☰ Media and devices › Camera Capture

Camera Capture

OpenClaw supports **camera capture** for agent workflows:

iOS node (paired via Gateway): capture a **photo** (jpg) or **short video clip** (mp4 , with optional audio) via node.invoke .

Android node (paired via Gateway): capture a **photo** (jpg) or **short video clip** (mp4 , with optional audio) via node.invoke .

macOS app (node via Gateway): capture a **photo** (jpg) or **short video clip** (mp4 , with optional audio) via node.invoke .

All camera access is gated behind **user-controlled settings**.

iOS node

User setting (default on)

iOS Settings tab → **Camera** → **Allow Camera** (camera.enabled)

Default: **on** (missing key is treated as enabled).

When off: camera.* commands return CAMERA_DISABLED .

Commands (via Gateway node.invoke)

camera.list

Response payload:

devices : array of { id, name, position, deviceType }

camera.snap

Params:

facing : `front|back` (default: `front`)

maxWidth : number (optional; default `1600` on the iOS node)

quality : `0..1` (optional; default `0.9`)

format : currently `jpg`

delayMs : number (optional; default `0`)

deviceId : string (optional; from `camera.list`)

Response payload:

format: "jpg"

base64: "<...>"

width , height

Payload guard: photos are recompressed to keep the base64 payload under 5 MB.

camera.clip

Params:

facing : `front|back` (default: `front`)

durationMs : number (default `3000`, clamped to a max of `60000`)

includeAudio : boolean (default `true`)

format : currently `mp4`

deviceId : string (optional; from `camera.list`)

Response payload:

format: "mp4"

base64: "<...>"

durationMs

`hasAudio`

Foreground requirement

Like `canvas.*`, the iOS node only allows `camera.*` commands in the **foreground**. Background invocations return `NODE_BACKGROUND_UNAVAILABLE`.

CLI helper (temp files + MEDIA)

The easiest way to get attachments is via the CLI helper, which writes decoded media to a temp file and prints `MEDIA:<path>`.

Examples:

```
openclaw nodes camera snap --node <id>              # default: both 1
openclaw nodes camera snap --node <id> --facing front
openclaw nodes camera clip --node <id> --duration 3000
openclaw nodes camera clip --node <id> --no-audio
```

Notes:

`nodes camera snap` defaults to **both** facings to give the agent both views.

Output files are temporary (in the OS temp directory) unless you build your own wrapper.

Android node

Android user setting (default on)

Android Settings sheet → **Camera** → **Allow Camera** (`camera.enabled`)

Default: **on** (missing key is treated as enabled).

When off: `camera.*` commands return `CAMERA_DISABLED`.

Permissions

Android requires runtime permissions:

`CAMERA` for both `camera.snap` and `camera.clip` .

`RECORD_AUDIO` for `camera.clip` when `includeAudio=true` .

If permissions are missing, the app will prompt when possible; if denied, `camera.*` requests fail with a `*_PERMISSION_REQUIRED` error.

Android foreground requirement

Like `canvas.*` , the Android node only allows `camera.*` commands in the foreground. Background invocations return `NODE_BACKGROUND_UNAVAILABLE` .

Payload guard

Photos are recompressed to keep the base64 payload under 5 MB.

macOS app

User setting (default off)

The macOS companion app exposes a checkbox:

Settings → General → Allow Camera (`openclaw.cameraEnabled`)

Default: **off**

When off: camera requests return "Camera disabled by user".

CLI helper (node invoke)

Use the main `openclaw` CLI to invoke camera commands on the macOS node.

Examples:

```
openclaw nodes camera list --node <id>             # list camera ids
openclaw nodes camera snap --node <id>             # prints MEDIA:<path>
openclaw nodes camera snap --node <id> --max-width 1280
openclaw nodes camera snap --node <id> --delay-ms 2000
openclaw nodes camera snap --node <id> --device-id <id>
openclaw nodes camera clip --node <id> --duration 10s       # prints MEDIA:<pat
openclaw nodes camera clip --node <id> --duration-ms 3000   # prints MEDIA:<pat
openclaw nodes camera clip --node <id> --device-id <id>
openclaw nodes camera clip --node <id> --no-audio
```

Notes:

> `openclaw nodes camera snap` defaults to `maxWidth=1600` unless
> overridden.

> On macOS, `camera.snap` waits `delayMs` (default 2000ms) after warm-
> up/exposure settle before capturing.

> Photo payloads are recompressed to keep base64 under 5 MB.

Safety + practical limits

> Camera and microphone access trigger the usual OS permission
> prompts (and require usage strings in Info.plist).

> Video clips are capped (currently `<= 60s`) to avoid oversized node
> payloads (base64 overhead + message limits).

macOS screen video (OS–level)

For *screen* video (not camera), use the macOS companion:

```
openclaw nodes screen record --node <id> --duration 10s --fps 15   # ∣
```

Notes:

Requires macOS **Screen Recording** permission (TCC).

Powered by mintlify