Capstone Presentations

Week 3 · Day 5: Showing What You Built

Yusuf Brima, Ph.D.

Introduction to Computer Vision · AIMS Cameroon, 2026–2027

Today

1. How the session runs
2. Presentations: 10 minutes + 5 minutes of questions per team
3. What the audience does
4. Course wrap-up and feedback

Reminders

- Code and one-page summary (with model card) submitted by the deadline announced in class.
- Quiz 3 runs in the tutorial slot, not during this session.

How the session runs

- Teams present in the order on the next slide. Be ready, with slides loaded, when the previous team starts its questions.
- **10 minutes** of talk, every member speaking. A signal at 8 minutes; we stop at 10.
- **5 minutes** of questions from the instructor, tutors and classmates.
- A demo video, if any, counts within the 10 minutes (at most 2 minutes).

Practitioner's Note

Test your slides on the room's projector before the session starts. Have a PDF copy on a USB stick and online. Do not rely on a live demo over the network.

Presentation order

#	Team	Project	Start
1	...	...	...
2	...	...	...
3	...	...	...
4	...	...	...
5	...	...	...
6	...	...	...
7	...	...	...
8	...	...	...

Filled in from the confirmed team list. With more than eight teams, presentations continue in the tutorial slot.

What a strong talk contains

1. **Problem**: who is the user, what decision does the output support, which errors matter? (1–2 min)
2. **Data**: source, licence, size, labelling, *how you split it*. (2 min)
3. **Methods**: baseline and main model, and why. (2 min)
4. **Results**: one clear table; per-class or per-group numbers. (2 min)
5. **Failures and limitations**: two real failure examples; robustness; what you would do next. (2 min)

Intuition

The talk mirrors the rubric. A listener should be able to answer: what problem, which data, how well, when does it fail, and would I trust it?

Practitioner's Note

One idea per slide; large fonts; axes labelled; the main number in words as well as in a table.

How projects are assessed

Criterion	Weight
Problem framing: user, decision, which errors matter	15%
Data: source and licence, labelling, size, splitting without leakage	20%
Methods: sensible baseline, justified main model, course tools used correctly	20%
Evaluation: appropriate metrics, per-class results, deployment-like test set, uncertainty	20%
Failure analysis and limitations: examples, explanations, next steps	15%
Presentation and reproducibility: clear story, readable figures, code that runs	10%

Intuition

An honest, modest result from a clean split scores higher than an impressive number from a leaky one.

Questions we are likely to ask

- How did you split the data, and could the same scene, patient or site appear in both train and test?
- How does your model compare with the simplest baseline?
- Which class or condition is hardest, and why?
- What happens with a darker, blurrier or different-camera image?
- Who would use this, and what would they do with a wrong output?
- If you had two more weeks, what would you do first?

Practitioner's Note

"We did not test that" is an acceptable answer, especially when followed by how you would test it.

What the audience does

- Listen as a reviewer: for each talk, note one strength and one question.
- Ask at least one question during the session.
- Fill in the short peer-feedback form for each team (shared in class); feedback goes to the team, anonymously.

Intuition

Questioning someone else's evaluation is the fastest way to learn to question your own. Be specific, be kind, be honest.

Before you present: a final checklist

- ☐ Title slide: project, team, every member's name.
- ☐ The split is stated and justified.
- ☐ Baseline and main model in one table.
- ☐ Per-class or per-group results.
- ☐ At least two failure examples, shown as images.
- ☐ A robustness or shift test.
- ☐ Limitations and next steps.
- ☐ Rehearsed and timed, under 10 minutes.

Part 1

Course wrap-up

Three weeks of computer vision

Week 1: foundations
human vision; images as data; filtering and frequency; edges, corners, features; from engineered to learned features

Week 2: deep vision
CNNs and training; architectures and transfer; detection; segmentation; transformers, captioning, open vocabulary

Week 3: creating and connecting
autoencoders and VAEs; diffusion; CLIP and vision–language; applications; robustness, fairness and ethics

Intuition

From a pixel to a model card: you can now take an image problem from question to data, model, honest evaluation and responsible deployment.

Course feedback

- An anonymous feedback form will be shared today.
- Tell us what helped you learn, what did not, and what you would change: pace, depth, examples, tutorials, the capstone.
- Your feedback shapes the course for the next cohort.

Staying in touch

- Questions about capstone follow-up, papers or datasets: by e-mail or during office hours announced in class.
- Turning a capstone into a paper, dataset or workshop submission: come and talk to me.

Thank you

Go and build things that see well, for people who need them.