
GitHub and LinkedIn, for robotics

The generic advice does not fit this field

Most advice about these two profiles is written for software engineers applying to product companies. Robotics is judged differently, by different people, looking for different evidence. This is what changes.

1. Who is actually reading

A professor Wants to know whether you can be left alone with an open problem for two weeks and return with something. They will open one repository, read for ninety seconds, and decide. They do not care about your contribution graph.

A recruiter A company does not read anything. A person or a screening model does, and both work from keywords and titles. They rarely open GitHub at all.

Optimise **GitHub for the professor** and **LinkedIn for the recruiter**. The second one costs you nothing with the first: a professor who searches for you will find the LinkedIn page too, and a page written in plain technical language reads correctly to both.

2. GitHub

The ninety-second test

Someone lands on your profile. What can they establish in ninety seconds? If the answer is “this person has taken some courses,” the profile has failed. If it is “this person built a system that ran on real hardware and can explain it,” it has worked.

Everything below serves that one test.

Pin four repositories, not six

Four real ones beat six padded with coursework. If a repository is a follow-along tutorial, an assignment everyone in your class submitted, or a fork you changed two lines in, unpin it. A sparse profile of real work reads better than a full one of noise.

Show the robot working

This is the single largest difference between robotics and software. **A video at the top of the README is worth more than the README.** Thirty seconds of the arm picking the object, the drone holding position, the rover crossing the terrain. Nobody will clone your repository to find out whether it works; the video is the only proof most readers will ever see.

If the hardware is not yours to film, show the simulation, and say plainly that it is simulation.

What a robotics README must contain

1. **One sentence saying what the system does.** Not what it is built with.

2. **The video or a GIF.** Above the fold.
3. **The hardware.** Platform, sensors, compute. “Franka Research 3, RealSense D435, Jetson Orin” tells a reader more than a paragraph of prose.
4. **The method, in two or three lines.** The algorithm or controller, and why that one.
5. **How to run it.** Dependencies, and the exact command. If it cannot be run, say so and say why.
6. **What does not work yet.** Stating a known limitation is the clearest evidence in the whole document that you understand your own system.

Commit like someone will read it

Because they will, if they are deciding whether to work with you. A history of one commit called “final code” says the work happened somewhere else and was dumped here. Commits that show the system being built, broken, and fixed say the opposite.

Be visible about the hardware

The most useful piece of advice I have been given about this came from an MIT alumnus, and it was simple: **most applicants have never touched industry hardware, and a professor would rather take someone who has.** Simulation-only candidates are common. Someone who has run a real arm, a real drone, a real sensor, and dealt with it failing, is not.

So make the hardware impossible to miss. Name the platforms explicitly, on the repository, on the profile, and in the statement of purpose. **Franka Research 3, Kinova, TurtleBot, Crazyflie, RealSense, Jetson, PX4.** These are not decoration, they are the evidence. A reader who knows the field recognises them instantly and draws the conclusion without being told.

If you have used industry hardware even once, that is a thing to be loud about. Most people cannot say it.

What to leave off

Streak counts, profile trophies, badge walls, auto-generated statistics cards, and the animated banner. None of these are read by anyone deciding anything, and together they take up the space where your work should be.

3. LinkedIn

The headline

This is the one field that is searched, so it is the one field where keywords matter. Say what you work on, not what you aspire to. A headline names the work: the subfield, and the platform or method if there is room. **Passionate about innovation** is not. If you have a position, name the institution; people search by it.

The About section

Three or four sentences. What you work on, where, and one line on what you want to do next. Write it in the first person and in plain sentences. Nobody has ever been hired for a metaphor about journeys.

Positions

Name the supervisor and the laboratory, not just the university. “Research Intern, University of X” is anonymous. “Research Intern, [Laboratory], University of X, with Prof. [Name]” is checkable, and checkable is the whole point.

Two lines per position, and make them technical. The reader should be able to tell what you personally did.

Posting

Post when something happened: a result, a competition, a paper, a talk. Do not post daily reflections. In robotics the audience is small and largely academic, and it notices the difference between someone doing work and someone describing work.

If you post about a competition, say what you built and what failed, not only where you placed. The failure is the part other people learn from, and it is the part nobody writes.

Connections

Connect with people whose work you have actually read. A connection request to an author with one line about which paper and what you found in it is answered far more often than a blank one. The same principle as a cold email, at a smaller scale.

4. A portfolio site

Build one. This is not optional, and it is the highest-return thing on this page.

GitHub and LinkedIn are both somebody else’s product, shaped by somebody else’s layout, and both bury the thing you want seen. A site of your own is the only surface where you decide what a reader sees first. It is also what you put in an email signature, on a CV, and under a QR code, and it is the single link that carries everything else.

- **Photographs and video, prominently.** You working on the hardware. The robot running. These do more than any paragraph, and almost nobody has them.
- **One page is enough.** Who you are, what you work on, where, and the projects. A reader should not have to navigate.
- **Host the documents on it.** CV, papers, certificates. A link is checkable; a claim is not.
- **Keep it current.** A site that stops in 2024 is worse than no site.

It takes a weekend and it is free to host. Almost nobody at your level has one, which is exactly why it works.

5. What both profiles are really for

Neither of these gets you a position. They remove the reason to say no.

A professor who has read your email and is mildly interested will open one of these to check whether you are real. What they are looking for is evidence that the things you claimed actually exist. Every part of both profiles should make that check easy and fast.