Disentangling Where & When

Factored Spatio-Temporal Explanations For Video Action Recognition

Bhavan Vasu[1] Giuseppe Raffa[2] Prasad Tadepalli[1]

[1] Oregon State University [2] Intel Labs

Oregon State University

A video model says “put down plate.” Why?



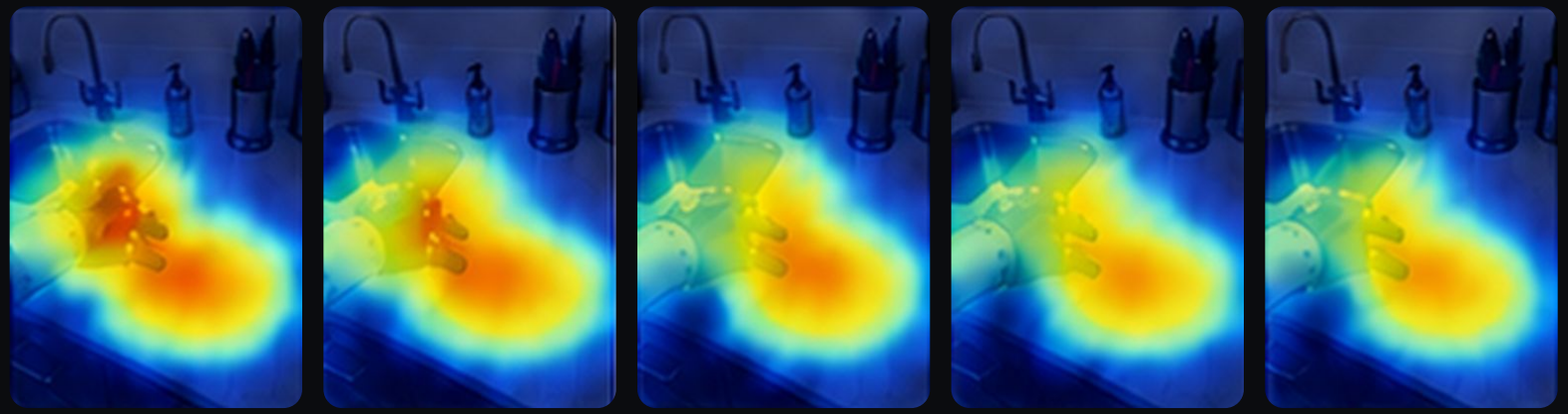
A It recognized the plate

B It tracked the downward motion

C Both

D No way to know

So you run a saliency method



WHAT YOU GET

Heat on the plate, in every frame

WHAT YOU ASKED

A, B, C or D?

Is that heat the plate, or the plate moving down?

A THOUGHT EXPERIMENT

Two models. Same accuracy. Same heatmap.

	Model A · reads the plate	Model B · reads the motion
Benchmark accuracy	Same	Same
Saliency map	Same	Same
A flat lid lowered the same way	RIGHT No plate, so no “put down plate”	WRONG Sees the motion, says “plate”
Plate hidden behind the gripper	WRONG Can't see the plate, misses it	RIGHT Motion still visible

Which one is on your robot? Nothing in today's toolbox can tell you.

Two questions, no tool to answer them

01

Where is the evidence?

Which pixels carry the noun

02

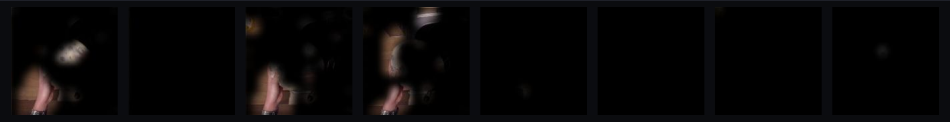
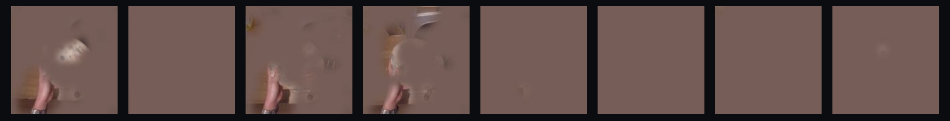
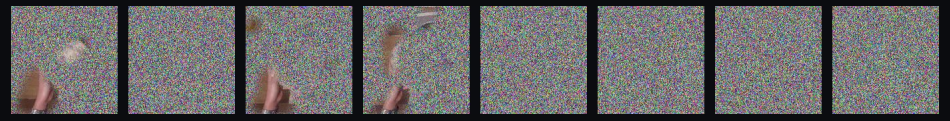
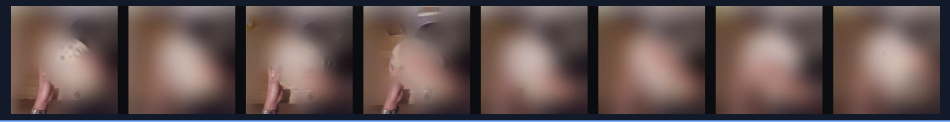
When does it matter?

Which moments carry the verb

Two factors, two parallel branches

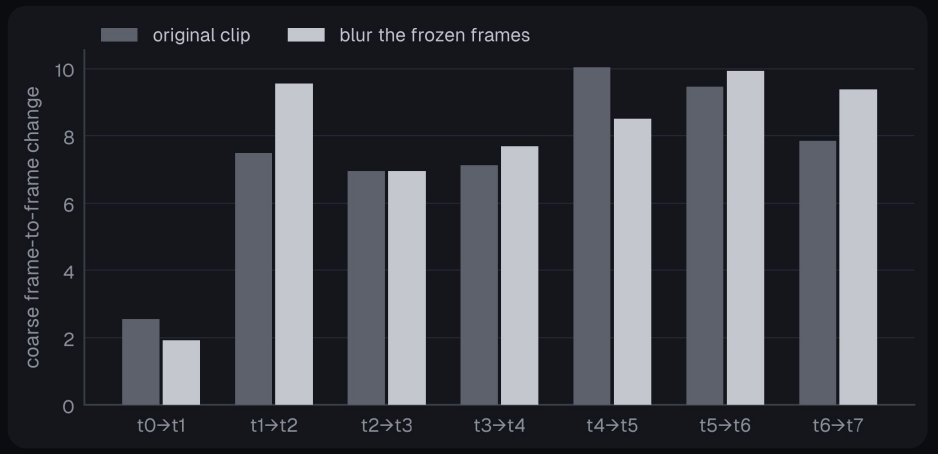


How do you take appearance out of a frame?

outside M_a	t0 → t7	WHAT GOES WRONG
Original		Reference
Black-out		Off-distribution; the mask's outline becomes a new cue
Mean fill		Flat color: off-distribution, no context left
Noise		Adds texture the model reacts to; changes every run
Blur		Keeps layout and color, removes the detail

Blurring a frame doesn't stop it moving





BLUR IS SPATIAL

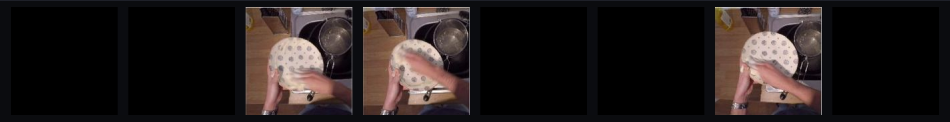
Blurring a frame also deletes its appearance, so “when” would leak into “where”

Freeze by copying the nearest kept frame



Three static runs: Motion only survives at the boundaries the mask chose.

Why this makes sense?

M□	0	0	1	1	0	0	1	0	WHAT GOES WRONG
Original									Reference
Black frames									Out of distribution
Mean frame									Ghost of the whole clip
Shuffle									Invents new trajectories, cant work for stirring
Soft blend									Ghosting from mixed neighbors
Nearest frame									Motion exactly zero, pixels real

Hard in the forward pass, soft in the backward

01 · THE OPERATOR

$$\hat{x}_t \;=\; M_t\, x_t \;+\; (1 - M_t) \sum_j \tilde{w}_{t \to j}\, x_j$$

Kept frames pass through; frozen ones copy another

02 · REPLACEMENT WEIGHTS

$$w_{t \to j} \;=\; \frac{\exp\!\left(-\frac{(t-j)^2}{2\sigma_R^2}\right) m_j}{\sum_{k \neq t} \exp\!\left(-\frac{(t-k)^2}{2\sigma_R^2}\right) m_k \;+\; \epsilon}$$

Only kept frames compete; nearer wins · σ_R = 2

03 · TWO STRAIGHT-THROUGH ESTIMATORS

$$j^*(t) = \arg\max_j\, w_{t \to j} \qquad \tilde{w}_{t \to j} = \mathbf{1}[\, j = j^*(t)\,] - \mathrm{sg}(w_{t \to j}) + w_{t \to j}$$

$$M_t \;=\; \mathbf{1}[\, m_t > 0.5\,] \;-\; \mathrm{sg}(m_t) \;+\; m_t$$

Forward: hard argmax copy · Backward: soft weights

04 · STE BIAS · PROP. 1

$$\| \hat{x}_t^{\,\mathrm{hard}} - \hat{x}_t^{\,\mathrm{soft}} \| \;\leq\; (1 - \Delta_t)\, \delta_t$$

Small when the pick is confident or candidates match

Four choices that keep the masks honest

01

Split supervision

Noun trains M_a, verb trains M□.

02

One-sided area budget

ReLU(sort(m) – t)² penalizes only excess. MSE forces masks to fill the budget.

03

Separation loss

$\Sigma M_a \cdot (1 - M□)$: no appearance budget on frames the model never sees.

04

Flow alignment

For single-head SSv2: align M□ with optical-flow magnitude.

One loss, four jobs

THE GOAL · EQ. 1

$$\max_{M_a,\,M_m} \; p_{n^*}(\hat{x}) + p_{v^*}(\hat{x}) \qquad \text{s.t.} \quad \tfrac{1}{THW}\textstyle\sum M_a \le B_a, \quad \tfrac{1}{T}\textstyle\sum M_m \le B_m$$

RELAXED INTO ONE LOSS · EQ. 7

$$\mathcal{L} = \underbrace{-\log p_{n^*}(\hat{x}) - \log p_{v^*}(\hat{x})}_{\text{split prediction}} + \underbrace{\lambda_A \mathcal{L}^a_{\text{area}} + \lambda_M \mathcal{L}^m_{\text{area}}}_{\text{sparsity}} + \underbrace{\lambda_S \mathcal{L}_K}_{\text{smoothness}} + \underbrace{\lambda_D \mathcal{L}_{\text{sep}}}_{\text{separation}}$$

SPLIT PREDICTION

Noun trains M_a, verb trains $M_{\square}$. –log p keeps gradients alive while masks are near-empty.

SPARSITY

A sorted area loss per mask, under budgets B_a and B_m. Next two slides.

SMOOTHNESS

3×31×31 ellipsoid kernel, same sorted loss: connected regions, not speckle.

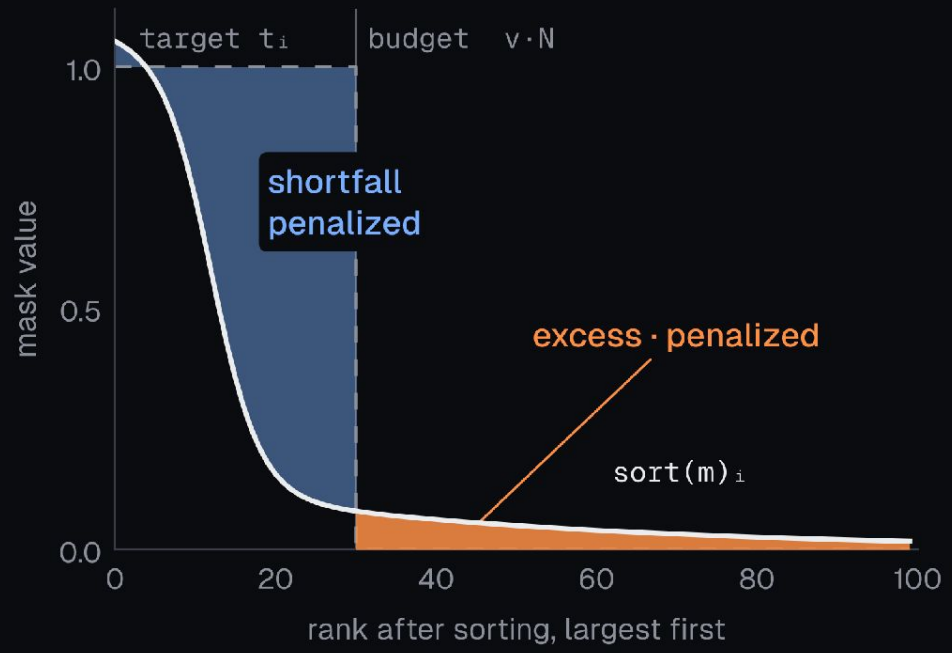
SEPARATION

$\Sigma\, M_a(1 - M_{\square})$: no appearance budget on frames the model never sees.

Sort the mask, then compare it to a budget

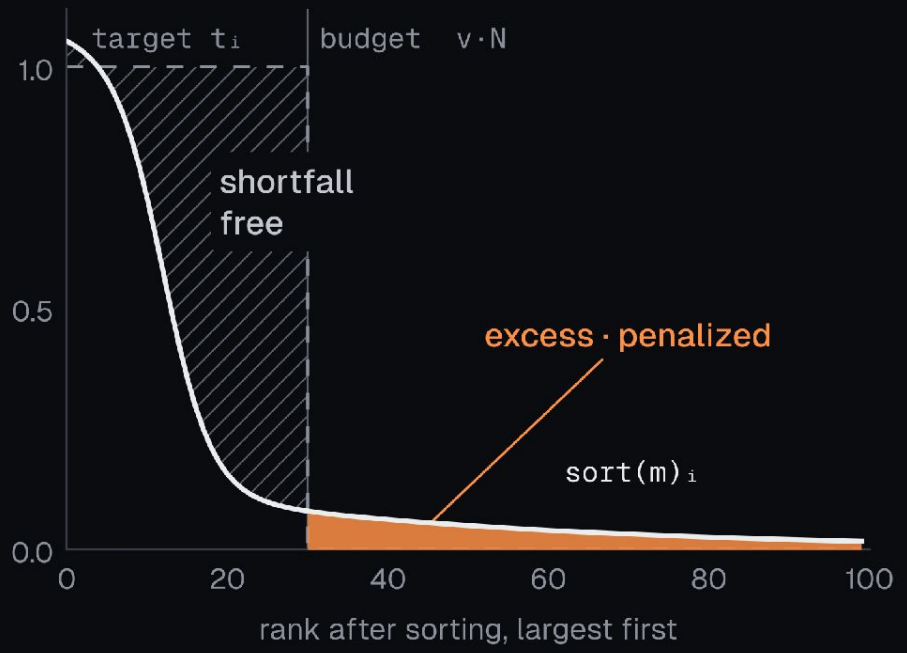
SYMMETRIC MSE · STEP

$$\mathcal{L}_{\text{area}}^{\text{MSE}} = \frac{1}{N}\sum_{i=1}^{N}\left(\text{sort}(m)_i - t_i\right)^2$$



ONE-SIDED ReLU · OURS

$$\mathcal{L}_{\text{area}}^{\text{ReLU}} = \frac{1}{N}\sum_{i=1}^{N}\text{ReLU}\left(\text{sort}(m)_i - t_i\right)^2$$



MSE fills the budget; ReLU can stop early



PROP. 3 · MSE SATURATES

$$|a^*_\lambda - v| \le \frac{L\sqrt{N}}{2\lambda}$$

Raise λ and the area is forced onto the budget

PROP. 4 · ReLU PERMITS LESS

$$\nabla\mathcal{L}_{\text{pred}}(m^*) = 0,\ \ a^* \le v \ \Rightarrow\ \nabla\big(\mathcal{L}_{\text{pred}} + \lambda\mathcal{L}^{\text{ReLU}}_{\text{area}}\big)(m^*) = 0\ \ \forall\lambda$$

Sub-budget stationary points survive any λ

~5×

below budget (EPIC, EGTEA)

.53 → .61

app. insertion, EGTEA

Lowest deletion AUC, ~5× sparser masks

APPEARANCE DELETION AUC ↓

0.10

vs 0.18 Extremal Perturbation · 0.23 STEP

MOTION DELETION AUC ↓

0.21

vs 0.50 Extremal Perturbation + temporal mask

PIXELS KEPT ↓

10.3%

vs 66.1% STEP · 21.1% unified mask

MOTION F⁻ ↑ · EGTEA

0.82

vs 0.12 STEP + M□ · 0.11 EP + M□

The pattern holds on every dataset

DATASET	APP. DELETION ↓	MOTION DELETION ↓	PIXELS KEPT ↓	MOTION F⁻ ↑
EGTEA Gaze+ 400 clips	0.27 vs 0.41	0.25 vs 0.77	12.4% vs 20.2%	0.82 vs 0.12
EPIC-Kitchens-55 9,385 clips	0.10 vs 0.18	0.21 vs 0.50	10.3% vs 21.1%	0.62 vs 0.21
Something-Something V2 200 clips · single head	0.07 vs 0.11	0.11 vs 0.42	22.8% vs 51.1%	0.62 vs 0.61

INSERTION AUC ↑ · OURS VS BEST · MASK SIZE

Here larger masks win. Why? →

EGTEA
App .75 vs .81 · 12% vs 20% px
Mot .79 vs .84 · 38% vs 82% fr

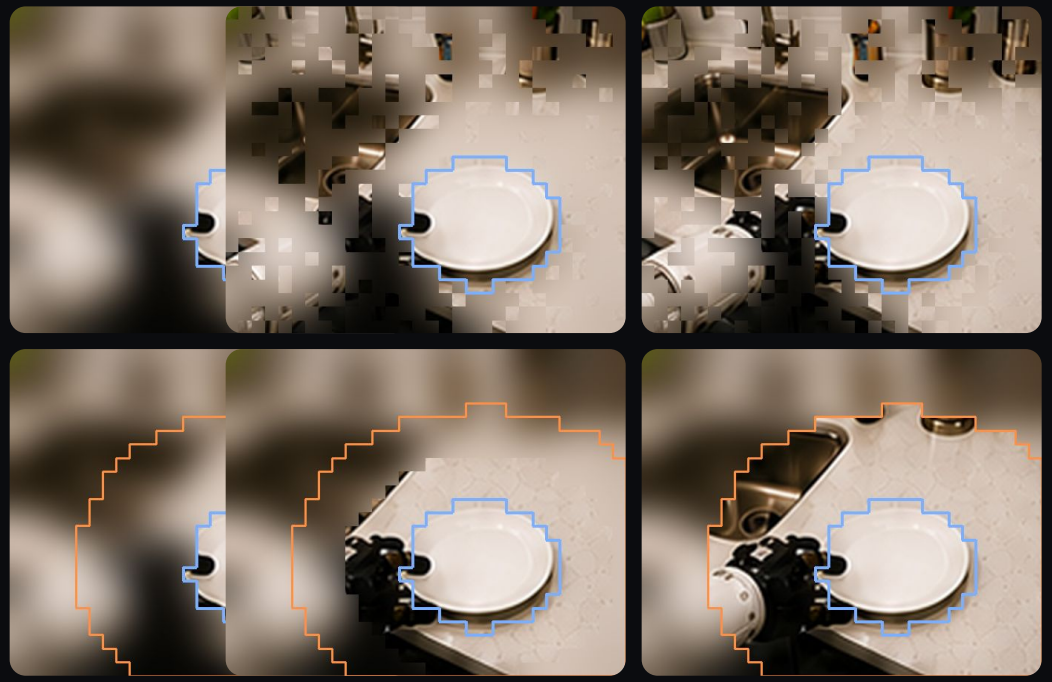
EPIC-KITCHENS
App .44 vs .59 · 10% vs 21% px
Mot .66 best · next .20

SSV2
App .19 vs .36 · 23% vs 79% px
Mot .14 vs .51 · 22% vs 87% fr

Same pixels, different order

marked by both (evidence)
marked only by mask 2

$\alpha = a_1 = 12\%$ $\alpha = 36\%$ $\alpha = a_2 = 60\%$



Mask 1
marks 12%

Mask 2
marks 60%

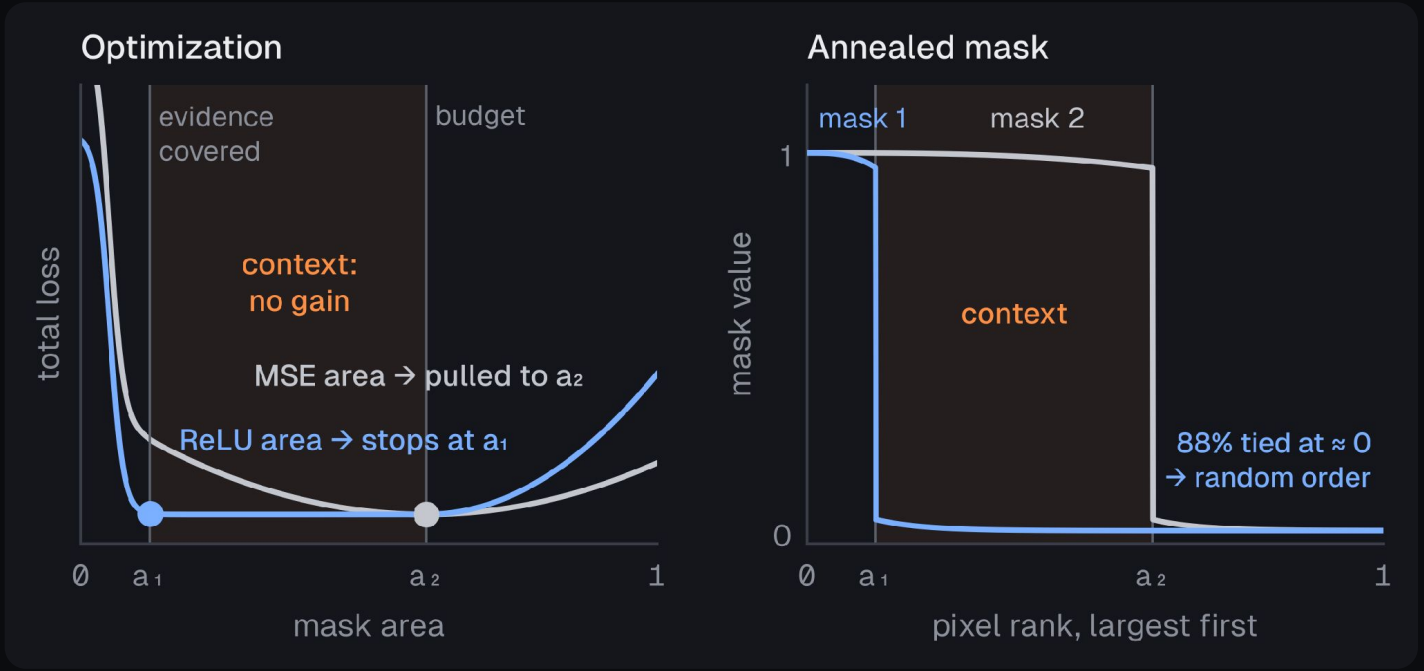
a_1, a_2
How much each mask marks. Each reveals its marked pixels first.

WHY NOT MASK 1?
After a_1 it has no ranking left: the other 88% come out in random order, so context arrives scattered and late.

WHY MASK 2 GAINS
Its extra 48% is context next to the evidence, revealed in order: confidence rises early and stays high.

Neither mask found more evidence. The large one just front-loads the context, and insertion pays for that.

Why the small mask skips the context



01 · NOTHING TO GAIN

With the plate kept, the model is already confident. ReLU area is flat below budget, so the mask stops at a_1 (Prop. 4).

02 · NO SECOND PLACE

Annealing makes the mask near-binary: context ends at ≈ 0, tied with the sink and the wall.

03 · FORCED TO FILL

MSE area pulls a mask to its budget (Prop. 3). Smoothing makes the filler the pixels next to the evidence.

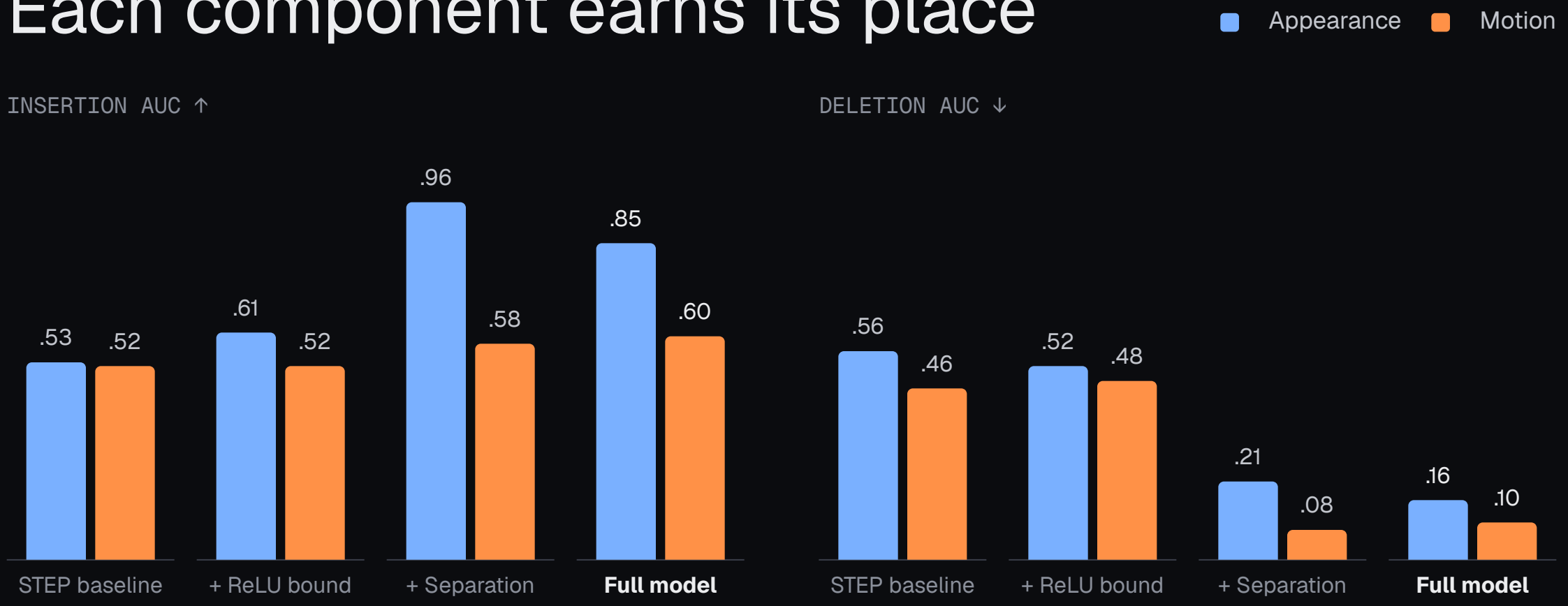
DELETION · IS IT NECESSARY?

Starts from the full clip. Remove the plate and confidence falls; context can't hold it up. Not needed.

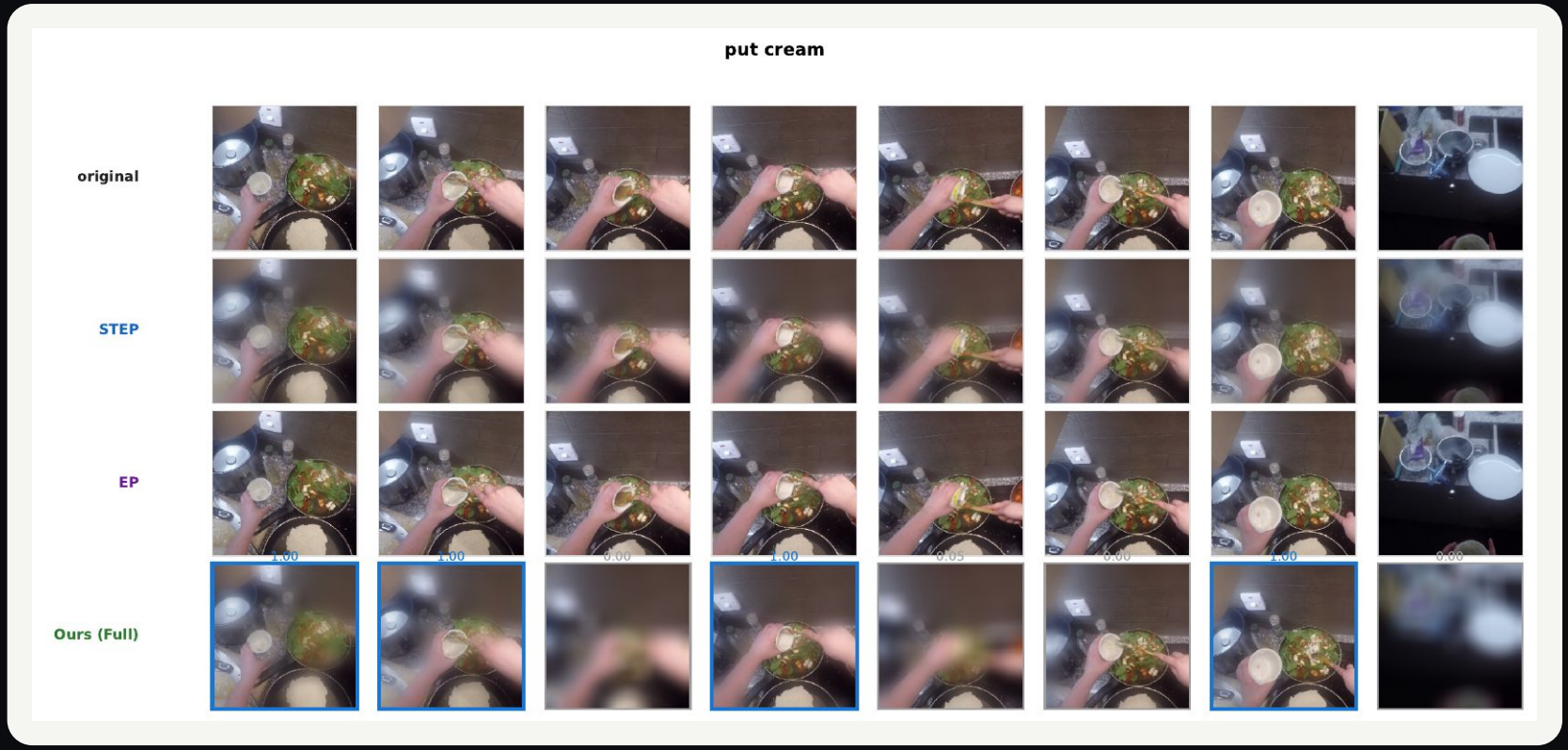
INSERTION · DOES IT HELP?

Starts from blur. The plate alone isn't fully convincing; nearby context speeds the rise. Rewarded.

Each component earns its place



Sparse in space, selective in time



“PUT CREAM”

Rows: input · STEP · EP · ours

Where

Only the hand and cream stay sharp

When

4 of 8 frames kept (outlined)

TAKEAWAYS

Explain where and when separately

01

Factor the explanation

Two masks expose failure modes one map hides

02

Do not penalize sparse masks

Sparse explanations should incur no loss

03

Nearest-frame replacement

New perturbation operator to perturb motion from videos

NEXT Multiple minimal explanations · Extend to VLMs · Localize where motion happens