UNIVERSITÄT
BERN

# Towards Systems that Learn by Themselves

Paolo Favaro

Computer Vision Group — University of Bern

# A long-standing goal

- To build machines that learn by themselves how to navigate environments and plan for tasks

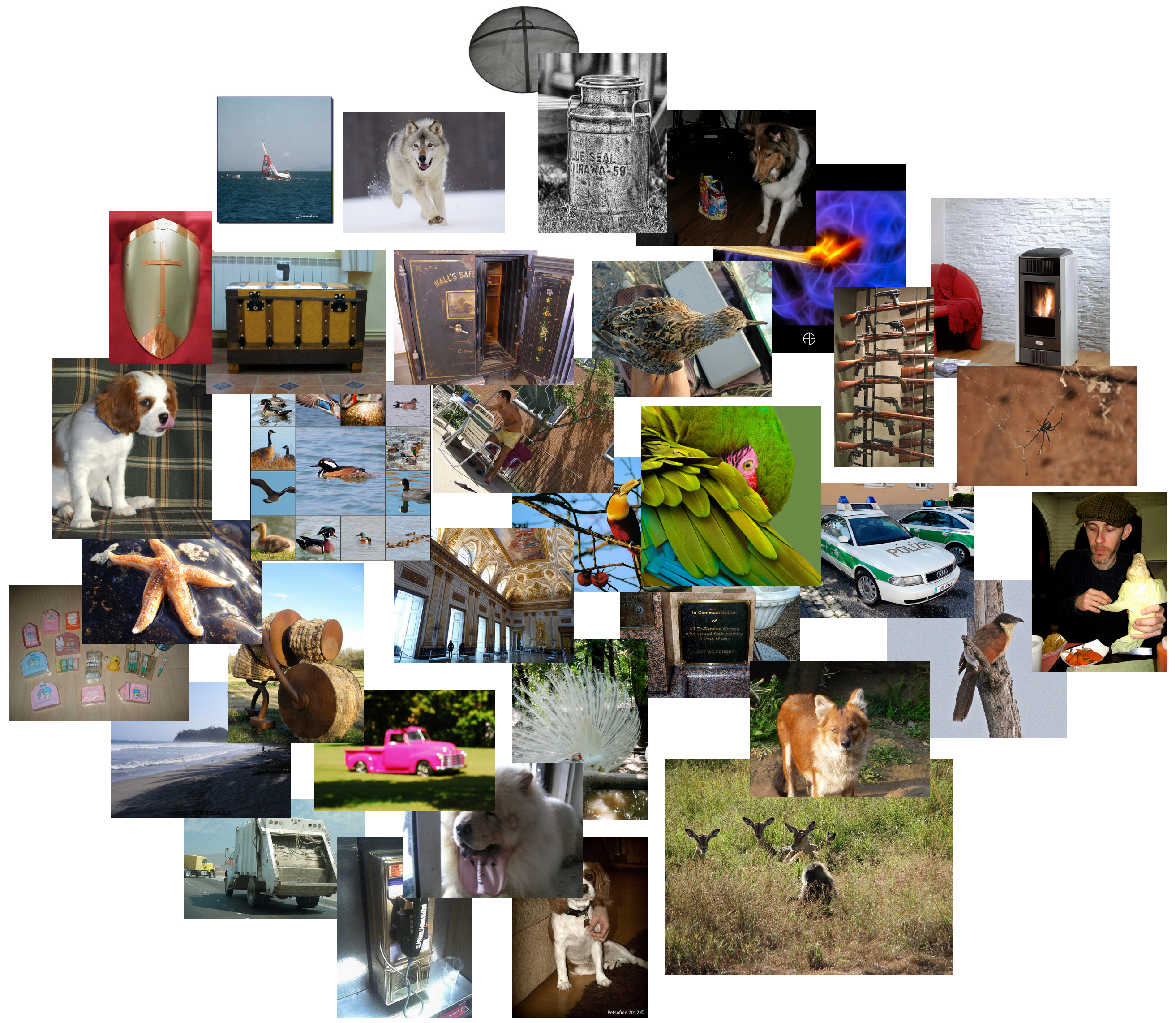
- We need to

  - Equip them with sensing devices for visual, auditory, tactile,… stimuli

  - Design algorithms to extract information from the observations

Image credits: MAAS Digital, NASA, JPL

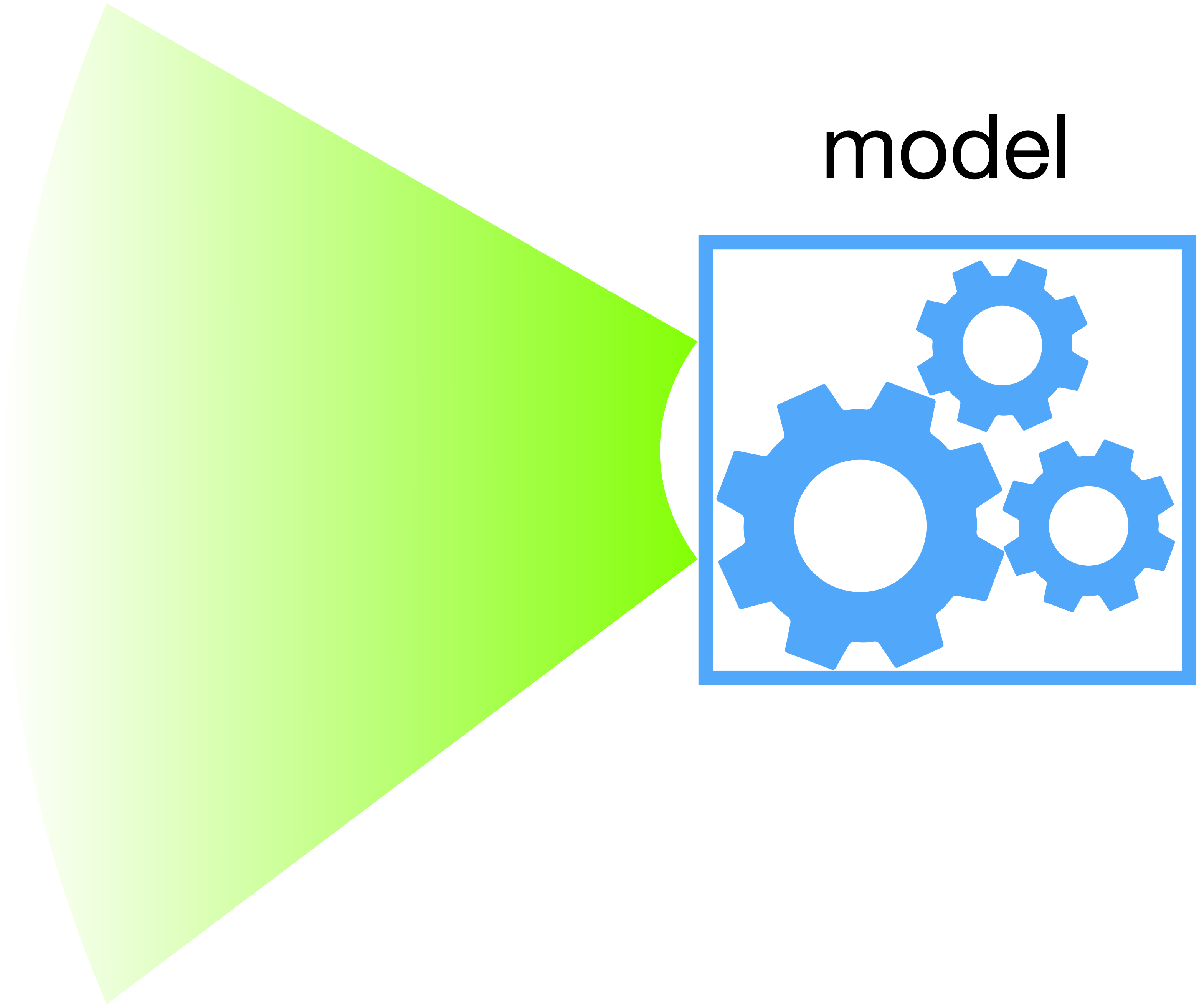
# Supervised learning



book car shield dog

data $d$

attributes $y$



model

task $p(y \mid d)$

Information is provided (manually) **per sample**

# Supervised learning

- Break down the problem into a set of tasks
- For each task provide a dataset with input-output pairings (supervision)
- Train a single model end-to-end to solve all tasks at once (or multiple models and then coordinate their operations)

# Does it sound familiar?

- Initially, we solved tasks by defining a set of pre-programmed rules and brute force search

- But we realized that we do not know what the best way of solving a task is…

# Should we also revise learning from examples?

- If we learn autonomous driving through examples…
- …we would also need to experience lots of accidents
- but is that how humans learn to drive*?

*although we certainly learn to walk through lots of falling!
Adolph et al, “How Do You Learn to Walk? Thousands of Steps and Dozens of Falls Per Day, Psychology Science, 2012

# some thoughts on Supervision

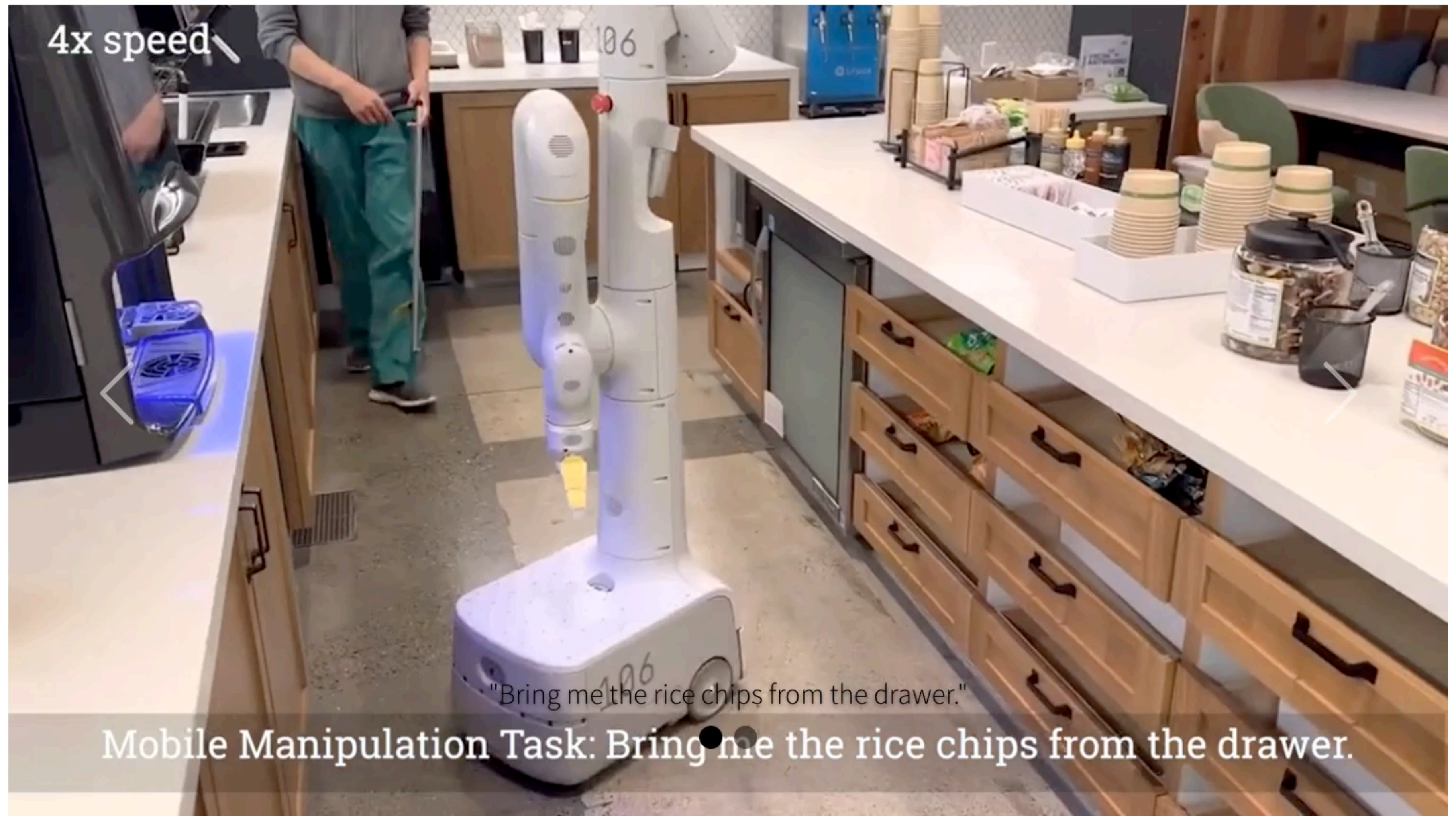
# Supervision today

- Multimodal learning shows that massive supervision is effective
  - Train with multiple signals (eg, images, videos, audio, text, segmentations, depth, normal maps, bounding boxes)
  - Example: PaLM-E
    (562B parms): 520B PaLM + 22B ViT
    Control loop with a robot
    Trained on single image + text prompts
  - Works also with a frozen PaLM

**Prompt:** `Human: <instruction>`
`Robot: <step history>. I see <img>`

Results

We show a few example videos showing how PaLM-E can be used to plan and execute long horizon tasks on two different real embodiments. Please note, that all of these results were obtained using the same model trained on all data. In the first video, we execute a long-horizon instruction "bring me the rice chips from the drawer" that includes multiple planning steps as well as incorporating visual feedback from the robot's camera. Finally, show another example on the same robot where the instruction is "bring me a green star". Green star is an object that this robot wasn't directly exposed to.




*Driess et al, PaLM-E: An Embodied Multimodal Language Model, ArXiv 2023

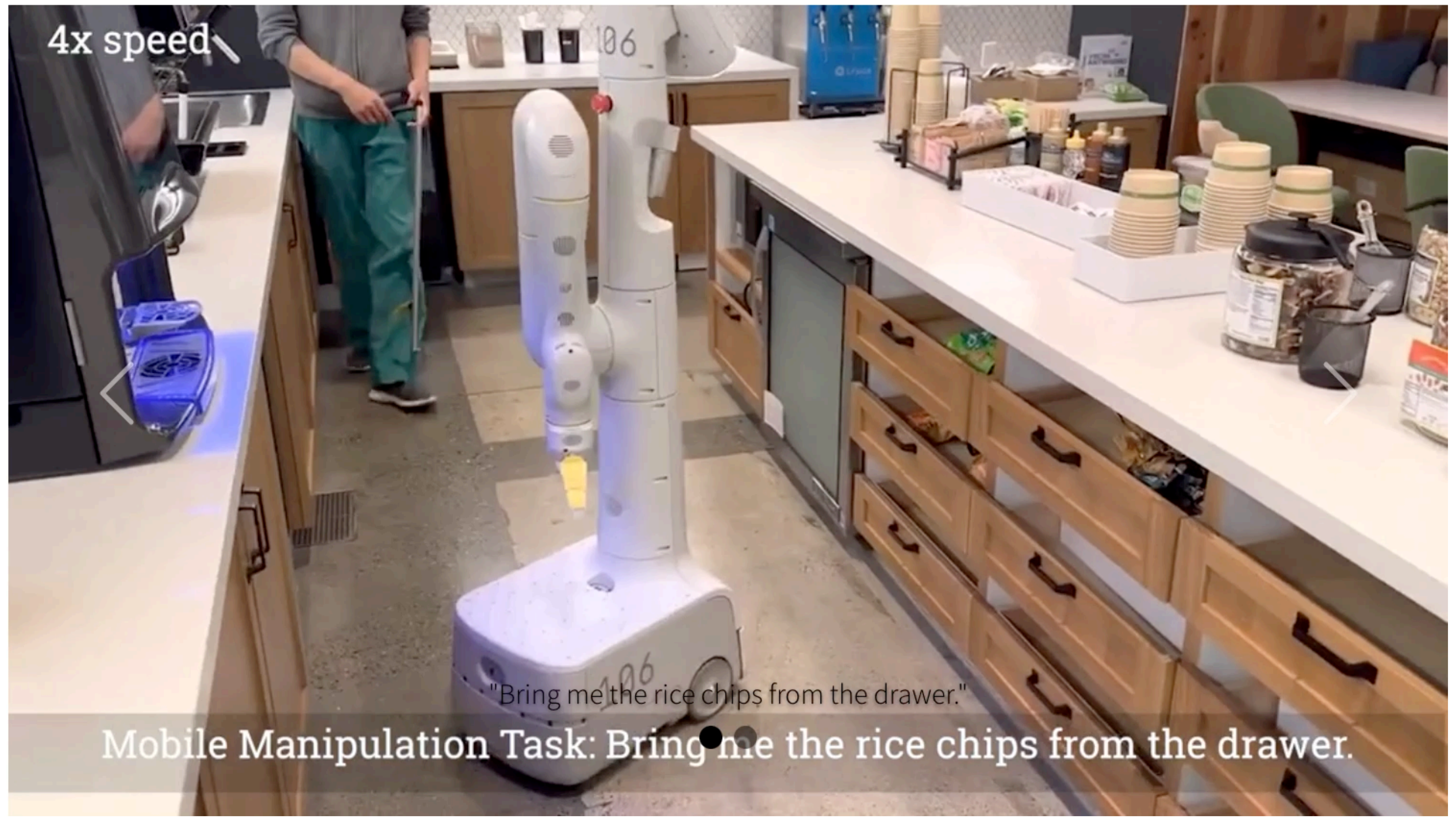
# Supervision today

- Multimodal learning shows that massive supervision is effective
    - Train with multiple signals (eg, images, videos, audio, text, segmentations, depth, normal maps, bounding boxes)
    - Example: PaLM-E
      (562B parms): 520B PaLM + 22B ViT
      Control loop with a robot
      Trained on single image + text prompts
    - Works also with a frozen PaLM

**Prompt:** `Human: <instruction>`
`Robot: <step history>. I see <img>`

Results

We show a few example videos showing how PaLM-E can be used to plan and execute long horizon tasks on two different real embodiments. Please note, that all of these results were obtained using the same model trained on all data. In the first video, we execute a long-horizon instruction "bring me the rice chips from the drawer" that includes multiple planning steps as well as incorporating visual feedback from the robot's camera. Finally, show another example on the same robot where the instruction is "bring me a green star". Green star is an object that this robot wasn't directly exposed to.

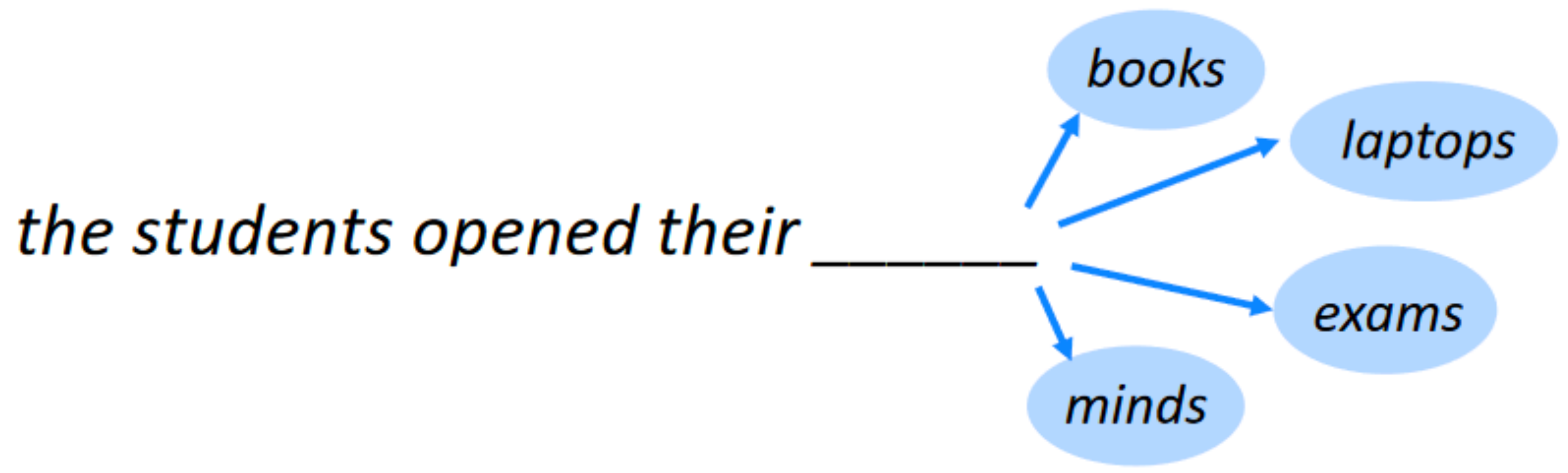



*Driess et al, PaLM-E: An Embodied Multimodal Language Model, ArXiv 2023

# Large Language Models

- LLMs are large models (billions to a trillion parameters) mostly trained on billions to trillion words/tokens to predict the next word

- LLMs are trained in an **unsupervised manner** (predict the next word task)




- LLMs such as GPT-X, PaLM-X, LLaMA have demonstrated surprising emergent abilities* not observed in small models

- Just learning the correlation in the data (ie, p(new word|previous words)) seems to go a very long way

*Wei et al, Emergent Abilities of Large Language Models, TMLR 2022

# Natural language supervision

- When is human annotation enough and not confusing to a model?

construction worker in orange
safety vest is working on road

?

man is pulling cables
behind orange machine

# A conjecture

- Human supervision will eventually limit the learning of large models
- Learning from raw data has the potential for the discovery of more patterns and knowledge than what is available in natural language
- Agents could use natural language to bootstrap their knowledge and to interface with human users, but not as the ultimate learning signal

# Self-learning

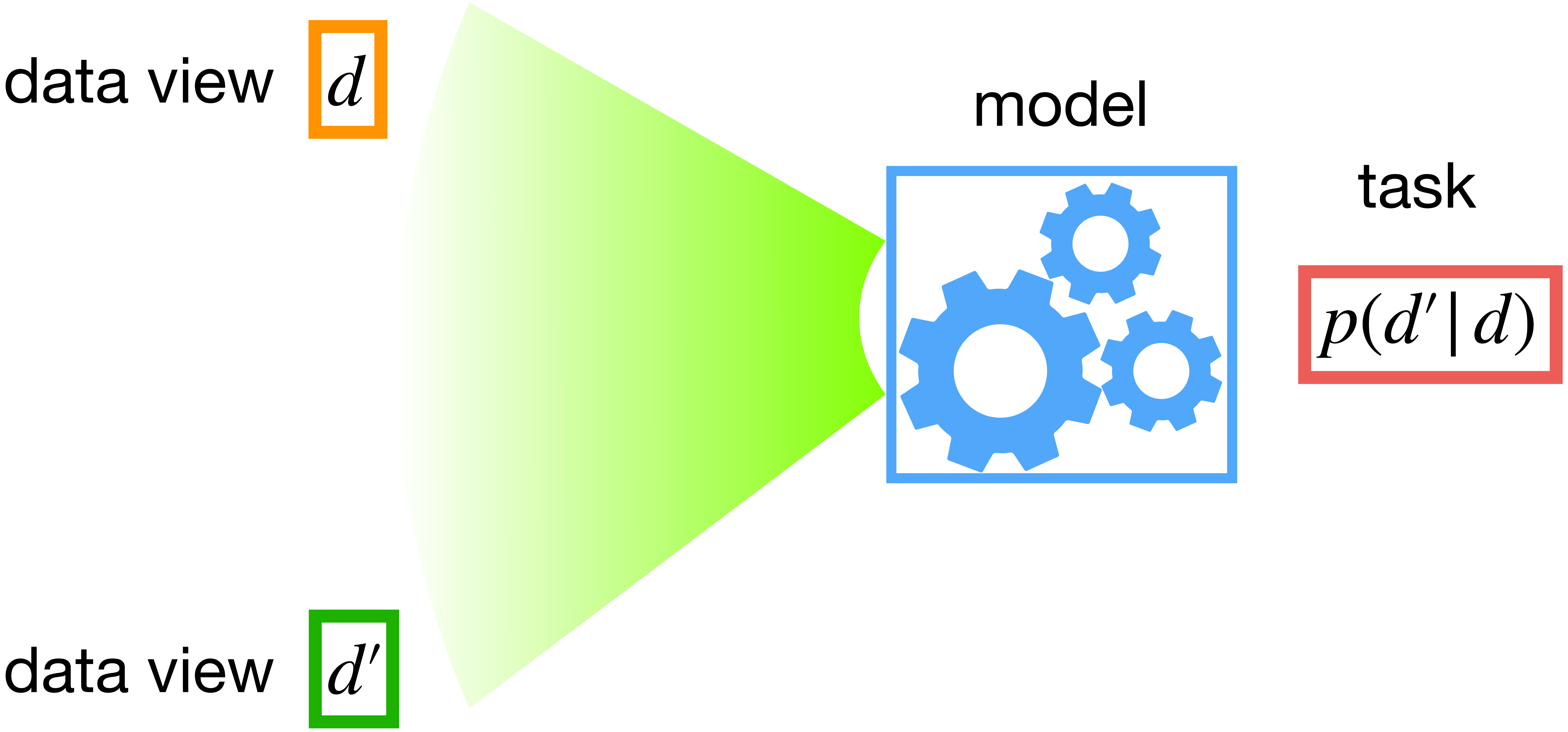
- Is it possible that there is an **uber-task** based on self-learning from which all the other capabilities emerge?

- Example: Given some past synchronized signals (eg, image frames, audio, tactile input), predict the future synchronized signals (eg, image frames, etc)

past frames → future frames

Images from Epstein et al, Oops! Predicting Unintentional Action in Video, CVPR 2020

# Unsupervised learning




Information is provided for the **whole dataset** (eg, a set of data augmentations)

# Why unsupervised learning?

- Why bother with UL when a lot of data with supervision* is readily available (eg, LAION)?
- Current SL methods work extraordinarily well
- The more supervision we combine, the better the performance (eg, multi-task learning in Flamingo [Alayrac et al 2022])

*although labelling may be unreliable and require further processing

# Why unsupervised learning?

- Performance increases with more data (a lot of data), so data collection costs can be high, time-consuming and error-prone

- Human annotation is not scalable

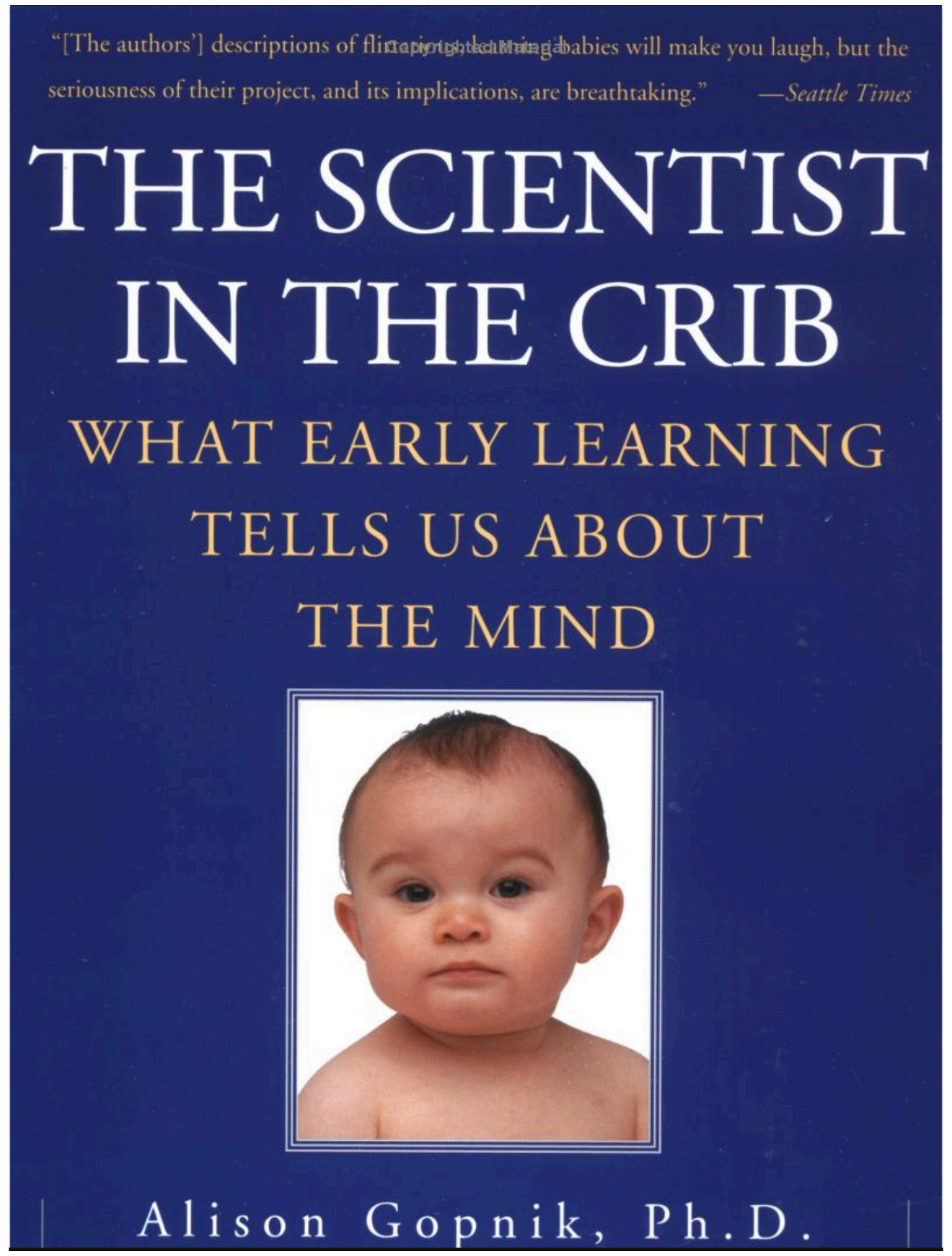
    - Every new task requires new human annotation

    - Specialized tasks require specialized humans (eg the medical domain) — they can be scarce and expensive

- We should at least minimize the human effort

# Why unsupervised learning?

- Babies learn a great deal in an unsupervised way before they develop natural language skills [Gopnik et al,, 2001]
- "What really reaches us from the outside world is a play of colours and shapes, light and sound."
- Babies make sense of the world even before they can communicate through language effectively



# Why unsupervised learning?

- Supervision seems to be more of an accelerator for learning
- Also, how efficient is it to learn from millions of examples?
  - Do children at school learn just from lots of tasks and solutions?
- Interesting properties emerge from general purpose tasks (eg, fine-tuning of LLMs or other SSL-trained models)

# Unsupervised learning

- Representation learning: Self-supervised learning
- Unsupervised segmentation learning
- Unsupervised learning of controllable systems
- Unsupervised learning of 3D shapes

neural network

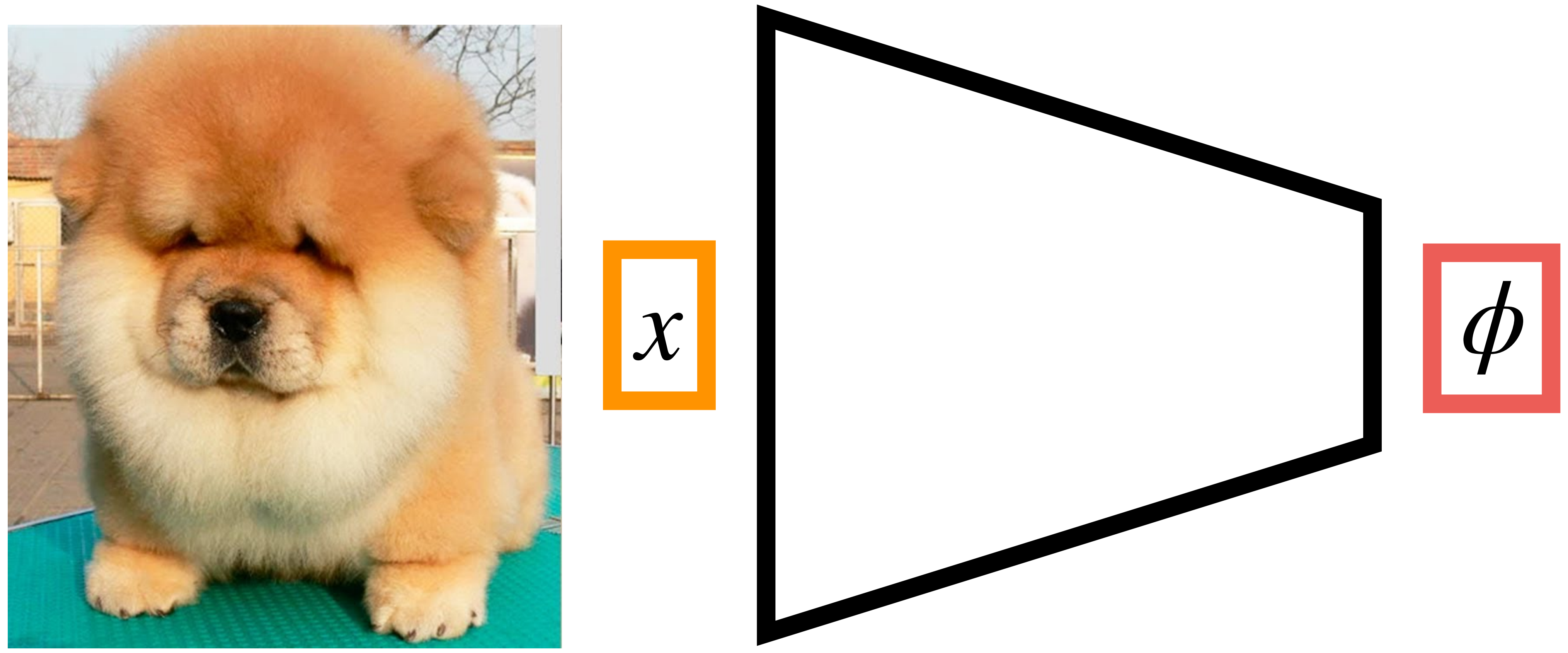
# Representation Learning

data

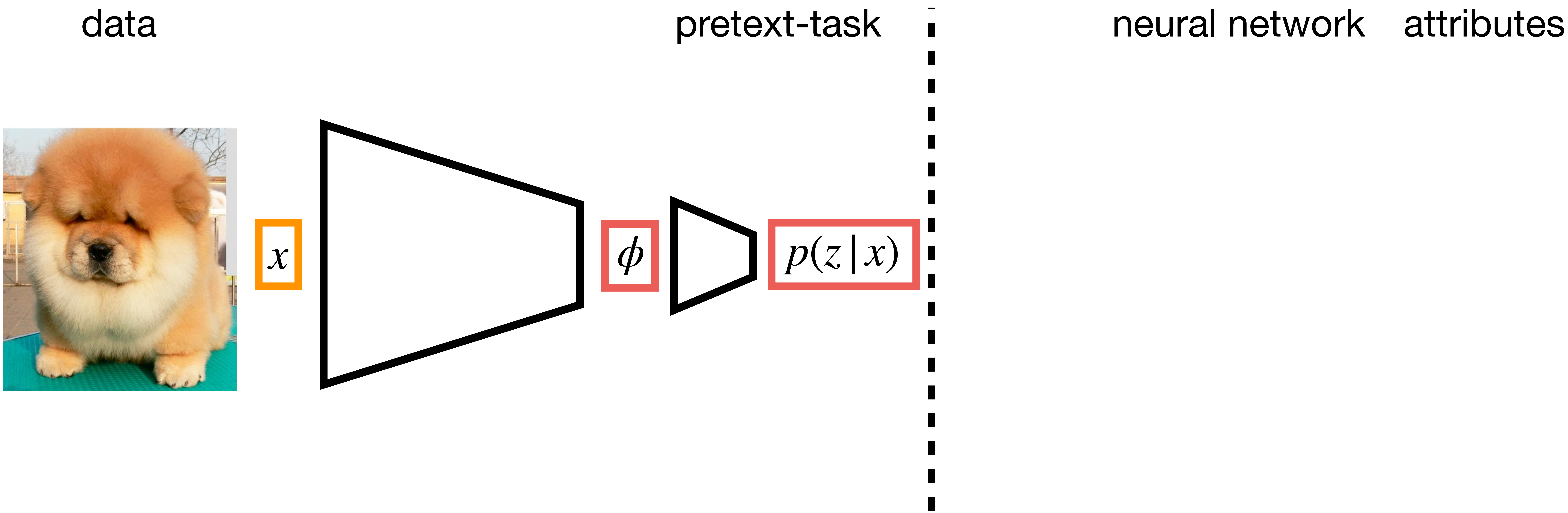
pretext-task

neural network

attributes



# Representation Learning



neural network

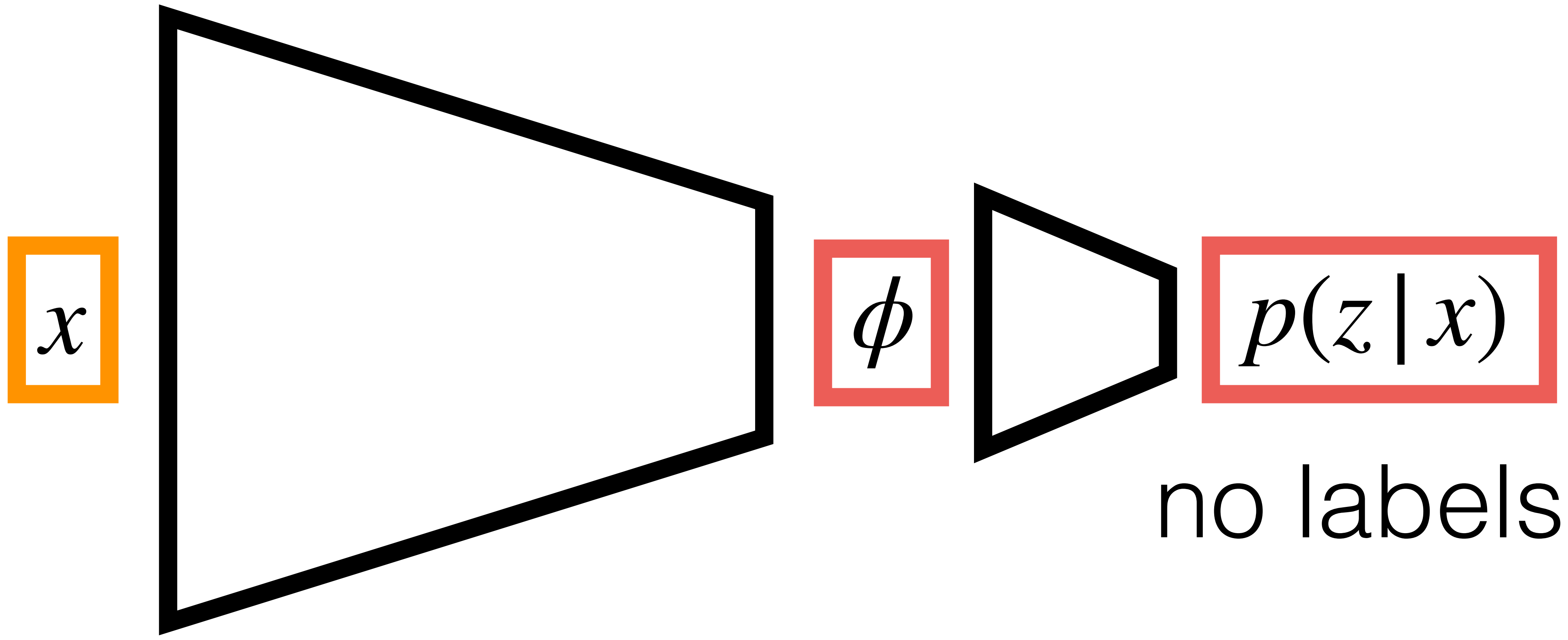
# Representation Learning

data

pretext-task

neural network

attributes



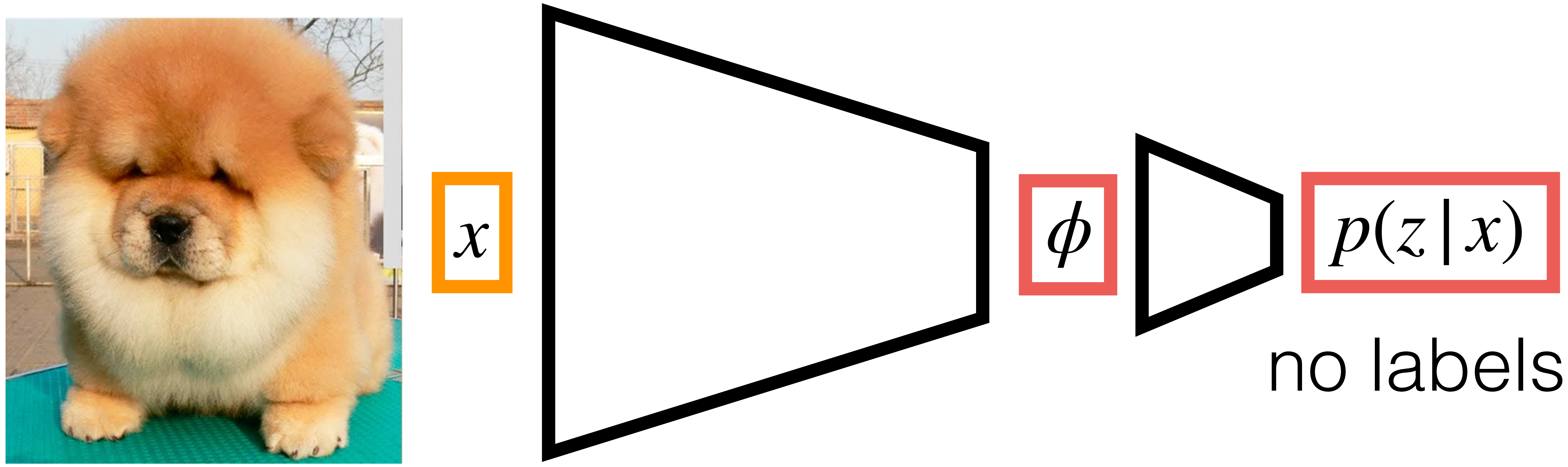
neural network

# Representation Learning

data

pretext-task

neural network attributes

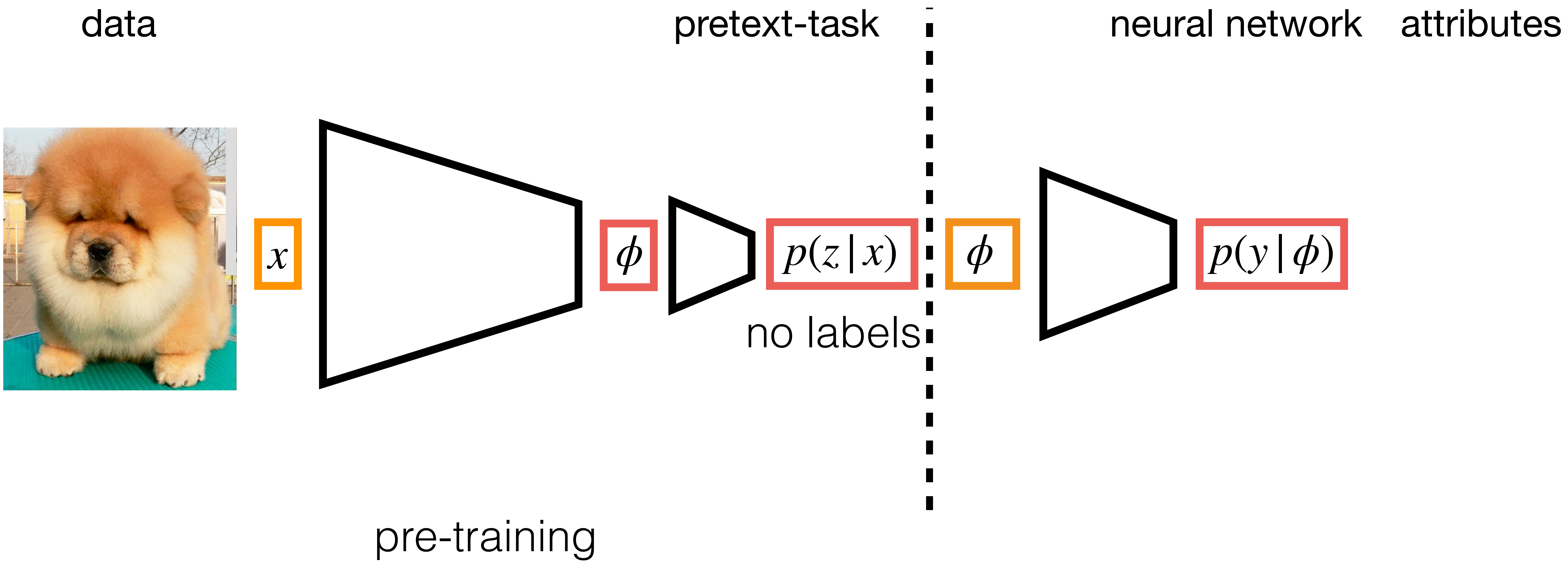



no labels

pre-training

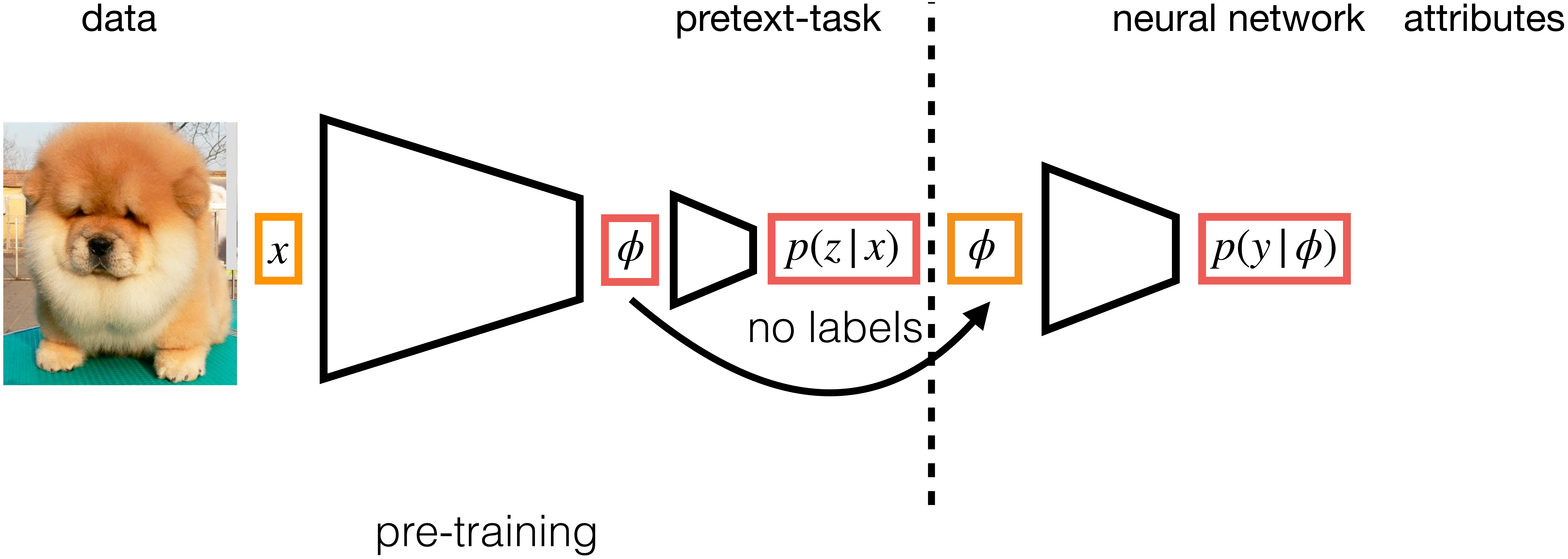
neural network

# Representation Learning



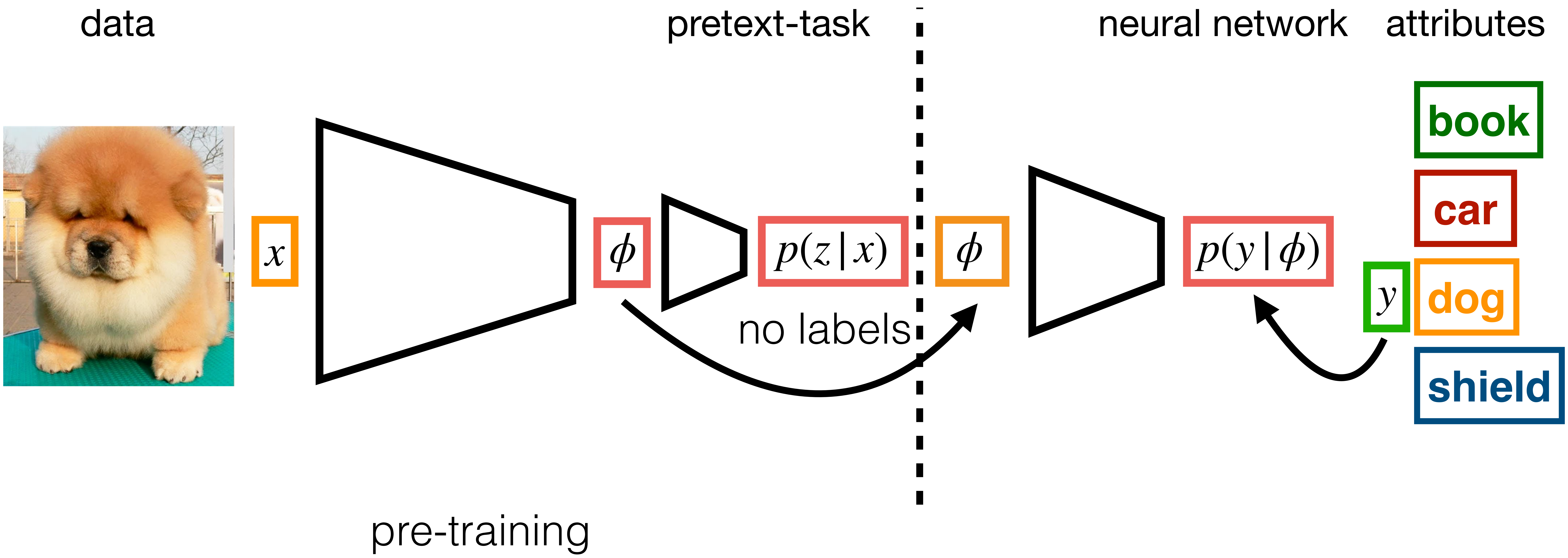
neural network

# Representation Learning



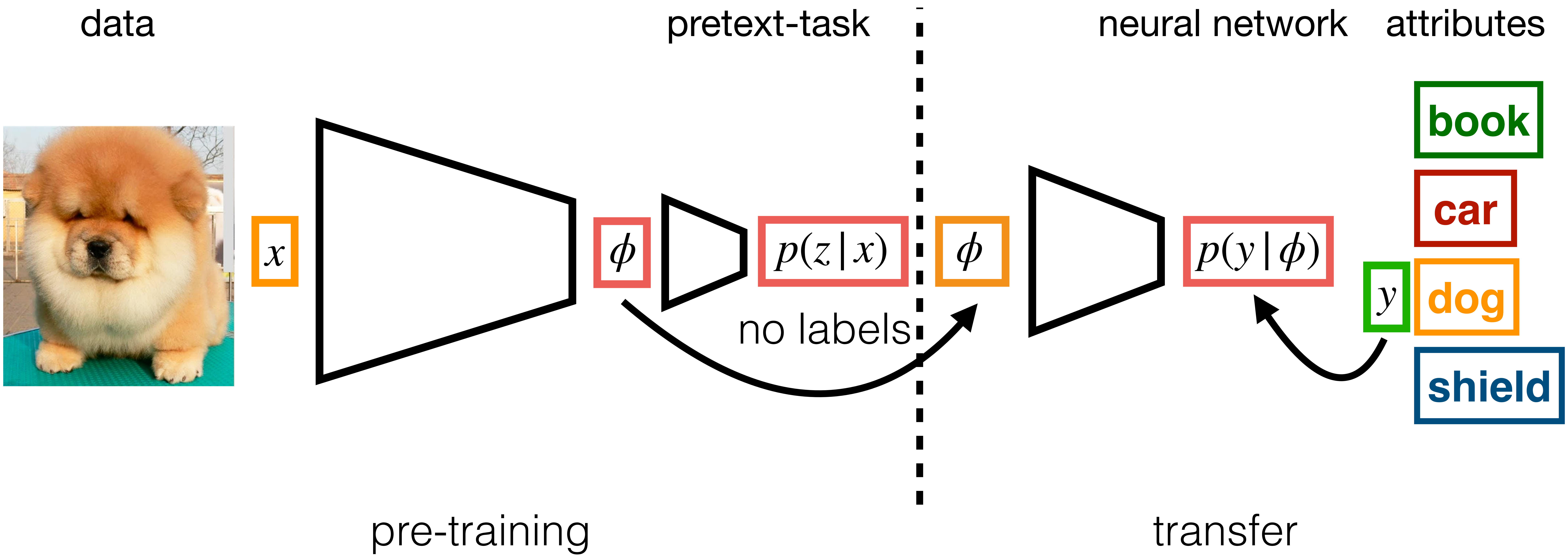
neural network

# Representation Learning



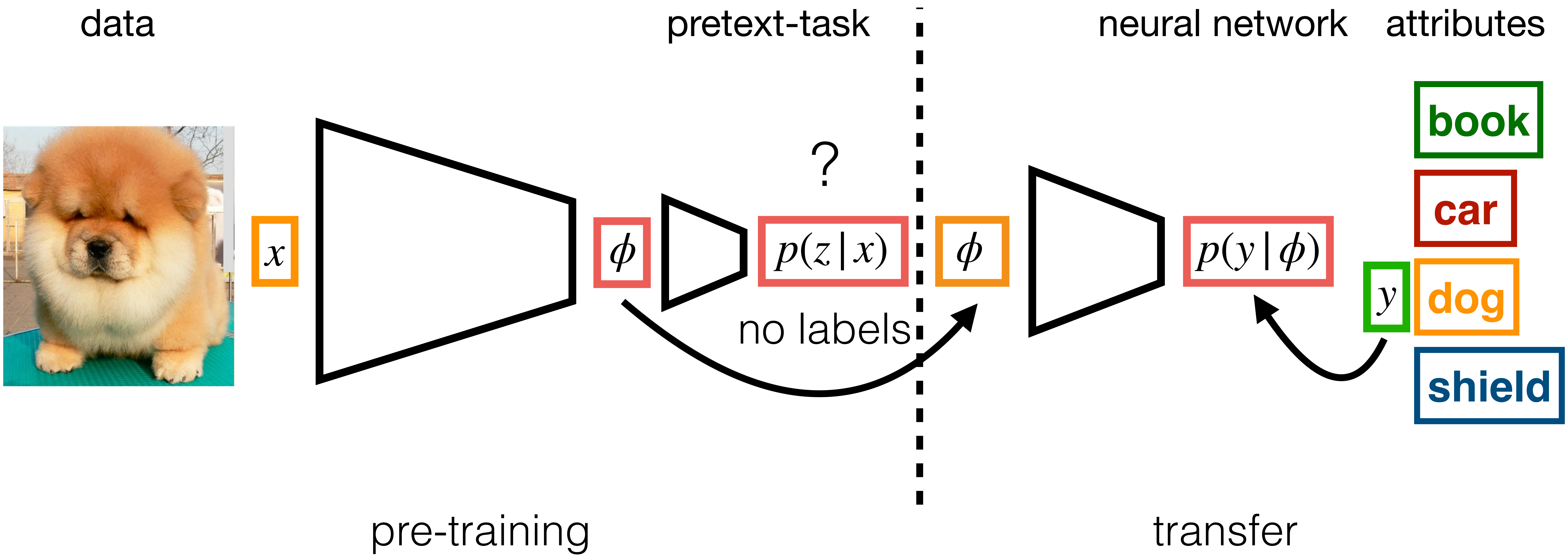
neural network

# Representation Learning



neural network

# Representation Learning



# Self-Supervised Learning

- The objective is to build features $\phi$ so that

$$p(y \mid \phi(x))$$

  is a good approximation of $p(y \mid x)$ for several tasks (and corresponding labels)

# Self-Supervised Learning

- The objective is to build features $\phi$ so that

$$p(y \mid \phi(x))$$

pre-training

is a good approximation of $p(y \mid x)$ for several tasks (and corresponding labels)

# Self-Supervised Learning

- The objective is to build features $\phi$ so that

$$p(y \mid \phi(x))$$

pre-training

is a good approximation of $p(y \mid x)$ for several tasks (and corresponding labels)

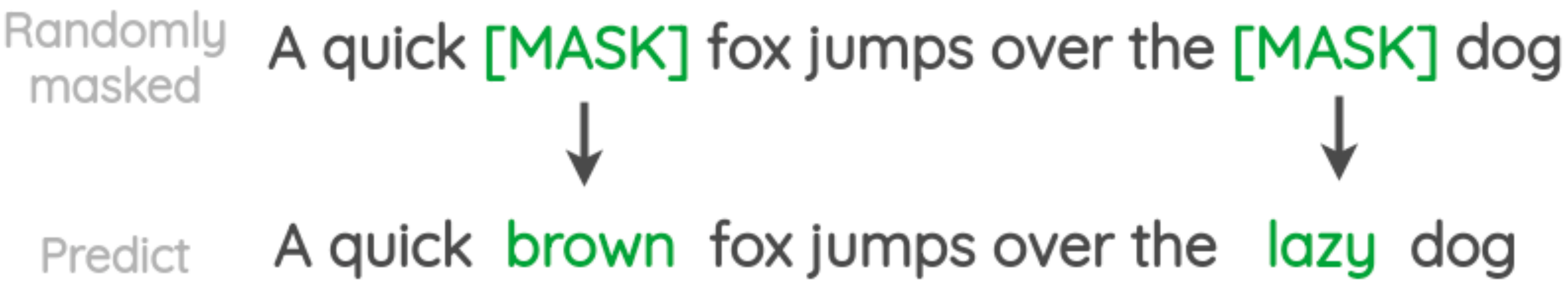
- Ideally, $\phi$ should be such that $p(y \mid \phi)$ can be "simple" (otherwise $\phi = x$ would be a trivial solution), e.g., a shallow neural network

# SSL in NLP

- Continuous Bag of Words


- Skip-gram

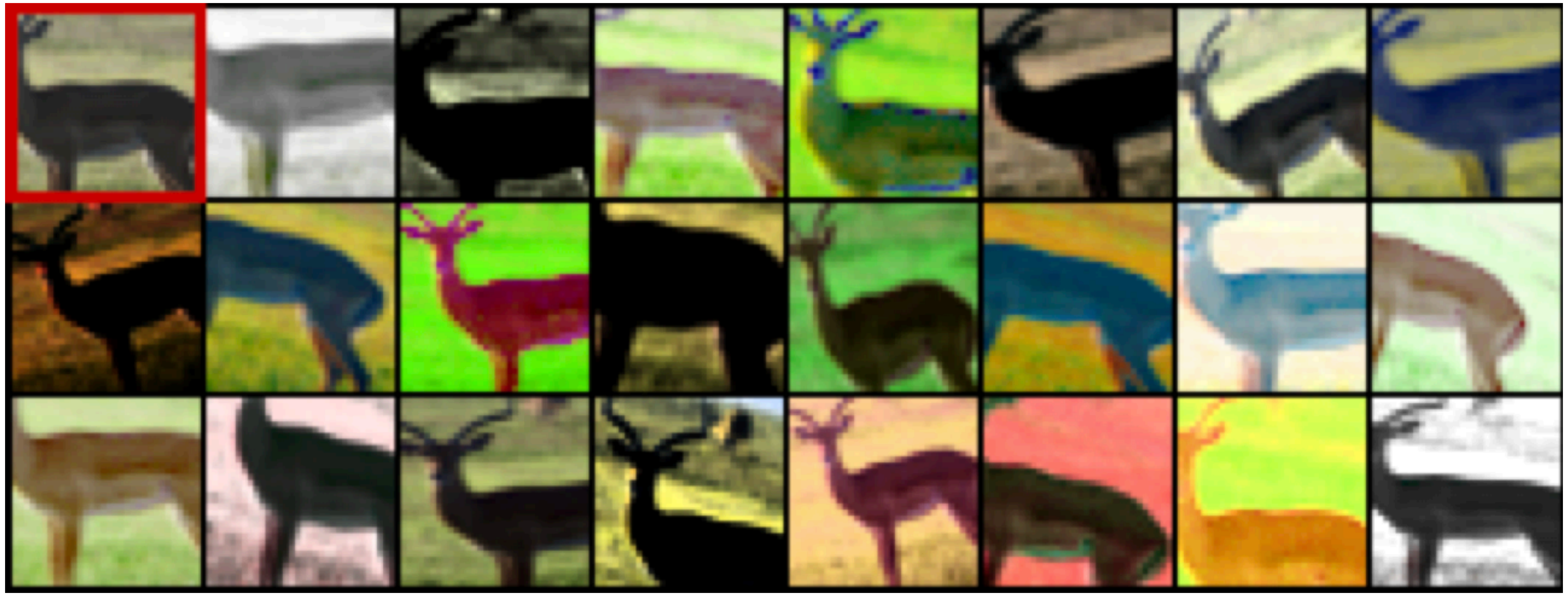

- BERT




Illustrations from https://amitness.com/2020/05/self-supervised-learning-nlp/

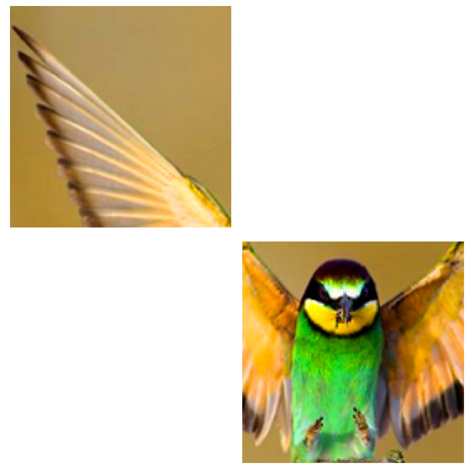
# The first known SSL in vision

- Exemplar-CNN proposed to build a category for each single image and to map all data augmentations of that image to this category

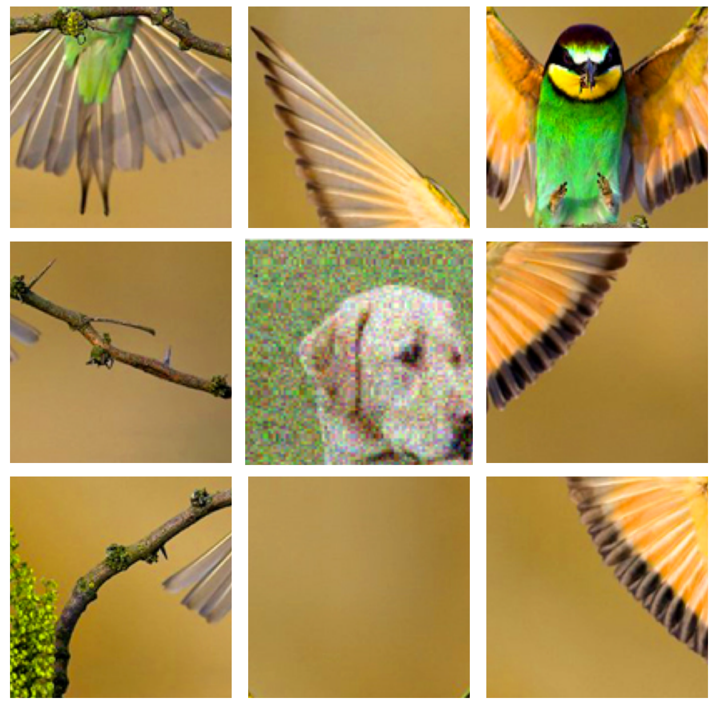


Dosovitski et al, Discriminative Unsupervised Feature Learning with Convolutional Neural Networks, NIPS 14

# Spatial configuration of parts



- Predict the relative position of object parts and identify outliers



- Features of different object parts must be distinguishable from each other
- but also more similar to each other than to outliers 

*Doersch et al 2015, Noroozi and Favaro 2016, Mundhenk et al. 2018, Noroozi et al 2018

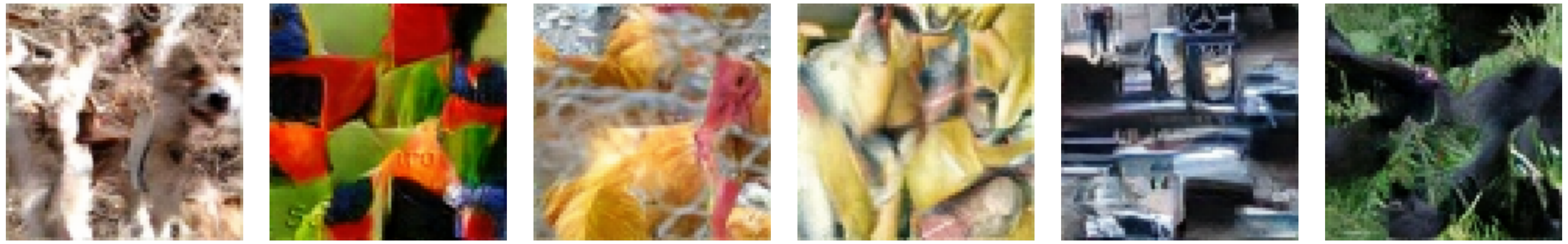
# Global vs local statistics

- Original data

# Global vs local statistics

- Original data

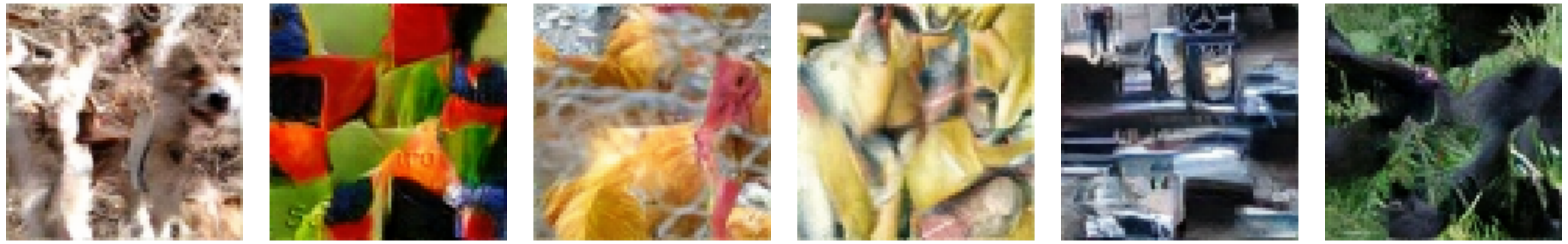
- Images where the local statistics are the same, but the global ones are not



# Global vs local statistics

- Original data

- Images where the local statistics are the same, but the global ones are not

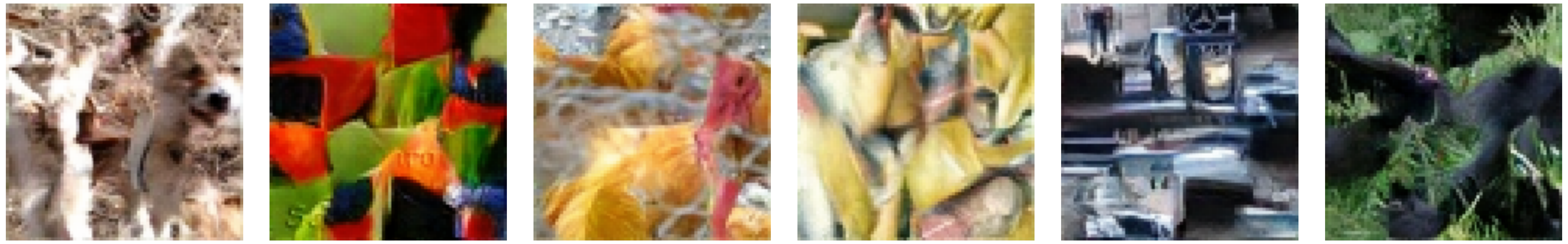


- Supervised learning features do not distinguish well between the two sets

# Global vs local statistics

- Original data

- Images where the local statistics are the same, but the global ones are not



- Supervised learning features do not distinguish well between the two sets

- Mid-range texture* classification is sufficient to solve the supervised task

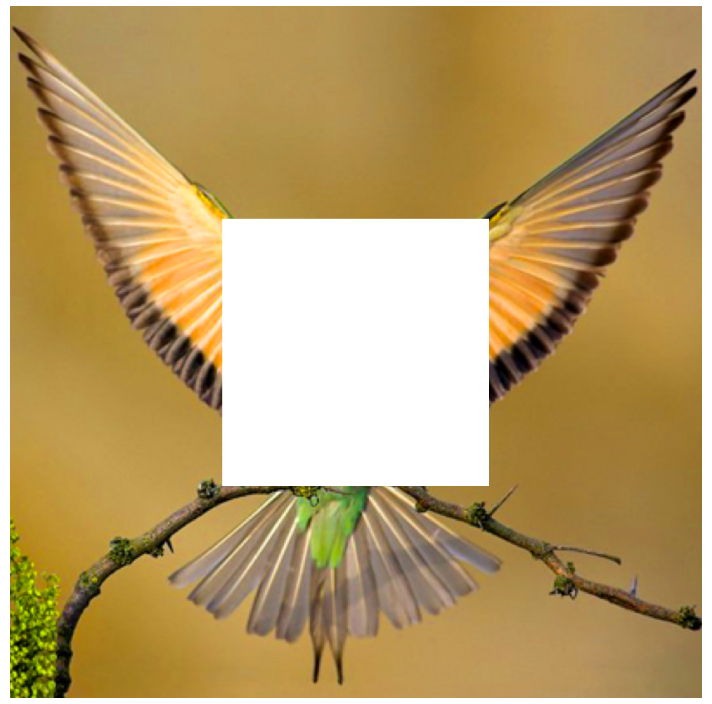
*See Jenni et al, Steering Self-Supervised Feature Learning Beyond Local Pixel Statistics, 2020 and
Geirhos et al, Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness, 2018

# Learning to discriminate global statistics

- Train a network to modify only the global statistics (e.g., missing face, disconnected limbs)
- Features of real objects should be distinguishable from features of unrealistic ones
- The feature representation should allow to discriminate global statistics (ie, shapes)

*S. Jenni and P. Favaro, Self-Supervised Feature Learning by Learning to Spot Artifacts, 2018
S. Jenni et al, Steering Self-Supervised Feature Learning Beyond Local Pixel Statistics, 2020

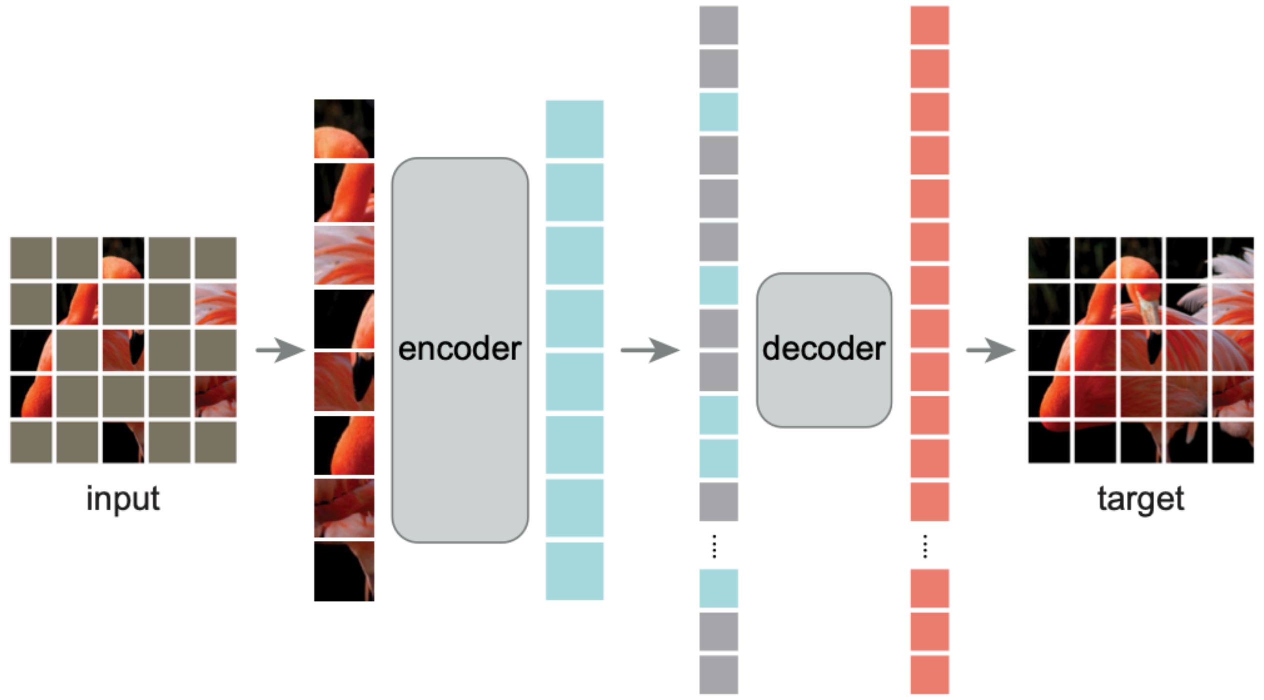
# Reconstruction-based



- Features should allow the reconstruction of a data sample from its context or other transformed versions of that sample

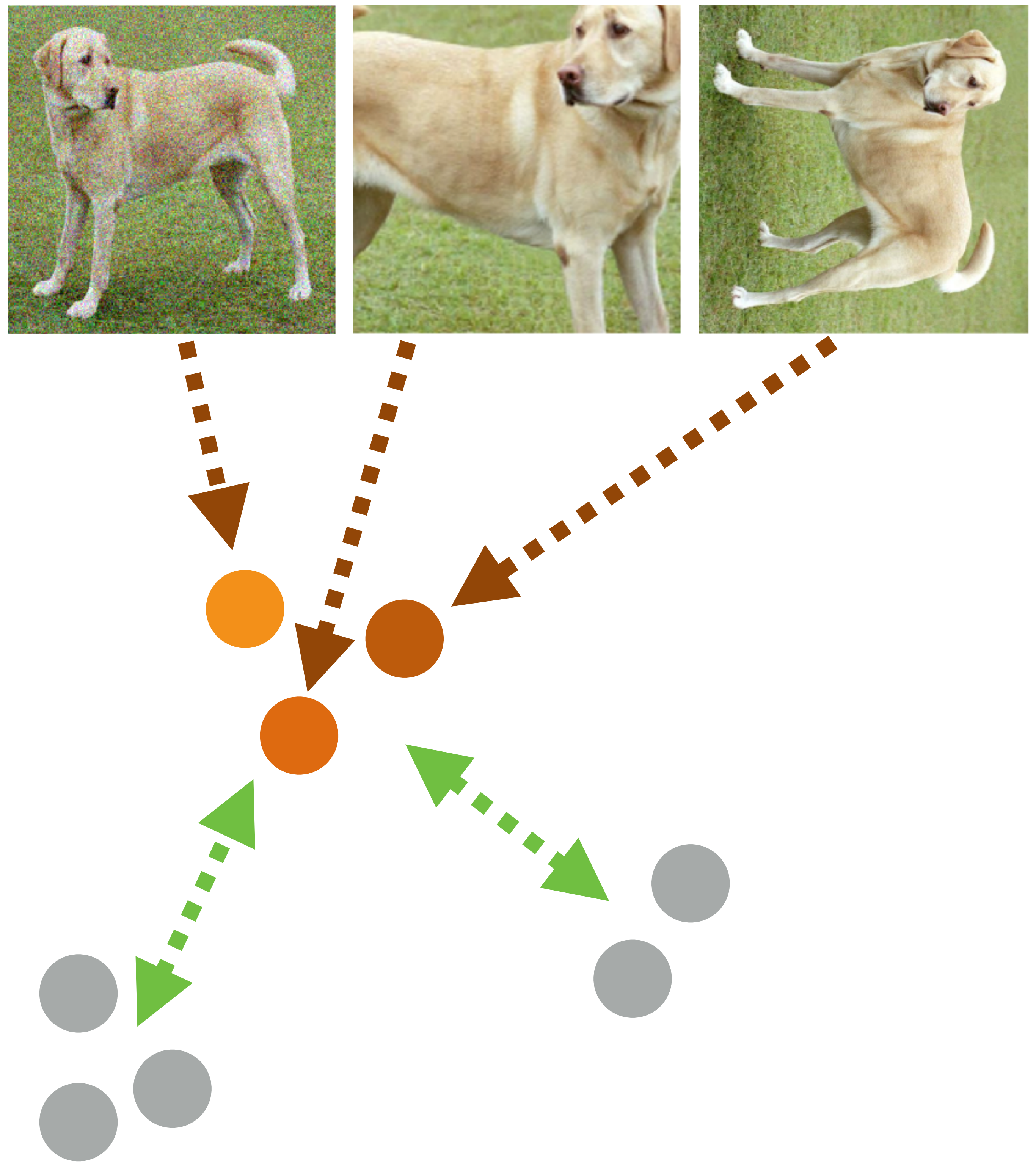
- Can be related to denoising AEs → Features are encouraged to be invariant to the added "noise"

- Images which differ by the transformation used in the pretext-task are mapped to similar features




*K. He et al, Masked Autoencoders Are Scalable Learners, CVPR 2022
D. Pathak et al, Context encoders: Feature learning by inpainting, 2016
G. Larsson et al, Learning representations for automatic colorization, 2016

# Contrastive Learning



- Pretext-task explicitly defines which images are similar based on data augmentation

- Network and optimization design provide non trivial performance boost (e.g., large minibatches, contrastive learning, additional network "head")

*Exemplar-CNN, SimCLR, MoCo, Deep Clustering, SeLa, SwAV
Noroozi et al, Representation Learning by Learning to Count, 2017
Wang and Gupta, Unsupervised Learning of Visual Representations Using Videos, 2015

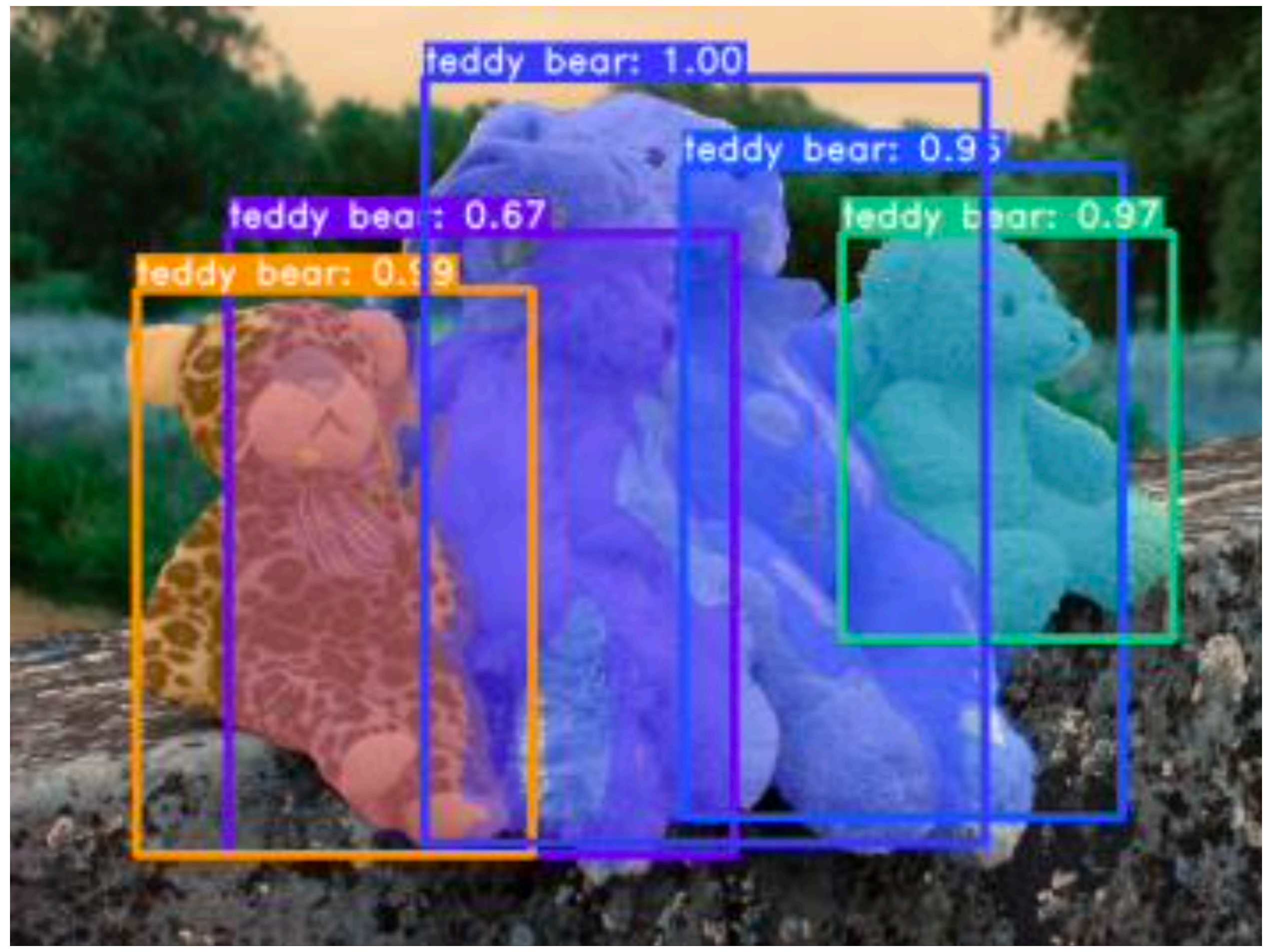
# Away from data augmentation

SSL by distilling generative models


Li et al, DreamTeacher: Pretraining Image Backbones with Deep Generative Models, ICCV 2023

# Object segmentation

- Object segmentation allows to identify pixels that belong to a single object




- In computer vision
  - More accurate than bounding boxes or single points
  - Better understanding of image content (shape information, removal of clutter, etc)

- In image processing
  - Allows advanced editing (background/object replacement, composition)

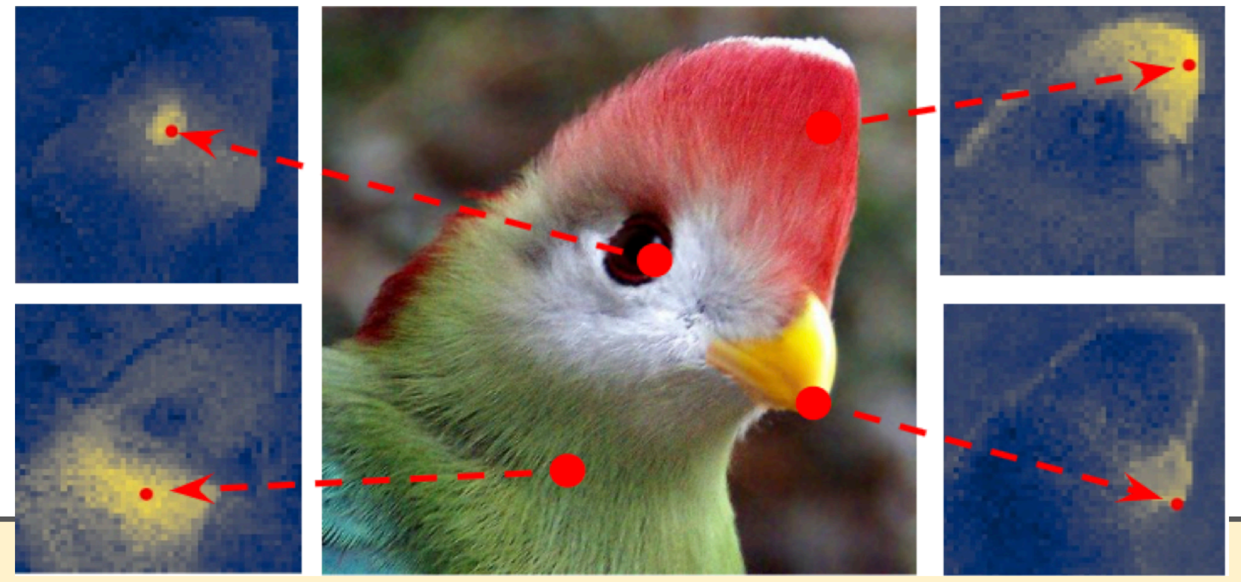
*Lan et al "DISCOBOX: Weakly Supervised Instance Segmentation and Semantic Correspondence from Box Supervision", ICCV 2021
†https://www.colorexpertsbd.com/blog/what-is-image-masking/

# Object segmentation labeling

- Manual labeling of segmentation masks in videos is unfeasible
- Prompted several attempts to learn object segmentation without labels
  - W-net, arxiv 2017
  - MONET, arxiv 2019
  - DeepUSPS, NeurIPS 2019
  - Autoregressive USL, ECCV 2020
  - LOST, BMVC 2021
  - FreeSOLO, CVPR 2022
  - TokenCut, CVPR 2022
  - DeepSpectral, CVPR 2022
  - Seong et al, CVPR 2023

# Object segmentation labeling

- Manual labeling of segmentation masks in videos is unfeasible
- Prompted several attempts to learn object segmentation without labels
  - W-net, arxiv 2017
  - MONET, arxiv 2019
  - DeepUSPS, NeurIPS 2019
  - Autoregressive USL, ECCV 2020
  - LOST, BMVC 2021
  - FreeSOLO, CVPR 2022
  - TokenCut, CVPR 2022
  - DeepSpectral, CVPR 2022
  - Seong et al, CVPR 2023



Built on top of pre-trained SSL features (eg, DINO, DenseCL)

# Realism as a segmentation signal

- **Key idea:** Use the segmentation mask to copy, shift and paste an object; then, use a “realism”-based metric to rate the composite image
- If the mask is incorrect, the composite image would have unrealistic artifacts (eg, repetitions or split objects that are typically joined)
- Prior work
  Cut&Paste ECCV 2018, PerturbGAN NeurIPS 2019, Copy-PastingGAN arxiv 2019, SEIGAN arxiv 2018

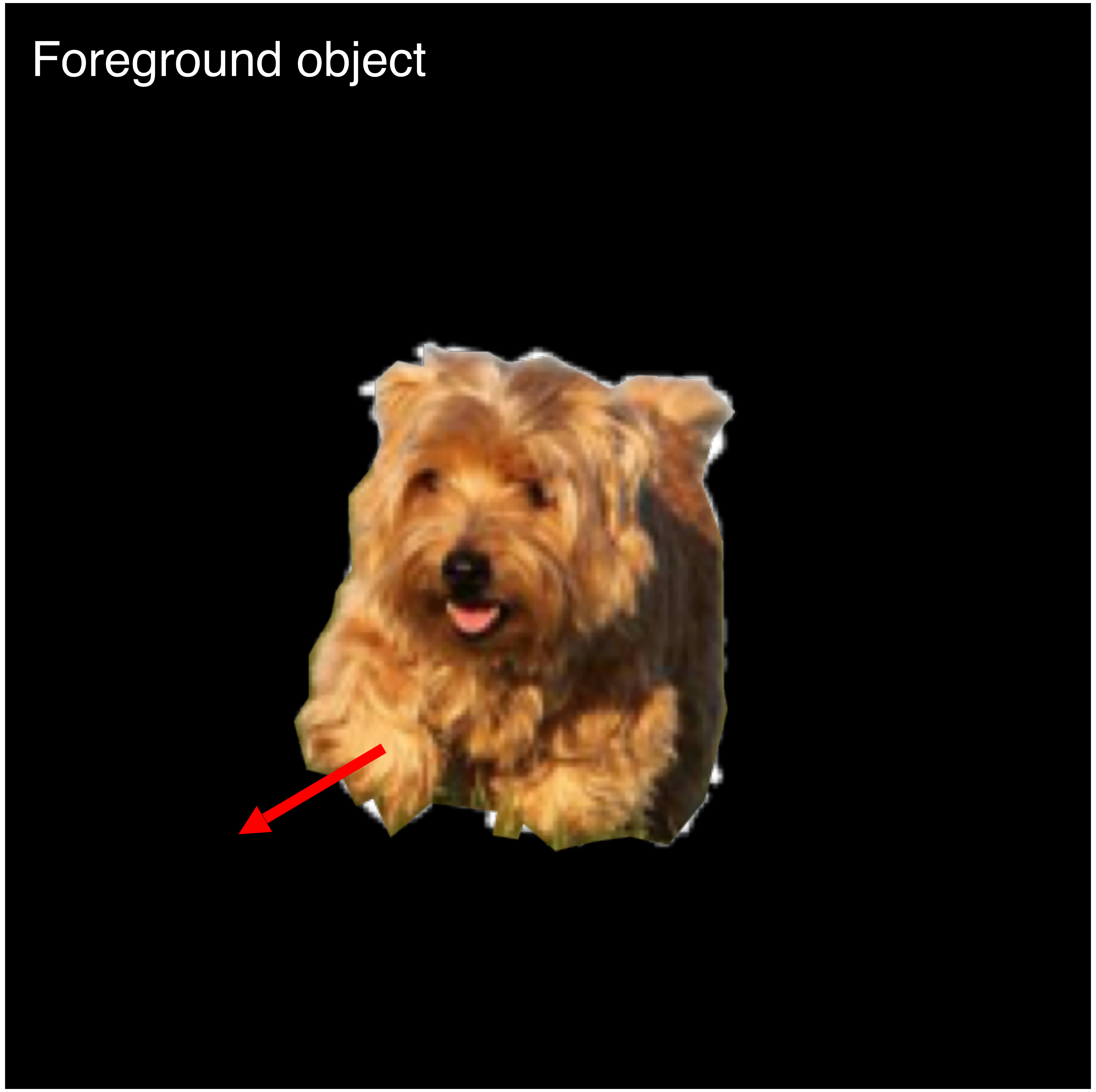
A. Bielski and P. Favaro, MOVE: Unsupervised Movable Object Segmentation and Detection, NeurIPS 2022

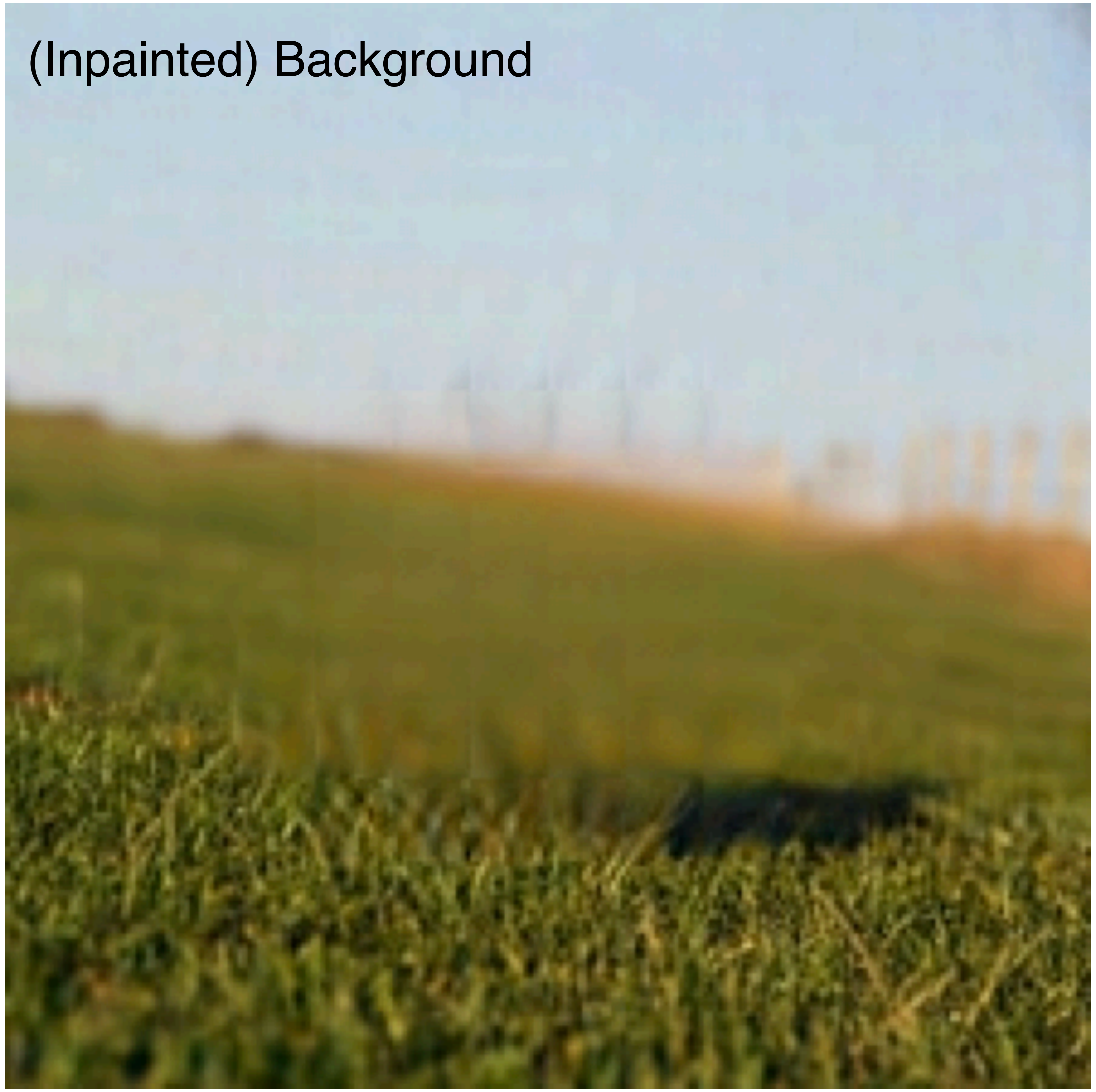
# Learning to MOVE

# Learning to MOVE

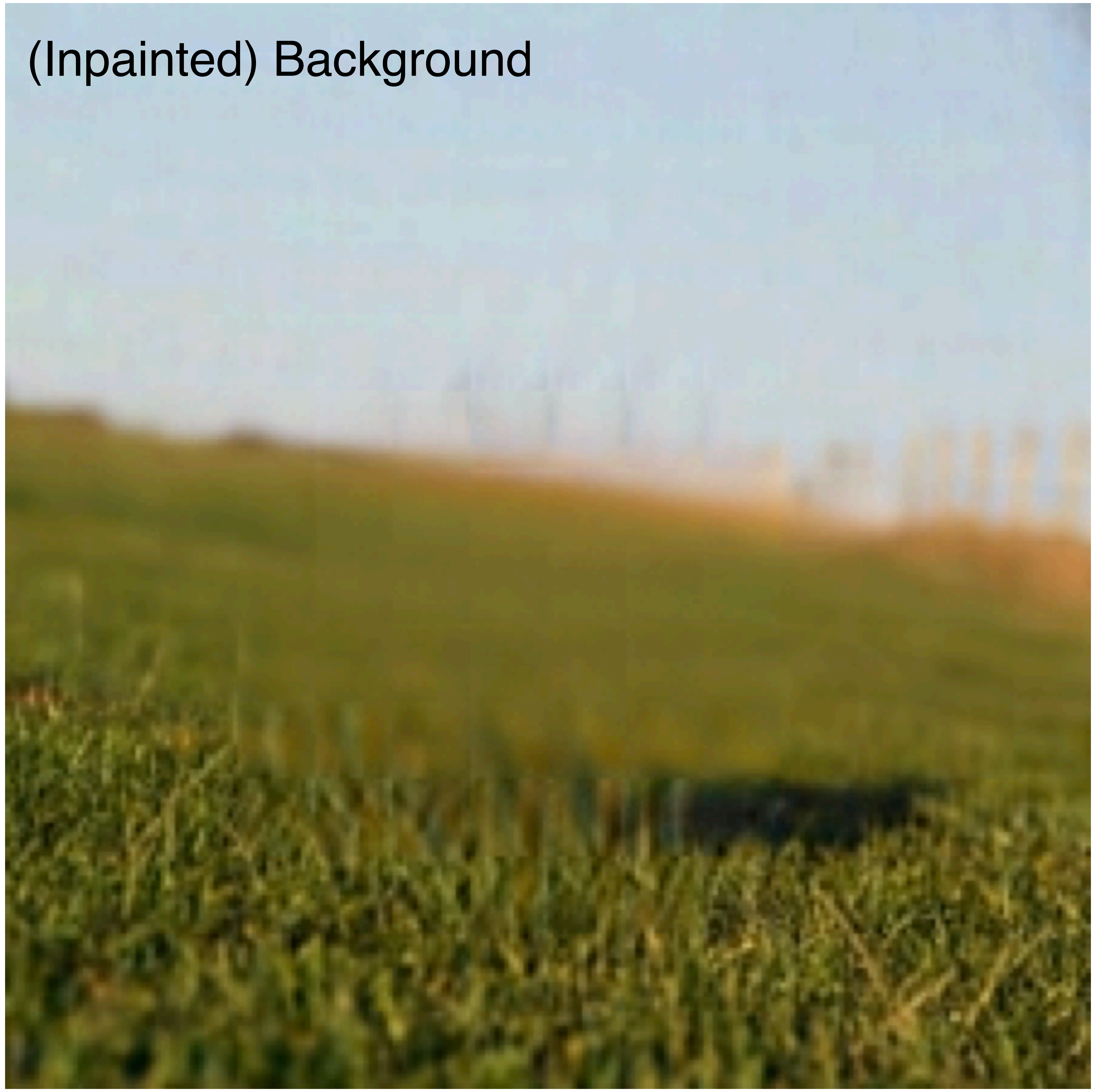
# Learning to MOVE

# Learning to MOVE



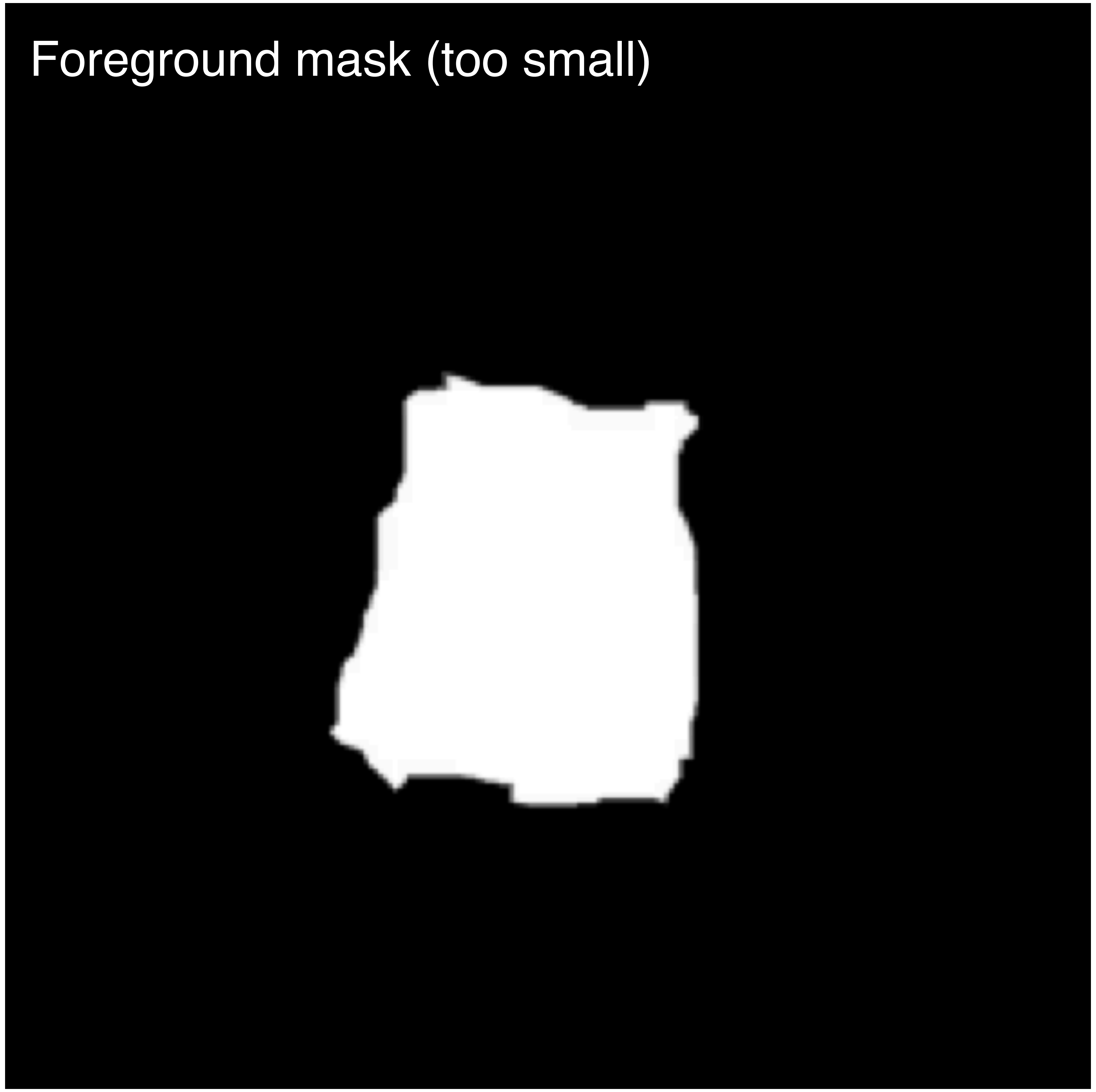



# Moveability is a signal for segmentation

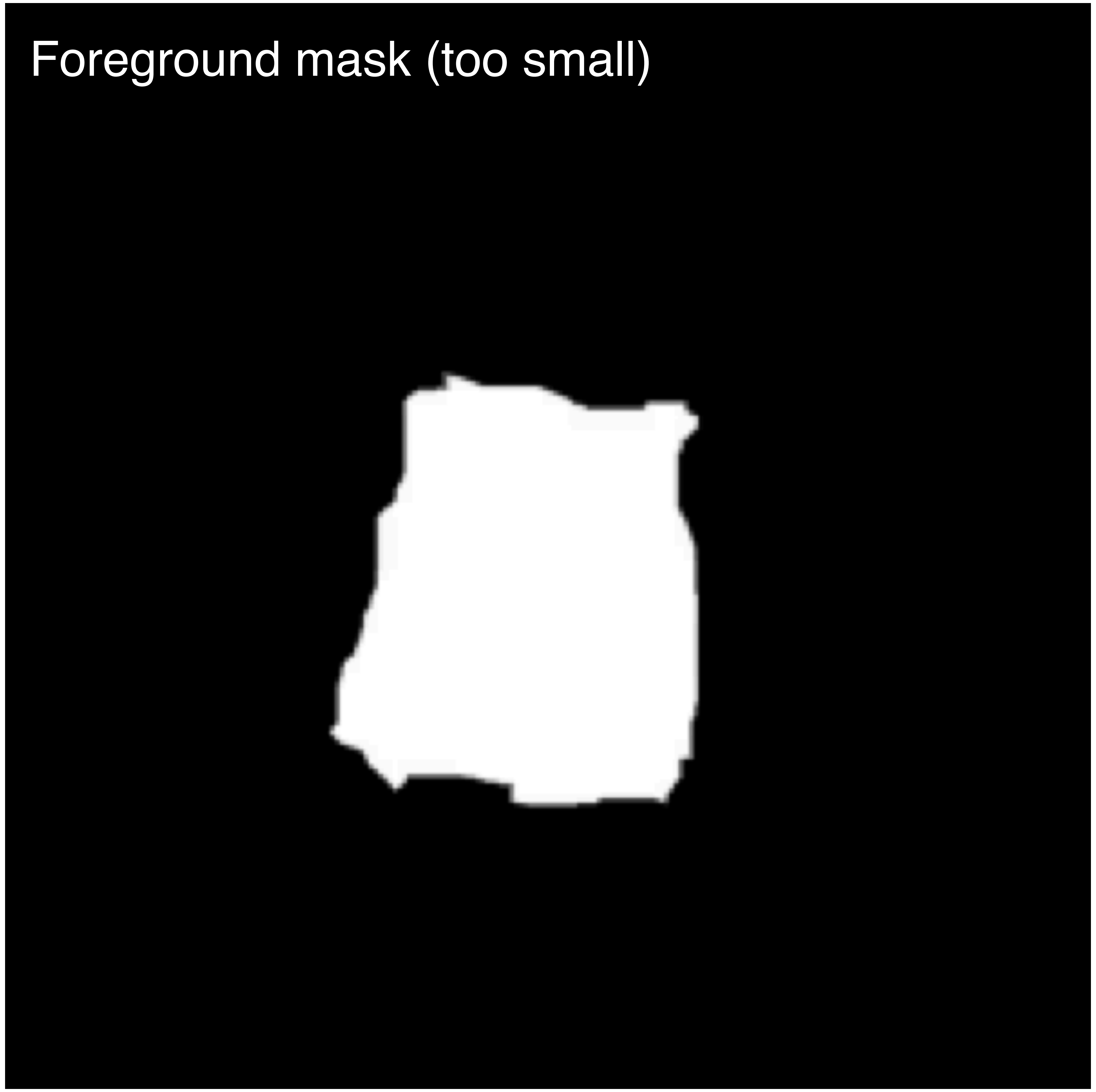



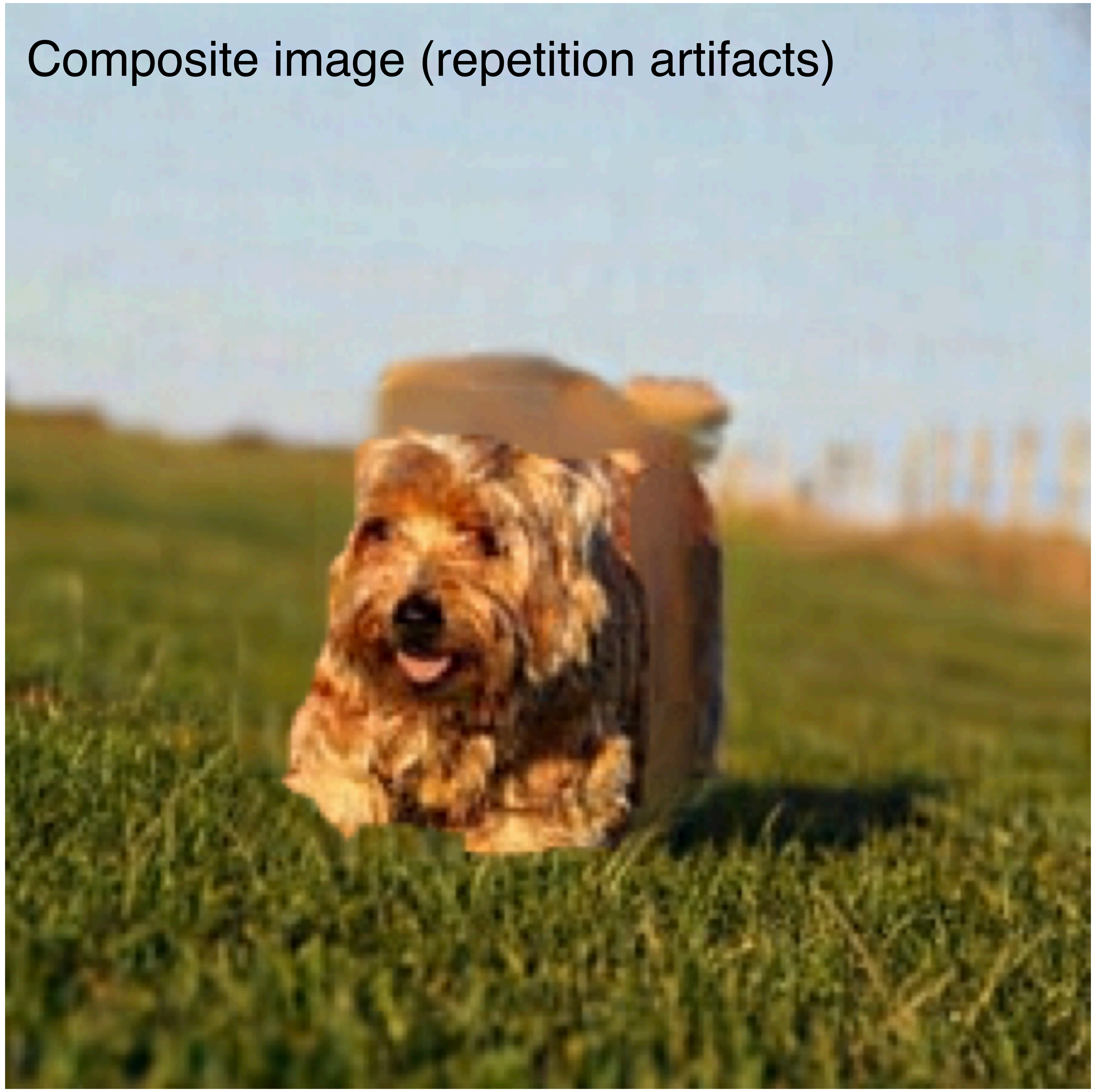
# Moveability is a signal for segmentation


# Moveability is a signal for segmentation

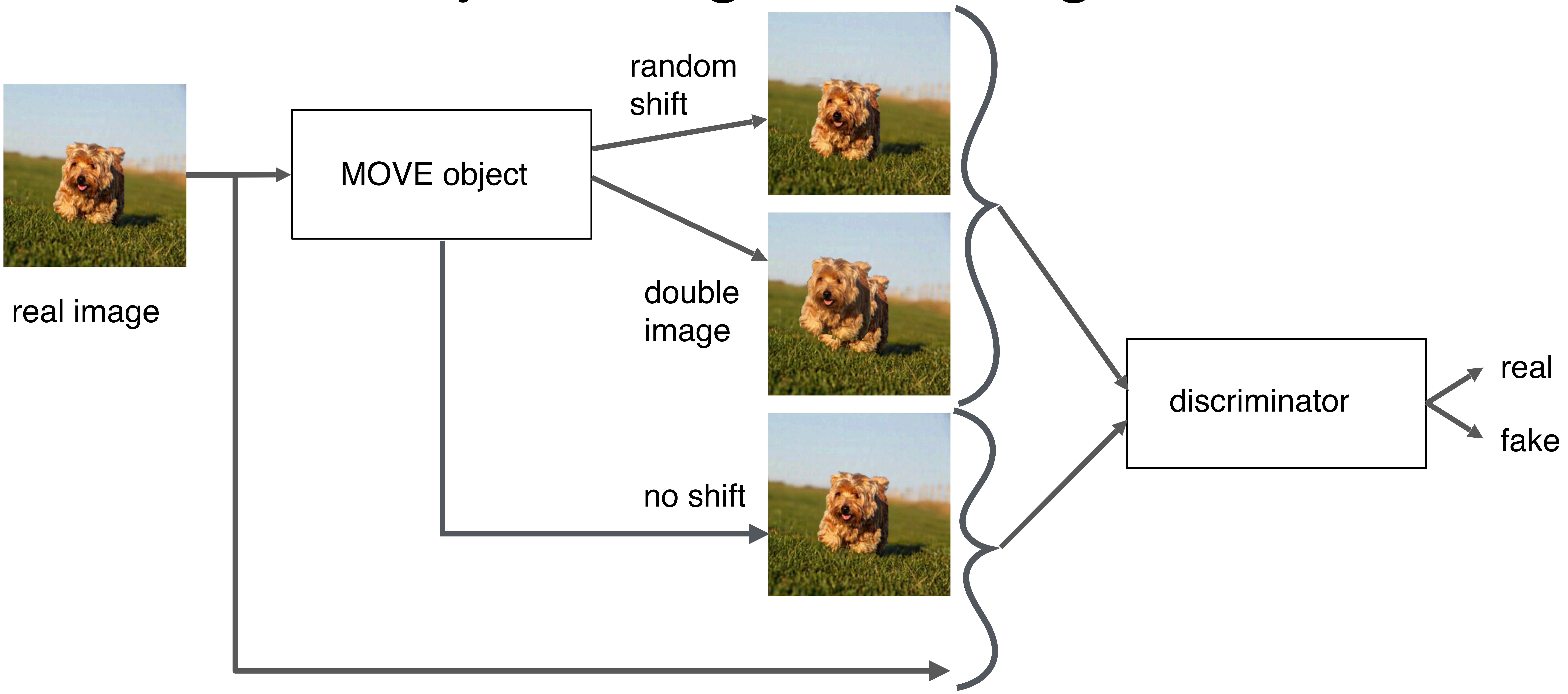



# Moveability is a signal for segmentation



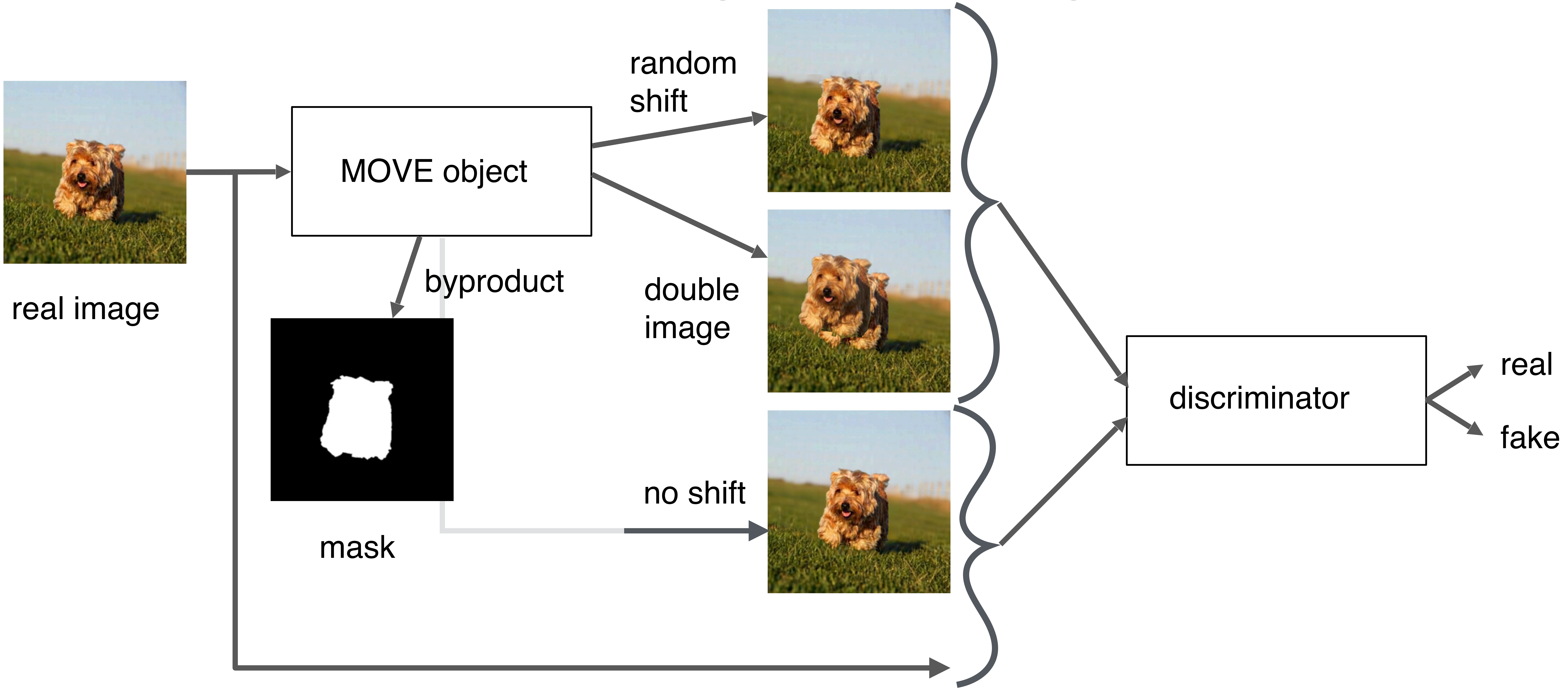



# Moveability is a signal for segmentation



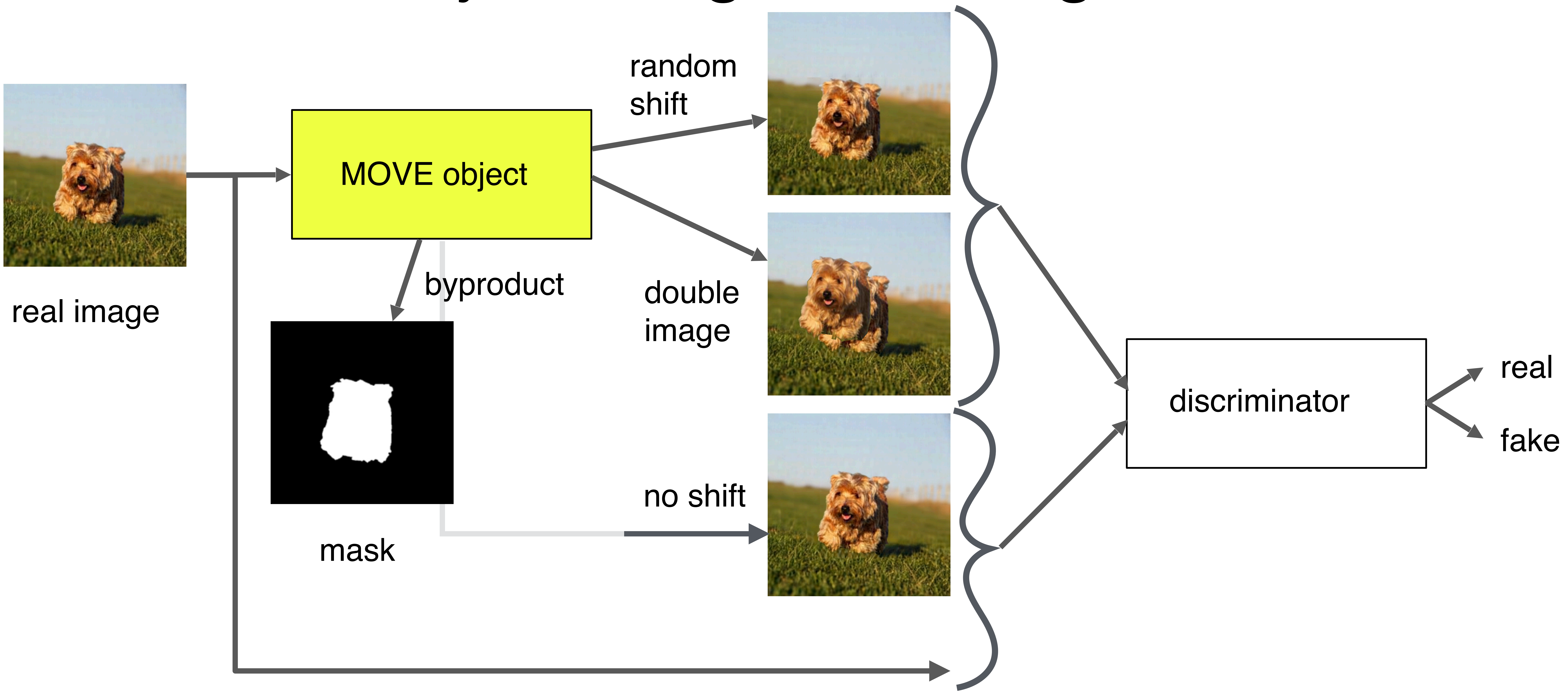

A. Bielski and P. Favaro, MOVE: Unsupervised Movable Object Segmentation and Detection, NeurIPS 2022

# Moveability is a signal for segmentation



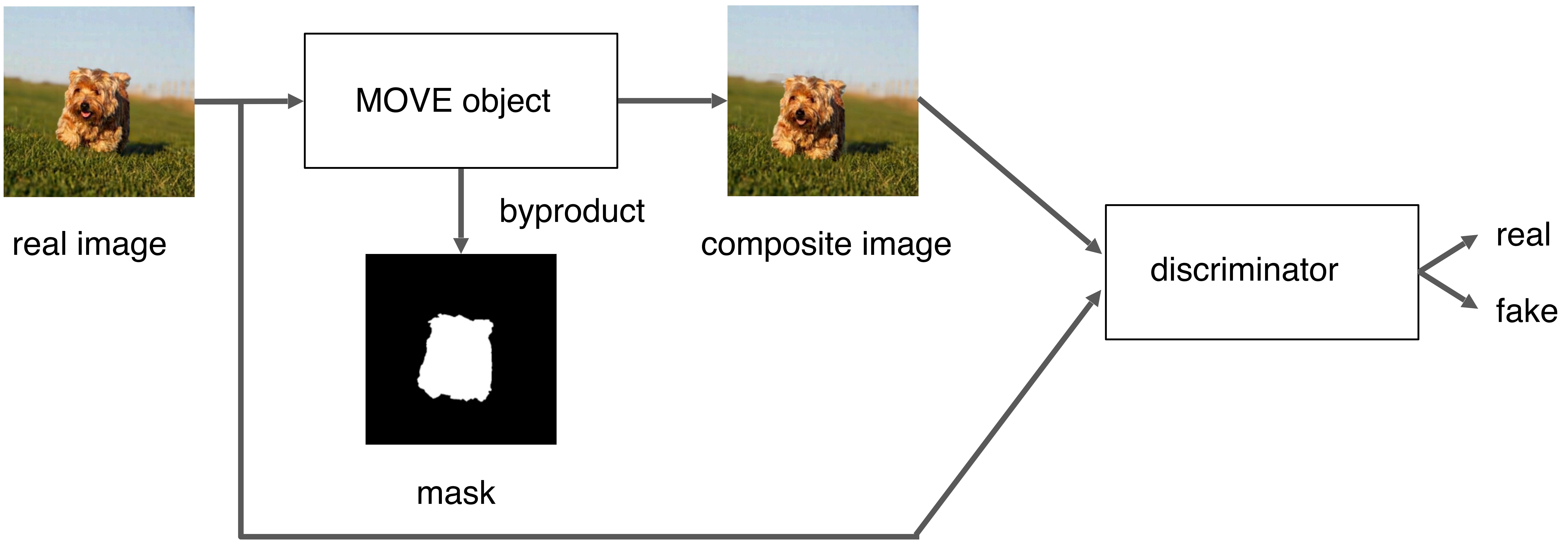

A. Bielski and P. Favaro, MOVE: Unsupervised Movable Object Segmentation and Detection, NeurIPS 2022

# Moveability is a signal for segmentation



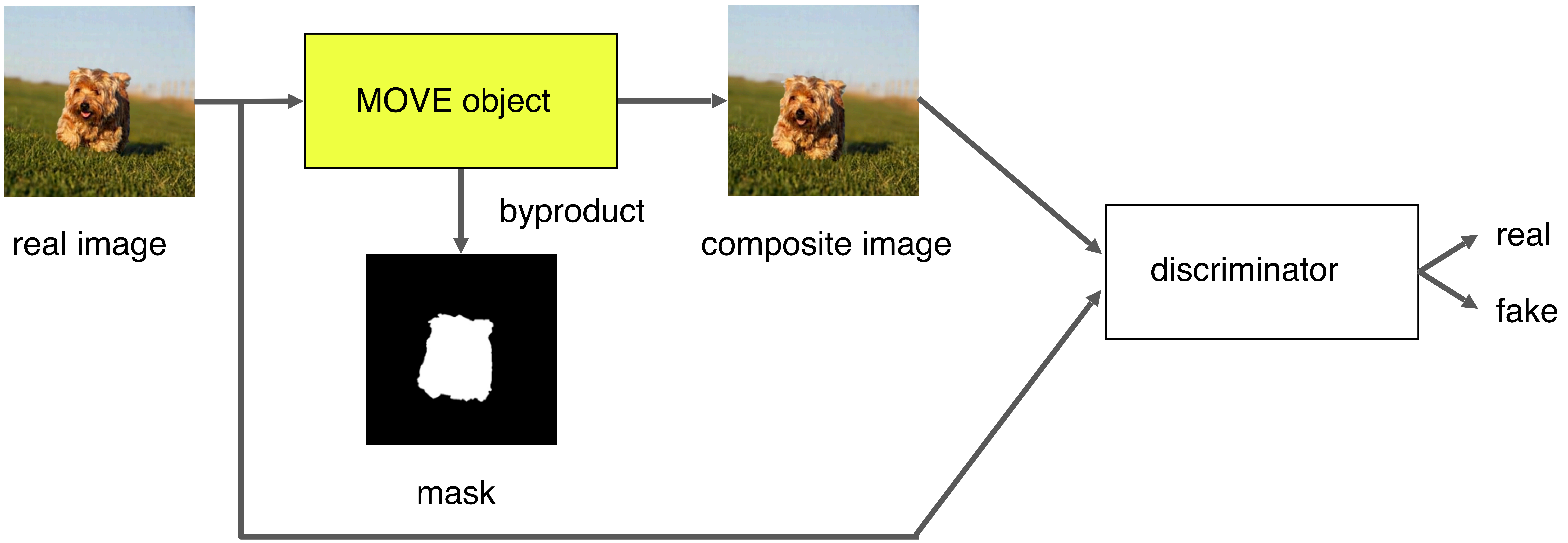

A. Bielski and P. Favaro, MOVE: Unsupervised Movable Object Segmentation and Detection, NeurIPS 2022

# Moveability is a Signal for Segmentation



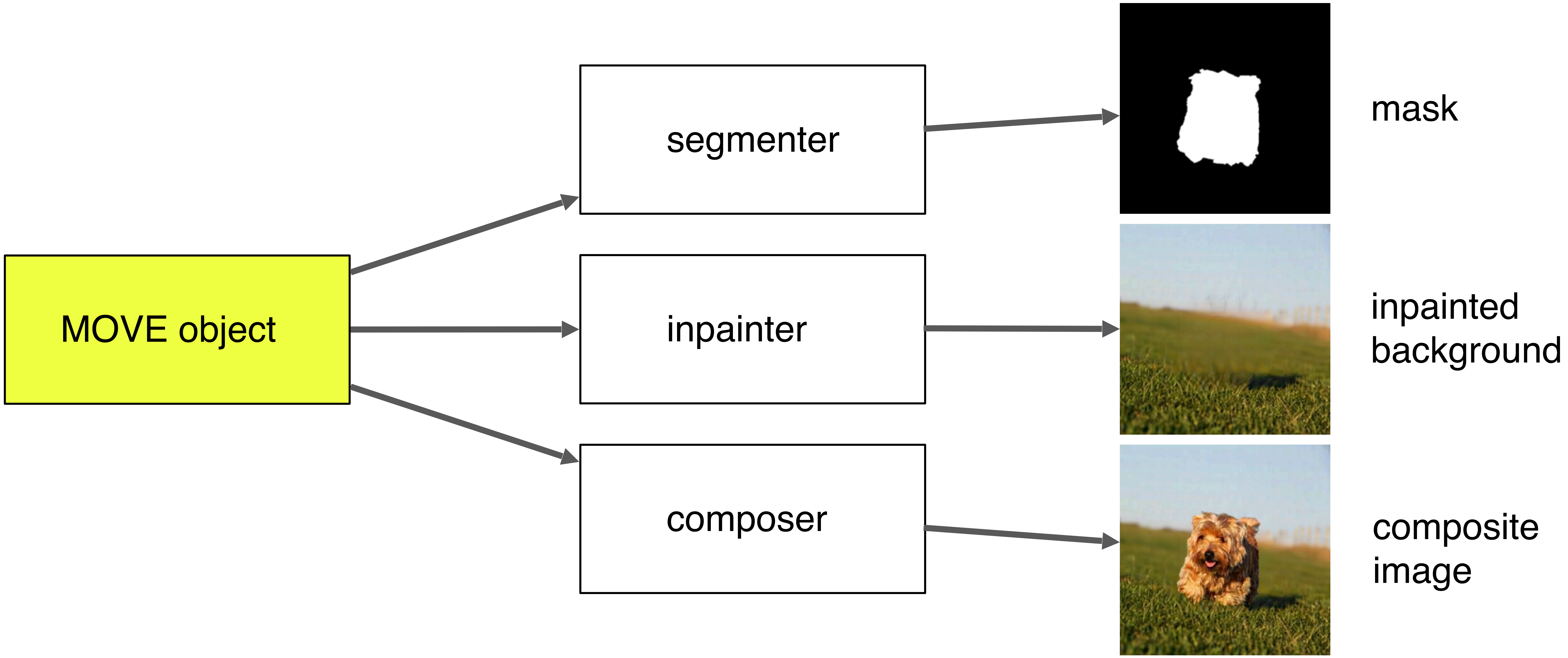

A. Bielski and P. Favaro, MOVE: Unsupervised Movable Object Segmentation and Detection, NeurIPS 2022

# Moveability is a Signal for Segmentation

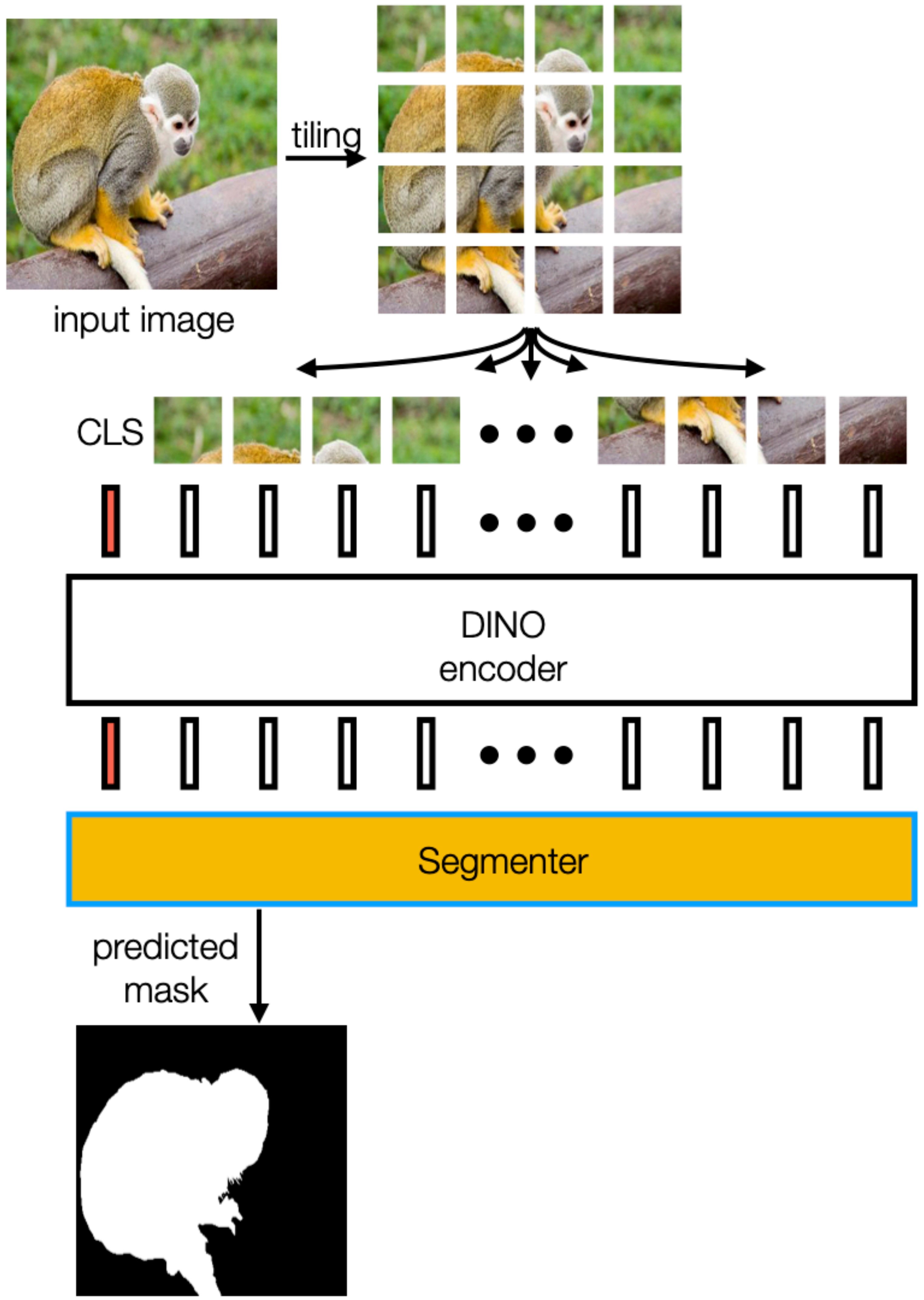



A. Bielski and P. Favaro, MOVE: Unsupervised Movable Object Segmentation and Detection, NeurIPS 2022

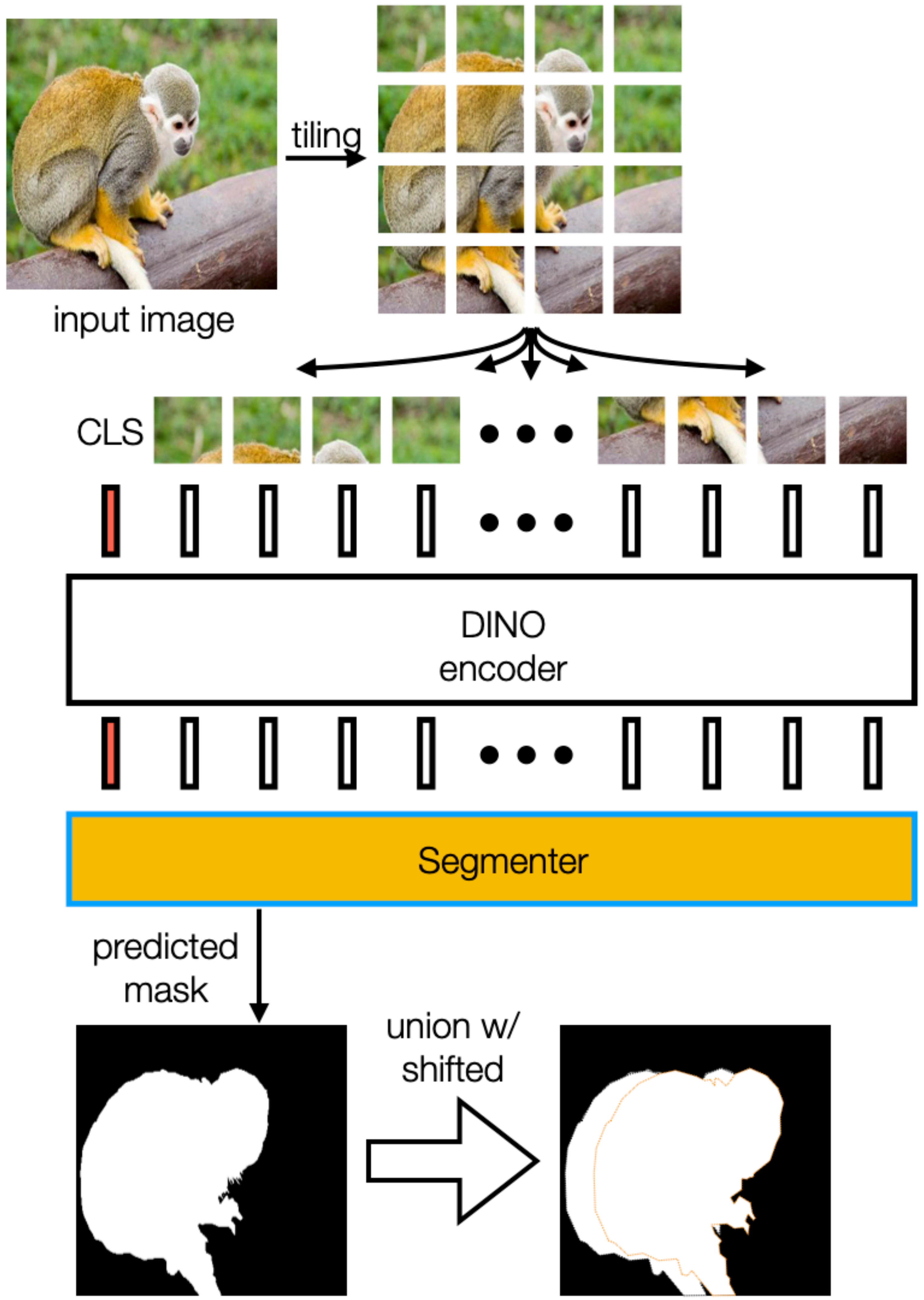
# Moveability is a signal for segmentation



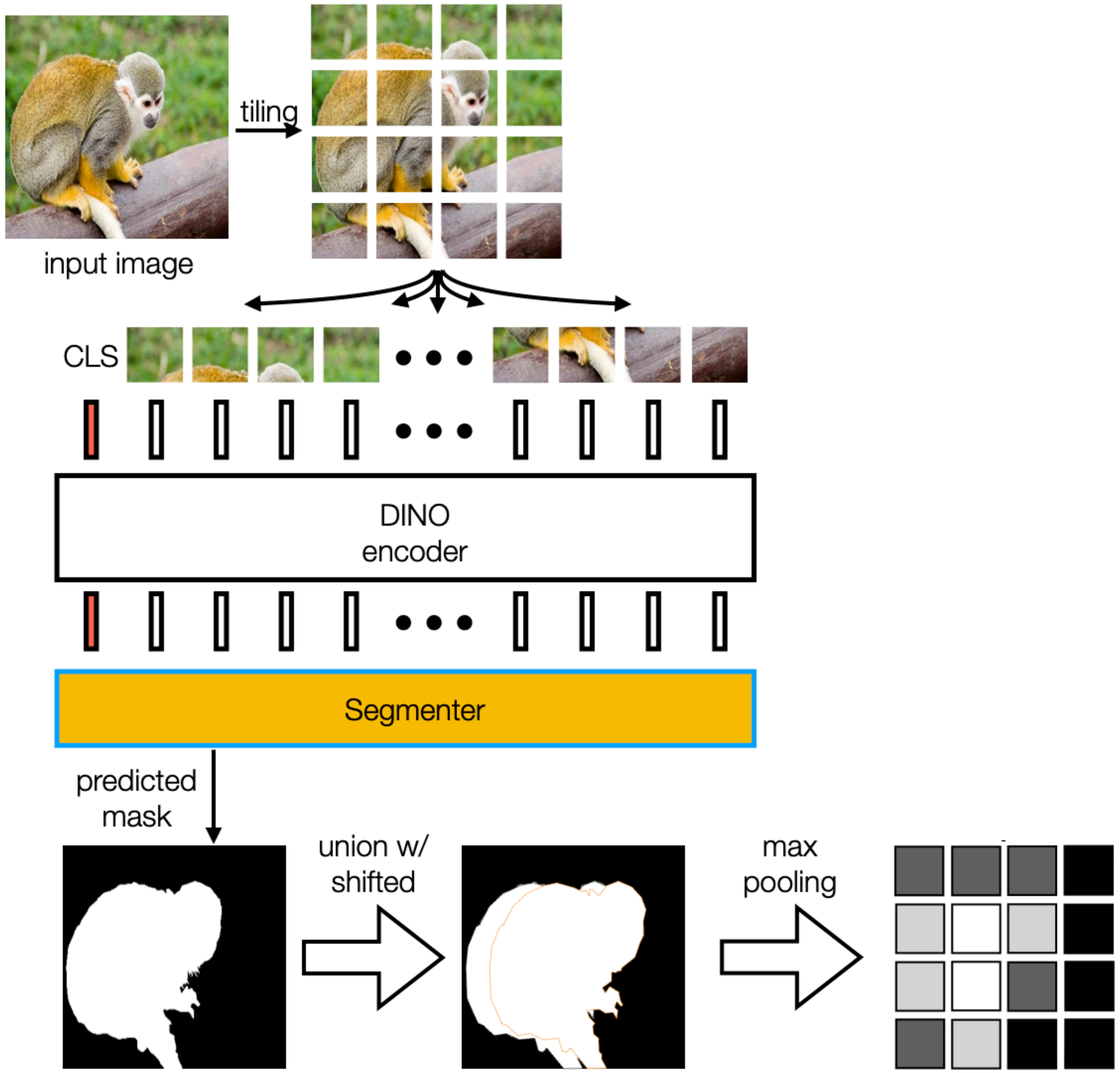
# Segmenting & Inpainting



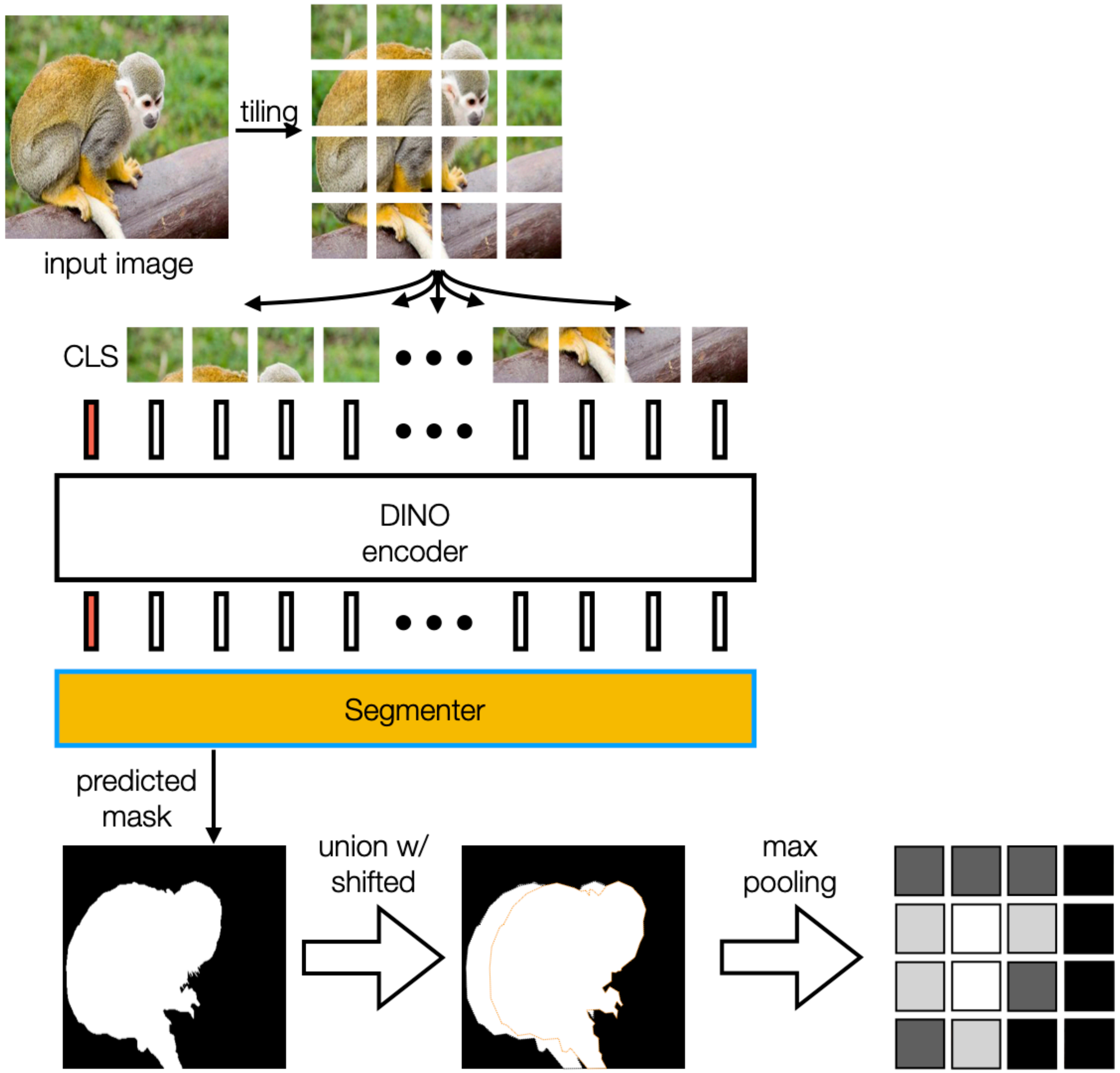
# Segmenting & Inpainting

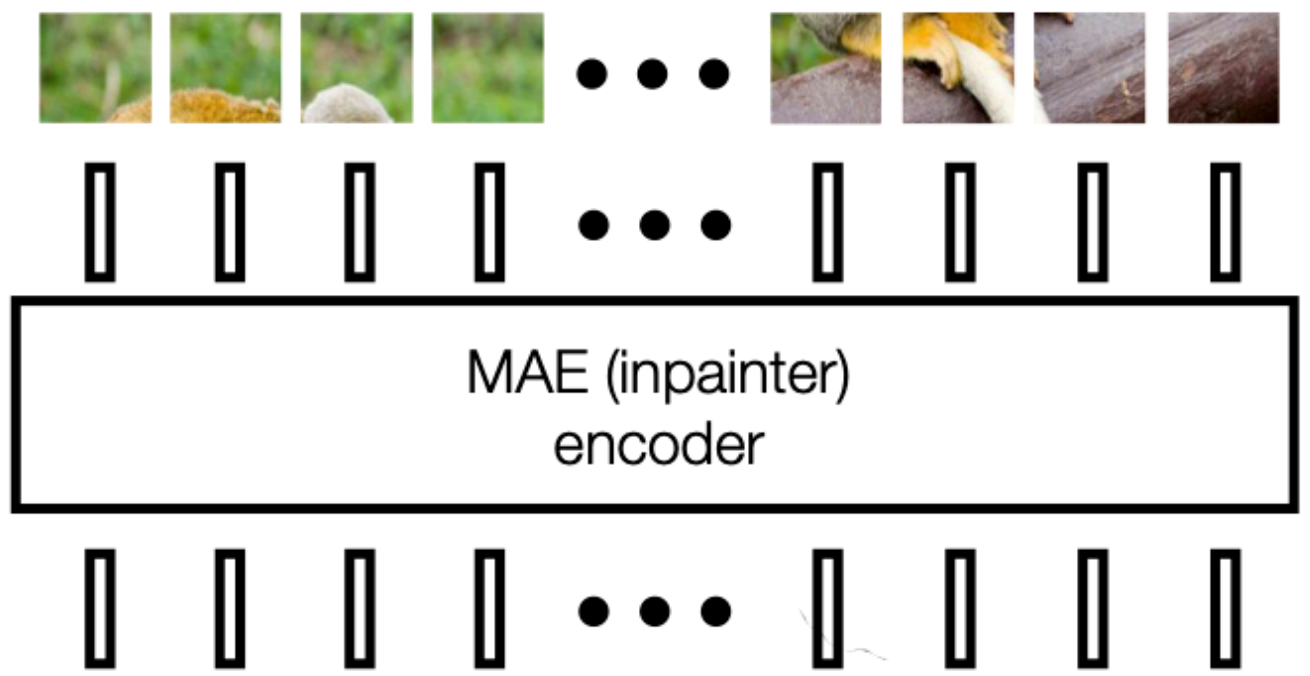


# Segmenting & Inpainting

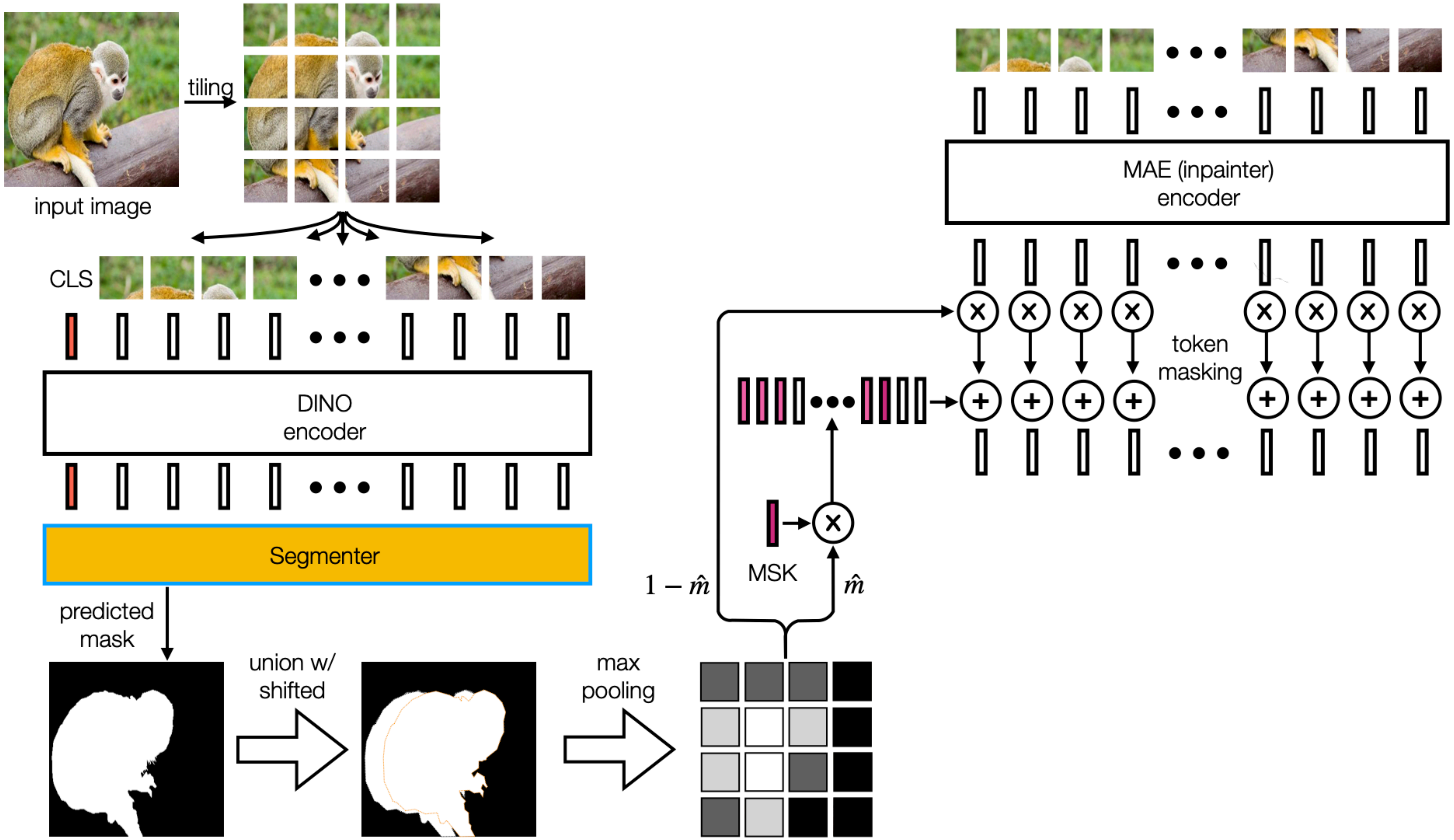


# Segmenting & Inpainting

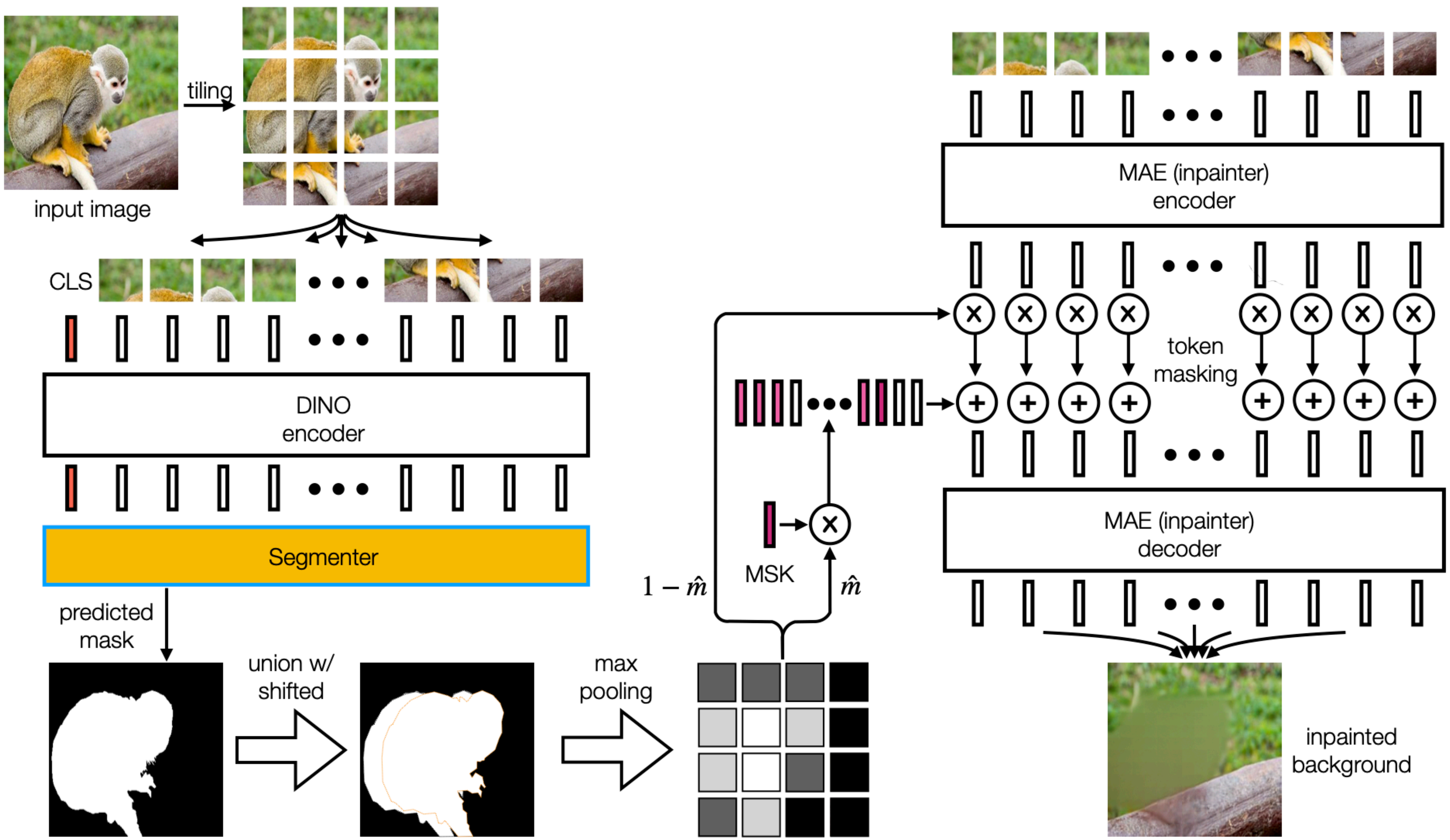


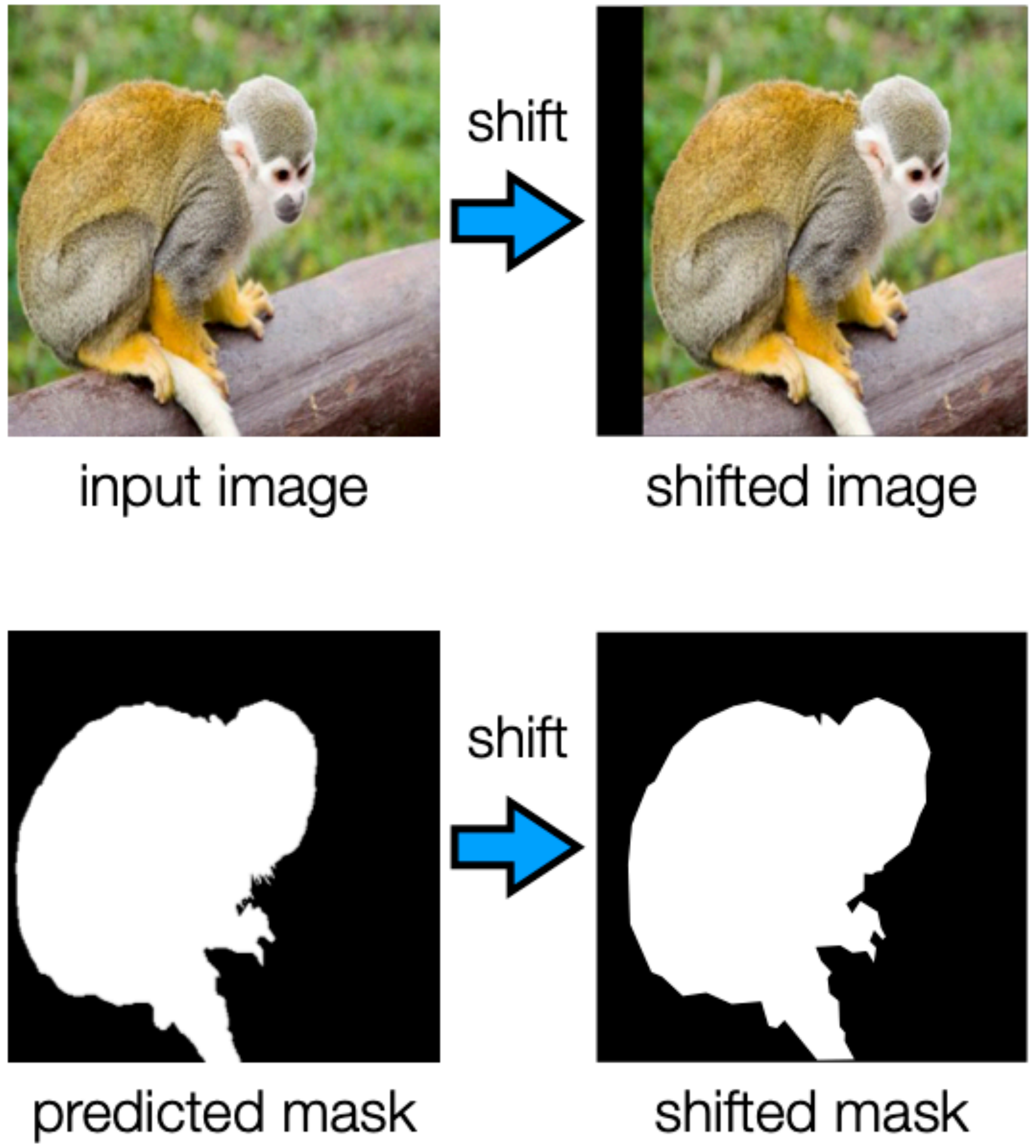
# Segmenting & Inpainting



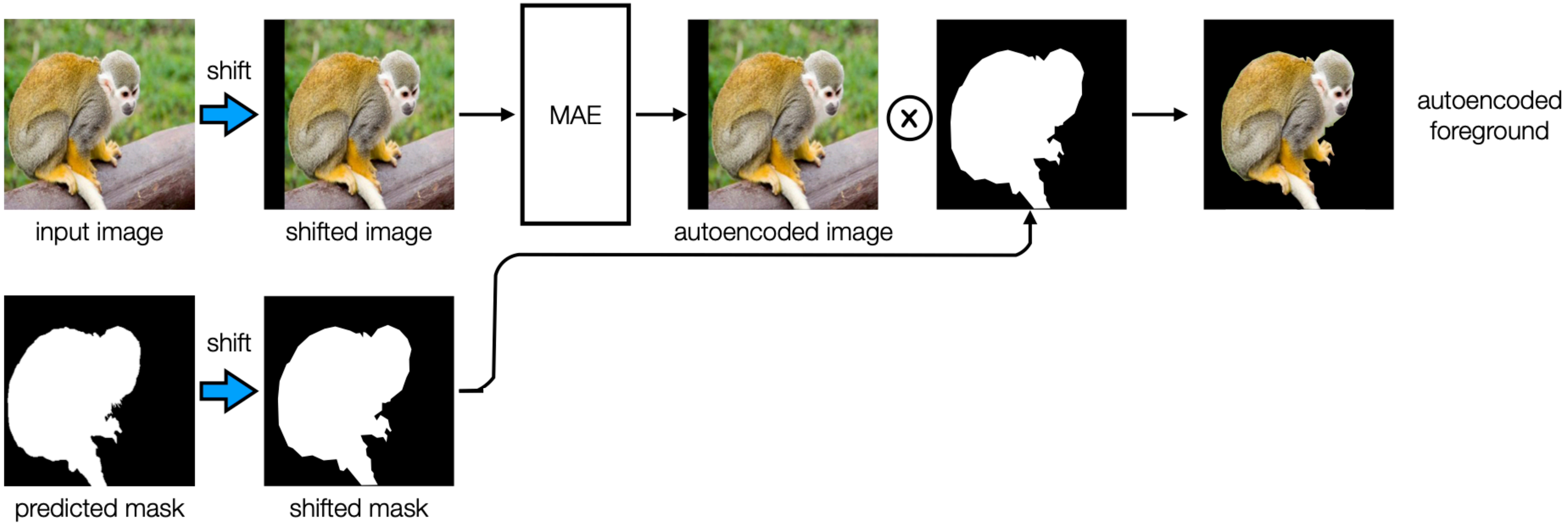
# Segmenting & Inpainting



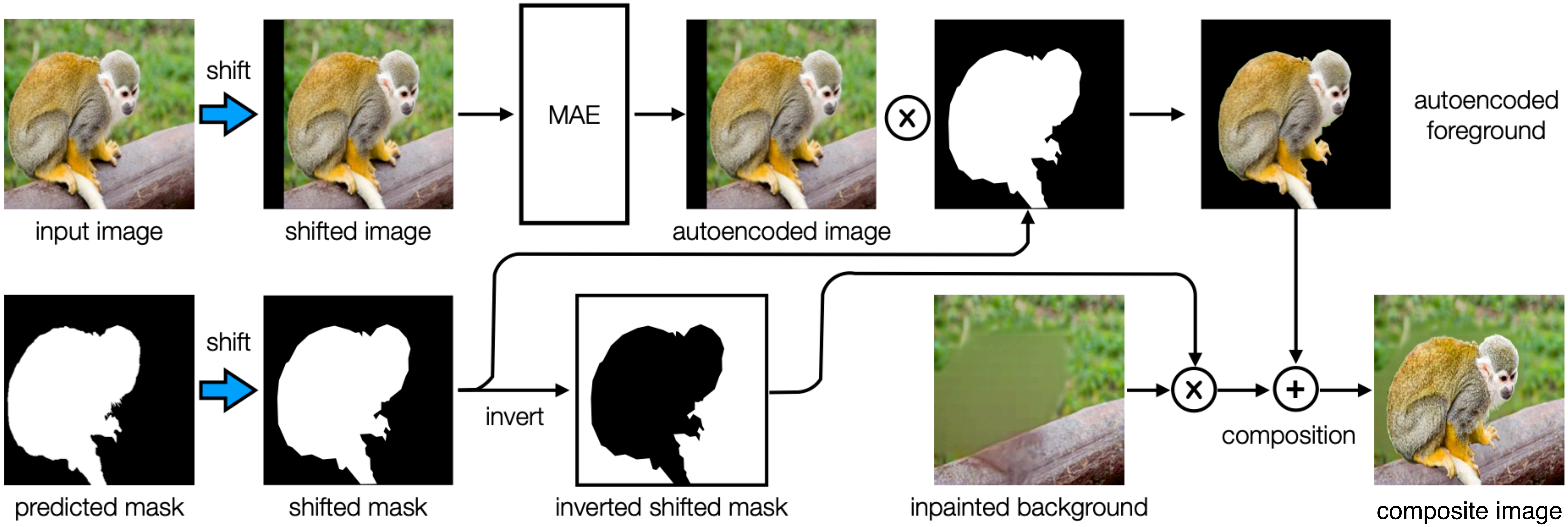
# Composition



# Composition



# Composition



# Saliency Results (ECSSD)

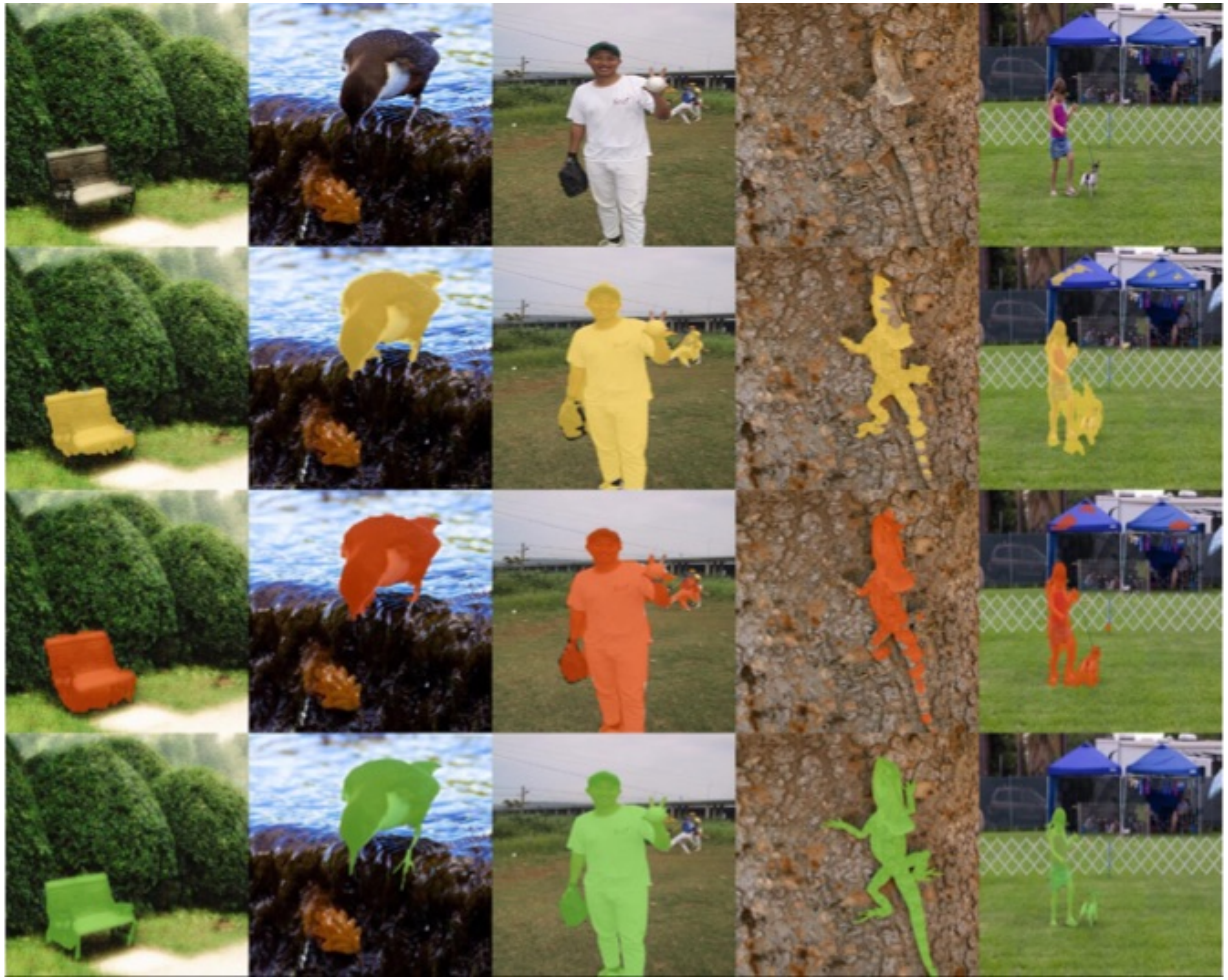
original

MOVE

SelfMask on MOVE

Ground truth

# Saliency Results (DUTS-TE)



original

MOVE

SelfMask on MOVE

Ground truth

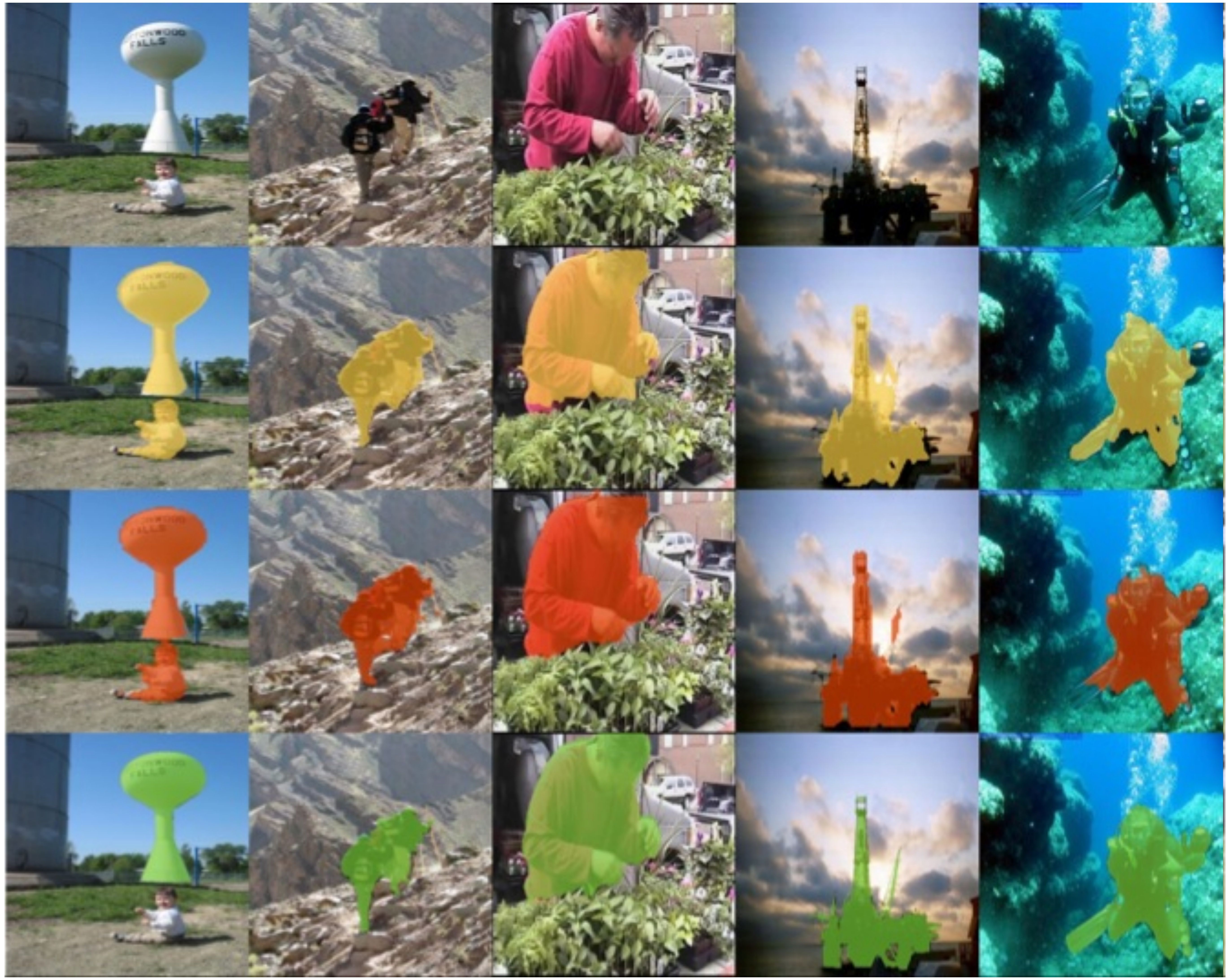
# Saliency Results (DUTS-OMRON)

original

MOVE

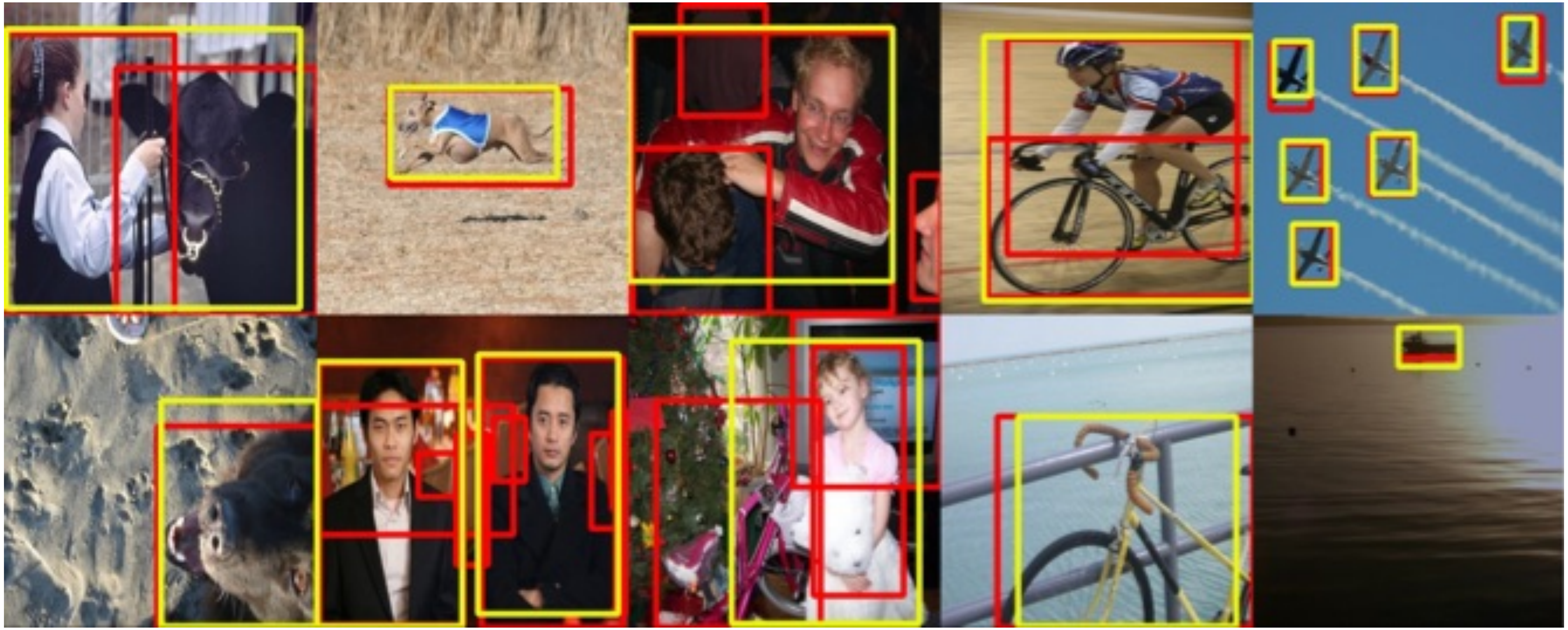
SelfMask on MOVE

Ground truth



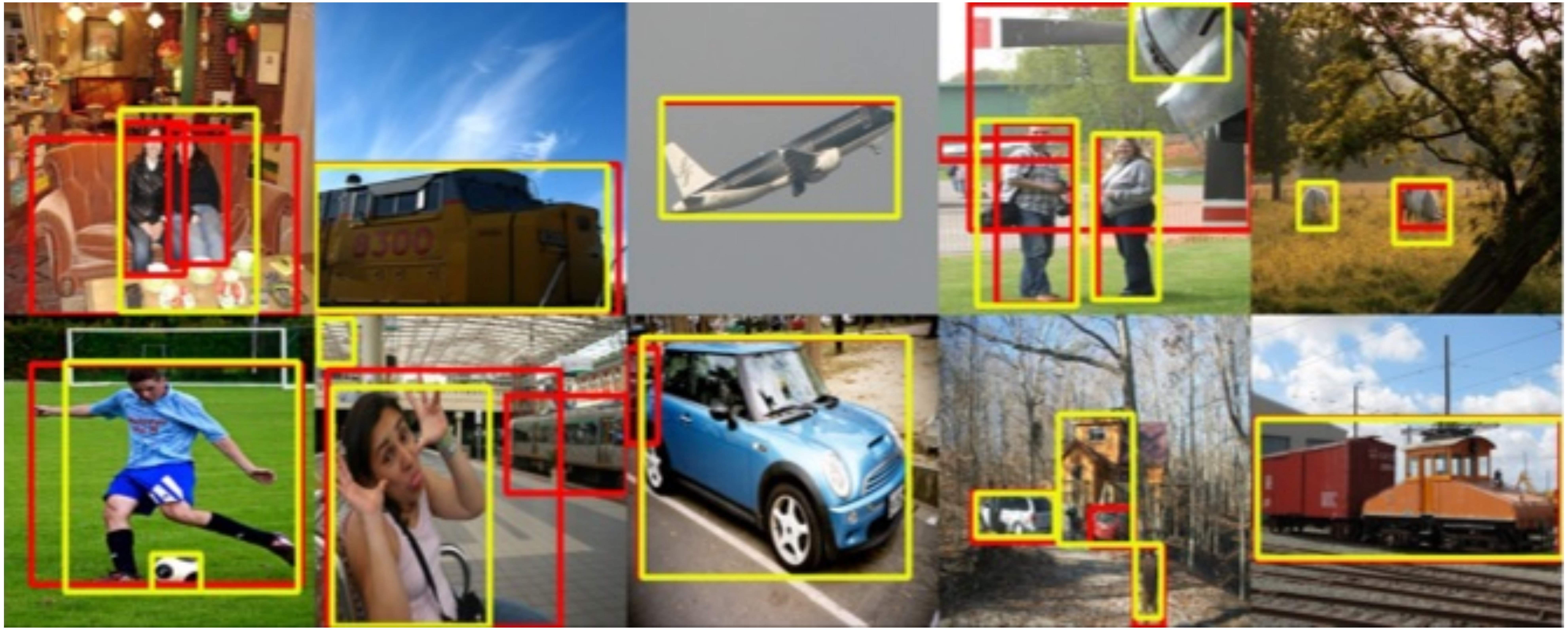
# Detection (VOC07)

Red is ground truth
Yellow is MOVE's prediction



# Detection (VOC12)

**Red** is ground truth
**Yellow** is MOVE's prediction



# Detection (COCO20K)

**Red** is ground truth
**Yellow** is MOVE's prediction

# Saliency Detection

| Model | DUT-OMRON | | | DUTS-TE | | | ECSSD | | |
|---|---|---|---|---|---|---|---|---|---|
| | Acc | IoU | max$F_\beta$ | Acc | IoU | max$F_\beta$ | Acc | IoU | max$F_\beta$ |
| Deep Spectral | - | .567 | - | - | .514 | - | - | .733 | - |
| TokenCut | .880 | .533 | .600 | .903 | .576 | .672 | .918 | .712 | .803 |
| FreeSOLO | .909 | .560 | .684 | .924 | .613 | .750 | .917 | .703 | .858 |
| **MOVE (Ours)** | **.913** | **.585** | **.690** | **.944** | **.680** | **.789** | **.950** | **.809** | **.901** |
| LOST + Bilateral | .818 | .489 | .578 | .887 | .572 | .697 | .916 | .723 | .837 |
| TokenCut + Bilateral | .897 | .618 | .697 | .914 | .624 | .755 | .934 | .772 | .874 |
| **MOVE (Ours) + Bilateral** | **.925** | **.627** | **.720** | **.949** | **.692** | **.811** | **.952** | **.804** | **.906** |
| SelfMask on pseudo* | .923 | .609 | .733 | .938 | .648 | .789 | .943 | .779 | .894 |
| SelfMask on pseudo* + Bilateral | **.939** | **.677** | **.774** | **.949** | .694 | .819 | .951 | .803 | .911 |
| **SelfMask on MOVE (Ours)** | .916 | .643 | .739 | .947 | **.720** | **.824** | **.957** | **.839** | **.917** |
| **SelfMask on MOVE (Ours) + Bilateral** | .922 | .657 | .743 | .948 | .699 | .817 | .956 | .819 | .912 |

# Unsupervised Single Object Discovery

| Method | VOC07 | VOC12 | COCO20K |
|---|---|---|---|
| FreeSOLO | 56.1 | 56.7 | 52.8 |
| LOST | 61.9 | 64.0 | 50.7 |
| Deep Spectral | 62.7 | 66.4 | 52.2 |
| TokenCut | 68.8 | 72.1 | 58.8 |
| **MOVE (Ours)** | **73.5** (↑ **4.7**) | **76.6** (↑ **4.5**) | **63.0** (↑ **4.2**) |
| LOD + CAD | 56.3 | 61.6 | 52.7 |
| rOSD + CAD | 58.3 | 62.3 | 53.0 |
| LOST + CAD | 65.7 | 70.4 | 57.5 |
| TokenCut + CAD | 71.4 | 75.3 | 62.6 |
| **MOVE (Ours) + CAD** | 73.6 | 77.1 | 65.0 |
| **MOVE (Ours) Multi + CAD** | **74.6** (↑ **3.2**) | **79.3** (↑ **4.0**) | **68.6** (↑ **6.0**) |

Correct Localization metric (CorLoc): percentage of images, where IoU>0.5 for a predicted single bounding box with at least one of the ground truth ones

# Unsupervised Single Object Discovery

| Method | VOC07 | VOC12 | COCO20K |
|---|---|---|---|
| LOD + CAD | 56.3 | 61.6 | 52.7 |
| rOSD + CAD | 58.3 | 62.3 | 53.0 |
| LOST + CAD | 65.7 | 70.4 | 57.5 |
| TokenCut + CAD | 71.4 | 75.3 | 62.6 |
| **MOVE (Ours)** + **CAD** | 73.6 | 77.1 | 65.0 |
| **MOVE (Ours) Multi** + **CAD** | **74.6** (↑ **3.2**) | **79.3** (↑ **4.0**) | **68.6** (↑ **6.0**) |

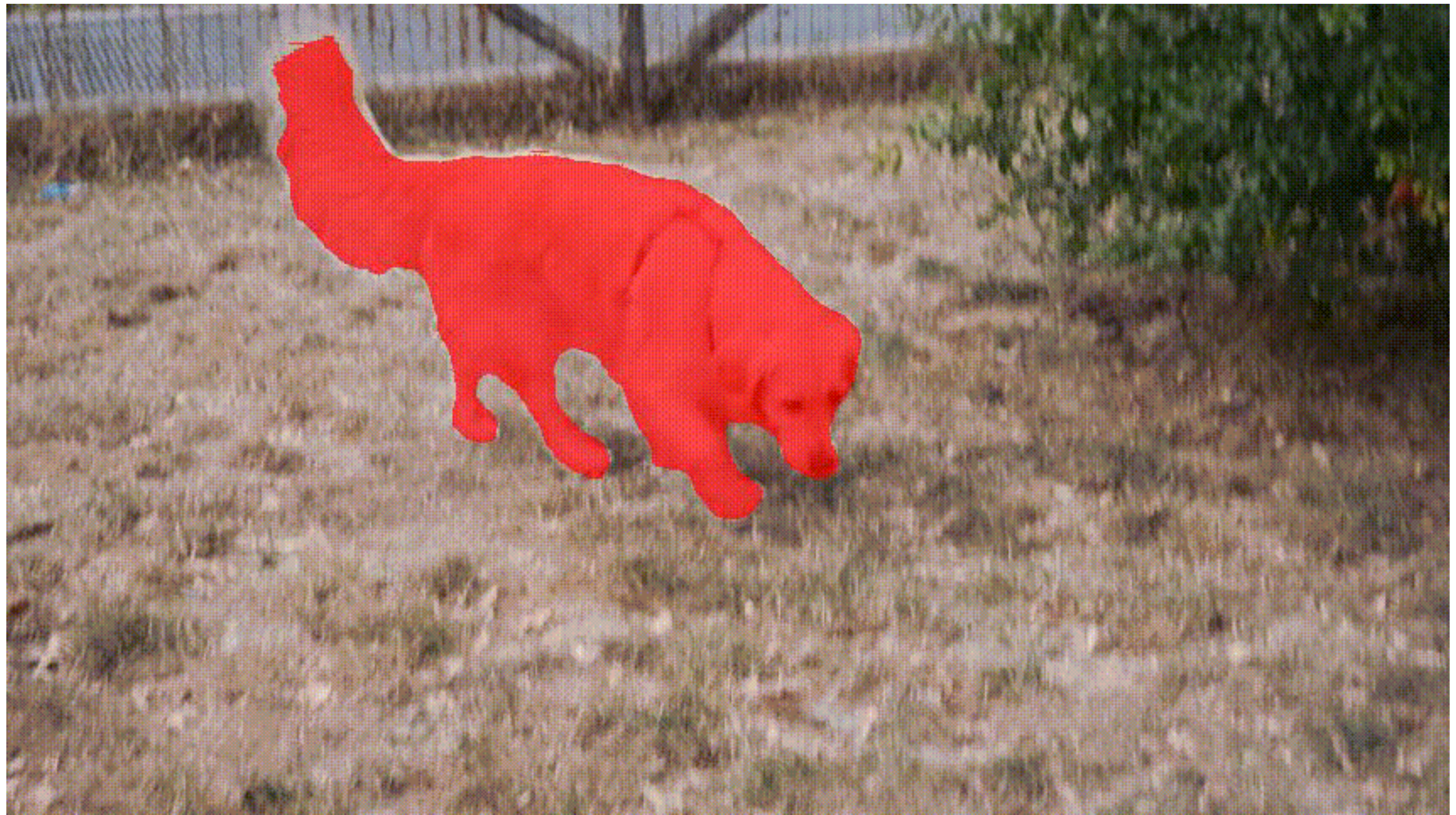
Correct Localization metric (CorLoc): percentage of images, where IoU>0.5 for a predicted single bounding box with at least one of the ground truth ones

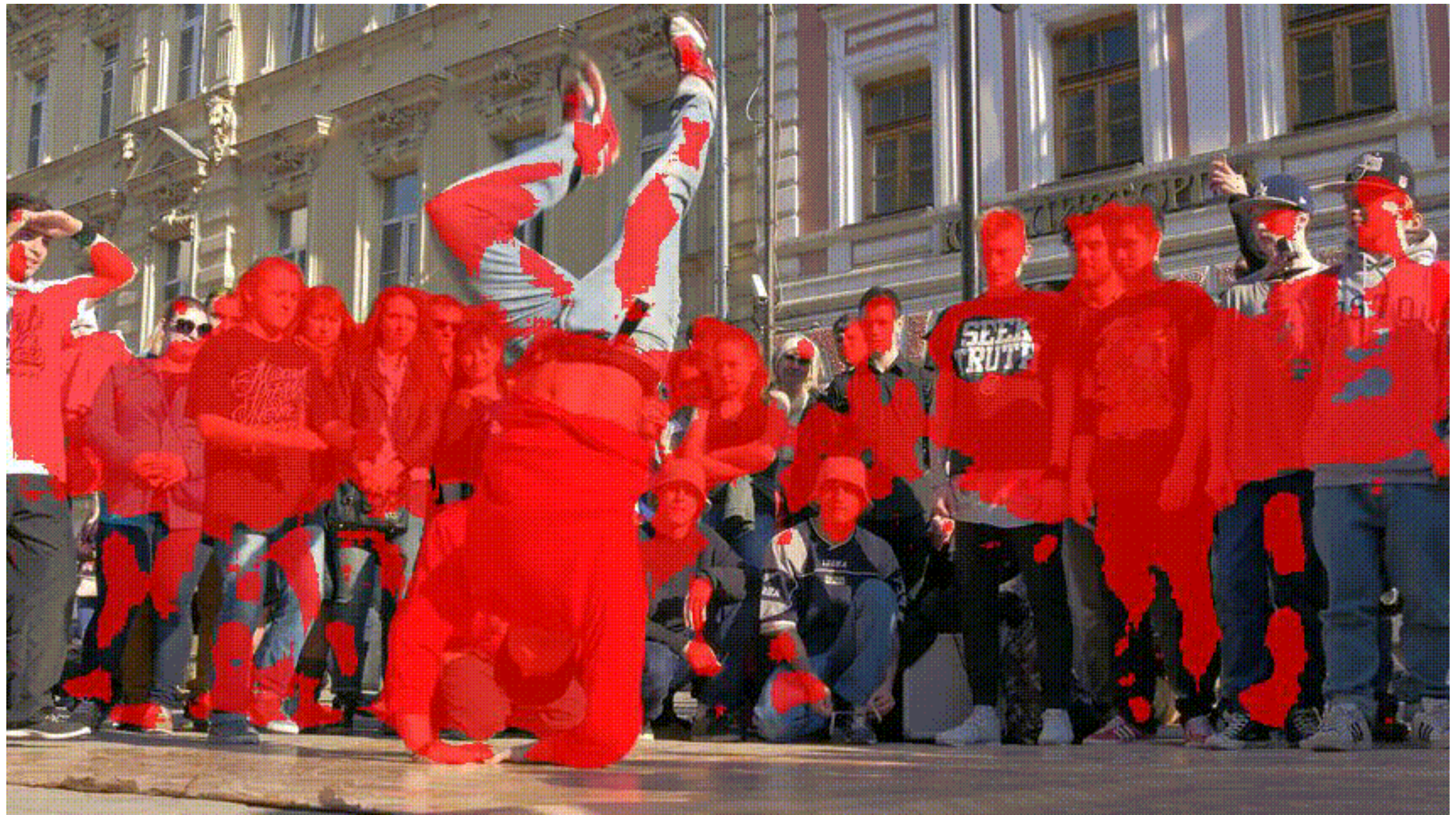
# Preliminary Tests on Videos

# Preliminary Tests on Videos

# Preliminary Tests on Videos

# Preliminary Tests on Videos



# Preliminary Tests on Videos



# Unsupervised learning of controllable systems

- So far only representations of single images: What about videos?

# Unsupervised learning of controllable systems

- So far only representations of single images: What about videos?
- We could represent each frame and the transitions across frames

# Unsupervised learning of controllable systems

- So far only representations of single images: What about videos?

- We could represent each frame and the transitions across frames

  - **States** are representations of static images

# Unsupervised learning of controllable systems

- So far only representations of single images: What about videos?
- We could represent each frame and the transitions across frames
  - **States** are representations of static images
  - **Actions** are representations of the changes/transitions

# Unsupervised learning of controllable systems

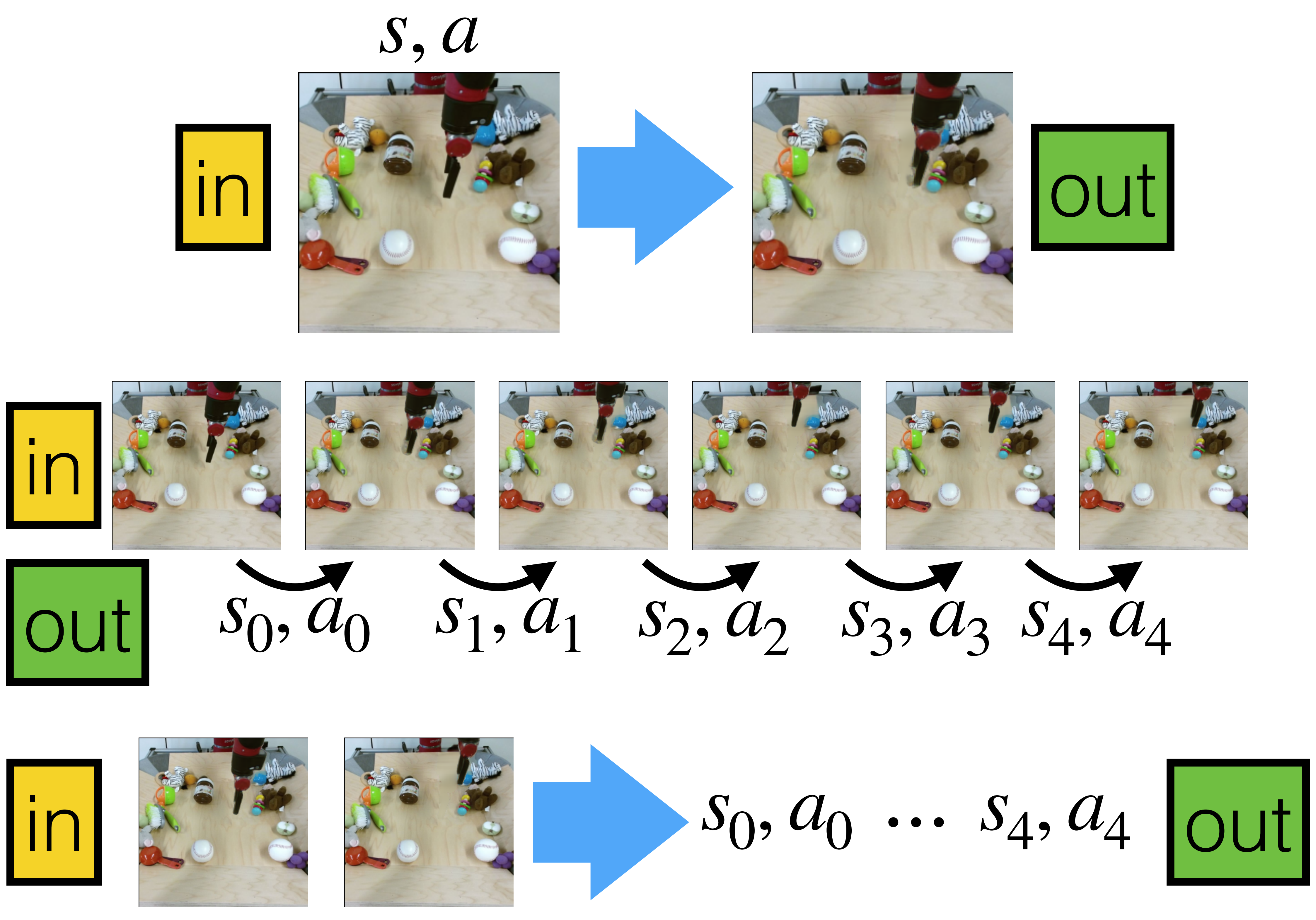
- So far only representations of single images: What about videos?
- We could represent each frame and the transitions across frames
  - **States** are representations of static images
  - **Actions** are representations of the changes/transitions
- One is usually given the actions, but they may not be easily available

# Unsupervised learning of controllable systems

- So far only representations of single images: What about videos?
- We could represent each frame and the transitions across frames
  - **States** are representations of static images
  - **Actions** are representations of the changes/transitions
- One is usually given the actions, but they may not be easily available
  - What about a model that **learns its action space**?

# Learning by predicting the future

- **Goal**: A generative controllable model with
  - **Predictions**: what is the future?
  - **Sequence parsing**: what is the representation of a video in terms of states and actions?
  - **Planning**: What sequence of actions takes an agent between these two states?
  - **Counterfactual**: eg, what would happen if?




Blattmann et al, ipoke: Poking a still image for controlled stochastic video synthesis, CVPR 2021
Menapace et al, Playable Video Generation, CVPR 2021
Menapace et al, Playable Environments: Video Manipulation in Space and Time, ArXiv 2022

# Object Interactions

simulated scenarios

What would happen if I placed a new object in front of the robot arm and moved the robot arm towards it?

Has not learned object-arm interactions

Has learned object-arm interactions

# Object Interactions

simulated scenarios

What would happen if I placed a new object in front of the robot arm and moved the robot arm towards it?

Has not learned object-arm interactions

Has learned object-arm interactions

# Object Interactions

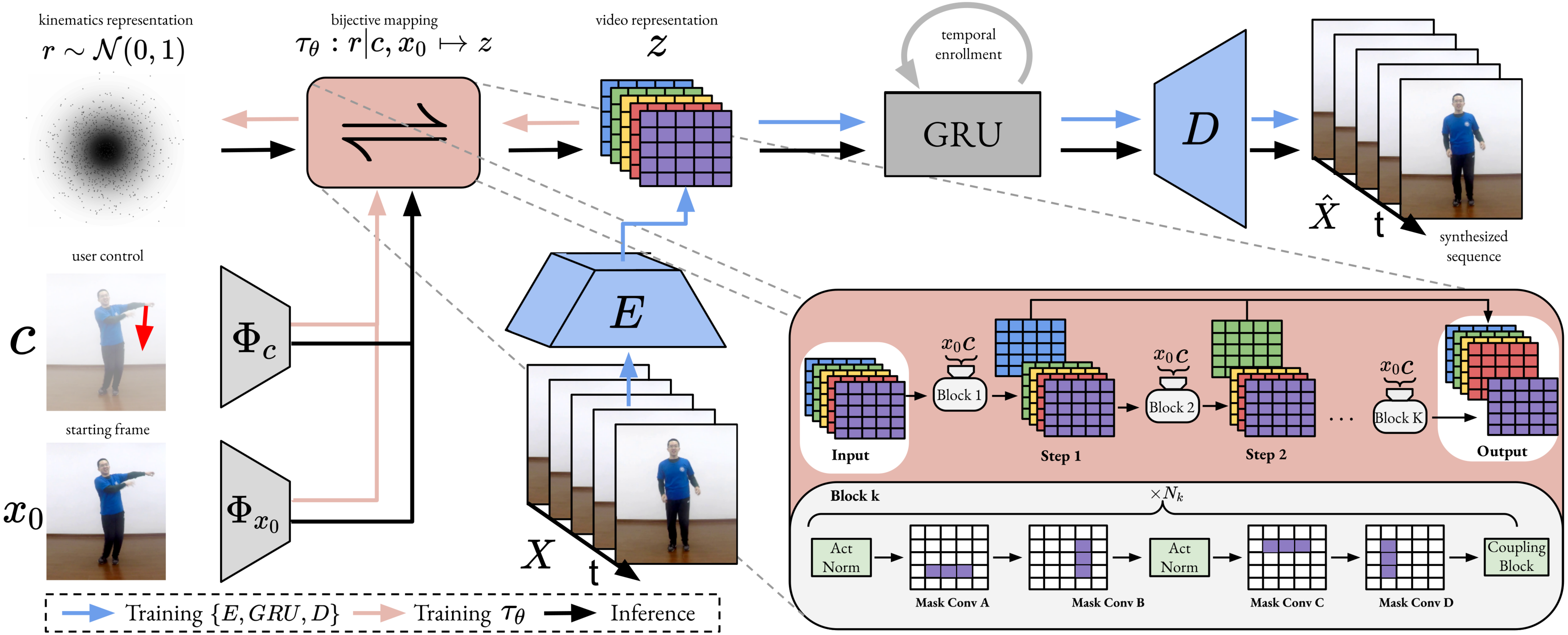
simulated scenarios

What would happen if I placed a new object in front of the robot arm and moved the robot arm towards it?

Has not learned object-arm interactions

Has learned object-arm interactions

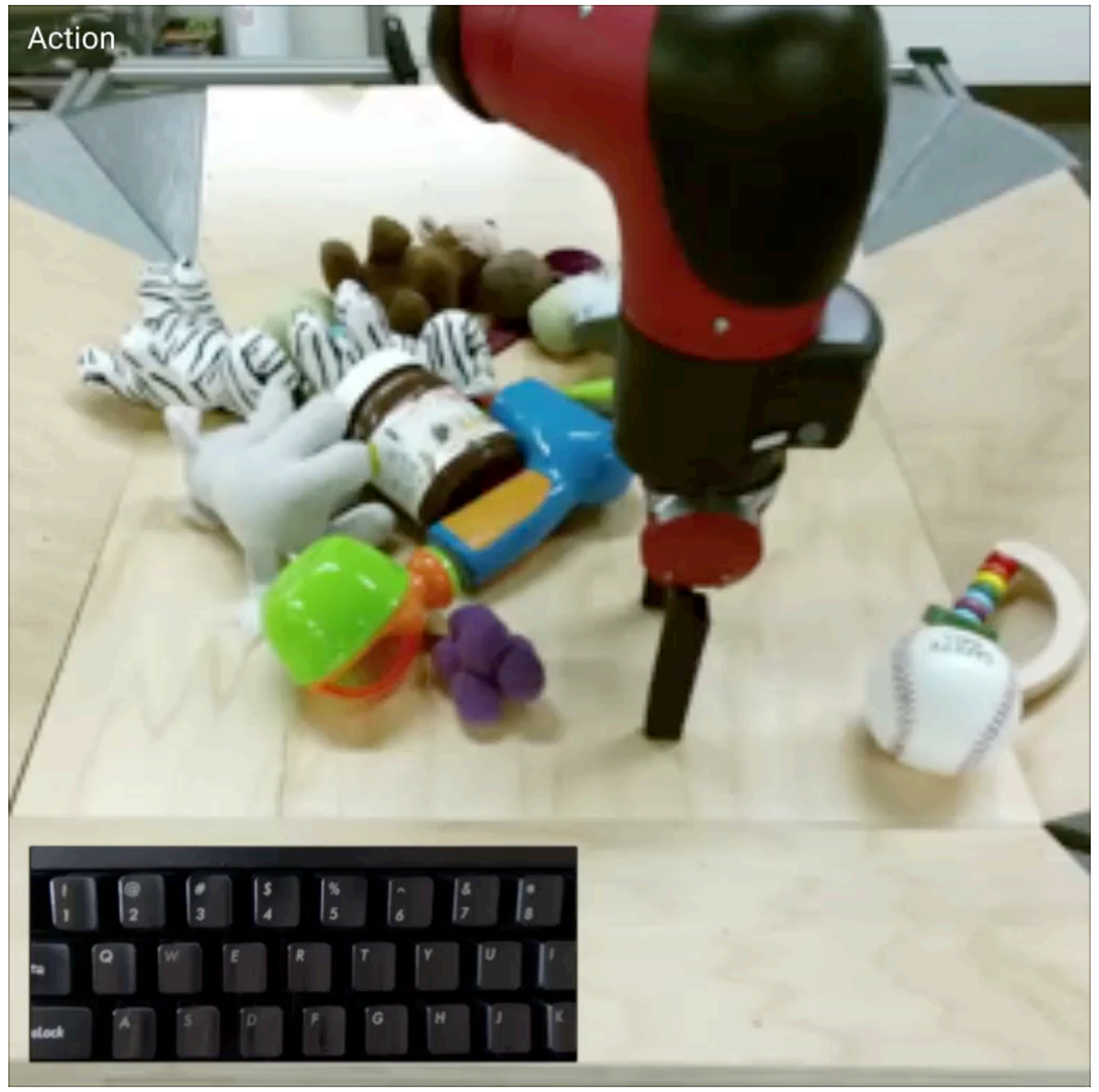
# Current Progress

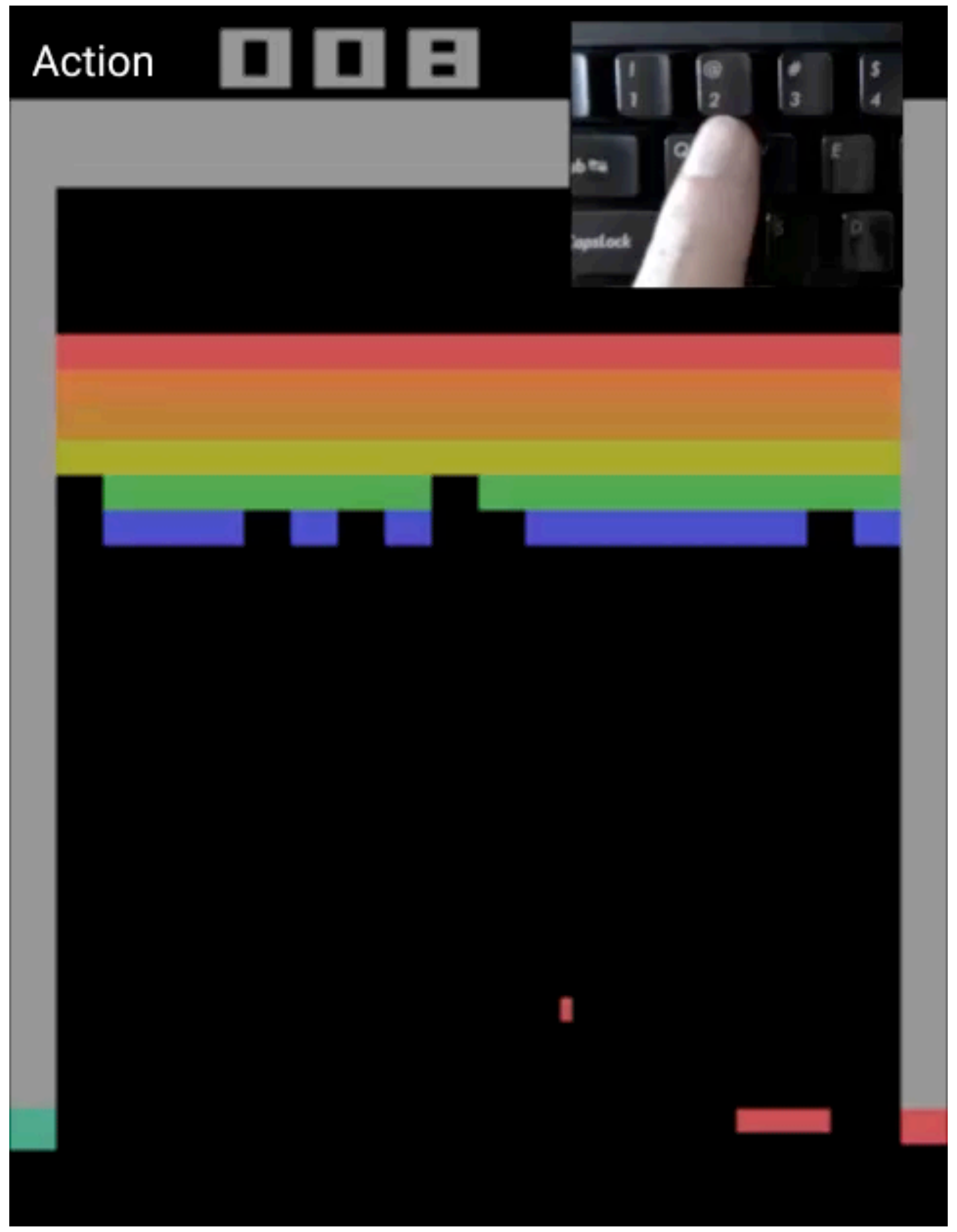



Blattmann et al, iPOKE: Poking a Still Image for Controlled Stochastic Video Synthesis, ICCV 2021

# Current Progress

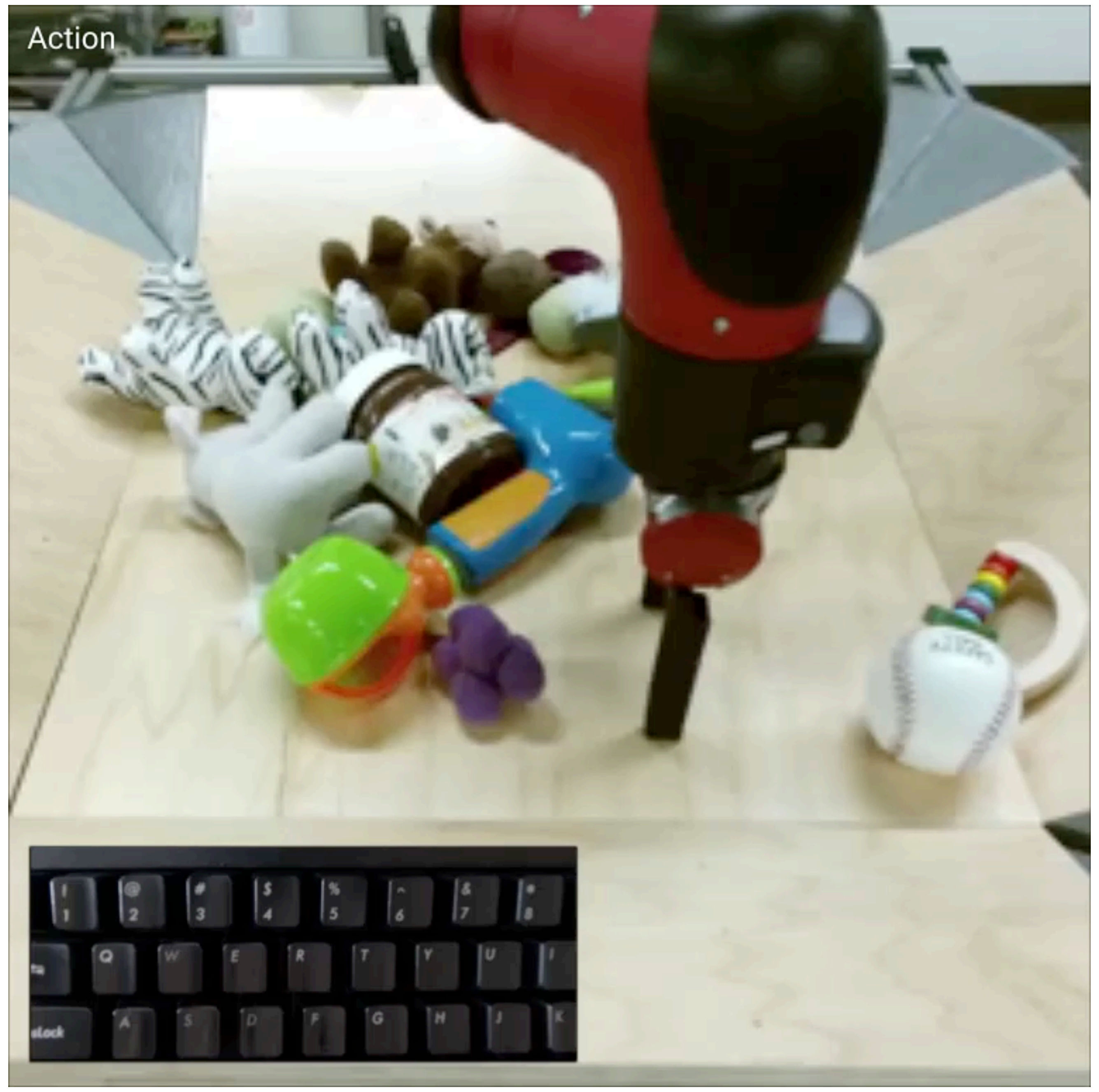
- Menapace et al, Playable Video Generation, CVPR 2021

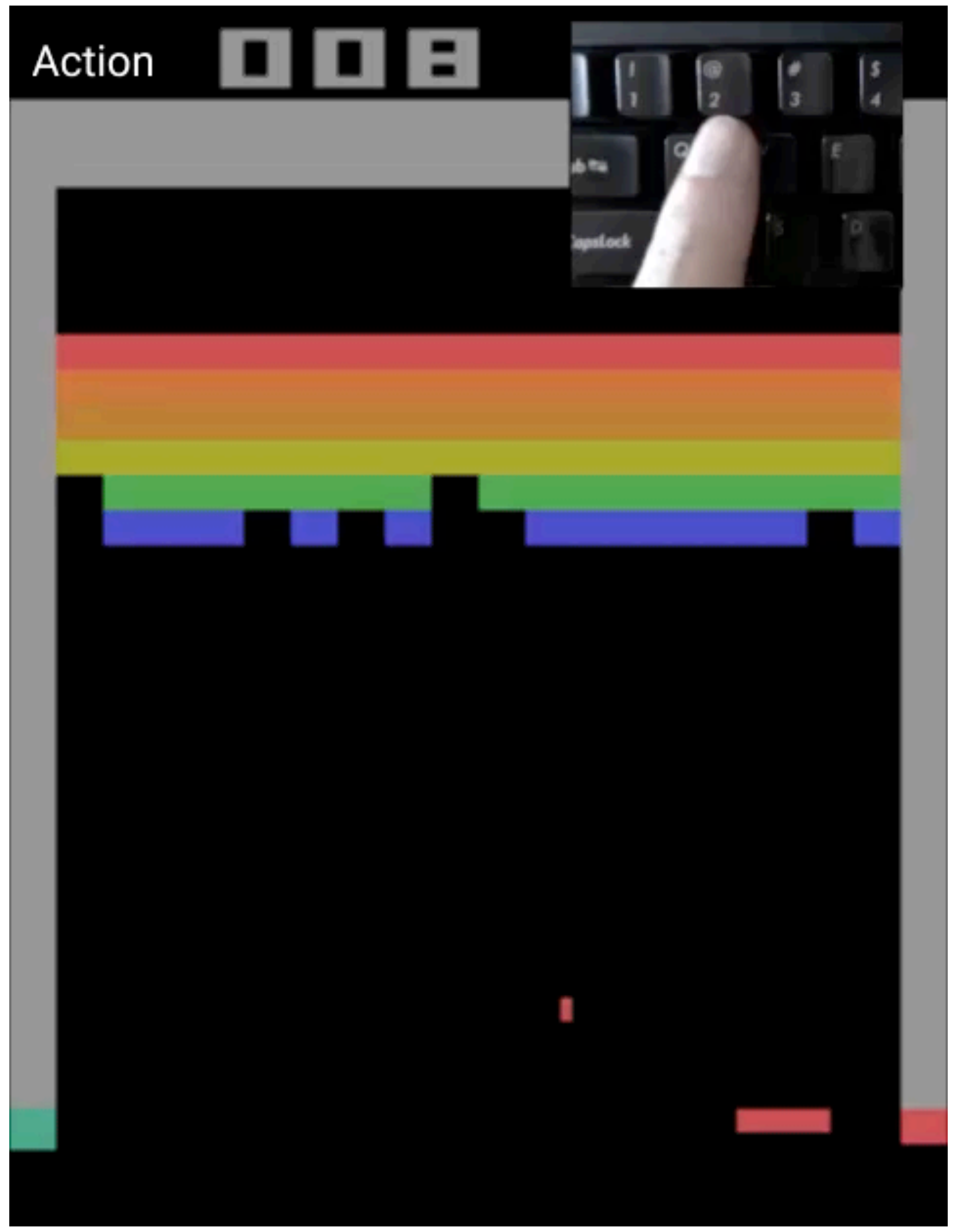



# Current Progress

- Menapace et al, Playable Video Generation, CVPR 2021



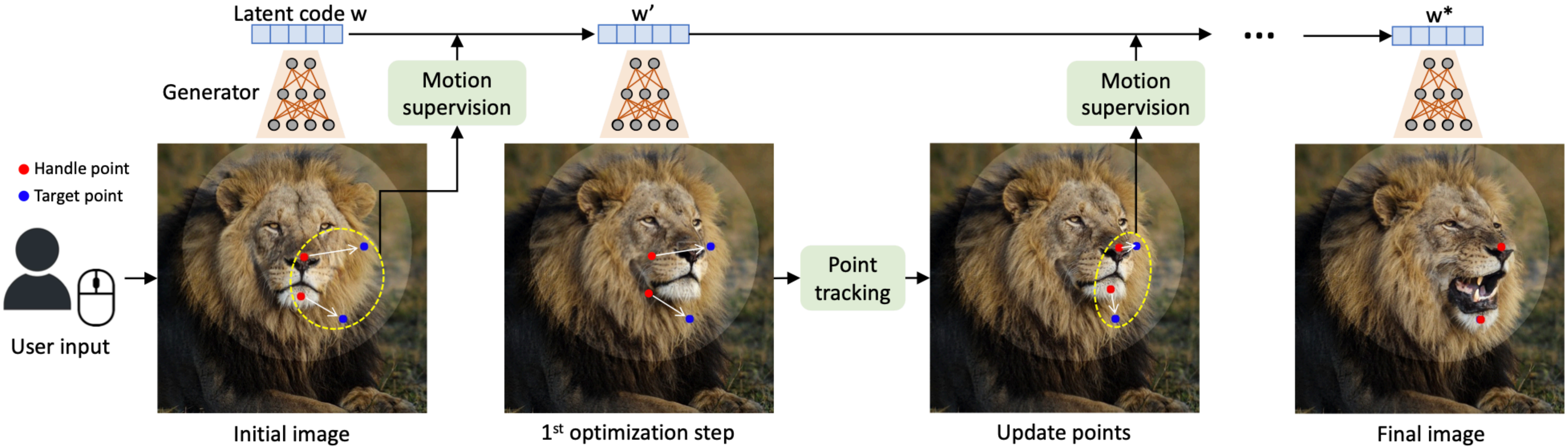

# Current Progress

- Menapace et al, Playable Environments: Video Manipulation in Space and Time, ArXiv 2022


# Current Progress

- Menapace et al, Playable Environments: Video Manipulation in Space and Time, ArXiv 2022


# Editable Models: DragGAN*




*Pan et al. Drag Your GAN: Interactive Point-based Manipulation on the Generative Image Manifold, SIGGRAPH 2023

# GLASS

- Global and Local Action-driven Sequence Synthesis (GLASS)

- Learns two action spaces:
  Global (explicit geometric transformations) and
  Local (photometric transformations)

- W-Sprites: New dataset to evaluate action identification

A. Davtyan and P. Favaro, Controllable Video Generation through Global and Local Motion Dynamics, ECCV 2022

# GLASS

- Global and Local Action-driven Sequence Synthesis (GLASS)

- Learns two action spaces:
  Global (explicit geometric transformations) and
  Local (photometric transformations)

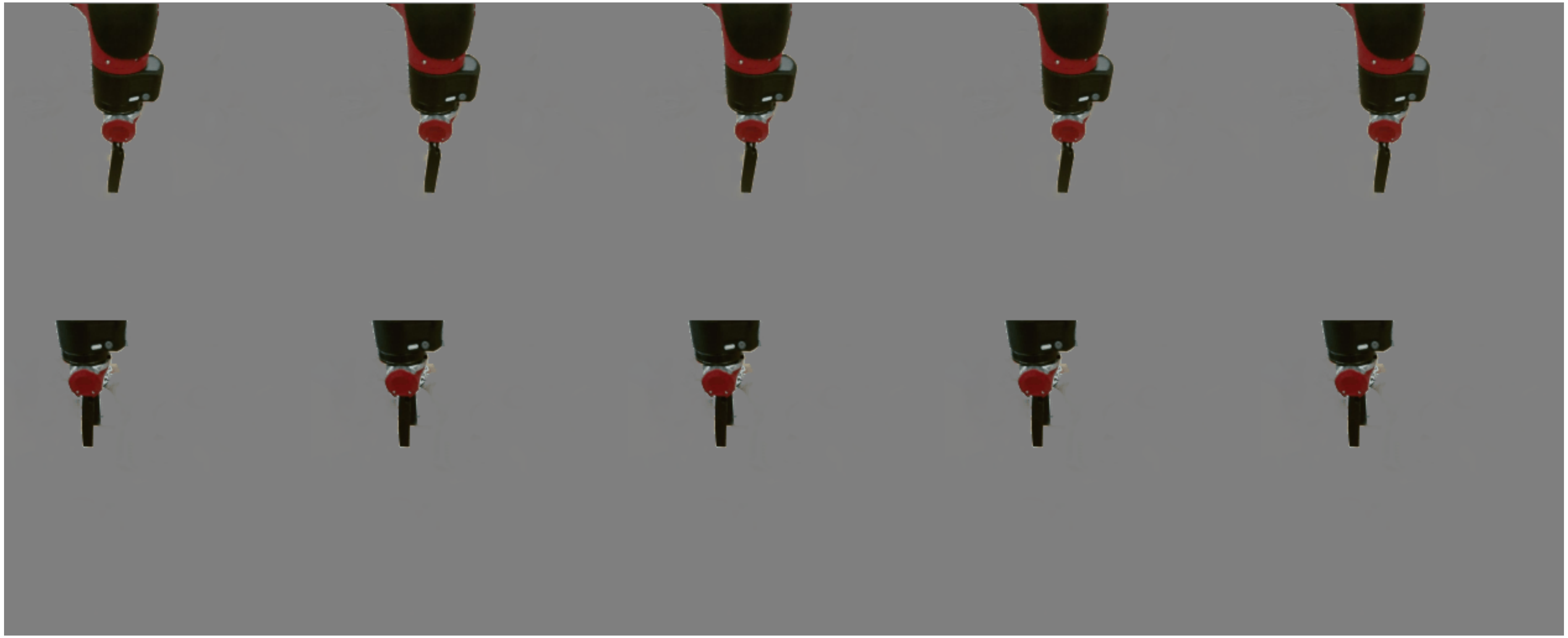
- W-Sprites: New dataset to evaluate action identification

A. Davtyan and P. Favaro, Controllable Video Generation through Global and Local Motion Dynamics, ECCV 2022

# GLASS: Global Action Analysis

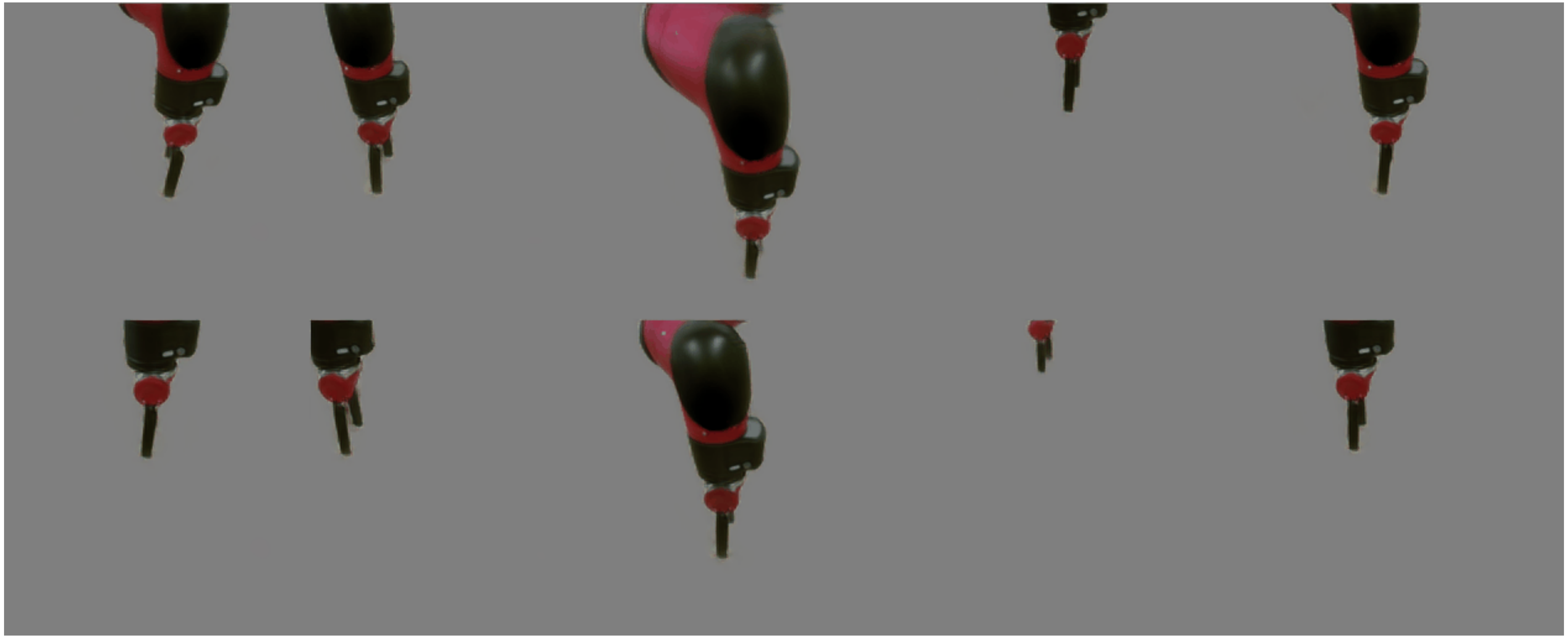
# GLASS: Local Action Analysis

# Learned Global Actions: BAIR



right left down up no motion

# Learned Global Actions: BAIR



right left down up no motion

# Learned Global Actions: Tennis

right left down up no motion

# Learned Global Actions: Tennis

right left down up no motion

# Action Transfer

actions

source

target

# Action Transfer

actions

source

target

# Action Transfer

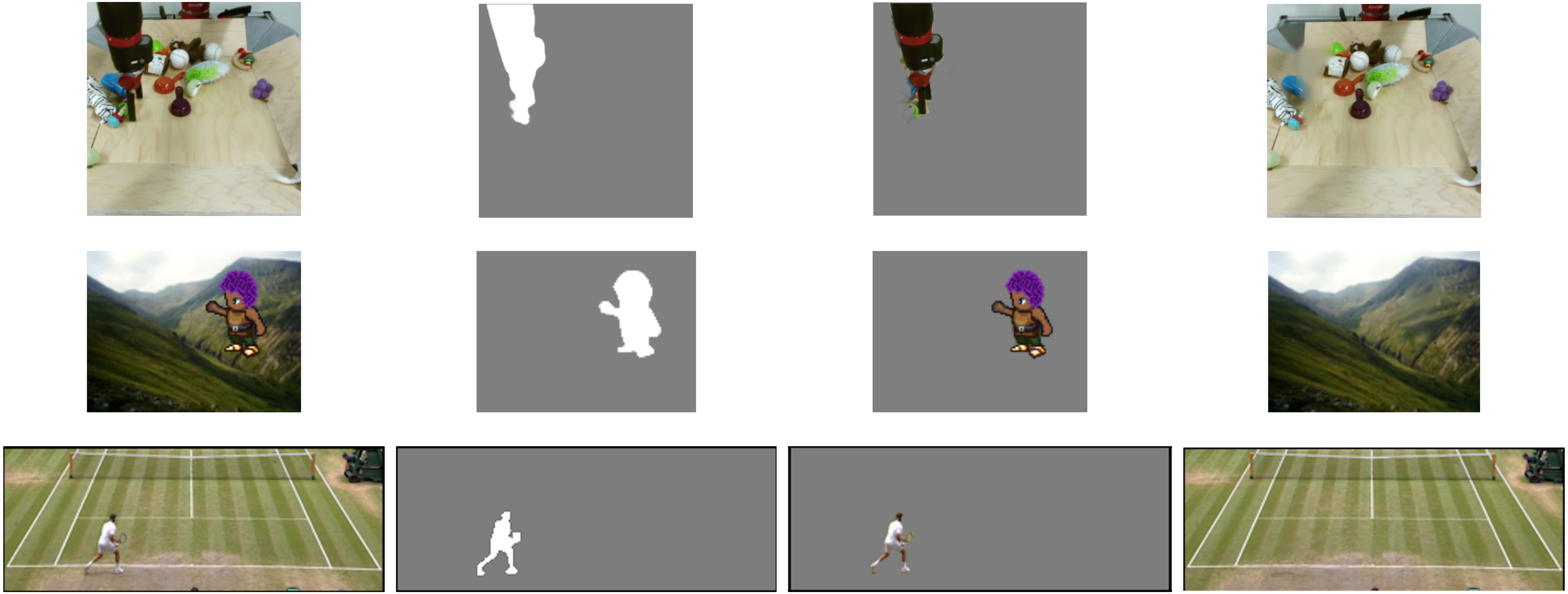
actions

source

target

# Qualitative Evaluation

input image | predicted segmentation | foreground | background



# Experiments: Video Generation

Image/Video reconstruction on BAIR (Robotic Arm)

conditional video generation

controllable

| Method | LPIPS↓ | FID↓ | FVD↓ |
|---|---|---|---|
| MoCoGAN [40] | 0.466 | 198 | 1380 |
| MoCoGAN+ [30] | 0.201 | 66.1 | 849 |
| SAVP [28] | 0.433 | 220 | 1720 |
| SAVP+ [30] | 0.154 | 27.2 | 303 |
| Huang et al. [21] w/ *non-param* control | 0.176 | 29.3 | 293 |
| CADDY [30] | 0.202 | 35.9 | 423 |
| Huang et al. [21] w/ *positional* control | 0.202 | 28.5 | 333 |
| Huang et al. [21] w/ *affine* control | 0.201 | 30.1 | **292** |
| GLASS | **0.118** | **18.7** | 411 |

# Experiments: Video Generation

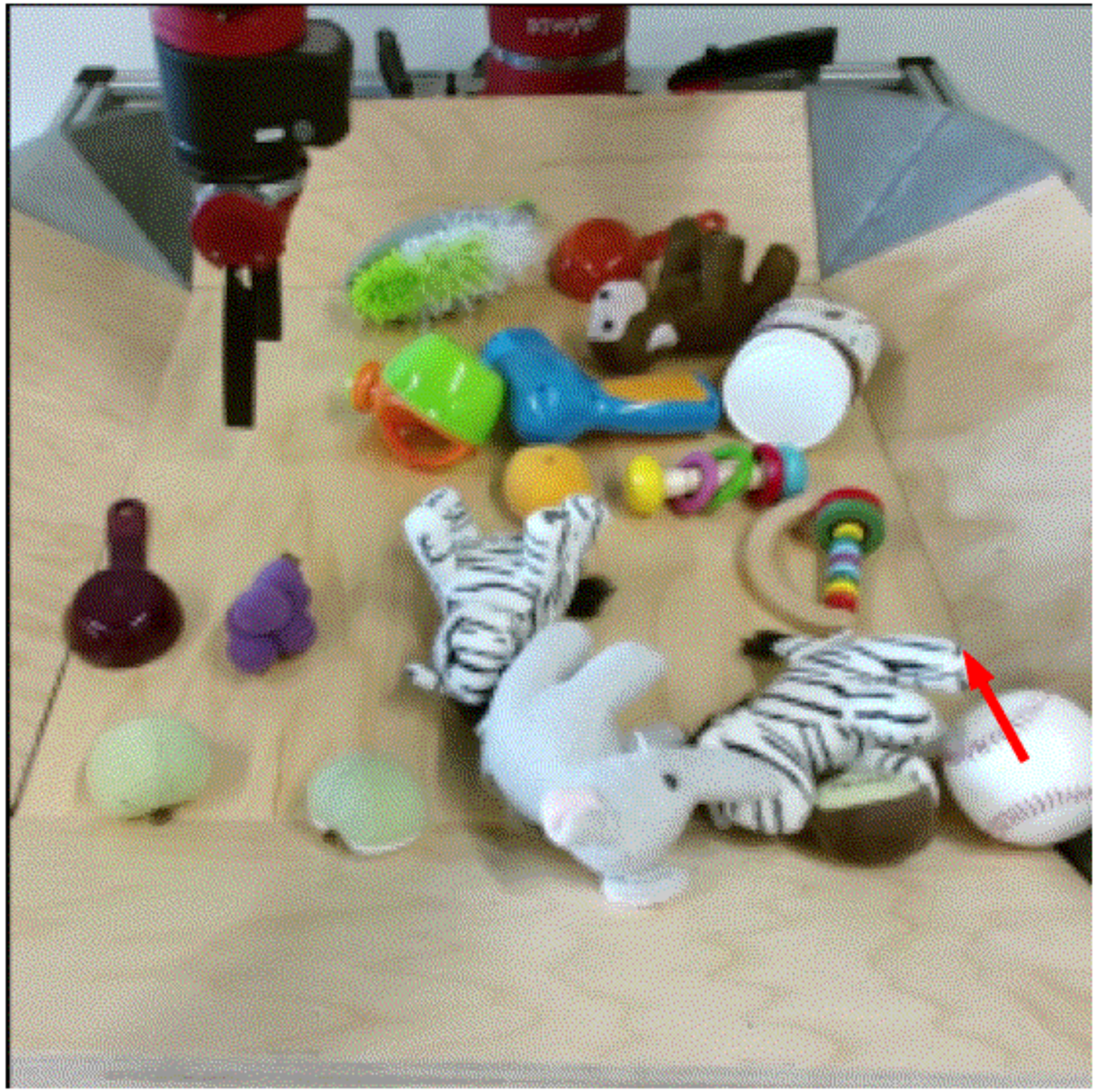
Image/Video reconstruction on Tennis

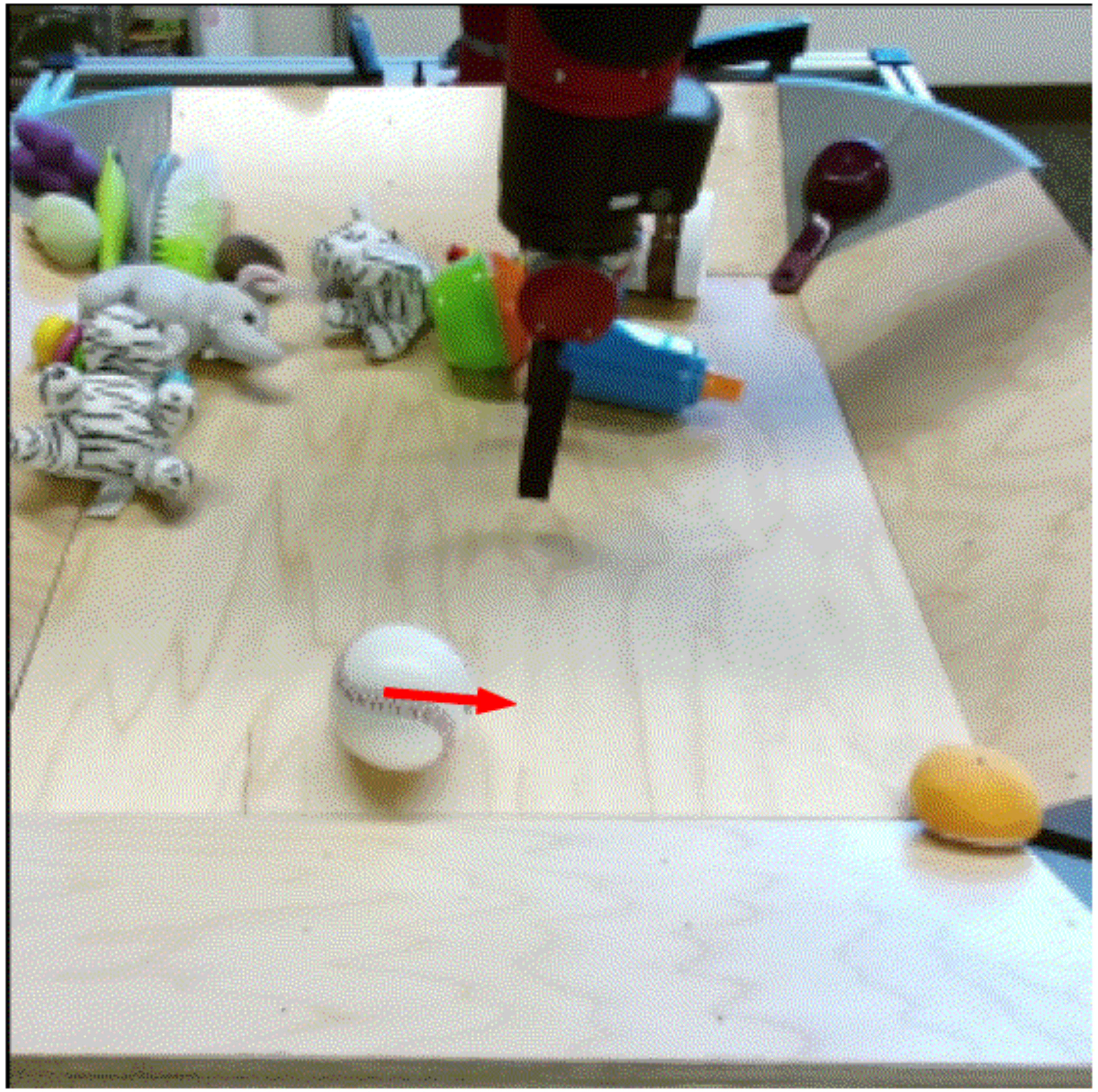
conditional video generation

controllable

| Method | LPIPS↓ | FID↓ | FVD↓ | ADD↓ | MDR↓ |
|---|---|---|---|---|---|
| MoCoGAN [40] | 0.266 | 132 | 3400 | 28.5 | 20.2 |
| MoCoGAN+ [30] | 0.166 | 56.8 | 1410 | 48.2 | 27.0 |
| SAVP [28] | 0.245 | 156 | 3270 | 10.7 | 19.7 |
| SAVP+ [30] | 0.104 | 25.2 | 223 | 13.4 | 19.2 |
| Huang et al. [21] w/ *non-param* control | 0.100 | 8.68 | 204 | 1.76 | 0.306 |
| CADDY [30] | 0.102 | 13.7 | 239 | 8.85 | 1.01 |
| Huang et al. [21] w/ *positional* control | 0.122 | 10.1 | 215 | 4.30 | 0.300 |
| Huang et al. [21] w/ *affine* control | 0.115 | 11.2 | **207** | 3.40 | 0.317 |
| GLASS | **0.046** | **7.37** | 257 | **2.00** | **0.214** |

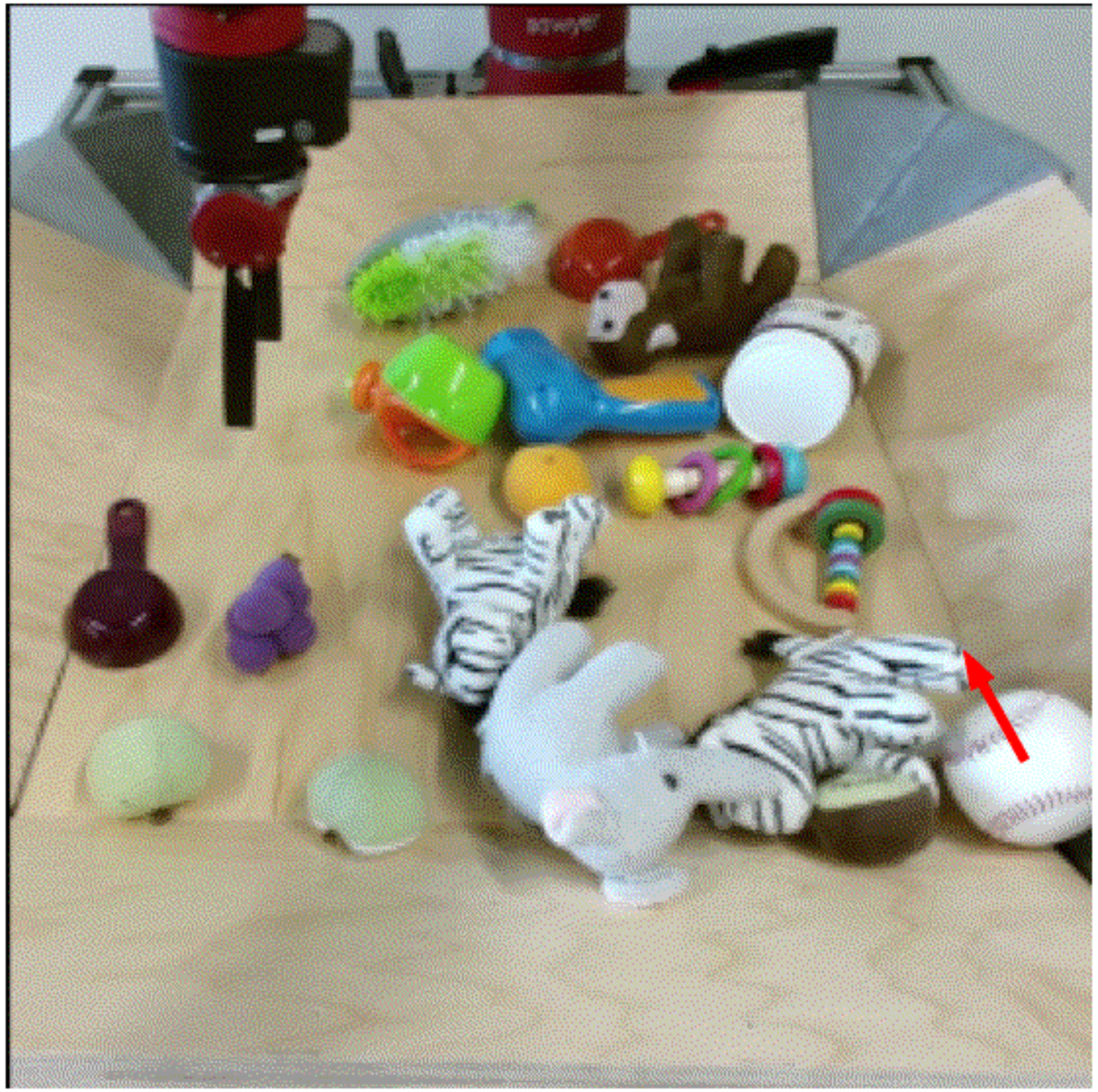
# Object Interactions via YODA*

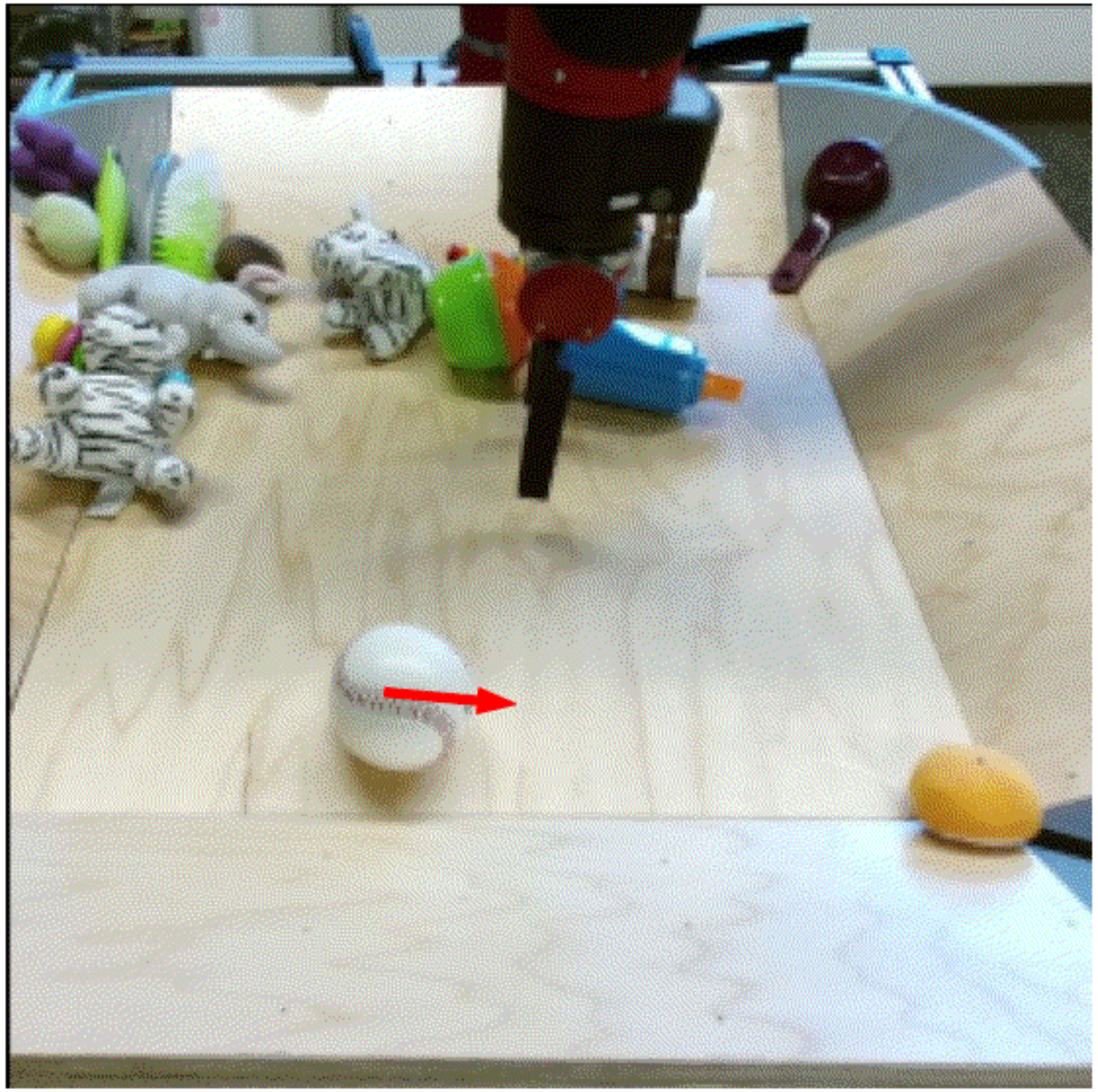




*Davtyan and Favaro, Learn the Force We Can: Multi-Object Video Generation from Pixel-Level Interactions, tech. report 2023

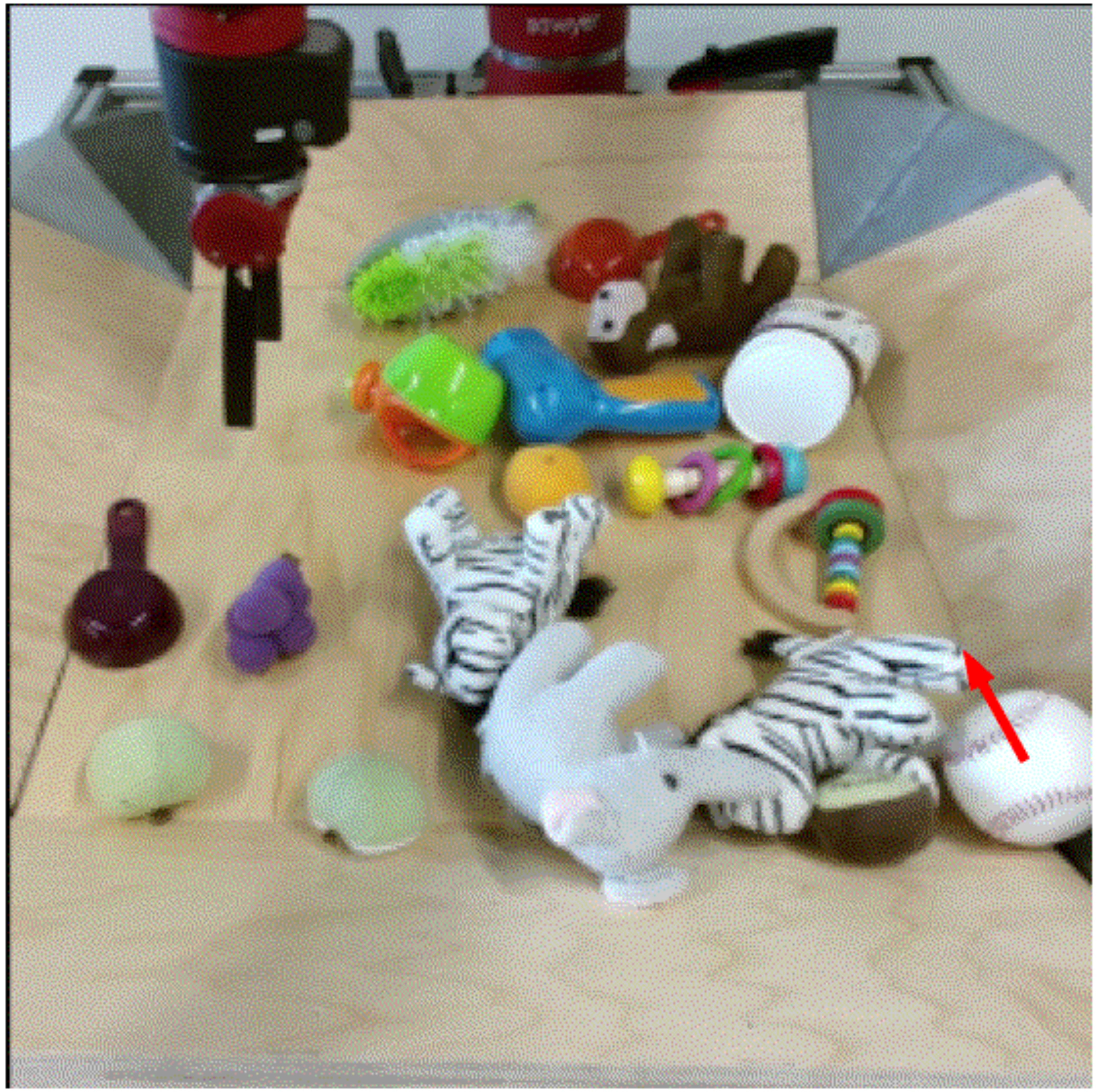
# Object Interactions via YODA*

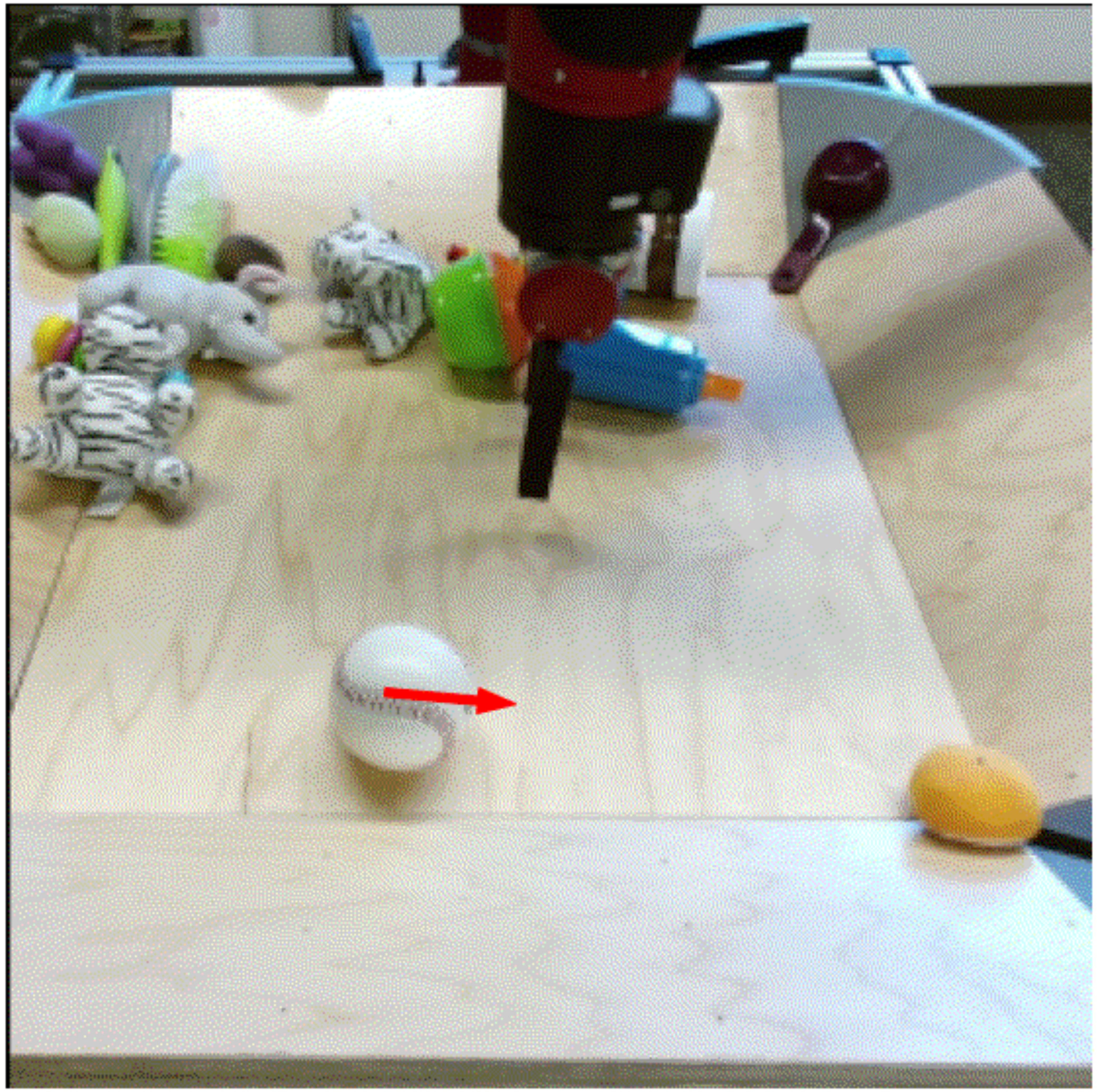




*Davtyan and Favaro, Learn the Force We Can: Multi-Object Video Generation from Pixel-Level Interactions, tech. report 2023

# Object Interactions via YODA*

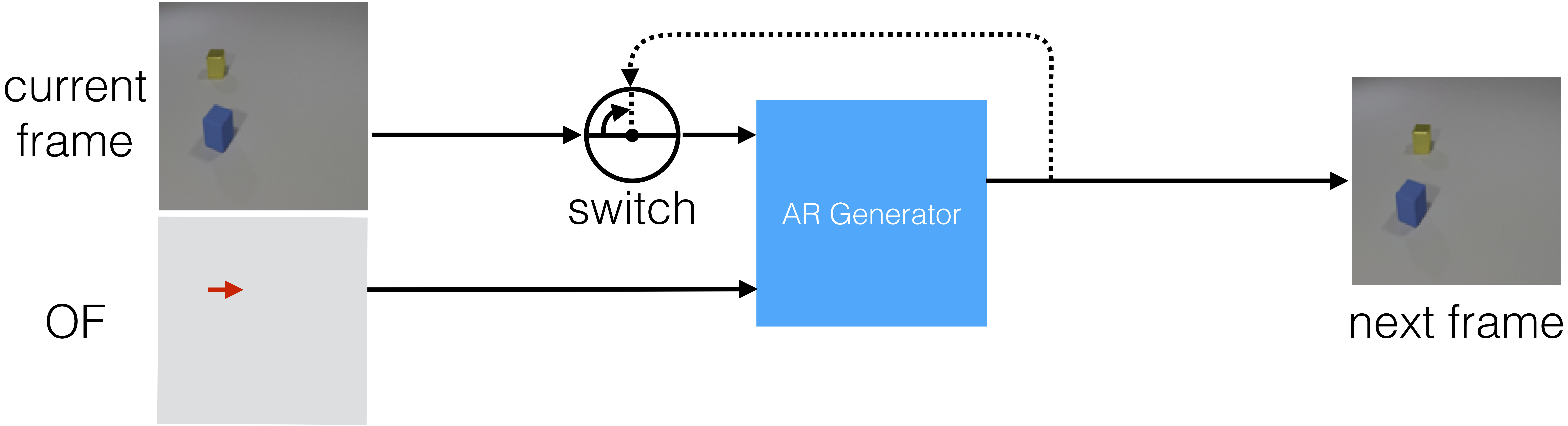




*Davtyan and Favaro,  Learn the Force We Can: Multi-Object Video Generation from Pixel-Level Interactions, tech. report 2023

# YODA

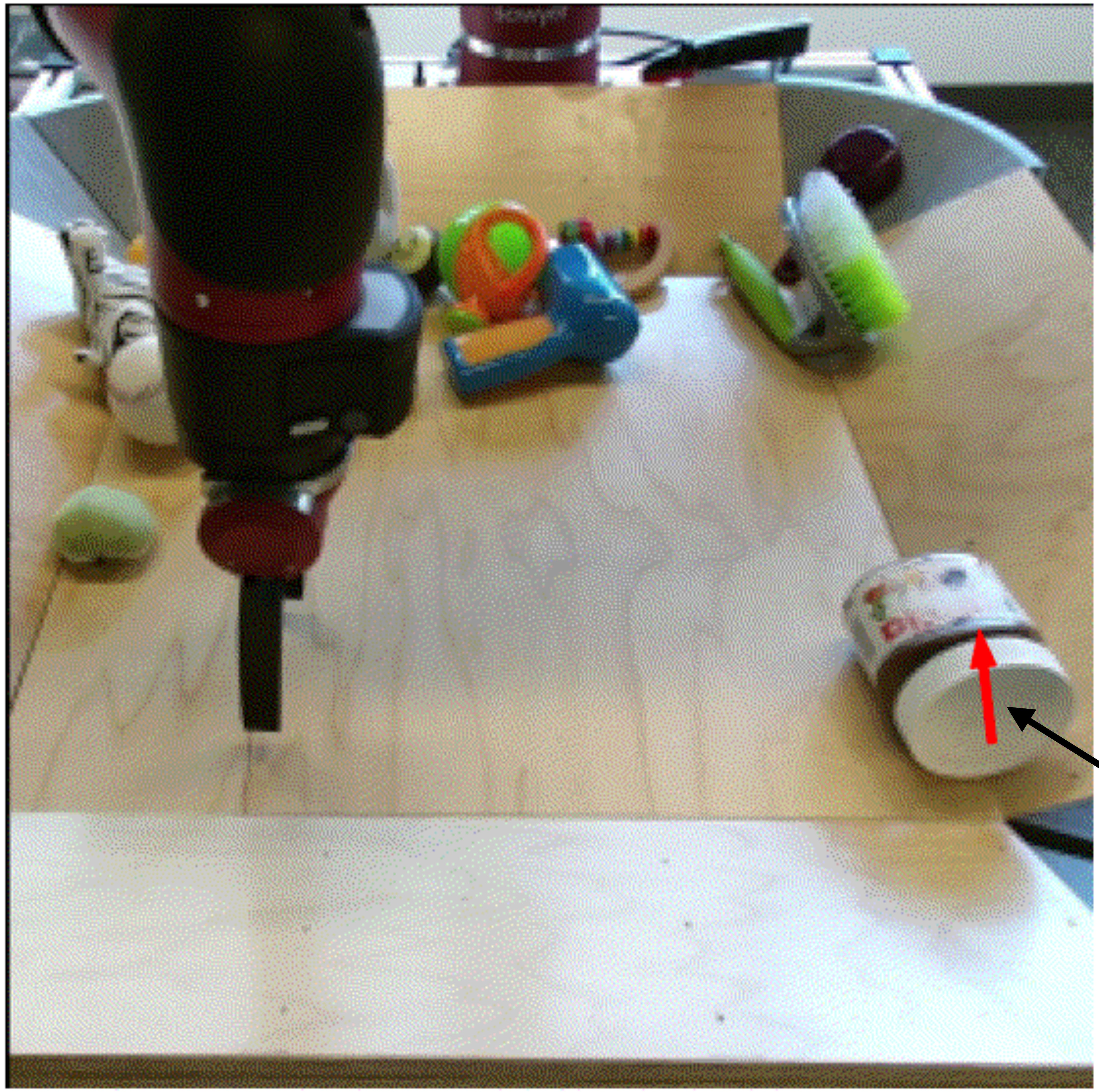
- **Step 1**: Train an auto-regressive generative model that outputs the next frame given the current frame and an encoding of optical flow




- **Step 2**: Use YODA to animate an image by editing the optical flow input

# Results

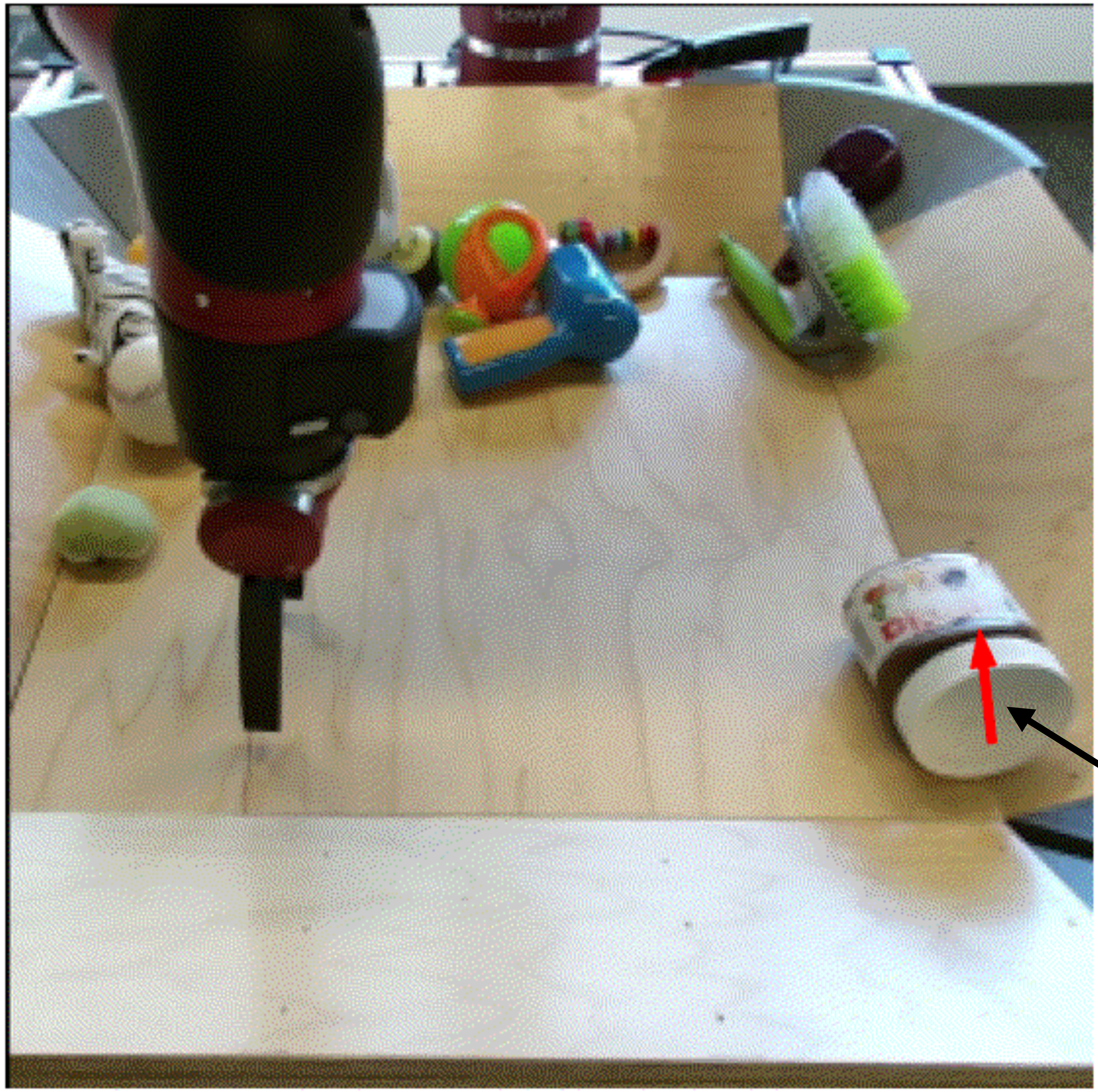
generated video from
a single input frame



optical flow input

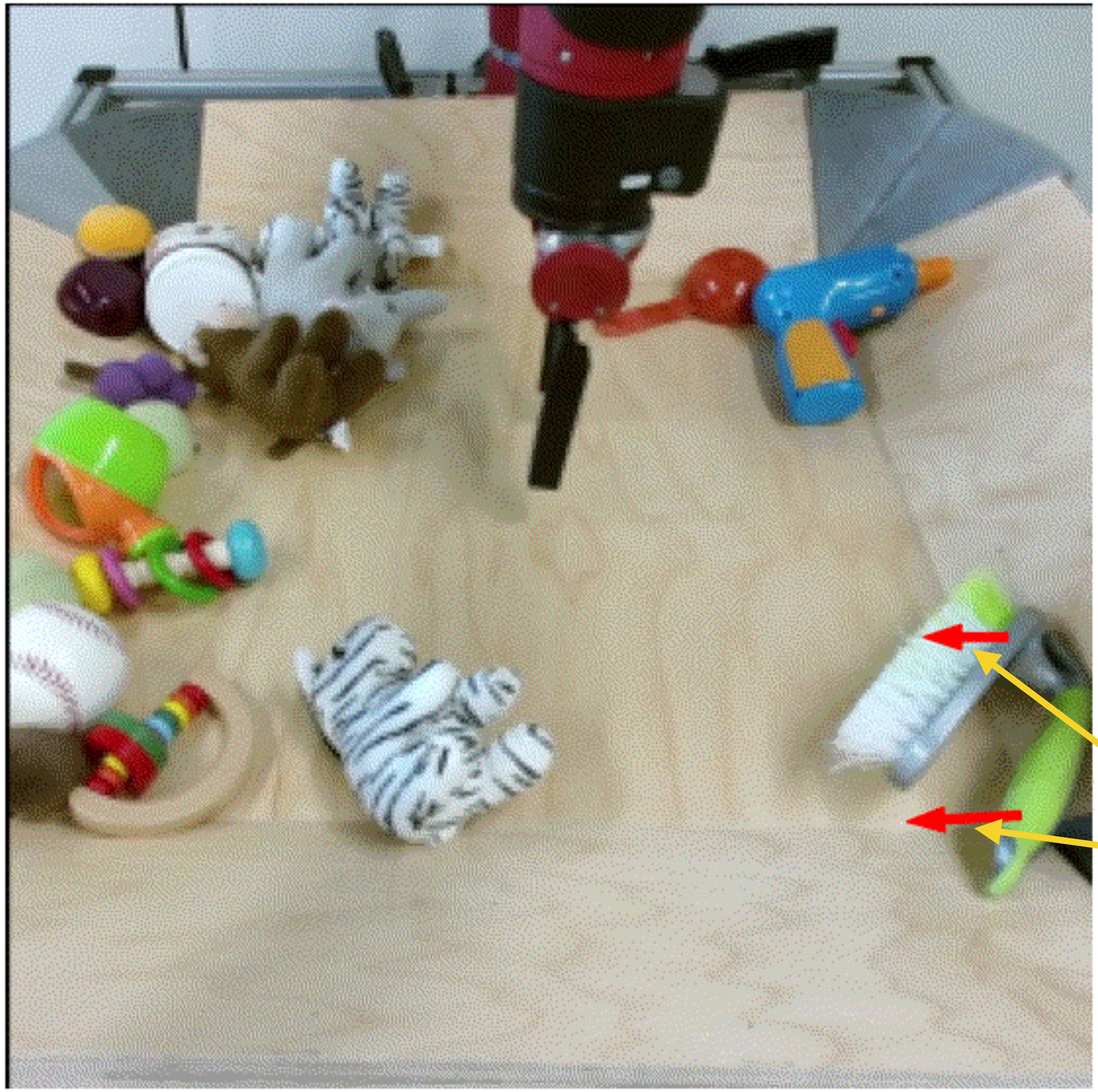
# Results

generated video from
a single input frame



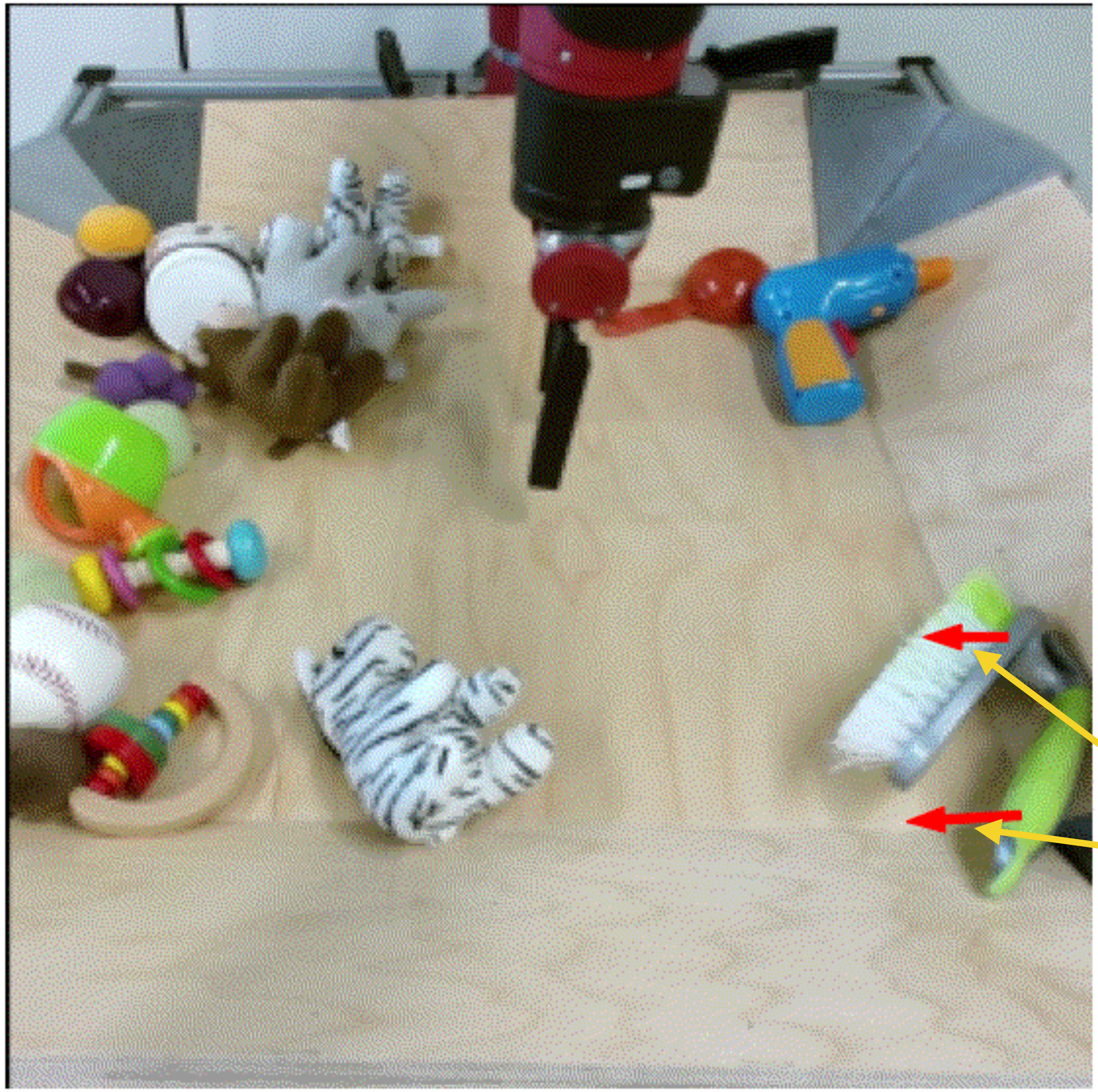
optical flow input

# Results



multiple optical flow inputs

# Results



multiple optical flow inputs

# Results

# Results

# Results

interaction shows an object pushes the other

interaction shows an object detaches from the other

# Results

interaction shows an object pushes the other

interaction shows an object detaches from the other

# Results

interaction shows an object pushes the other

interaction shows an object detaches from the other

# Results

interaction shows an object pushes the other

interaction shows an object detaches from the other

# Results

interaction shows an object pushes the other

interaction shows an object detaches from the other

# Results

interaction shows an object pushes the other

interaction shows an object detaches from the other

# Results

interaction shows an object pushes the other

interaction shows an object detaches from the other

# Sneak Preview

# Sneak Preview

# Sneak Preview

same initial frame but different inputs: result in different generated videos

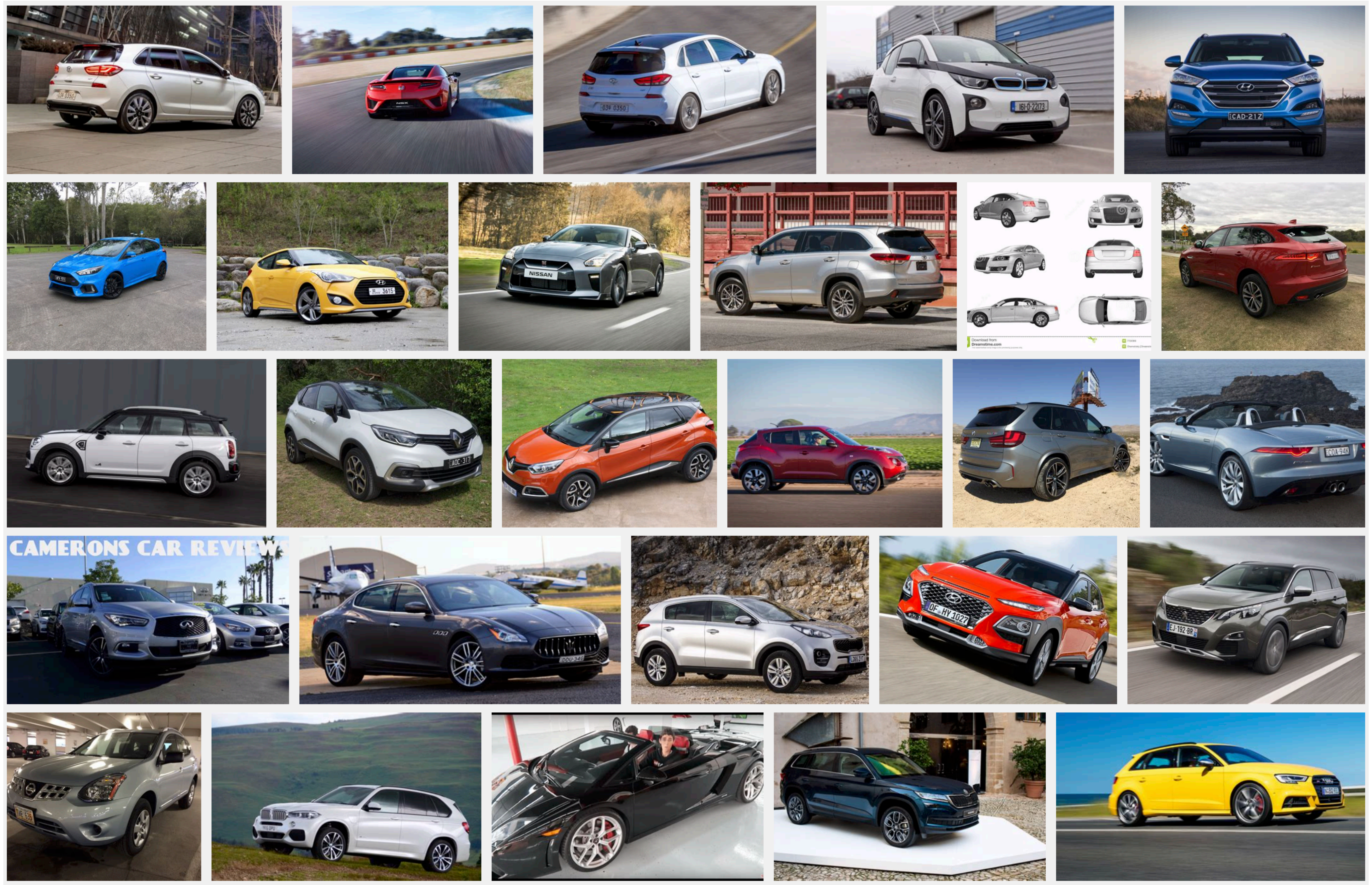
# Sneak Preview

same initial frame but different inputs: result in different generated videos

# Unsupervised learning of 3D shapes

# 3D in the wild

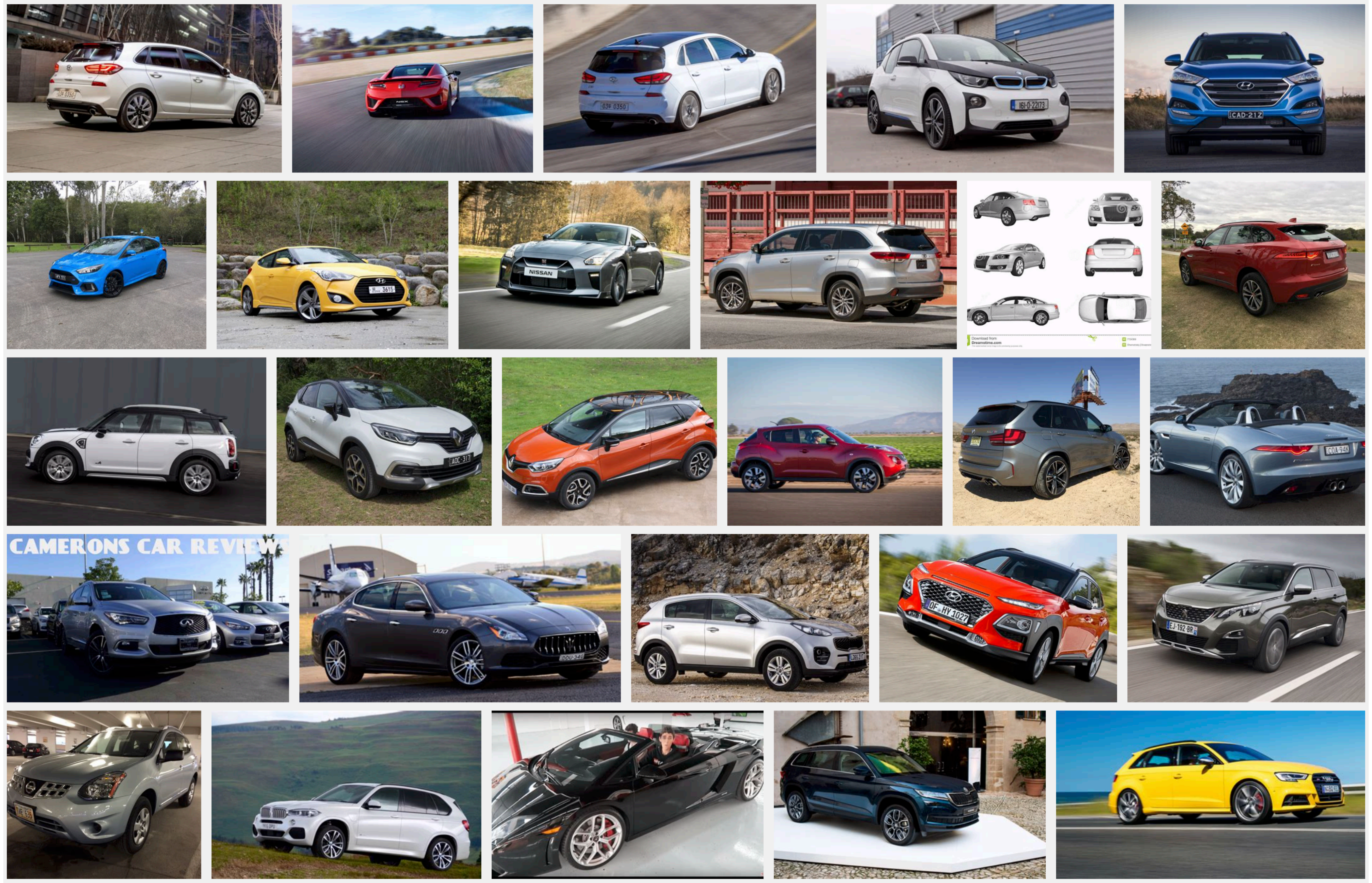
Given a dataset of real images **without**:
1) Multiple views of the same object instance
2) Annotation: no landmarks, no 3D templates, no viewpoints, no masks, etc




**Goal**
Learn to map 1 image with 1 object to its **3D**, **texture** and **viewpoint**

# 3D in the wild

Given a dataset of real images **without**:
1) Multiple views of the same object instance
2) Annotation: no landmarks, no 3D templates, no viewpoints, no masks, etc



**Goal**
Learn to map 1 image with 1 object to its **3D**, **texture** and **viewpoint**

**A first step**
Learn to map 1 image with 1 object to its **viewpoint**

# Unsupervised Viewpoint Estimation

compare images globally

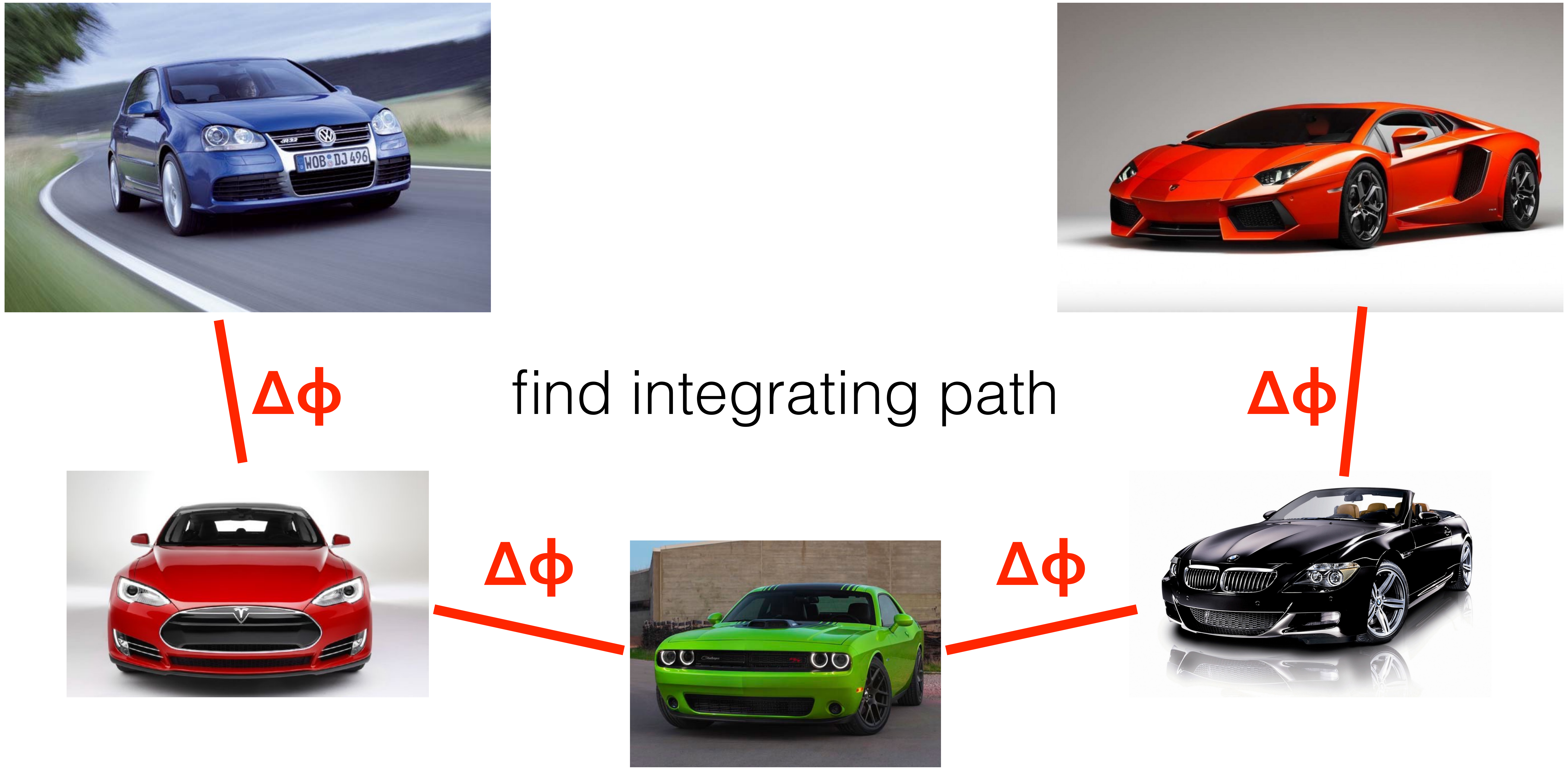
# Estimate Relative Viewpoints

$\Delta\phi$

estimate small
viewpoint changes

$\Delta\phi$

# Estimate Relative Viewpoints



# Results


A. Szabó, A.Vedaldi and P. Favaro, Building the View Graph of a Category by Exploiting Image Realism, ICCV Workshop, 2015

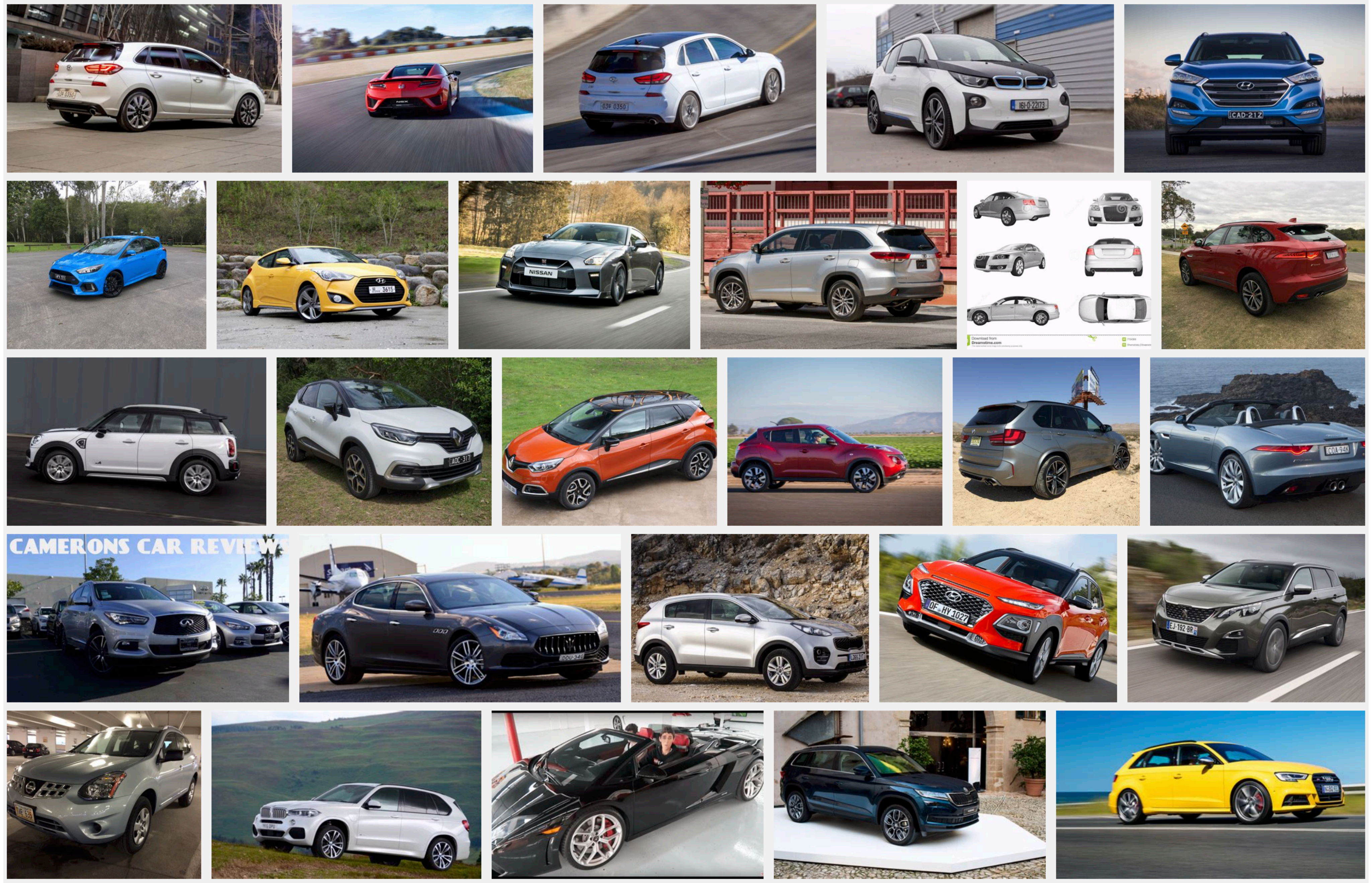
# Results


A. Szabó, A.Vedaldi and P. Favaro, Building the View Graph of a Category by Exploiting Image Realism, ICCV Workshop, 2015

# 3D in the wild

Given a dataset of real images **without**:

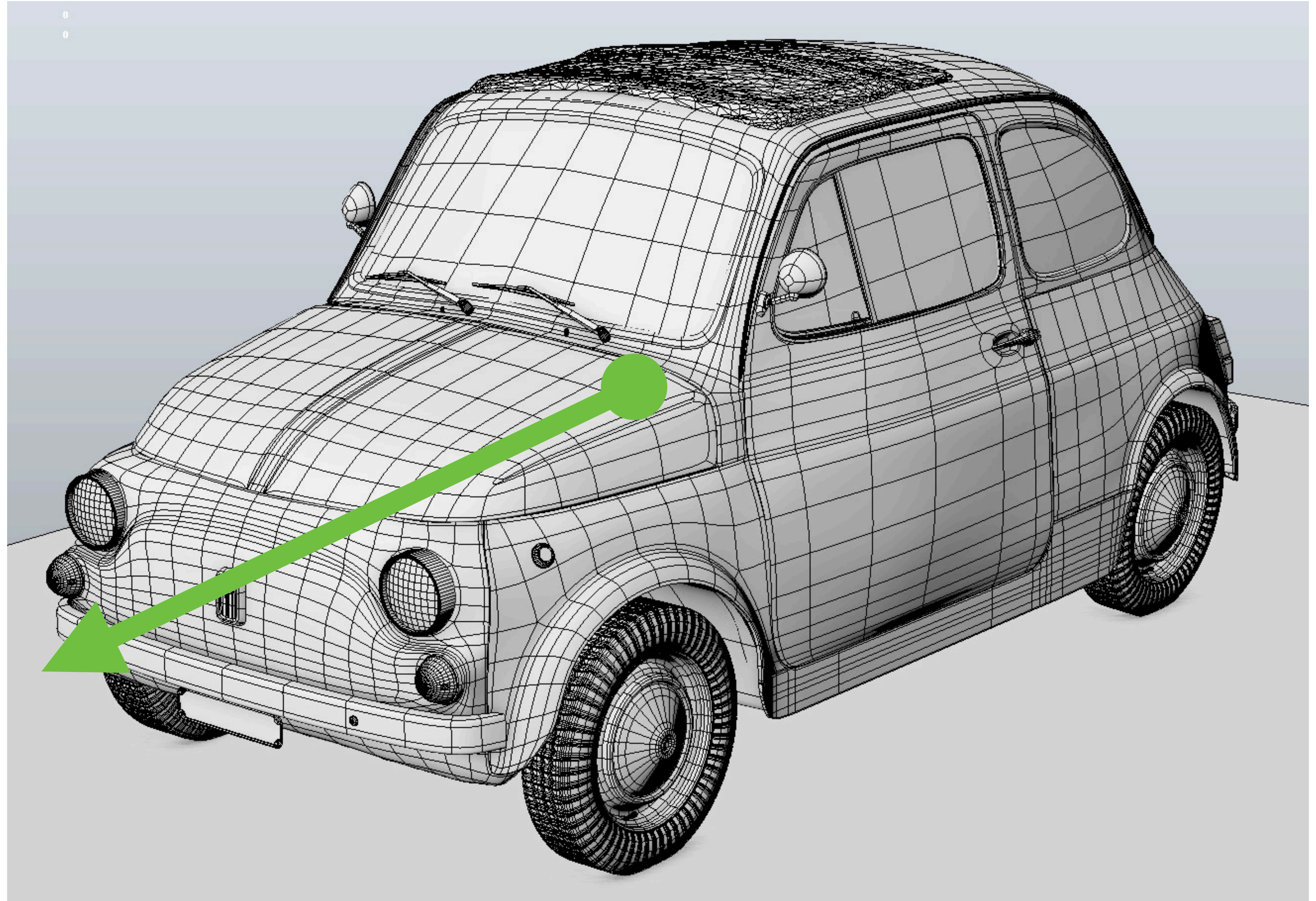
1) Multiple views of the same object instance
2) Annotation: no landmarks, no 3D templates, no viewpoints, no masks, etc




**Goal**
Learn to map 1 image with 1 object to its **3D**, **texture** and **viewpoint**

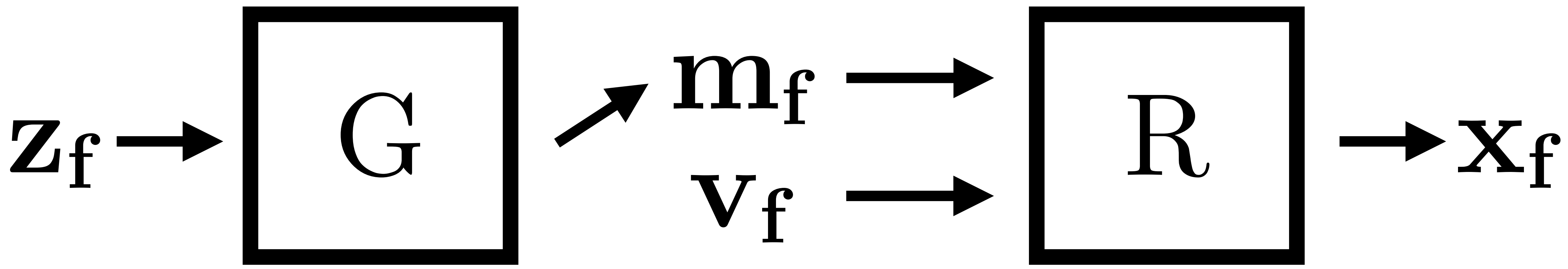
# Unsupervised Learning of 3D from an Uncurated Image Collection

Map 1 image with 1 object to its 3D, texture and viewpoint

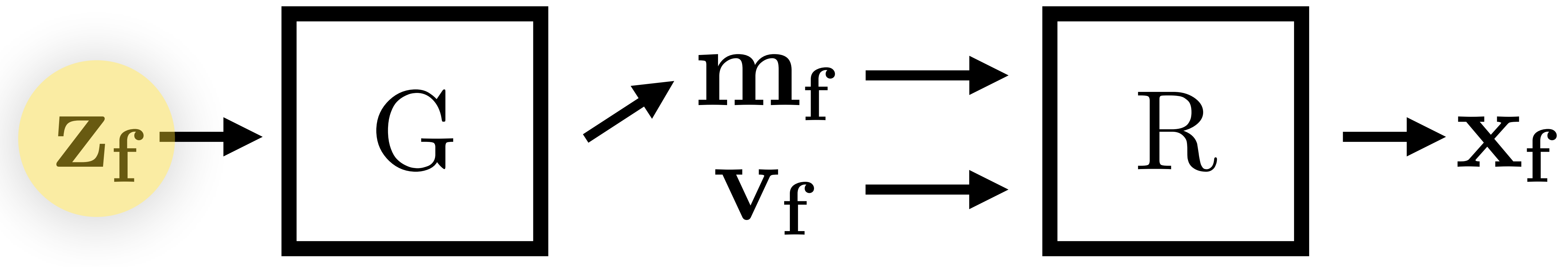
# A 3D Generative Model

- The generator G generates 3D, texture and background
- We render a view via a **differentiable renderer** from a **random viewpoint**
- It should look **realistic**



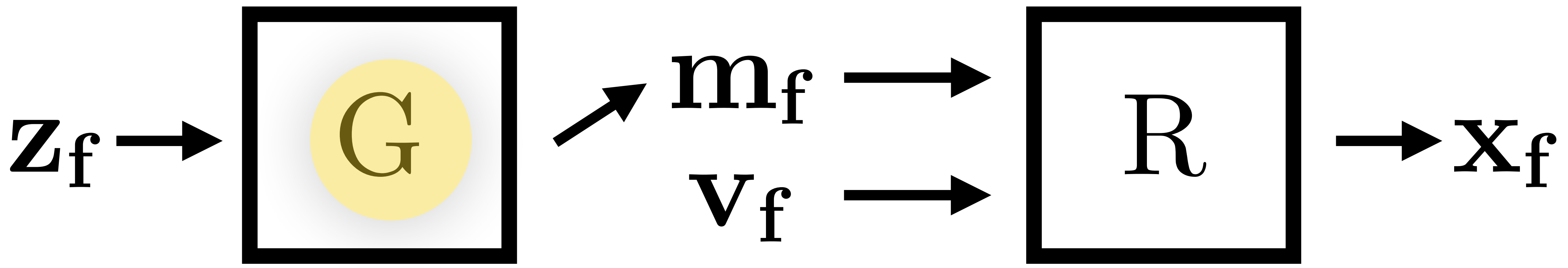
# A 3D Generative Model

- The generator G generates 3D, texture and background
- We render a view via a **differentiable renderer** from a **random viewpoint**
- It should look **realistic**



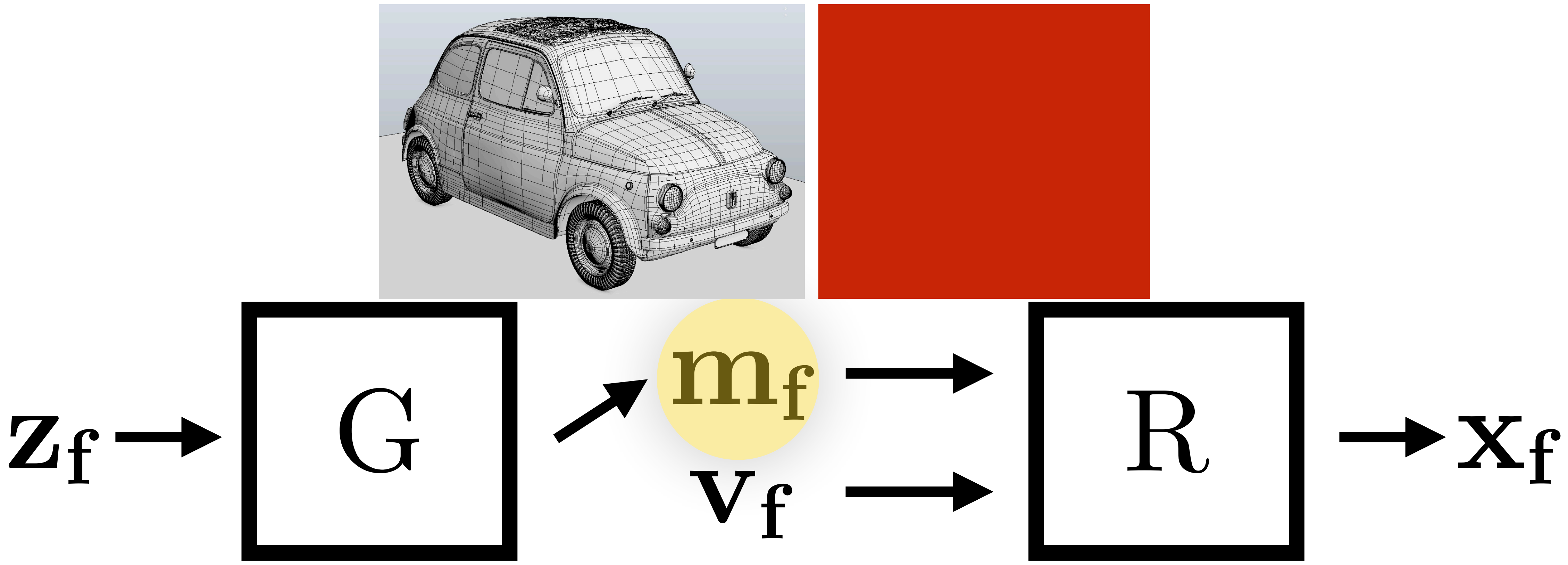
# A 3D Generative Model

- The generator G generates 3D, texture and background
- We render a view via a **differentiable renderer** from a **random viewpoint**
- It should look **realistic**



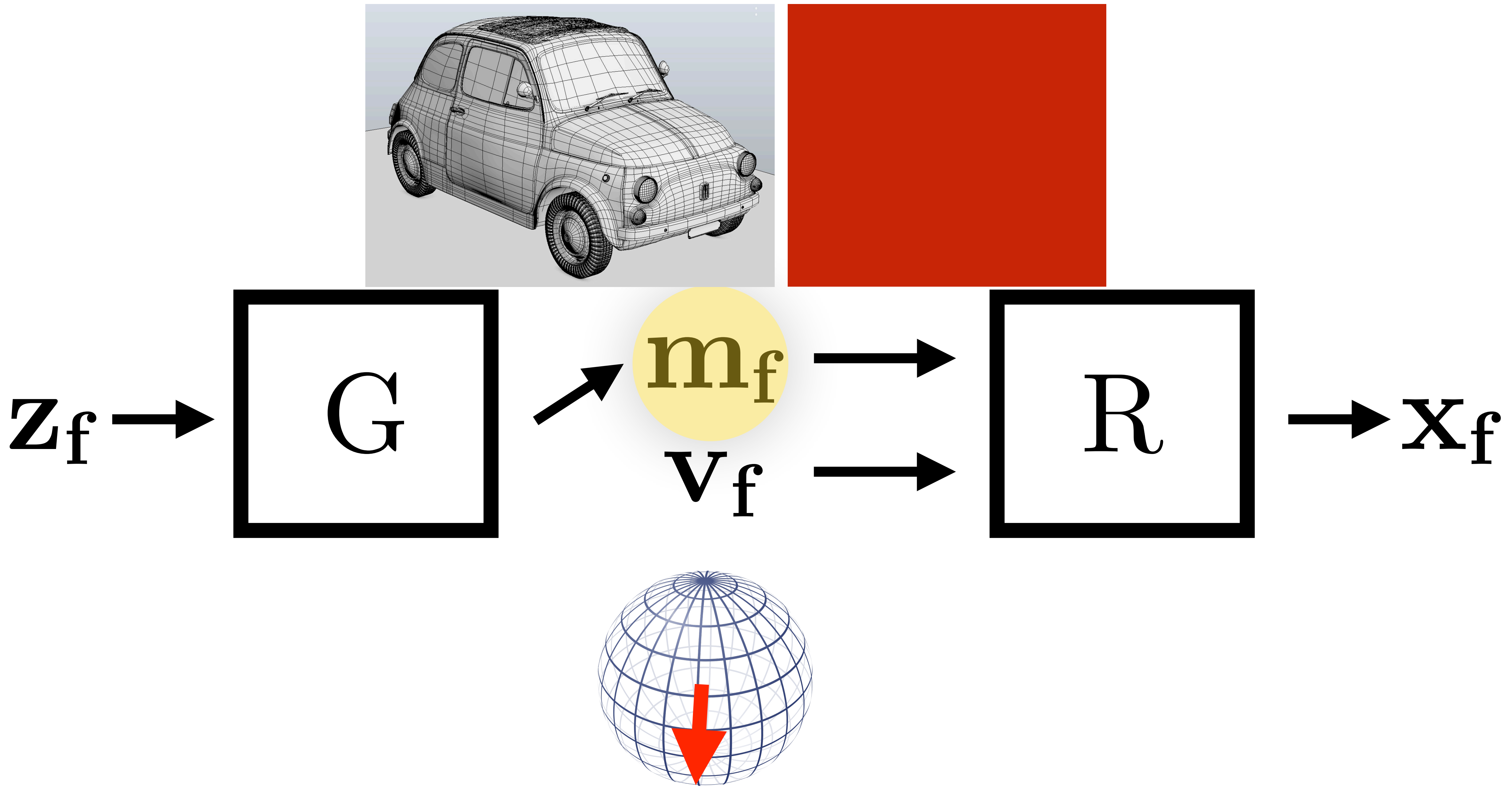
# A 3D Generative Model

- The generator G generates 3D, texture and background
- We render a view via a **differentiable renderer** from a **random viewpoint**
- It should look **realistic**



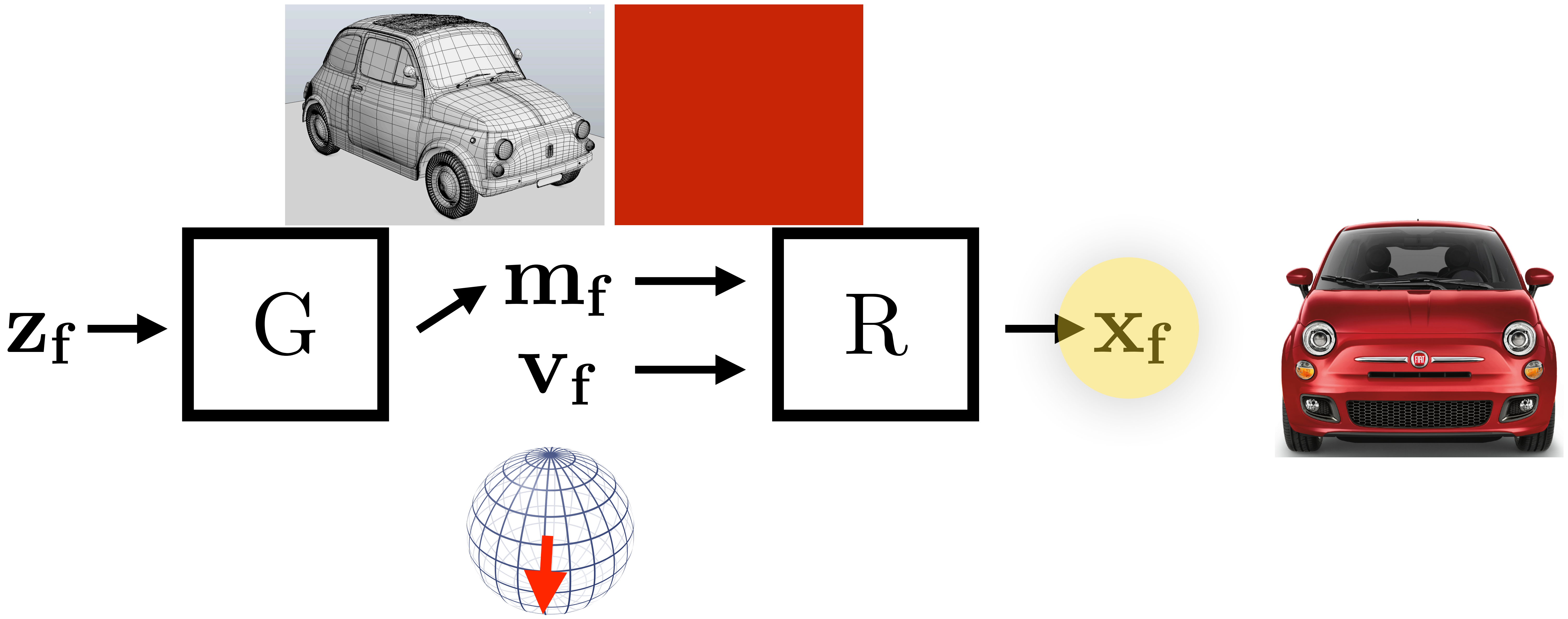
# A 3D Generative Model

- The generator G generates 3D, texture and background
- We render a view via a **differentiable renderer** from a **random viewpoint**
- It should look **realistic**



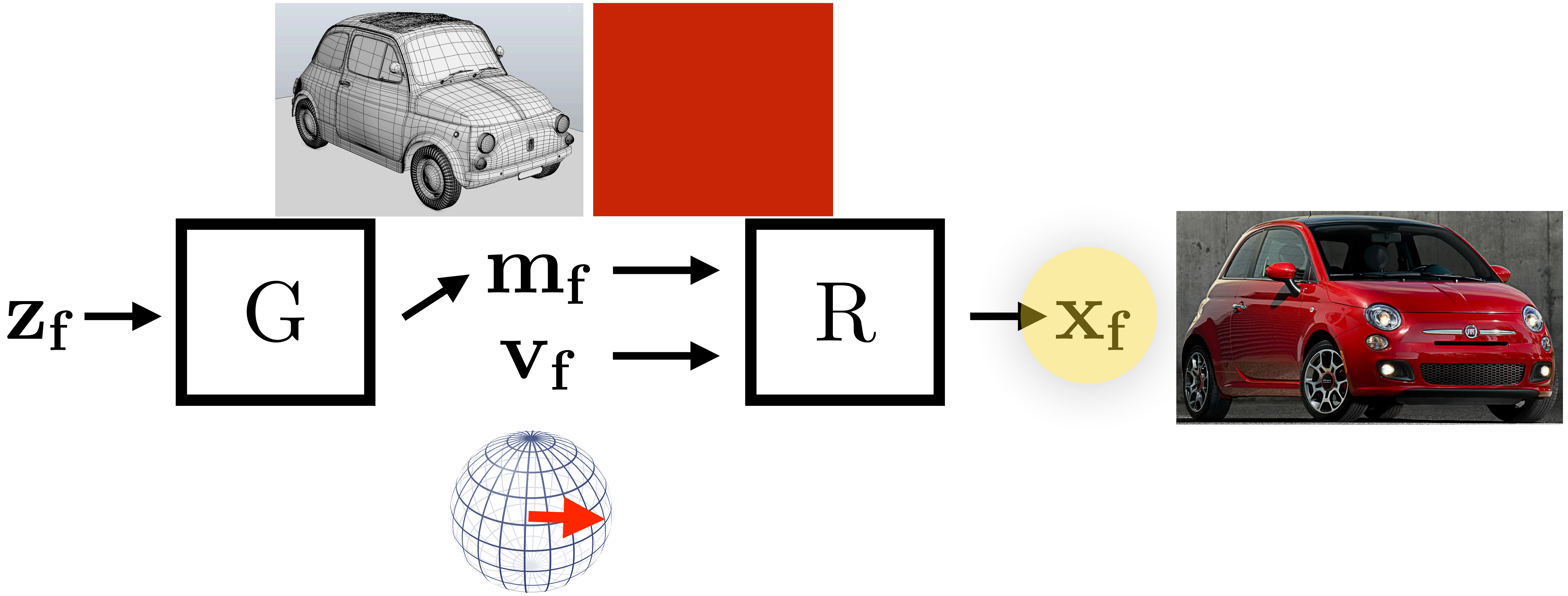
# A 3D Generative Model

- The generator G generates 3D, texture and background
- We render a view via a **differentiable renderer** from a **random viewpoint**
- It should look **realistic**



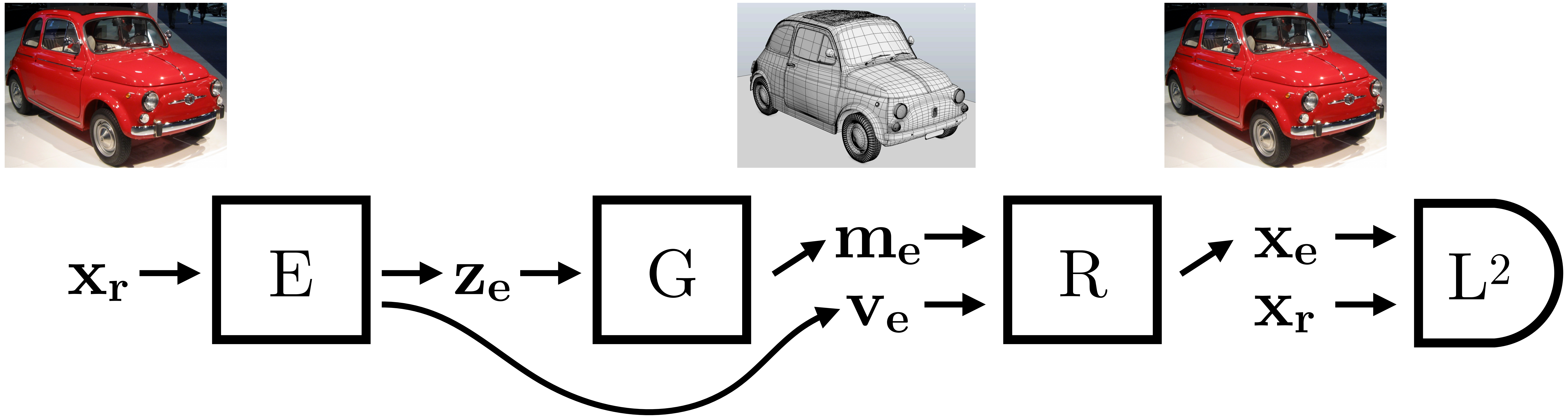
# A 3D Generative Model

- The generator G generates 3D, texture and background
- We render a view via a **differentiable renderer** from a **random viewpoint**
- It should look **realistic**



# Mapping Images to 3D and Pose

- Combine an encoder with the previous generator to autoencode images



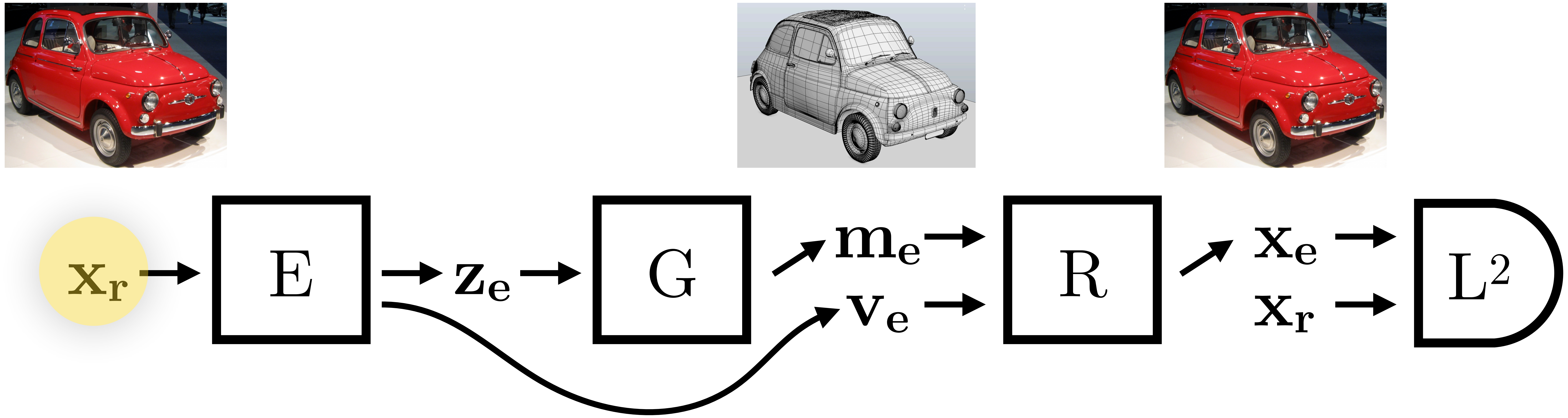

- Encoder learns to map images to their 3D, texture, pose and background

Szabó and Favaro, "Unsupervised 3D Shape Learning from Image Collections in the Wild", arXiv 2018

Szabó et al, Unsupervised Generative 3D Shape Learning from Natural Images, arXiv 2019

# Mapping Images to 3D and Pose

- Combine an encoder with the previous generator to autoencode images



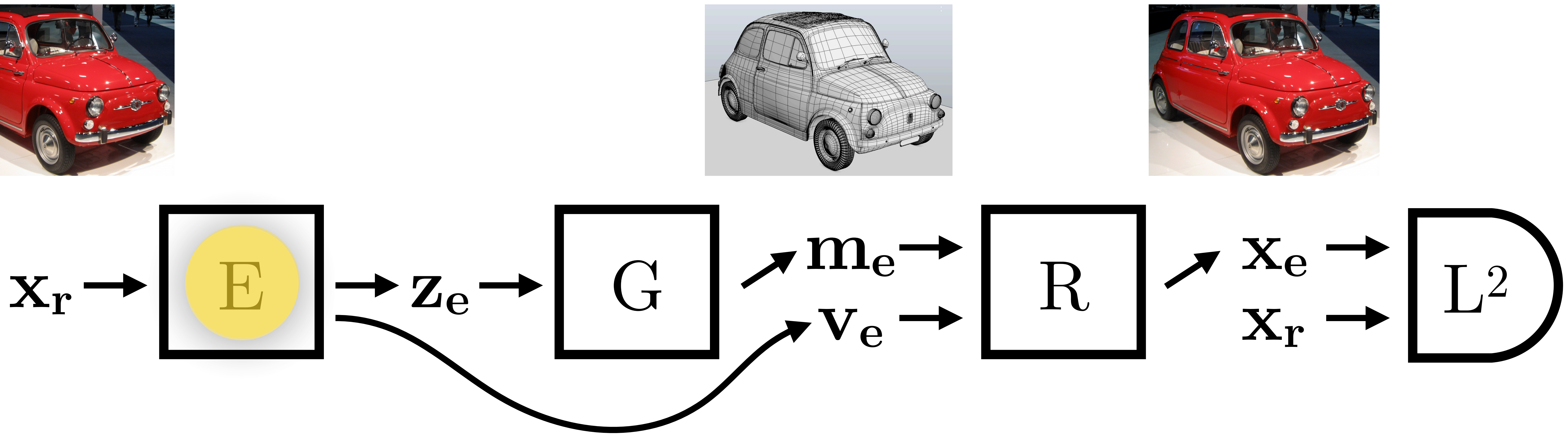

- Encoder learns to map images to their 3D, texture, pose and background

Szabó and Favaro, “Unsupervised 3D Shape Learning from Image Collections in the Wild”, arXiv 2018

Szabó et al, Unsupervised Generative 3D Shape Learning from Natural Images, arXiv 2019

# Mapping Images to 3D and Pose

- Combine an encoder with the previous generator to autoencode images

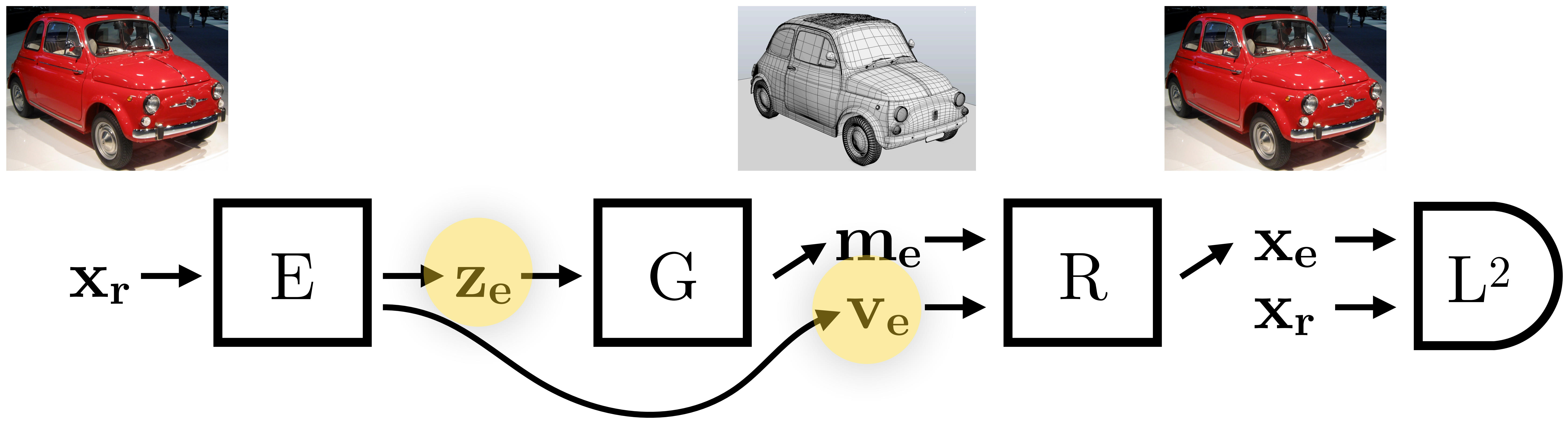



- Encoder learns to map images to their 3D, texture, pose and background

Szabó and Favaro, "Unsupervised 3D Shape Learning from Image Collections in the Wild", arXiv 2018
Szabó et al, Unsupervised Generative 3D Shape Learning from Natural Images, arXiv 2019

# Mapping Images to 3D and Pose

- Combine an encoder with the previous generator to autoencode images



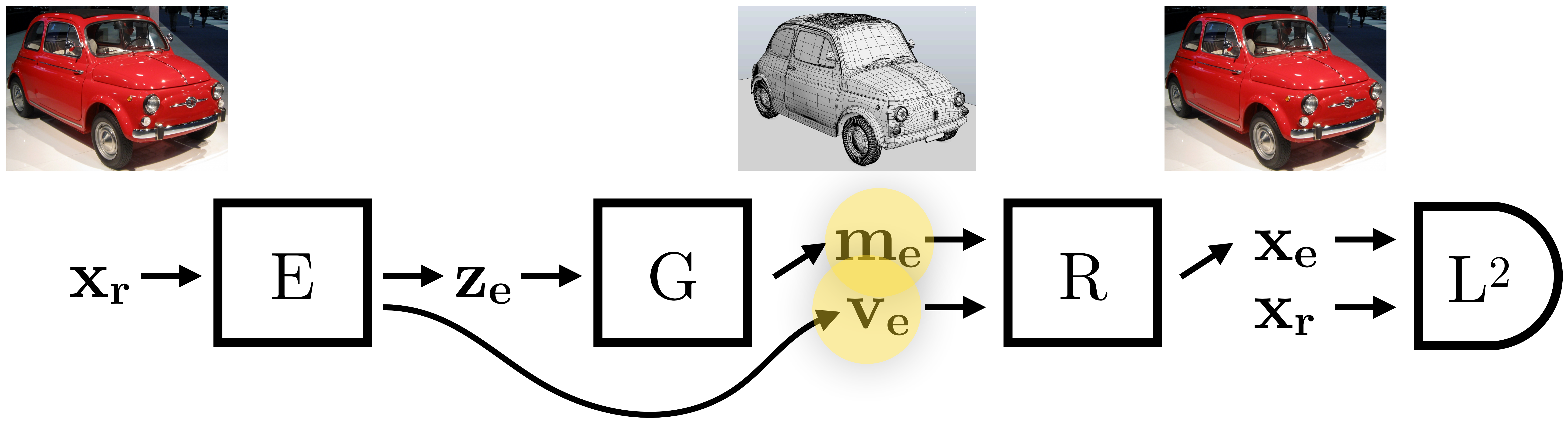

- Encoder learns to map images to their 3D, texture, pose and background

Szabó and Favaro, "Unsupervised 3D Shape Learning from Image Collections in the Wild", arXiv 2018

Szabó et al, Unsupervised Generative 3D Shape Learning from Natural Images, arXiv 2019

# Mapping Images to 3D and Pose

- Combine an encoder with the previous generator to autoencode images



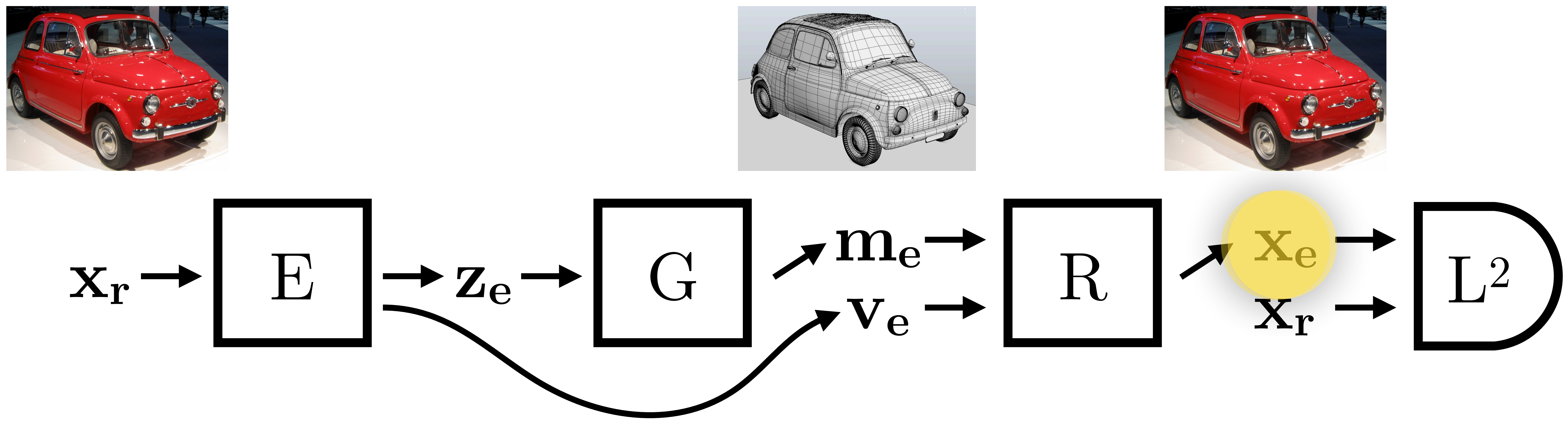

- Encoder learns to map images to their 3D, texture, pose and background

Szabó and Favaro, “Unsupervised 3D Shape Learning from Image Collections in the Wild”, arXiv 2018

Szabó et al, Unsupervised Generative 3D Shape Learning from Natural Images, arXiv 2019

# Mapping Images to 3D and Pose

- Combine an encoder with the previous generator to autoencode images




- Encoder learns to map images to their 3D, texture, pose and background

Szabó and Favaro, "Unsupervised 3D Shape Learning from Image Collections in the Wild", arXiv 2018

Szabó et al, Unsupervised Generative 3D Shape Learning from Natural Images, arXiv 2019

# Generative Model on CelebA

output image | 3D | texture | backgr. | views without background

# Generative Model on CelebA

generated image | generated 3D | generated texture | generated background | generated viewpoints

# Autoencoder on CelebA

input rec. 3D texture backgr. views without background

# Conclusions

- Unsupervised learning allows scaling and possibly a better generalization
- Poses lots of interesting and challenging problems
- It forces a drastic change in how problems are solved
- In my view a key building block for machines that learn by themselves