hardware.camera2.CameraManager
import android.hardware.camera2.CameraMetadata

fun isBokehSupported(context: Context): Boolean {
    val cameraManager = context.getSystemService(Context.CAMERA_SERVICE) as
    CameraManager

    try {
        for (cameraId in cameraManager.cameraIdList) {
            val characteristics =
            cameraManager.getCameraCharacteristics(cameraId)

            // 获取设备支持的特效模式
            val availableEffects =
            characteristics.get(CameraCharacteristics.CONTROL_AVAILABLE_EFFECTS)
            if (availableEffects != null) {
                if
            (availableEffects.contains(CameraMetadata.CONTROL_EFFECT_MODE_BOKEH)) {
                    return true // 设备支持散景效果
                }
            }

            // 检查是否有特定的散景模式支持
            val availableSceneModes =
            characteristics.get(CameraCharacteristics.CONTROL_AVAILABLE_SCENE_MODES)
            if (availableSceneModes != null) {
                if
            (availableSceneModes.contains(CameraMetadata.CONTROL_SCENE_MODE_PORTRAIT))
            {
                    return true // 设备支持人像模式（可能带有散景效果）
                }
            }
        }
    } catch (e: Exception) {
        e.printStackTrace()
    }

    return false
}

```

2. 检查 CONTROL_AVAILABLE_SCENE_MODES

一些设备可能将散景效果放在场景模式中，比如“人像模式”可能会使用散景效果。在这种情况下，你可以检查 CONTROL_AVAILABLE_SCENE_MODES 列表中是否存在 CameraMetadata.CONTROL_SCENE_MODE_PORTRAIT 这样的模式。

```

fun isPortraitModeSupported(context: Context): Boolean {
    val cameraManager = context.getSystemService(Context.CAMERA_SERVICE) as
    CameraManager

    try {
        for (cameraId in cameraManager.cameraIdList) {
            val characteristics =
            cameraManager.getCameraCharacteristics(cameraId)

            // 获取设备支持的场景模式
            val availableSceneModes =
            characteristics.get(CameraCharacteristics.CONTROL_AVAILABLE_SCENE_MODES)
            if (availableSceneModes != null) {
                if
            (availableSceneModes.contains(CameraMetadata.CONTROL_SCENE_MODE_PORTRAIT))
            {
                return true // 设备支持人像模式，可能包含散景效果
            }
        }
    } catch (e: Exception) {
        e.printStackTrace()
    }

    return false
}

```

3. 检查多摄像头配置

如果设备有多个摄像头（例如主摄像头和长焦或深度摄像头），则可以通过组合多个摄像头的数据来实现散景效果。你可以检查 `CameraCharacteristics.REQUEST_AVAILABLE_CAPABILITIES` 是否包含 `CameraCharacteristics.REQUEST_AVAILABLE_CAPABILITIES_LOGICAL_MULTI_CAMERA`，这意味着设备可能支持类似散景的效果。

```

fun isMultiCameraSupported(context: Context): Boolean {
    val cameraManager = context.getSystemService(Context.CAMERA_SERVICE) as
    CameraManager

    try {
        for (cameraId in cameraManager.cameraIdList) {
            val characteristics =
            cameraManager.getCameraCharacteristics(cameraId)

            val capabilities =

```

```
characteristics.get(CameraCharacteristics.REQUEST_AVAILABLE_CAPABILITIES)
    if (capabilities != null &&
        capabilities.contains(CameraCharacteristics.REQUEST_AVAILABLE_CAPABILITIES_
            LOGICAL_MULTI_CAMERA)) {
        return true // 设备支持多摄像头
    }
}
} catch (e: Exception) {
    e.printStackTrace()
}

return false
}
```

总结

- **特效模式**：通过 `CONTROL_AVAILABLE_EFFECTS` 检查是否支持 Bokeh 模式。
- **场景模式**：检查是否支持 Portrait 场景模式。
- **多摄像头配置**：检查是否支持多摄像头配置来可能实现散景效果。

这些检查可以帮助你判断设备是否支持硬件级别的散景效果。每个设备的支持情况可能不同，因此最好综合考虑这些因素。