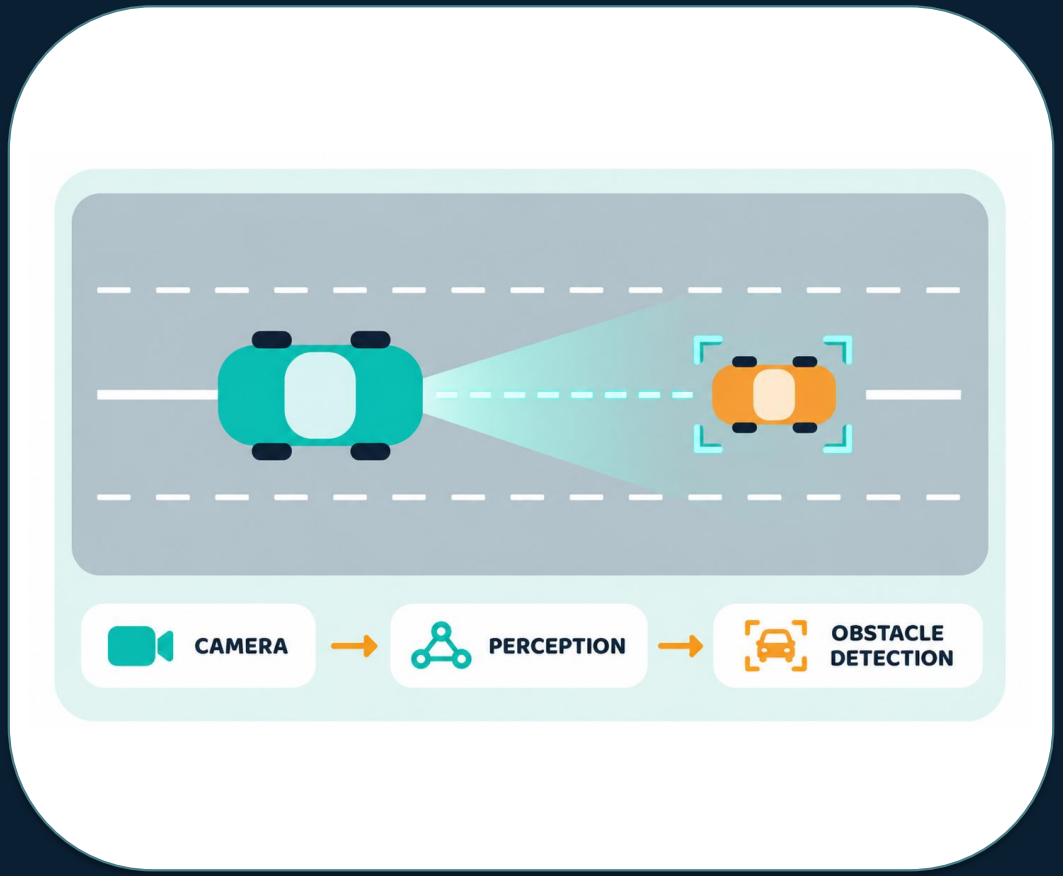
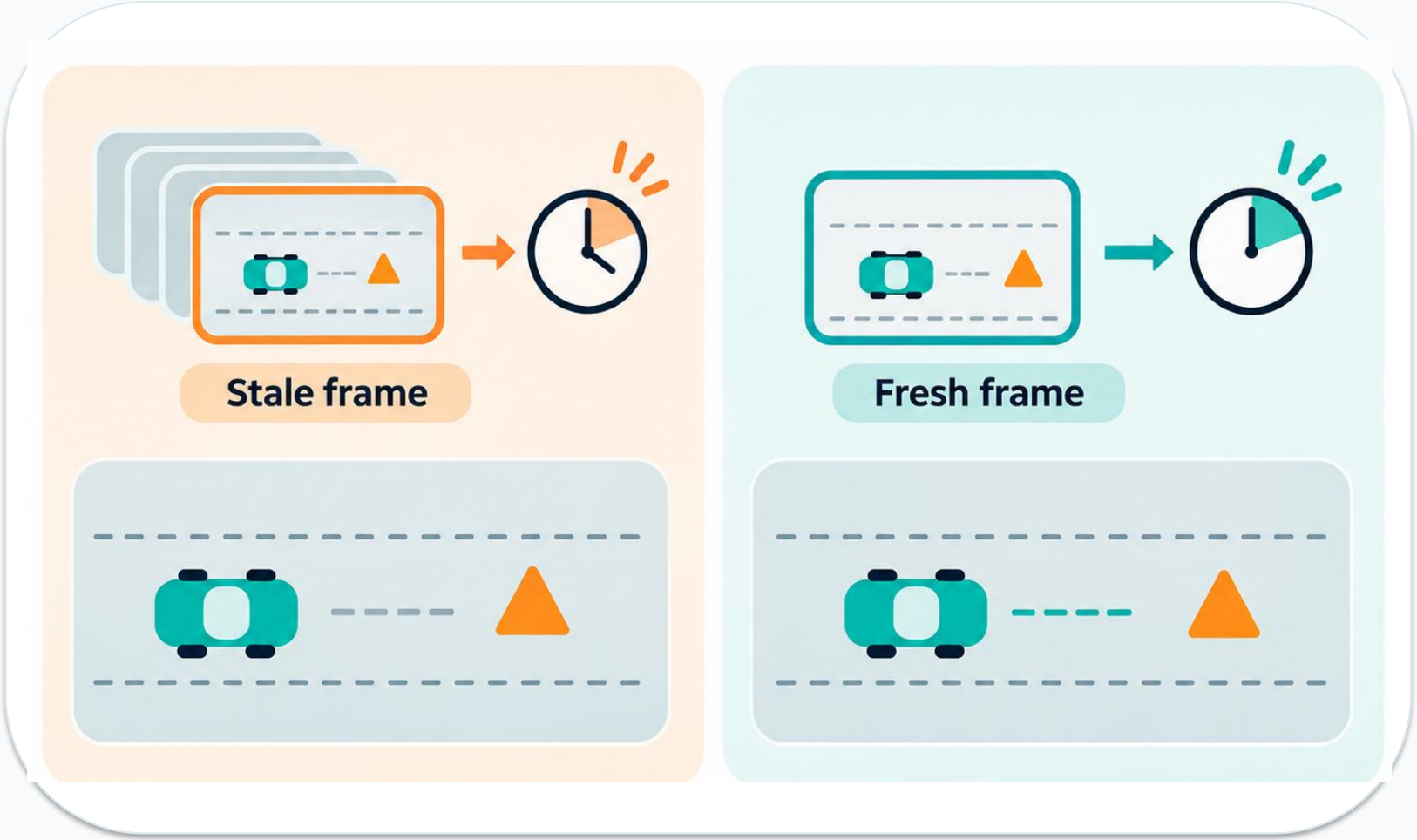
LM-RtA · RESEARCH WORKSHOP · 3 NOV 2026

Adaptive Real-Time Visual Processing Under Latency and Resource Constraints

Related work and research hypotheses



When camera information arrives too late



- Old frames can delay decisions.
- Skipping frames can save work.
- Safer driving must be tested.

Two studies on information freshness



VISTA · 2026

- Prioritizes recent frames.

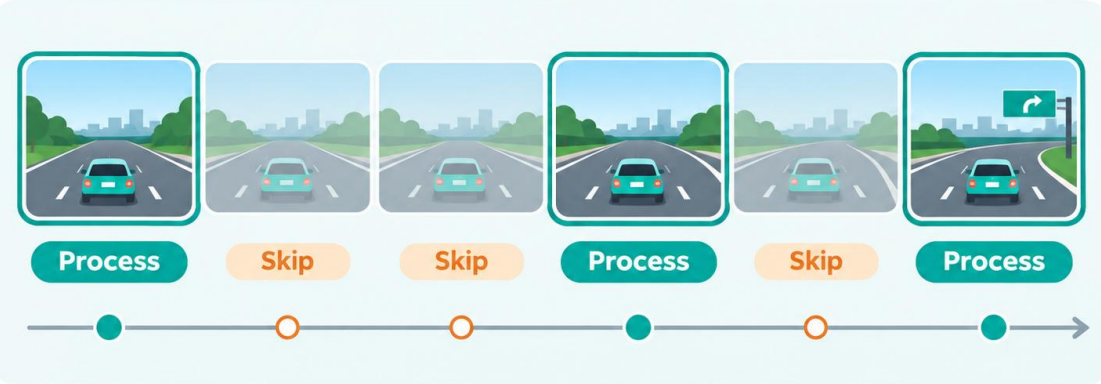
AoI Scheduling · 2025

- Reduces the age of task data.

Neither directly proves safer driving.

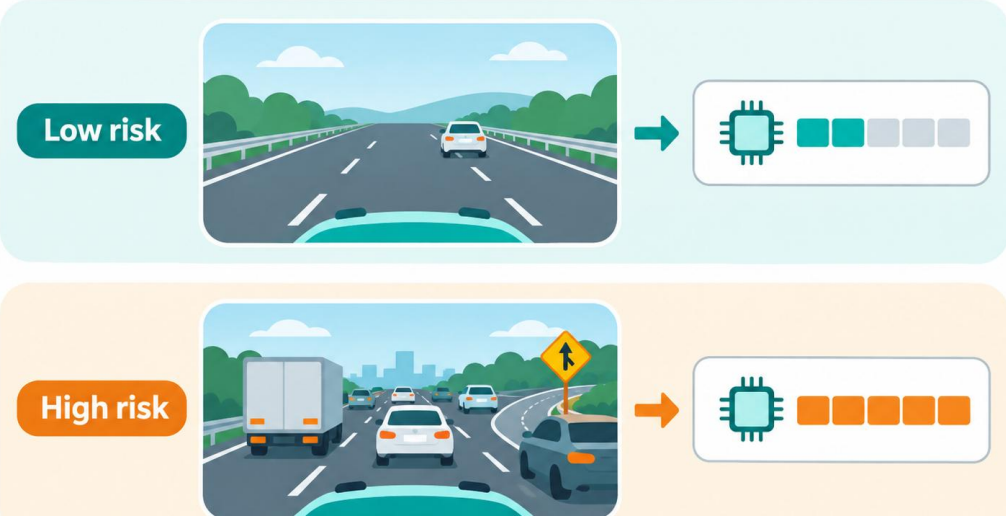
Two studies on using computation wisely

FrameHopper · 2022



- Skips less useful video frames.

Tonic Meta-Control · 2026



- Changes effort with driving risk.

What we want to find out

RQ1 / H1

Will fresher frames improve driving safety?

- The effect may change with traffic and delay.

RQ2 / H2

Can we process fewer frames without losing safe control?

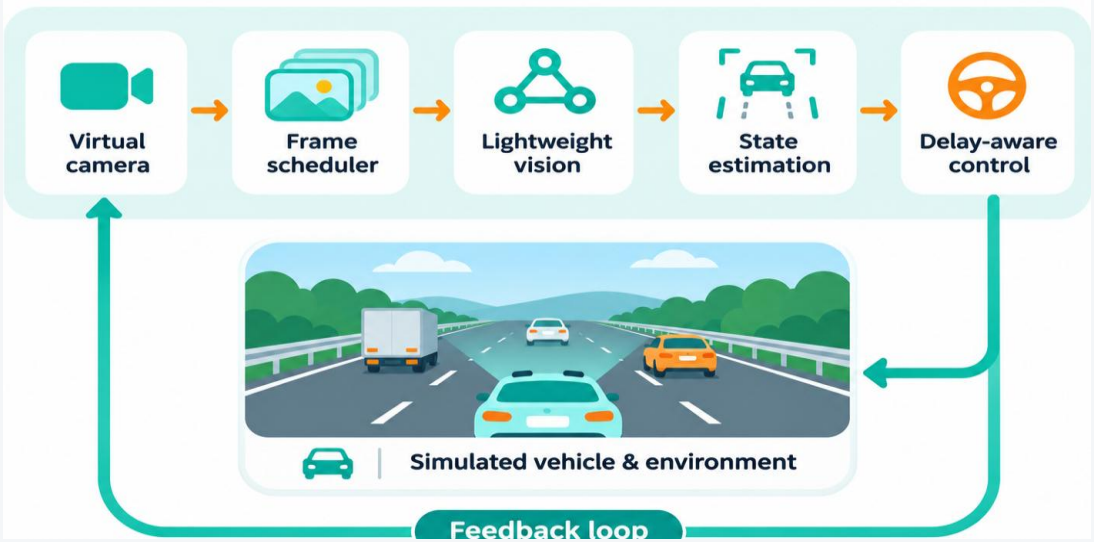
- We may save work within a safety limit.



How we could test the hypotheses

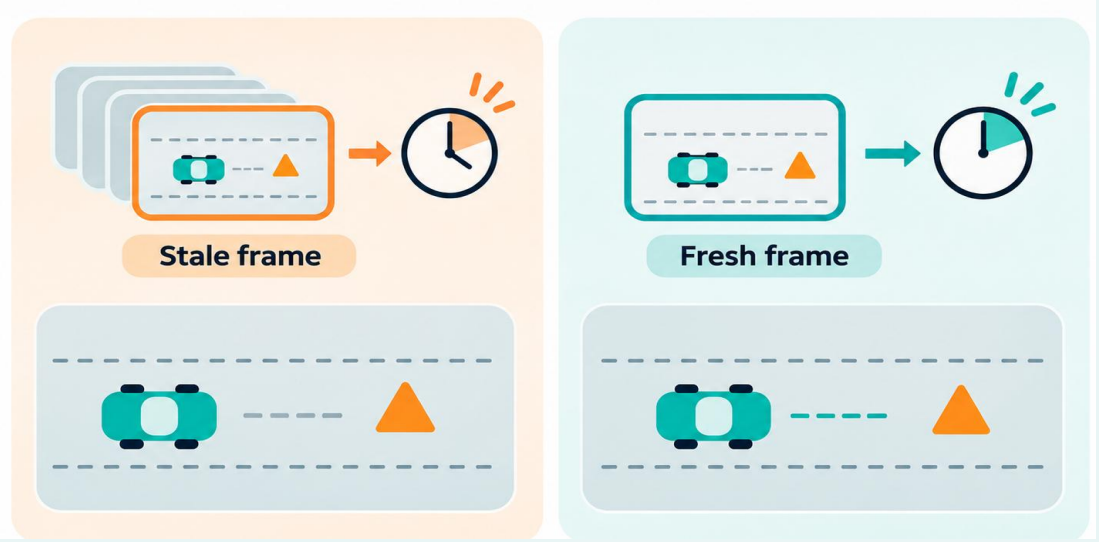
Simple visual plan of the experiment.

1. Closed-loop workflow



Same route and same seeds
Add delay, jitter, frame loss

2. Strategy comparison



FIFO baseline
Fresh-frame method
Optional frame skipping

What should we measure?

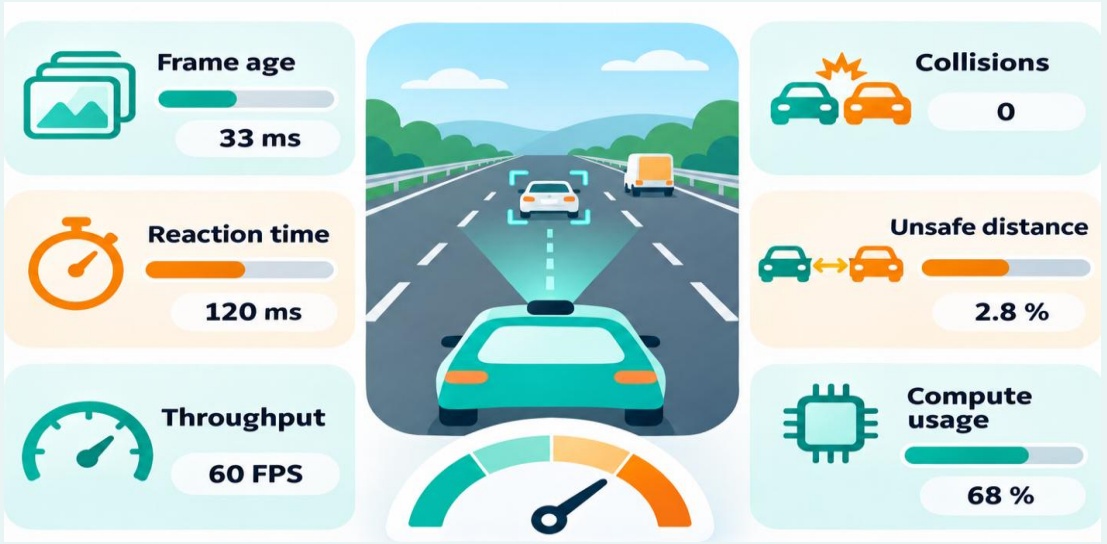
Key conditions and outcomes.

3. Driving conditions



Light traffic and dense traffic
Vary delay and compute limits

4. Outcomes to report



Freshness
Workload
Driving safety and progress

LM-RtA · PART 2 · 3 NOV 2026

Research plan: from hypotheses to experiments

What we will do, who does what, and what we need to decide

Three parts, one research question

Fatai

Related work

Scientific background, literature review, hypotheses

Roman

Simulation

Simulator, baseline controller, delay injection

Nikita

Visual processing

Perception, evaluation method, metrics

Roles are a starting point. We make final decisions together.

Where we are and what comes next



We are here: between hypotheses and the main study.

Before the main study we run six short pilots to check our setup and our metrics.

What we do, step by step (proposal)

Before 3 Nov

Prepare

- Finish literature table
- Fix metric list
- Agree roles

Team (all)

Week 1

Pilots

- P1 delay calibration
- P2 perception ceiling
- P3 hazard micro-scenarios
- P4 seed variance
- P5 TTC sanity check
- P6 stale-rule ablation

Nikita: P1, P2, P4, P5

Roman: P3, P6

Week 2

Main study

- Frozen plan in study.json
- All policies, all cells
- 10+ seeds per cell

Roman: runs

Nikita: metrics

Week 3

Analysis

- Bootstrap CIs over seeds
- Figures and tables
- Honest limitations note

All, Fatai: write-up

Six cheap checks before the main study

P1

Delay calibration

Where does overload start on our hardware?

P2

Perception ceiling

How much loss comes from perception alone?

P3

Hazard micro-scenarios

Is the lead-time signal big enough to see?

P4

Seed variance

How many seeds do we need?

P5

TTC sanity check

Does our TTC match a reference on logs?

P6

Stale-estimate rule

Does the 2-second rule decide the ranking?

How we will compare the policies

Policies

FIFO baseline

Newest frame

Fixed rate

Adaptive (RAB-1)

Conditions

Load sweep (ρ)

Light and dense traffic

Sudden braking

Cut-in

Measures

Primary: collisions in hazard window

Min TTC, lead time

Decision age

Processed-frame share

Same tape for every arm · 10+ seeds per cell · held-out seeds 101–130 for final test

Milestones (proposal, dates to agree)



Workshop

Present plan

Pilots P1–P6

Calibrate, check metrics

Main study

All arms, all cells

Analysis

Figures, honest note

Perception S5

then held-out S6

What we need from the team

1

Primary metric

Collisions in the hazard window, with min TTC as a check. Agree?

2

Load points

Which ρ values are in scope for the main study?

3

Perception ceiling

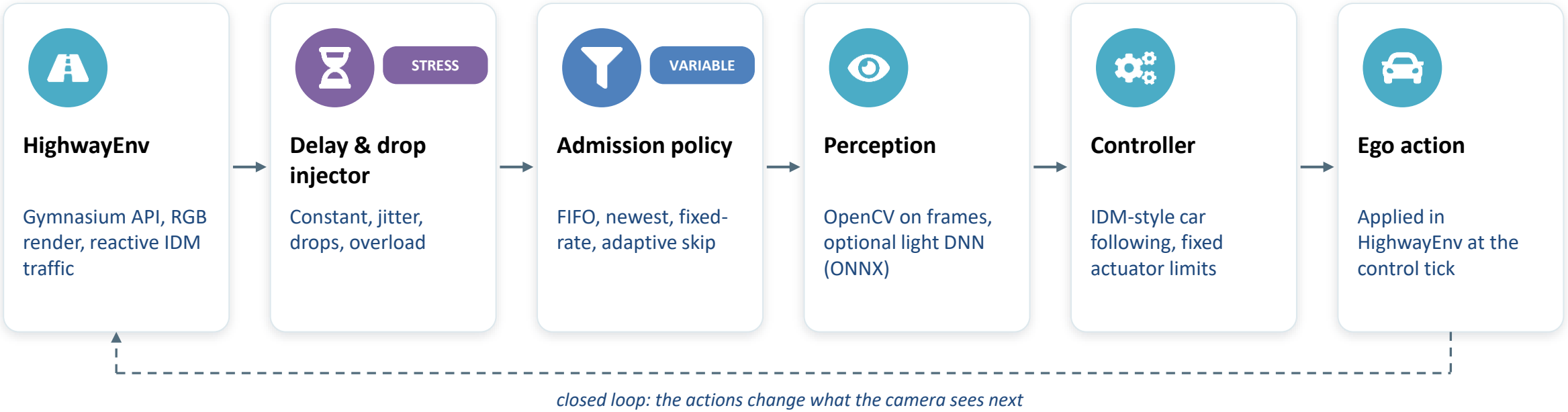
Do we accept it as a limitation, or report it as a result?

4

H8 this cycle?

Delay-aware prediction now, or in the next cycle?

Testbed: delay injected before perception



Same for every arm

Seeds and one recorded delay tape, control rates, controller and actuator limits. Only the admission policy changes.

Rates (proposed)

20 Hz physics and 10 Hz control. HighwayEnv defaults are 15 Hz and 1 Hz, too coarse for a latency study.

Tooling

Python, NumPy, pandas, SciPy bootstrap CIs, matplotlib. Frozen study JSON in, runs.csv out.

Delay is not one number: four kinds to inject

Constant latency d

Every frame arrives d late, with d swept in control ticks. The simple baseline.

Jitter

Random per-frame delay with a heavy tail. DNN inference time often varies by more than 100 ms (Liu et al.).

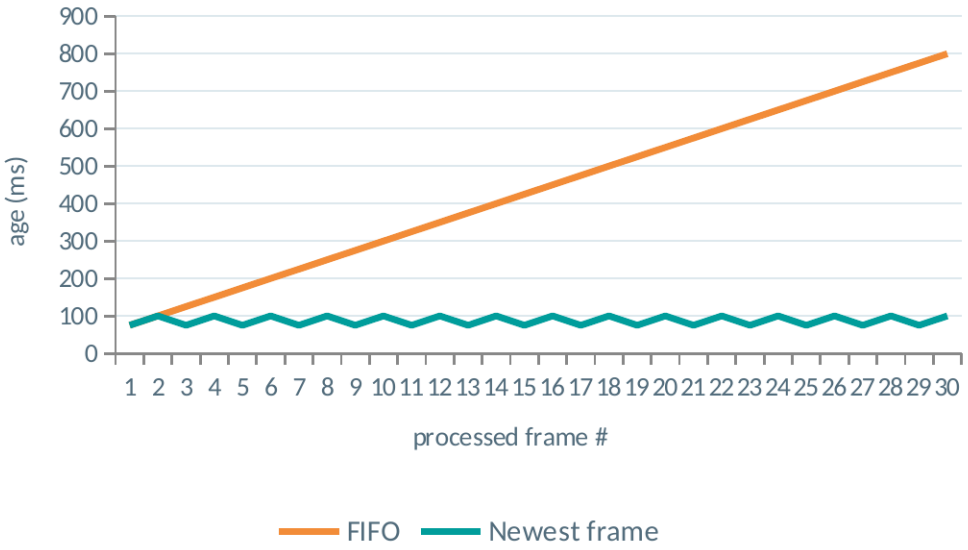
Frame drops

Random and bursty loss, as when a pipeline sheds frames under load.

Overload ρ

ρ = arrival rate × service time. Sweep ρ ≈ 0.8, 1.0, 1.5, 1.86 (the VISTA replay used 1.86).

Output age under overload (toy model)



Analytic illustration, not a result: 20 Hz camera, 75 ms service time (ρ = 1.5).

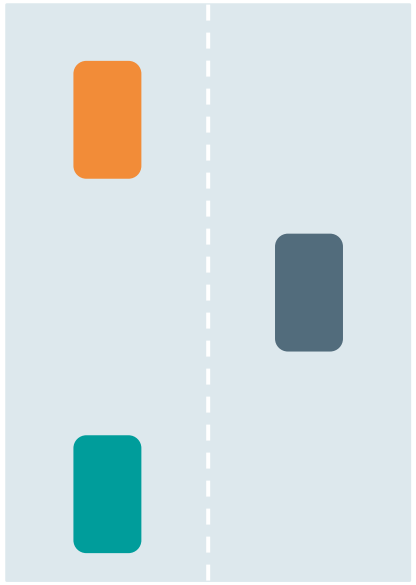
Decision age = age of the estimate at the moment the controller acts. Our primary freshness metric.

Baseline controller and three hazard scenarios

Baseline controller

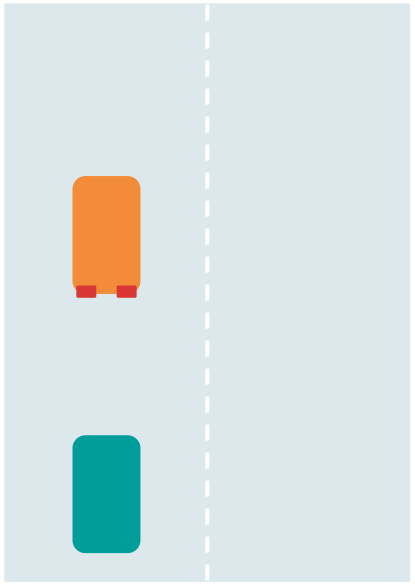
- Input: state estimate from perception (lead gap, relative speed, lane occupancy)
- Longitudinal: IDM-style car following
- Lateral: lane keeping, lane change only on a fresh estimate
- Other traffic is reactive (IDM), so evaluation is closed-loop
- Parameters and actuator limits fixed for every arm

Stale-estimate rule: estimate older than 2 s means cautious speed and no lane change. It may decide the ranking on its own, so we ablate it (P6).



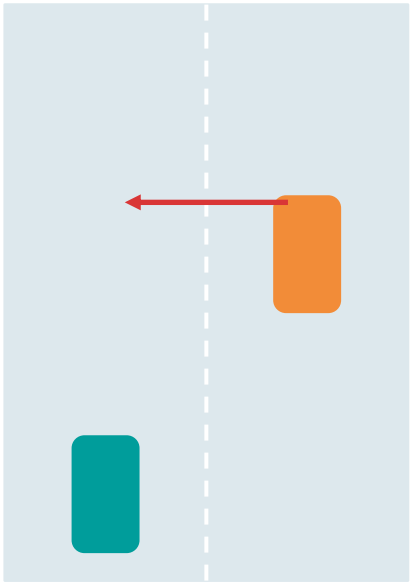
Normal flow

Overload without a hazard. Shows what freshness costs or saves in calm traffic.



Lead braking

Hazard with a clear warning: the lead car slows over many frames.



Cut-in

Hazard visible in only a few frames. Where frame skipping is most likely to fail (H2b).

EXTRA LITERATURE

Extra sources that shape our setup

HIGHWAYENV · LEURENT, 2018

Fast kinematic simulator with reactive IDM/MOBIL traffic. Perception is abstracted away.

For us: We add RGB rendering and OpenCV perception, so results hold for a simplified visual model.

DNN TIMING · LIU ET AL., 2022

Most tested driving DNNs vary in inference time by more than 100 ms.

For us: Inject jitter, not only fixed delay. Basis for H3.

OPEN VS CLOSED LOOP · WANG ET AL., 2026

Distance metrics (ADE/FDE) showed no reliable correlation with closed-loop driving score.

For us: Closed-loop with reactive traffic is our main evidence. Open-loop is only a proxy.

TONIC META-CONTROL · HAN ET AL., 2026

Adapts compute depth to accumulated risk. Part of its saving is a shallow first response at hazard onset.

For us: Tests whether a shallow onset costs lead time. Basis for H5.

Details taken from abstracts and our metrics notes; full-text check pending before quoting numbers.

ADDITIONAL HYPOTHESES

New hypotheses from the simulation side

H3

Delay shape matters

CLAIM

At the same mean delay, jittery or bursty delay hurts control more than constant delay.

TEST

Constant vs jitter vs bursty drops at matched mean age. Compare min TTC and collisions on the same tape.

FALSIFIED IF

Outcomes depend only on mean age.

H4

A critical delay exists

CLAIM

Beyond some delay, relative to braking time, no admission policy keeps the car safe.

TEST

Sweep d and ρ for FIFO, newest and adaptive. Look for the point where all policies fail.

FALSIFIED IF

Some policy keeps collisions at baseline over the whole sweep.

H5

Shallow onset costs lead time

CLAIM

Responding cheaply at hazard onset loses lead time. Forcing full depth on the first trigger recovers it.

TEST

Tonic meta-control vs a variant forcing S2 on the first trigger, on cut-in and braking.

FALSIFIED IF

Lead time to hazard is equal for both.

Numbering continues from H1 and H2 (RQ1/RQ2 on slide 4). Each hypothesis needs a pilot (P3, P4) before the main study.

NEXT STEPS

What I build next, and what we should decide

1. **Environment wrapper**
   HighwayEnv with rendered frames, fixed seeds, 20 Hz / 10 Hz rates.

2. **Delay injector and tape**
   Constant, jitter, drops, overload. One recorded tape replayed for all arms.

3. **Baseline controller**
   First on ground-truth state, then on OpenCV perception (pilot P2).

4. **Calibration with Nikita**
   Find where overload starts on our hardware (P1), then the stale-rule ablation (P6).

Questions for the discussion

1. Are 20 Hz physics and 10 Hz control the right rates?
2. Which delay kinds go into the main study?
3. FIFO baseline: unbounded or bounded queue?
4. First pilot: how much does delay alone hurt control?
5. Is rendered RGB plus OpenCV an acceptable simplified visual model?
6. Lightweight DNN detector now, or next cycle?