ML system design interview

Crop disease from a phone photograph — features by hand, or learned end to end

LECTURE 15 MOCK INTERVIEW

Applications of Machine Learning

BSc Mathematics of Artificial Intelligence · Third year

Where we are

Today's system is built from pieces you have already measured on images.

The piece	Where we saw it
Convolutions, weight sharing, a network trained from scratch on few labels per class	Lecture 12
A pretrained backbone: probe first, fine-tune second; augmentation, and where it must not reach	Lecture 13
Finding the object before naming it: detection, segmentation, IoU	Lecture 14
What a model relies on, and for whom it fails	Lecture 10
The interview itself: nine steps, trade-offs out loud	Lecture 9

✓ Nothing new is needed. What is new is defending one choice against another when the interviewer will not let go.

Today

1. **The prompt** — a farmer photographs a sick leaf; the app names the disease
2. **The overview** — clarify, frame, metrics and labels, at the pace of a real interview
3. **The deep dive** — what the model looks at. Two solutions: features engineered by hand, and a network trained end to end. Then the interviewer makes you choose
4. **The rest of the system**, the whole design on one slide, the follow-ups and the red flags

No notebook today: a whiteboard and a conversation.

The format, with one change: the deep dive

- A real interview has two halves. First the candidate sketches the whole system, a sentence per step
- Then the interviewer **picks one box and digs**: “tell me more about that”, “what if …”, “why not the other way?” — until the candidate reaches the edge of what they know
- The deep dive is where senior and junior candidates separate. Anyone can name a technique; few can say when it breaks
- Today the box is **the representation**: who decides what the model looks at — you, or the network?

> **HOW TO SURVIVE A DEEP DIVE**
>
> Answer the question asked, then say what it costs. When you do not know, say how you would find out: which experiment, on which data.

THE INTERVIEWER

“We are a farmers’ cooperative. A farmer photographs a sick leaf with a phone. Design the system that tells them what disease it is.”

What do you say first?

Step 1 · Your turn

Before you design anything: what do you ask?

Write down the questions that would change the design.

Which plants, which diseases, which phones, where the photo is taken, what happens after the answer, and what you can learn from.

Step 1 — The questions, and why each one matters

Question	Why it changes the design
Which crops, which diseases?	the classes, and whether two of them look alike
Where is the photo taken, and by whom?	a lab table or a field: the hardest fact for any representation
Is there a network connection in the field?	on the phone or on a server: the size of the model
What does the farmer do with the answer?	a wrong treatment costs a harvest; the output is advice
Who checks a doubtful case?	whether the app may say “I am not sure”
What labelled photos exist today?	how much a learned representation can learn

Step 1 — What the interviewer answers

- **Crops:** tomato, potato and grapevine; about twenty diseases between them, plus “healthy”. Some look alike: early and late blight, downy and powdery mildew
- **Photos:** taken in the field, on whatever phone the farmer owns, in sun or shade, often with soil, hands and other leaves in the frame
- **Connection:** often none in the field. The diagnosis must run on the phone; photos upload later
- **Next:** the app shows the likely diseases and the cooperative’s treatment advice; an agronomist answers the cases the app sends on
- **Labels:** a few thousand field photos from past consultations, labelled by the agronomist who answered, mostly of the common diseases; and a large public set of single leaves shot on a plain table in a lab

> **WRITE IT ON THE BOARD**
>
> Two facts will decide the deep dive: *field photos*, and *few field labels* for the rare diseases.

Step 2 · Your turn

State the machine learning task

What goes in, what comes out, and what may the app answer when it cannot tell?

Step 2 — Input, output, decision

- **Input:** the photo, and the crop the farmer selected — region and date as context
- **Output:** a probability for each disease of that crop and for “healthy”: one softmax per crop. The agronomists label the main disease, so one answer per leaf is a fair assumption — say it aloud
- **Decision:** show the top three with their treatments; if the top probability is low, or the photo is not a usable leaf, **abstain** and send it to an agronomist

> **RED FLAG**
>
> An app that always answers. A confident wrong diagnosis sends a farmer to spray the wrong product; “I am not sure, an agronomist will look” is part of the output, not a failure of it.

Steps 3 and 4 — Metrics and labels, at interview pace

Metrics

- Recall per disease, the dangerous ones read first; top-three accuracy, because the app shows three
- Mistakes weighted by what they cost: two diseases with the same treatment are a cheap confusion; a fungus called a virus is not
- Accuracy against coverage: how right the app is on the cases it answers, and how many it sends on
- Online: the agronomists' verdicts on a sample of answered cases

Labels

- Consultation photos: farmers ask when worried, so severe cases are over-represented — selection bias
- Agronomists disagree on look-alikes: label a slice twice and measure it; that is the ceiling
- The lab set is plentiful, and taken in conditions the app will never see
- **Split by farm and by season**, never by photo: one farm sends many photos of the same field

✓ The interviewer nods and stops you here: "Fine. Now tell me what your model looks at."

THE DEEP DIVE · STEPS 5 AND 6

“What does your model look at — and who decided that?”

Two solutions, then a choice

Steps 5–6 · Your turn

Two ways to turn a photo into a diagnosis

Sketch a system in which *you* decide what the model measures, and one in which *the network* decides. Do not pick one yet.

The two solutions, side by side

	The pipeline	Who decides what is measured
A · Features by hand	photo → find the leaf → measure colour, spots, texture, shape → a small classifier on a few dozen numbers	the agronomist and the engineer, in advance
B · End to end	photo → a convolutional network → one probability per disease	gradient descent, from the labelled photos

- A is how plant pathology was automated before convolutional networks; B is what most such apps ship today
- Neither is obsolete. Each wins under conditions you can state — that is what the interviewer wants to hear by the end

Solution A · Your turn

You are the agronomist. What would you measure?

An agronomist tells blight from mildew by looking. Turn what they look at into numbers a classifier can use.

And before measuring anything on the leaf: how do you know which pixels are leaf?

Solution A — First, find the leaf

- **Segment by colour:** convert to a colour space where “green” is one channel’s range (hue, or the a* axis of Lab), threshold it, keep the largest connected region
- **Separate the lesions** inside the leaf the same way: brown, yellow, black and white pixels that are not healthy green
- **Normalise:** correct the white balance, so the same yellow under sun and under shade gives the same numbers

> **NOTE WHAT JUST HAPPENED**
>
> Every feature on the next slide is measured on this mask. If the mask is wrong, every feature is wrong, and the classifier is never told.

Solution A — The features, and the symptom each one reads

Group	Measured as	The symptom it reads
Colour	histograms of the lesion pixels' colour; the share of the leaf that is yellow	chlorosis; the brown of blight against the white of mildew
Spots	number of lesions, their sizes, how round they are, concentric rings	the target-shaped spots of early blight
Texture	statistics of local intensity patterns inside the lesions	the powdery film of powdery mildew; a mosaic pattern of a virus
Shape and position	leaf outline and curl; lesions on the margin or along the veins	leaf curl; spots bounded by the veins
Severity	lesion area over leaf area	how far it has gone — useful advice in itself

Then a classifier that suits a few dozen numbers and a few labels: logistic regression, or gradient-boosted trees for the interactions between colour and spot shape.

The interviewer pushes

“Here is a real photo: three leaves, soil, a thumb, the sun behind them. Walk me through your pipeline.”

- The colour threshold keeps the green of the other leaves and drops the brown dead tissue — which is the disease. The largest region is the wrong leaf
- Backlight turns healthy green to yellow: the “chlorosis” feature fires on a healthy plant
- Soil is brown and spotted: the spot counter counts it

> **A STRONG ANSWER**
>
> Say that the *first* step fails, and fix the capture before the model: the app asks for one leaf, held against the sky or a sheet of paper, and a quality check refuses dark, blurred or cluttered photos. The user interface is part of the feature engineering.

Solution A — What it buys, and what it costs

It buys	It costs
few labels: a few dozen numbers need tens of examples per disease, not hundreds	a ceiling: it can only see what the agronomist could name in advance
an explanation: “large brown spots with rings”, and a severity in percent	brittle capture: every feature inherits the segmentation’s mistakes
a tiny model that runs on any phone	a new disease means new features, written and tested by hand
failures you can debug, one feature at a time	look-alike diseases that differ in ways nobody wrote down stay confused

Solution B · Your turn

Let the network decide. What exactly do you train?

On which photos, starting from which weights, at which resolution — and how does it fit on a phone?

Solution B — The model

- **Start pretrained.** A few thousand field photos cannot train a network from scratch; a backbone pretrained on a large image collection already knows edges, textures and blobs. Probe first: a linear head on frozen features, the number to beat; then fine-tune the last blocks
- **Small enough for the phone:** a backbone designed for mobile devices, not the largest one; measure its accuracy against its size
- **Resolution:** an early lesion is a few pixels once the photo is shrunk to the network's input. Find the leaf first with a small detector, crop it, then classify the crop
- **Output:** one head per crop, or the crop as an input; a softmax, and abstention below a threshold

✓ **Solution B keeps A's first step, done by a learned detector instead of a colour threshold.**

The interviewer pushes

“You have few field photos. Which augmentations would you use — and which would you refuse?”

Augmentation	Use it?	Why
flips, rotations	**yes ✓**	a leaf has no “up”
random crops and scale	**yes ✓**	farmers hold the phone at every distance
brightness, contrast, blur	**yes, mildly ✓**	sun and shade; phones that do not focus
hue shifts, strong colour jitter	**no ✗**	yellow against green *is* the symptom: shifting it relabels the photo
random erasing	**careful ✗**	it can erase the only lesion and leave a “diseased” label on a healthy leaf

An augmentation is a claim that the label does not change. Make it only where an agronomist would agree. And on the training loader only: an augmented validation set no longer measures the model.

The interviewer pushes

“You trained on the large lab set and got a near-perfect test score. Ship it?”

- No. The test photos came from the same lab: one leaf, the same table, the same light, often each disease photographed in its own session
- A network takes whatever separates the classes most cheaply. If the background or the light differs between diseases, it learns those — a **shortcut**. The test set shares the shortcut, so it cannot see it
- This has happened in published work: near-perfect on lab photos, and far lower on photos from the field

> **HOW TO FIND OUT**
>
> Test on field photos, split by farm and by season. Paste the leaves onto other backgrounds and see if the answer changes. Look at what the network attends to, with the explanation tools of the auditing lecture: the lesion, or the table?

Solution B — What it buys, and what it costs

It buys	It costs
a higher ceiling: it finds the cues nobody wrote down, and separates look-alikes better	labels: hundreds of field photos per disease to fine-tune well; the rare diseases have tens
robustness to clutter and light — if, and only if, it was trained on photos that have them	shortcuts, learned silently, invisible to a test set that shares them
a new disease is new labelled photos, not new code	no explanation a farmer can read; no severity for free
one pipeline for every crop	a larger model on the phone, and a retraining pipeline to run

The interviewer pushes · Your turn

“You have one engineer and one season. Pick one, and defend it.”

Name the facts your choice depends on, and what would make you change your mind.

A against B, on the facts of this prompt

The fact	Favours	Because
field photos, cluttered, any phone	B	A's colour segmentation fails first, and everything inherits it
look-alike diseases	B	the difference is in cues nobody has named
a few thousand field labels, rare diseases with tens	A, or B pretrained	few labels favour few parameters to fit
runs offline on the phone	A, or a small B	a mobile backbone fits; the largest one does not
advice must be explained; severity wanted	A	its features are the explanation

✓ There is no free winner. The interviewer is waiting for the conditions, not for a name.

A strong answer: B, built on A’s parts

- **Ship B**: a mobile backbone, pretrained, fine-tuned on the field photos; the lab set only as extra training data, never as the test
- **Keep A’s pieces where they win:** the capture guide and the quality check in front; the leaf found before it is classified; lesion area as the severity shown to the farmer
- **A is the baseline B must beat**, on field photos split by farm, disease by disease. A rare disease where B does not beat A keeps A’s answer, or goes to an agronomist

I would switch to A if …	I would stay with B because …
the photos were taken at a fixed station — same light, plain background — and labels were scarce	the photos are taken in the field, and each season brings more labelled ones

What the interviewer writes down in the deep dive

- “Hand-made features are obsolete” — or “networks are a black box, so hand features are safer”: either one, with no conditions
- “The network learns its own features, so the data does not matter”: what it learns is decided by the photos
- A test score on lab photos reported as the field accuracy
- A hue or colour augmentation on a task where colour is the label
- Training from scratch on a few thousand photos
- No baseline: B with nothing to beat
- Cannot say what would change the choice

Steps 7 to 9 — The rest of the system, briefly

Step	The strong answer, in a line or two
7 · Thresholds	the abstention threshold set from accuracy against coverage, with the agronomists: as many answers as the cooperative will stand behind; check calibration per crop first
8 · Serving	on the phone, offline; the model shipped with the app and updated with it; photos and answers uploaded when there is signal
9 · Launch	shadow first: the app answers, an agronomist answers too, compare. Then one region at a time
9 · Drift	a new season, a new disease, a new phone’s colour processing: watch the abstention rate and the agronomists’ sample
9 · Feedback	abstained cases come back labelled, but only the hard ones; also label a random sample of answered cases

The whole system, on one slide

Photo, with the crop selected → capture guide and quality check (dark, blurred, no leaf: ask again) → a small detector finds the leaf and crops it → a fine-tuned mobile network: one probability per disease of that crop → above the threshold, show the top three with treatments, and the lesion area as severity; below it, send to an agronomist → the agronomists' answers, and a random sample they check, become labels → retrain each season, test on the newest season by farm, disease by disease, against the hand-feature baseline.

Measured by	Offline	Online
Each disease	recall on field photos split by farm and season; top three; accuracy against coverage	agronomists' verdicts on a sample; abstention rate; farmers' reports of a treatment that failed

The follow-up questions interviewers ask

“What if …”	The gist of a good answer
a new disease reaches the region?	agronomists first; in A new features, in B labelled photos and a new output; the new class measured before it is shown
a farmer photographs a fruit, or the dog?	the quality check and the detector refuse it; a softmax always names a disease, so abstain
the cooperative adds olives?	a new crop is a new head; the field photos start the cycle again, with A as the first baseline
a phone brand saturates its colours?	a slice by phone model; white balance in the capture step; photos from that phone in training
two agronomists disagree?	a third one, or a lab test; and the agreement rate as the ceiling you report

The red flags an interviewer writes down

- Designs before asking where the photos are taken
- An app that always answers
- Accuracy on a random split of photos: the same farm on both sides
- Lab photos as the test of a field system
- One solution named, no baseline, no condition under which the other would win
- Augmentations that change the label
- A model too big for the phone it must run on, offline
- Stops at training: no capture guide, no abstention, no feedback loop

What we did

1. Ran a second mock interview, with a deep dive: the whole design sketched quickly, then one box — the representation — pushed on until it broke
2. Framed the task as one softmax per crop with abstention, measured by recall per disease, top three and accuracy against coverage, on field photos split by farm and season
3. Solution A, features by hand: find the leaf, measure colour, spots, texture and shape, a small classifier. Few labels and an explanation; brittle capture and a ceiling
4. Solution B, end to end: a pretrained mobile network fine-tuned on field photos, the leaf cropped first, augmentations that keep the label. A higher ceiling; more labels, and shortcuts that a lab test cannot see
5. Chose B, kept A’s capture step, severity and baseline, and said under which conditions A would win

NOT PRESENTED — FOR AFTERWARDS

Exercises Lecture 15

five, in the style of the written paper

Exercises — Lecture 15

1 A crop-disease classifier is fine-tuned on a few field photos per disease, with random hue shifts among its augmentations. Say why this augmentation is wrong here, and state the rule it breaks. [4]

2 A network trained and tested on a large public set of leaves photographed one at a time on a lab table reaches a near-perfect test score. Explain why this does not tell you how the app will do in the field, and say how you would find out. [5]

3 Solution A finds the leaf by thresholding green pixels, then measures colour, spots and texture on that mask. Name the step that fails first on a cluttered field photo, why the classifier cannot notice, and two ways to fix it. [4]

Solutions on Lecture 16’s deck.

Exercises — Lecture 15 (continued)

4 Two cooperatives want the same diagnosis app. One photographs leaves at a fixed station with the same light and a plain background, and has tens of labelled photos per disease. The other has farmers photographing in the field, and hundreds per disease. Recommend hand-made features or an end-to-end network for each, and justify both. [5]

5 The diagnosis app ends in a softmax over the diseases of the selected crop. Say why it must be able to abstain, and what sets the abstention threshold. [4]

Solutions on Lecture 16’s deck.

THE FIVE SET AT THE END OF LECTURE 14

Solutions Lecture 14

not part of this lecture

Solutions — Lecture 14

1. Two boxes are disjoint and moved further apart. State what IoU and its derivative do, and why no learning…

✓ Both are identically zero; a constant has no descent direction.

- Past contact the intersection is exactly zero for every separation
- The gradient is absent rather than small, so no optimiser can act on it

2. A detector counts objects with MAE 3.00 at the conventional score cut-off of 0.5. A sweep over the same 128…

✓ The cut-off was chosen on the same images the error is reported on, so 1.92 is optimistic by construction; quote it as a diagnostic, beside the 3.00 at the threshold fixed before looking.

- Picking 0.75 after seeing the error curve is fitting a hyperparameter on the test data
- With no held-out images, only a threshold fixed in advance gives an honest figure

Solutions — Lecture 14 (2)

3. Average precision is defined with a maximum inside it. State what that maximum repairs, and which earlier…

✓ It replaces the non-monotone precision by its running maximum — the non-monotonicity proved in Lecture 3.

- Precision falls at every false positive and can rise again
- Integrating the raw curve would reward the sawtooth rather than the ranking

4. mAP is a mean over classes of a mean over IoU thresholds. Give one thing each averaging hides.

✓ The class mean hides that a one-instance class weighs as much as a 350-instance one; the threshold mean hides the spread between loose and tight matching.

- An unweighted mean ignores how many instances each class has
- A single number in the middle stands for two very different regimes

Solutions — Lecture 14 (3)

5. Your mAP over IoU 0.50 to 0.95 on 128 COCO images is 43.9; torchvision reports 37.0 for the same weights on…

✓ Run a reference evaluator on the same predictions and annotations (ours agreed to 0.0014); the gap is then the sample — 128 images, many easy, and classes with a single instance.

- An implementation you wrote yourself is checked against one you did not before its number is quoted
- A small evaluation set moves the score, here in the flattering direction, with the detector unchanged

LECTURE 16 · GÉRON CH. 13

Recurrent networks

A one-day-ahead forecast, the baselines it must beat, and a split that respects time
A recurrent cell, unrolled, and trained by backpropagation through time
LSTM and GRU gates, and the 2020 regime change