

 semantic-release	chore: release v0.21.3	21858c4 · 19 hours ago	
 .agents	chore(assets): attribut...	4 days ago	
 .github	chore(ci): gate the rele...	4 days ago	
 assets	chore(assets): #44 ad...	19 hours ago	
 detector	fix: #29 report authen...	last week	
 docs	docs: add FAQ + contri...	2 days ago	
 examples	feat: #39 ship a contai...	4 days ago	
 hooks	docs: render GitHub c...	last month	
 src/fugleramme	chore: release v0.21.3	19 hours ago	
 tests	test: stop waiting out t...	19 hours ago	
 tools	perf: #39 ship artwork...	4 days ago	
 .dockerignore	feat: #39 ship a contai...	4 days ago	
 .git-blame-ignore-revs	chore: ignore the for...	last month	
 .gitignore	feat: #29 choose the d...	2 weeks ago	
 .python-version	feat: #1 scaffold fram...	2 months ago	
 AGENTS.md	chore: #68 land artwo...	4 days ago	
 CHANGELOG.md	chore: release v0.21.3	19 hours ago	
 CLAUDE.md	chore: CLAUDE.md -> ...	3 weeks ago	
 CONTRIBUTING.md	chore(assets): attribut...	4 days ago	
 Dockerfile	feat: #39 ship a contai...	4 days ago	
 LICENSE	Initial commit	2 months ago	
 README.md	docs: add FAQ + contri...	2 days ago	
 install.sh	fix: #29 take the next f...	last week	
 mkdocs.yml	docs: add FAQ + contri...	2 days ago	
 pyproject.toml	chore: release v0.21.3	19 hours ago	
 run.sh	fix: #29 report authen...	last week	
 uv.lock	chore: release v0.21.3	19 hours ago	

fugleramme

E-ink bird frame for Raspberry Pi - real-time bird detection by audio, fully local AI, rendered as real, hand-cut 1800s bird illustrations.



Sorry about the dirty window - squirrels have been stealing the bird food.

live demo

online

release

v0.21.3

ci

passing

last commit

yesterday

stars

328

contributors

3

license

MIT + art

CC-BY-SA

Note

Still in early development: expect the odd bug and a few unpolished edges, with plenty more features to come.

Live on [fugleramme.arnegiacomo.dev](#) running from my kitchen window and displaying the actual birds currently heard in my garden (Bergen, Norway).

Hardware, install and operations docs: [arnegiacomo.dev/fugleramme](#)

How it works

[BirdNET-Go](#) listens on a mic and handles the classifier. Fugleramme polls its api, matches each species to an illustration, then packs them onto a page, and redraws only when the birds change - on an [Inky Impression](#) e-ink panel, and as a web kiosk serving the same view. There's an admin page that lets you configure what to show, and automatic updates and such.

If you already run BirdNET-Go, point the frame at it instead - on the same machine or anywhere else reachable from your network.

Tip

The e-ink panel is not required, although it's recommended for the intended experience. Without one, Fugleramme runs web-only - show the kiosk on a display over HDMI, or open it from any device on the network.

Hardware

A Raspberry Pi 5, an [Inky Impression 13.3"](#) (Spectra 6), a mic and an A4 frame. Full parts list, recommendations and alternatives: [Hardware](#).

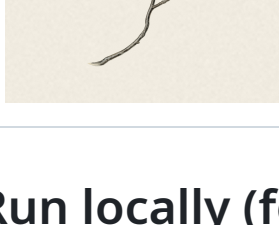
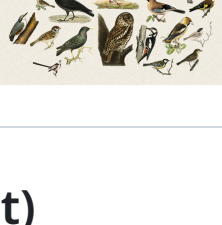
Art

Half the point of this project is showing off some amazing public-domain natural-history illustrations. Over 800 cut-outs covering more than 400 species, every one taken from a real plate and hand-curated for this project (no art is AI-generated, though some has been retouched with AI).

Each detected species is matched to its illustration, background-removed, and packed onto a textured paper page with the larger birds toward the centre, sized by body mass. An empty window shows a bare perch.

The plates are Scandinavian, British and central European, so the Nordics, the British Isles and Germany are best covered. Elsewhere not so much (yet). Broader European and North American coverage is in the works!

See [Adding artwork](#) for manual cutout steps.

No detections	A few visitors	A full garden
		

Run locally (for development)

```
uv sync
uv run fugleramme-fake-detector
uv run fugleramme-dev
```

The fake detector's flags, and working against a real station instead: [Running it without a Pi](#).

Install on a Raspberry Pi

From the pi (assuming you have the hardware up and running):

```
curl -fsSL https://raw.githubusercontent.com/arnegiacomo/fugleramme
```

Asks where BirdNET-Go should live and which ports to use, clones the repo, installs the required deps, and starts the frame as a systemd service. **NB!** Will probably require a reboot on a fresh system.

From a blank SD card, see the full [install guide](#).

Run in a container

```
docker run -d -p 8080:8080 -v fugleramme:/data \
-e FUGLERAMME_DETECTOR_URL=http://birdnet.local:
ghcr.io/arnegiacomo/fugleramme
```

Or build the image from a checkout:

```
docker build -t fugleramme .
docker run --rm -p 8080:8080 -v fugleramme:/data \
-e FUGLERAMME_DETECTOR_URL=http://birdnet.local:
```

Kiosk on :8080 , admin on :8080/admin , everything it persists in /data .

On a Linux box with a USB mic, this brings up BirdNET-Go alongside it:

```
curl -fsSL https://raw.githubusercontent.com/arnegiacomo/fugleramme
docker compose up -d
```

See [Container](#) for more info.

Contributing

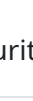
Contributions are very welcome and encouraged - fixes, docs and artwork most of all. Thanks to [everyone who has contributed](#) so far 

- **Something is broken** - a [bug report](#)
- **A question, an idea, or a frame you have built** - the [FAQ](#) first, then [Discussions](#)
- **A fix, a doc change, or a bird you have cut** - open a PR, no issue needed

See [Contributing](#) for more info.

License

- Code: MIT - see [LICENSE](#) .
- Detection ([BirdNET-Go](#), installed separately as a container): CC BY-NC-SA 4.0, non-commercial only. BirdNET model by the Cornell Lab of Ornithology and

 semantic-release	chore: release v0.21.3	21858c4 · 19 hours ago	
 .agents	chore(assets): attribut...	4 days ago	
 .github	chore(ci): gate the rele...	4 days ago	
 assets	chore(assets): #44 ad...	19 hours ago	
 detector	fix: #29 report authen...	last week	
 docs	docs: add FAQ + contri...	2 days ago	
 examples	feat: #39 ship a contai...	4 days ago	
 hooks	docs: render GitHub c...	last month	
 src/fugleramme	chore: release v0.21.3	19 hours ago	
 tests	test: stop waiting out t...	19 hours ago	
 tools	perf: #39 ship artwork...	4 days ago	
 .dockerignore	feat: #39 ship a contai...	4 days ago	
 .git-blame-ignore-revs	chore: ignore the for...	last month	
 .gitignore	feat: #29 choose the d...	2 weeks ago	
 .python-version	feat: #1 scaffold fram...	2 months ago	
 AGENTS.md	chore: #68 land artwo...	4 days ago	
 CHANGELOG.md	chore: release v0.21.3	19 hours ago	
 CLAUDE.md	chore: CLAUDE.md -> ...	3 weeks ago	
 CONTRIBUTING.md	chore(assets): attribut...	4 days ago	
 Dockerfile	feat: #39 ship a contai...	4 days ago	
 LICENSE	Initial commit	2 months ago	
 README.md	docs: add FAQ + contri...	2 days ago	
 install.sh	fix: #29 take the next f...	last week	
 mkdocs.yml	docs: add FAQ + contri...	2 days ago	
 pyproject.toml	chore: release v0.21.3	19 hours ago	
 run.sh	fix: #29 report authen...	last week	
 uv.lock	chore: release v0.21.3	19 hours ago	

fugleramme

E-ink bird frame for Raspberry Pi - real-time bird detection by audio, fully local AI, rendered as real, hand-cut 1800s bird illustrations.



Sorry about the dirty window - squirrels have been stealing the bird food.

live demo

online

release

v0.21.3

ci

passing

last commit

yesterday

stars

328

contributors

3

license

MIT + art

CC-BY-SA

Note

Still in early development: expect the odd bug and a few unpolished edges, with plenty more features to come.

Live on [fugleramme.arnegiacomo.dev](#) running from my kitchen window and displaying the actual birds currently heard in my garden (Bergen, Norway).

Hardware, install and operations docs: [arnegiacomo.dev/fugleramme](#)

How it works

[BirdNET-Go](#) listens on a mic and handles the classifier. Fugleramme polls its api, matches each species to an illustration, then packs them onto a page, and redraws only when the birds change - on an [Inky Impression](#) e-ink panel, and as a web kiosk serving the same view. There's an admin page that lets you configure what to show, and automatic updates and such.

If you already run BirdNET-Go, point the frame at it instead - on the same machine or anywhere else reachable from your network.

Tip

The e-ink panel is not required, although it's recommended for the intended experience. Without one, Fugleramme runs web-only - show the kiosk on a display over HDMI, or open it from any device on the network.

Hardware

A Raspberry Pi 5, an [Inky Impression 13.3"](#) (Spectra 6), a mic and an A4 frame. Full parts list, recommendations and alternatives: [Hardware](#).

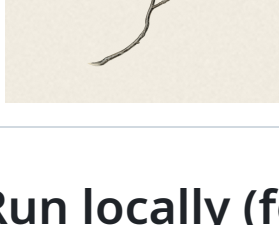
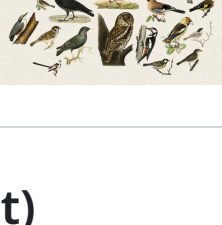
Art

Half the point of this project is showing off some amazing public-domain natural-history illustrations. Over 800 cut-outs covering more than 400 species, every one taken from a real plate and hand-curated for this project (no art is AI-generated, though some has been retouched with AI).

Each detected species is matched to its illustration, background-removed, and packed onto a textured paper page with the larger birds toward the centre, sized by body mass. An empty window shows a bare perch.

The plates are Scandinavian, British and central European, so the Nordics, the British Isles and Germany are best covered. Elsewhere not so much (yet). Broader European and North American coverage is in the works!

See [Adding artwork](#) for manual cutout steps.

No detections	A few visitors	A full garden
		

Run locally (for development)

```
uv sync
uv run fugleramme-fake-detector
uv run fugleramme-dev
```

The fake detector's flags, and working against a real station instead: [Running it without a Pi](#).

Install on a Raspberry Pi

From the pi (assuming you have the hardware up and running):

```
curl -fsSL https://raw.githubusercontent.com/arnegiacomo/fugleramme
```

Asks where BirdNET-Go should live and which ports to use, clones the repo, installs the required deps, and starts the frame as a systemd service. **NB!** Will probably require a reboot on a fresh system.

From a blank SD card, see the full [install guide](#).

Run in a container

```
docker run -d -p 8080:8080 -v fugleramme:/data \
-e FUGLERAMME_DETECTOR_URL=http://birdnet.local:
ghcr.io/arnegiacomo/fugleramme
```

Or build the image from a checkout:

```
docker build -t fugleramme .
docker run --rm -p 8080:8080 -v fugleramme:/data \
-e FUGLERAMME_DETECTOR_URL=http://birdnet.local:
```

Kiosk on :8080 , admin on :8080/admin , everything it persists in /data .

On a Linux box with a USB mic, this brings up BirdNET-Go alongside it:

```
curl -fsSL https://raw.githubusercontent.com/arnegiacomo/fugleramme
docker compose up -d
```

See [Container](#) for more info.

Contributing

Contributions are very welcome and encouraged - fixes, docs and artwork most of all. Thanks to [everyone who has contributed](#) so far 

- **Something is broken** - a [bug report](#)
- **A question, an idea, or a frame you have built** - the [FAQ](#) first, then [Discussions](#)
- **A fix, a doc change, or a bird you have cut** - open a PR, no issue needed

See [Contributing](#) for more info.

License

- Code: MIT - see [LICENSE](#) .
- Detection ([BirdNET-Go](#), installed separately as a container): CC BY-NC-SA 4.0, non-commercial only. BirdNET model by the Cornell Lab of Ornithology and

BirdNET model by the Cornell Lab of Ornithology and

Chemnitz University of Technology, taxonomy data
powered by eBird.org.

- Bird images: each style folder carries its own terms and sources, and its manifest links the plate every file was cut from. `classic` is CC BY-SA 4.0 - see [assets/artwork/classic/ATTRIBUTION.md](#) .
- Label fonts (`assets/fonts/`); SIL OFL 1.1 - see [assets/fonts/ATTRIBUTION.md](#) .
- Bird sizes (`assets/bird_sizes.csv`); body mass from AVONET (Tobias et al. 2022, Ecology Letters, [doi:10.1111/ele.13898](#)), CC BY 4.0.
- BirdNET scientific-name aliases (`assets/birdnet_aliases.json`): [OpenFauna](#)'s compiled taxonomic alias map, CC BY-SA 4.0 - see [assets/ATTRIBUTION.md](#) .

Prebuilt frames

I've built a few of these. If you'd like one rather than building it yourself, please [get in touch](#).

