

FastJAM: a Fast Joint Alignment Model for Images



Omri Hirsch*



Ron Shapira Weber*



Shira Ifergane



Oren Freifeld

Presented By:

Omri Hirsch

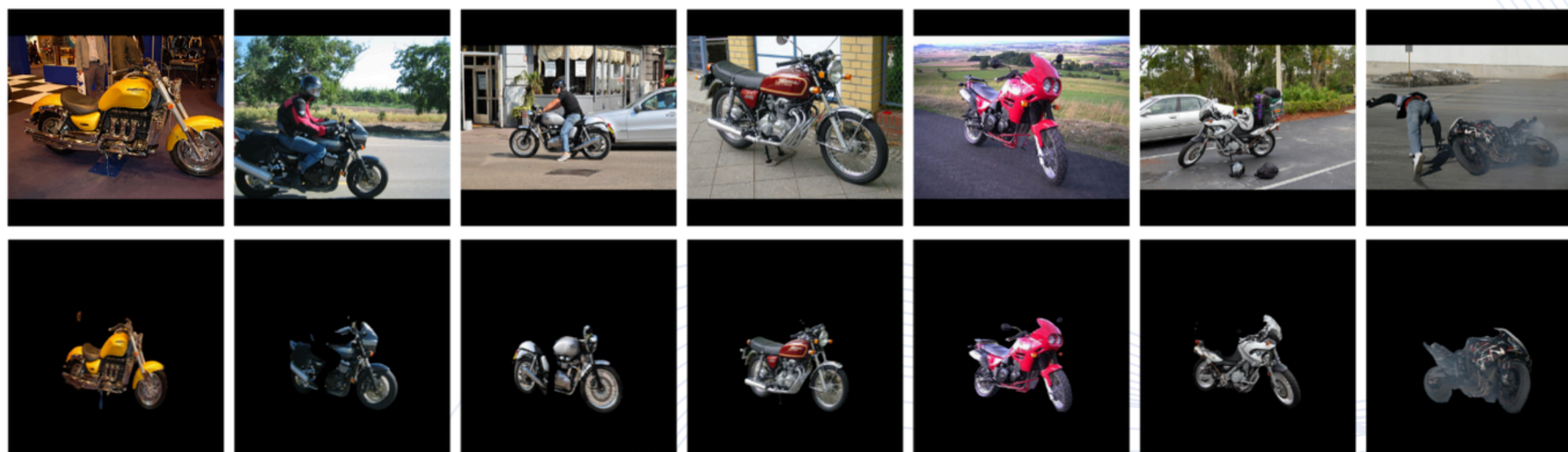


FastJAM's Website

• <https://bgu-cs-vil.github.io/FastJAM/>

What is Joint Alignment (JA)?

Joint Alignment by FastJAM



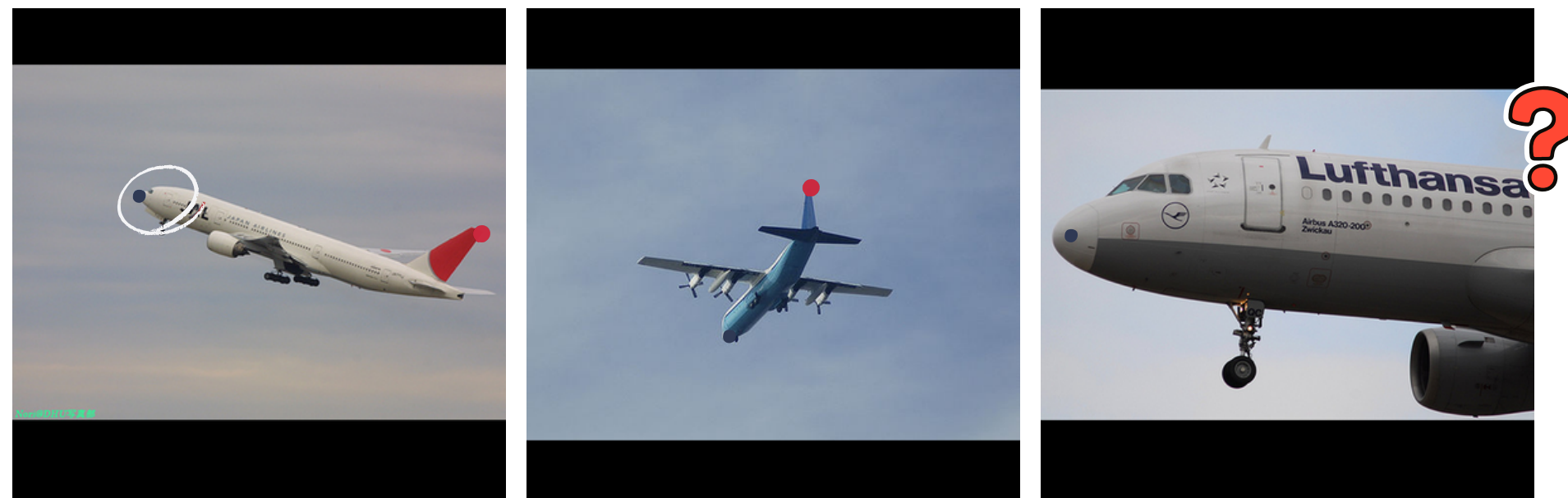
Motivation

Joint Alignment is crucial for many vision tasks

01 Pairwise = Error Accumulation

02 Inefficiency of Existing Joint Alignment Methods

03 JA based on Dense Feature Losses



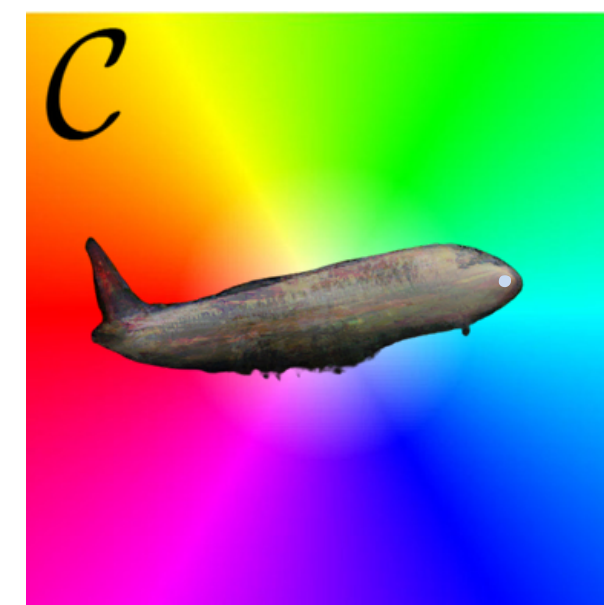
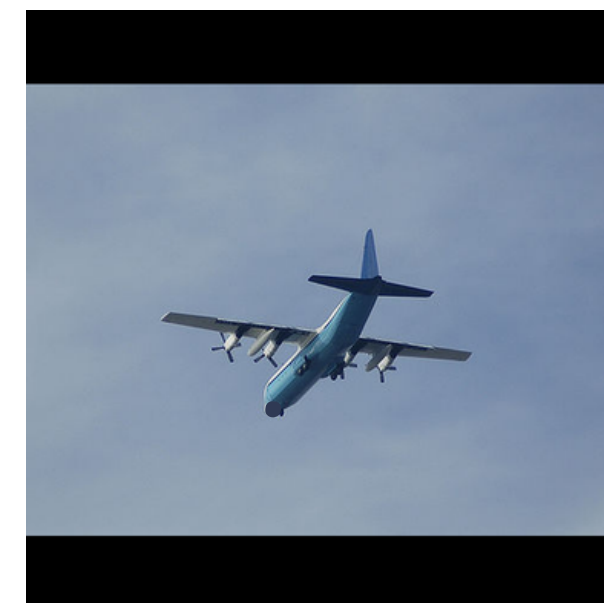
Motivation

Joint Alignment is crucial for many vision tasks

01 Pairwise = Error Accumulation

02 Inefficiency of Existing Joint Alignment Methods

03 JA Based on Dense Feature Losses



Motivation

Joint Alignment is crucial for many vision tasks

01 Pairwise = Error Accumulation

02 Inefficiency of Existing Joint Alignment Methods

03 JA Based on Dense Feature Losses



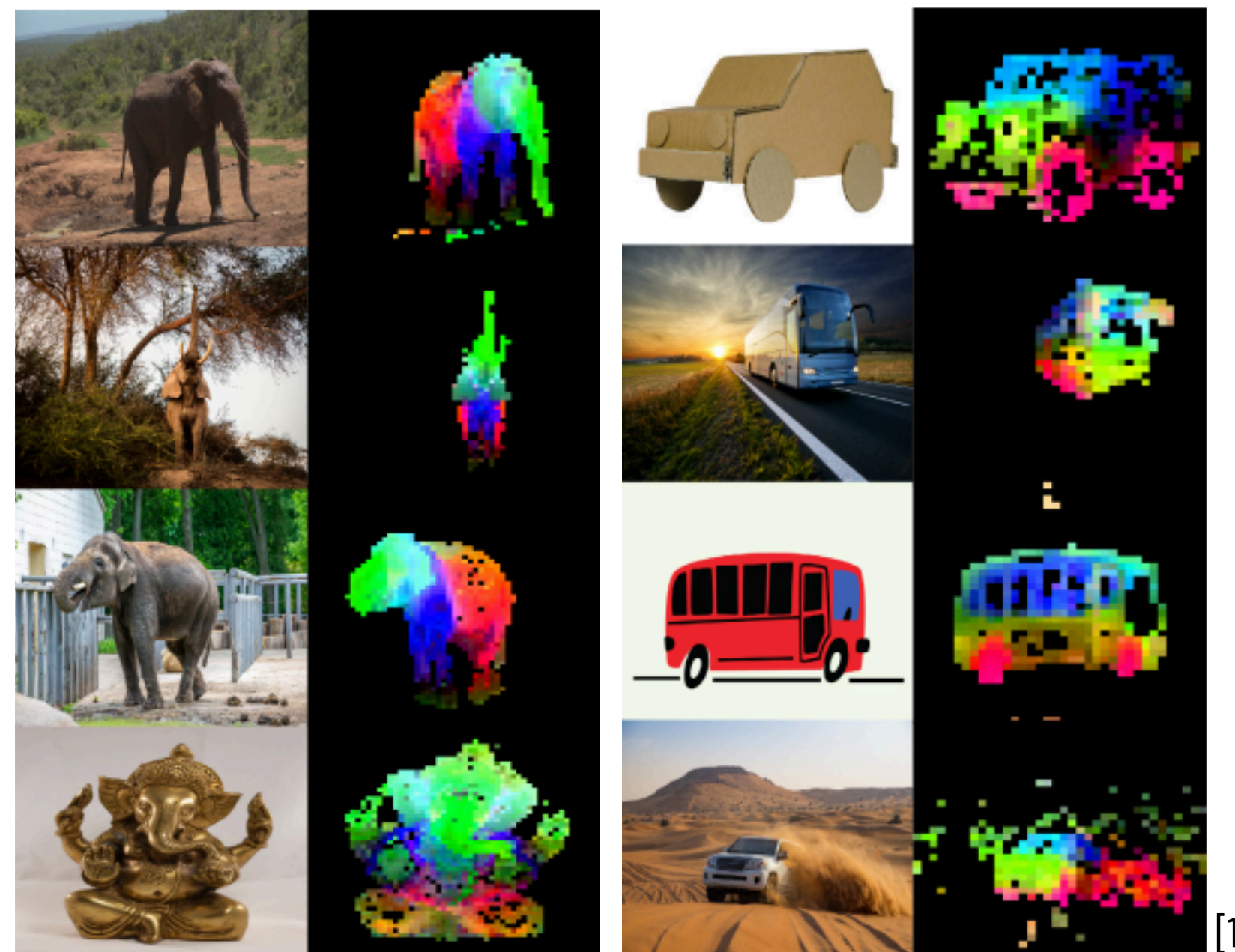
Motivation

Joint Alignment is crucial for many vision tasks

01 Pairwise = Error Accumulation

02 Inefficiency of Existing Joint Alignment Methods

03 JA Based on Dense Feature Losses



[1] Oquab et al., TMLR 2024

Challenges



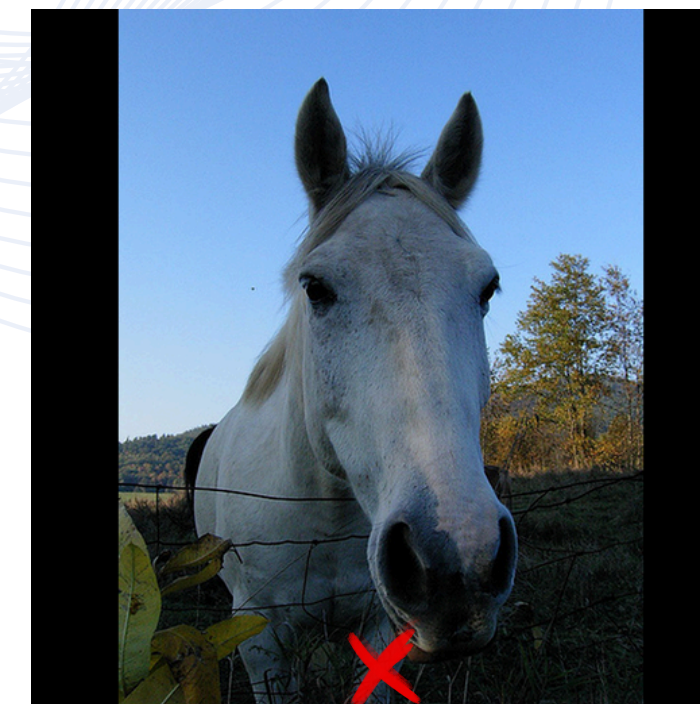
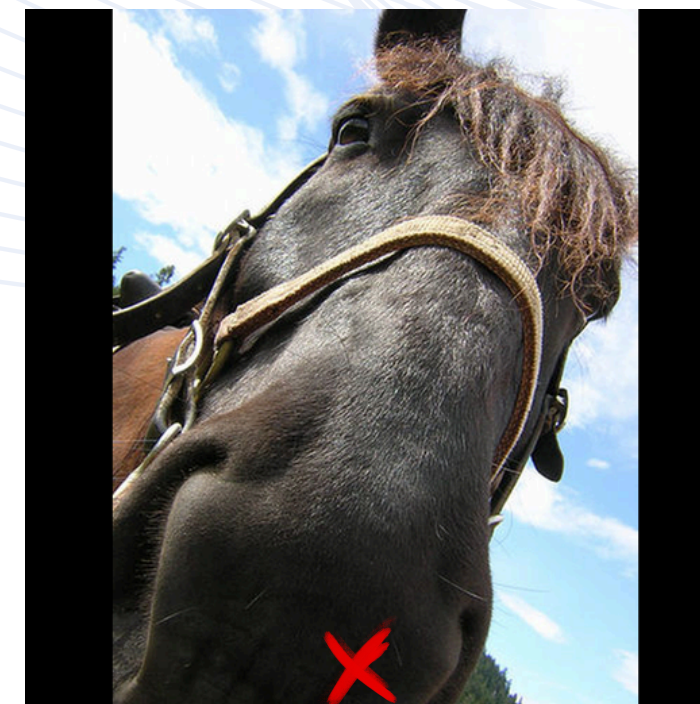
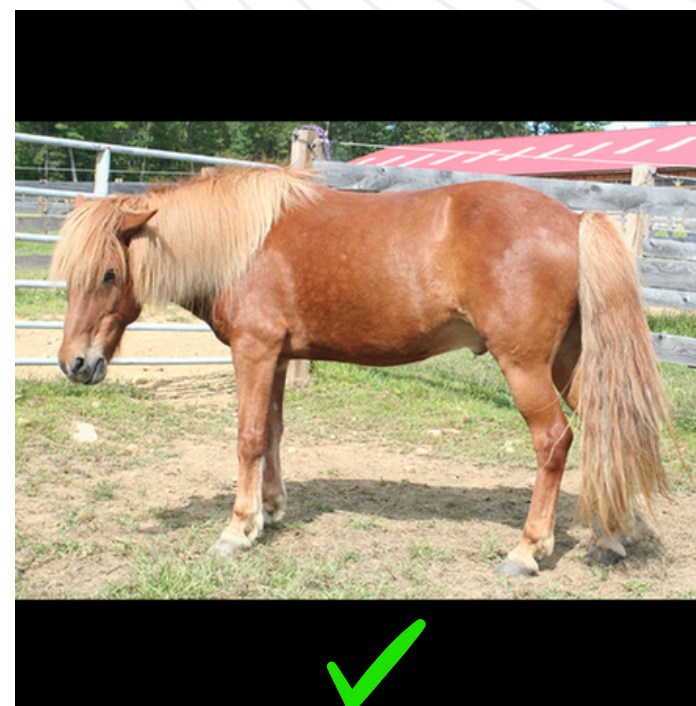
Unsupervised: 01

No "Known" Reference



Regularization: 02

Due to #1, Usually Previous
Methods Require Regularization



Cross Instance: 03

Cross Instance JA is Even
More Difficult

Challenges



Unsupervised: 01

No "Known" Reference



Regularization: 02

Due to #1, Usually Previous
Methods Require Regularization



Cross Instance: 03

Cross Instance JA is Even
More Difficult



[1]

[1] Barel et al., ECCV 2024

Challenges



Unsupervised: 01

No "Known" Reference



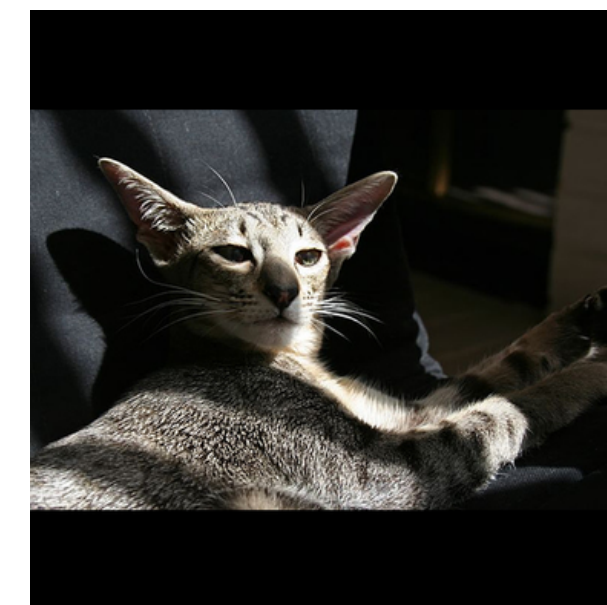
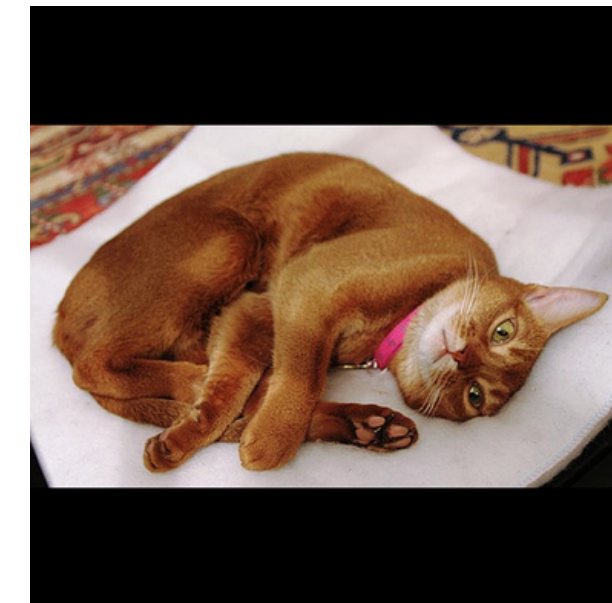
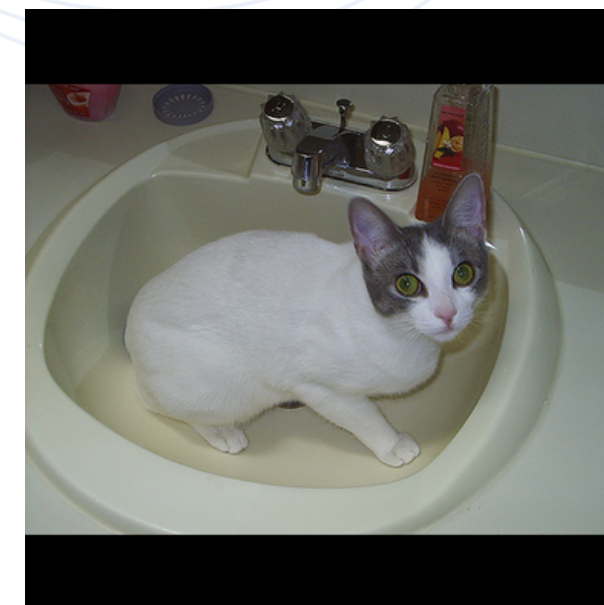
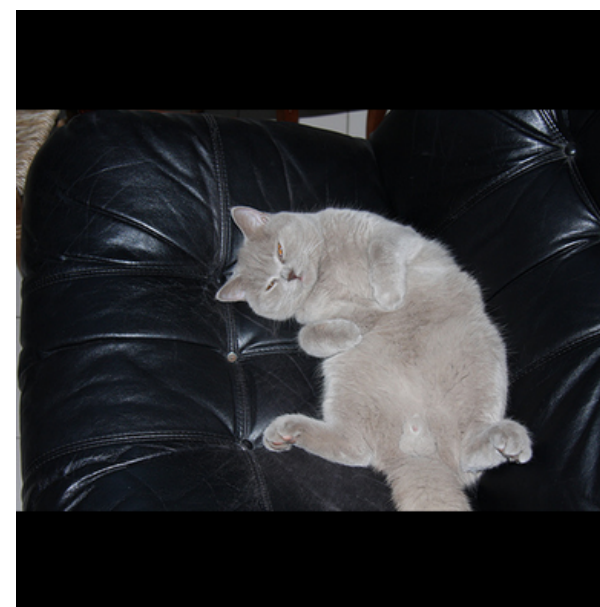
Regularization: 02

Due to #1, Usually Previous
Methods Require Regularization

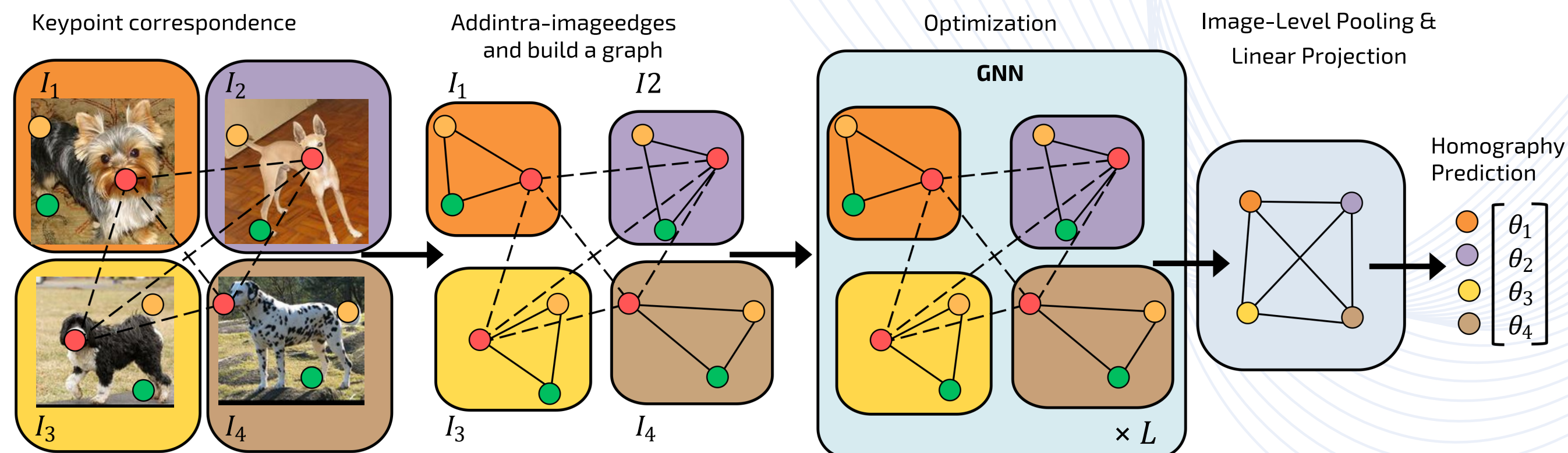


Cross Instance: 03

Cross Instance JA is Even
More Difficult



FastJAM Architecture



INPUT:
Keypoint
Correspondence
Graph



OUTPUT:
Homographies
For Each Image

Legend:

— Intra-image edge

- - - Inter-image edge

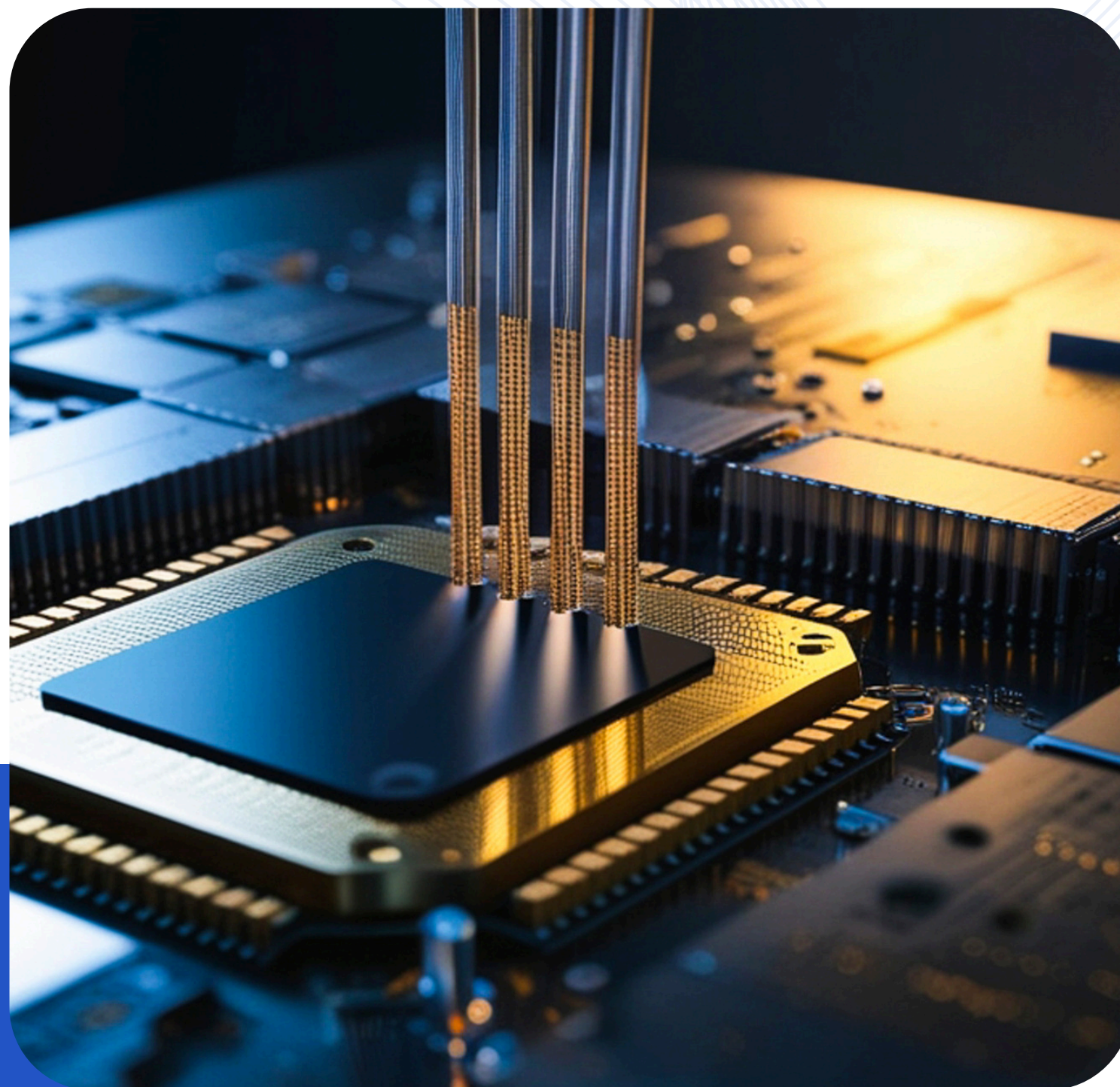
○ ○ ○ Corresponding Keypoints Between Images

Runtime Comparison

FastJAM runtime is in **Seconds** Compared to Minutes or Hours (hh:mm:ss)

→ Neural Congealing	01:18:30 ± 00:06:18
→ ASIC	01:06:38 ± 00:00:38
→ SpaceJAM	00:06:00 ± 00:00:12
→ FastJAM (Ours)	00:00:49 ± 00:00:04

*For the Full Table Please Check Our Paper



Quantitative Results

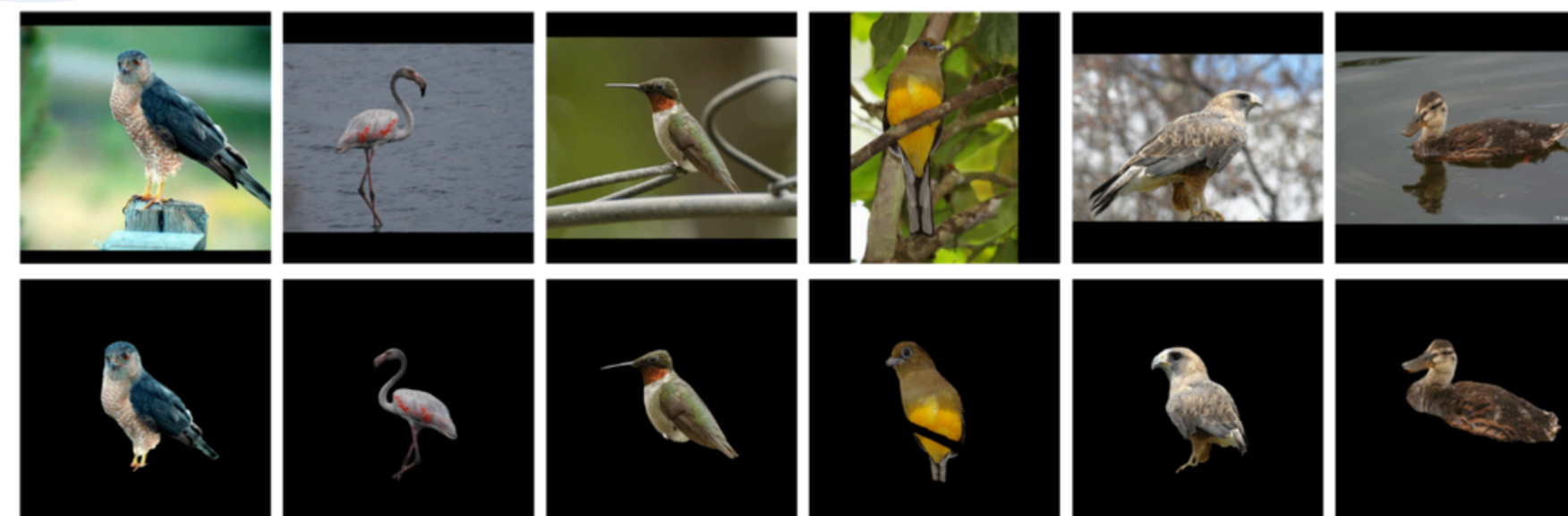
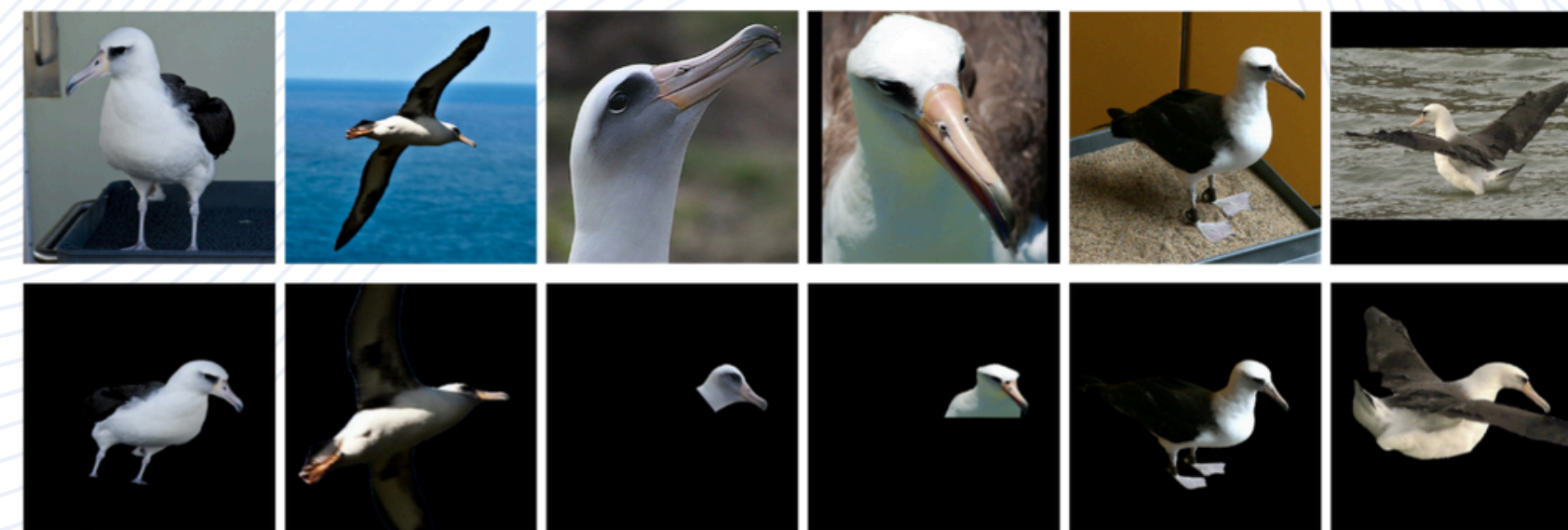
FastJAM gets either **better** or comparable **results**.
Based on **PCK**

	SPair-71K Avg. All Classes	CUB-200 Subsets
→ Neural Congealing	–	63.6
→ ASIC	37.0	–
→ SpaceJAM	<u>45.7</u>	<u>69.9</u>
→ FastJAM (Ours)	53.4	73.6

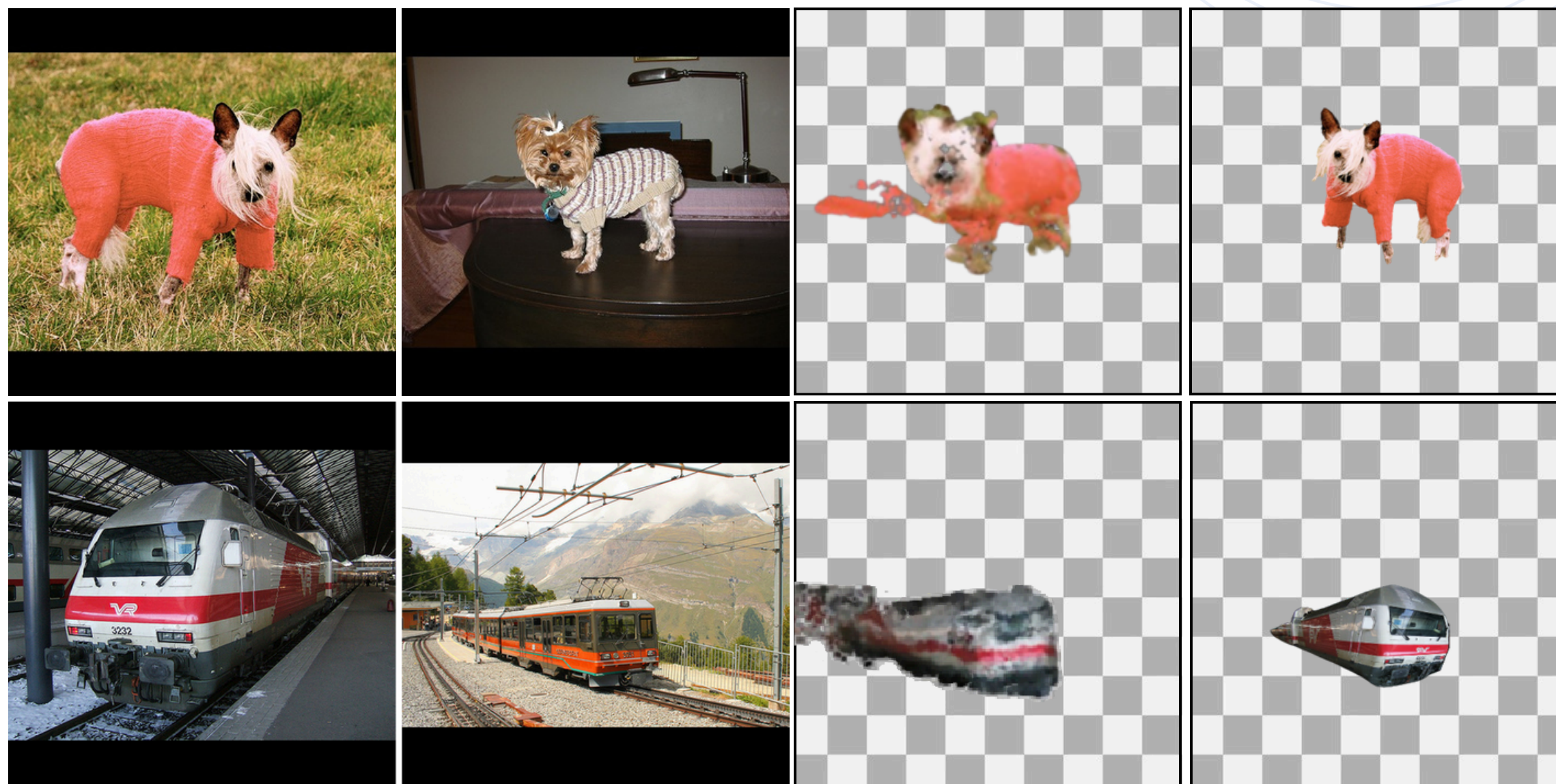
*For the Full Table Please Check Our Paper



Qualitative Results



Qualitative Comparison



Source

Target

ASIC

FastJAM (Ours)

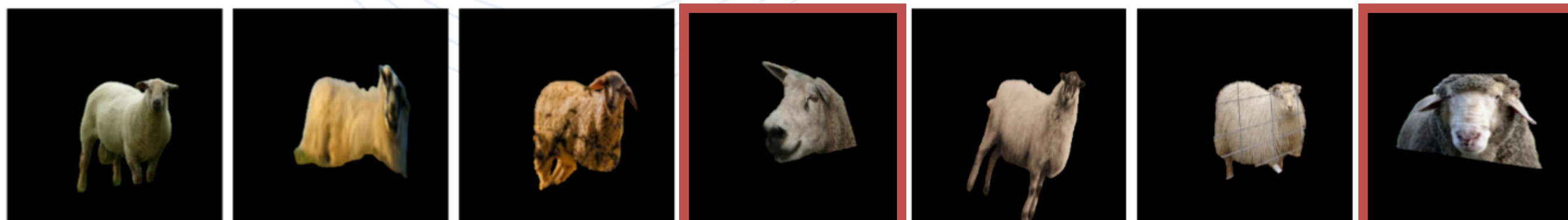
***For More Results
Please Check Our Paper**

Qualitative Comparison

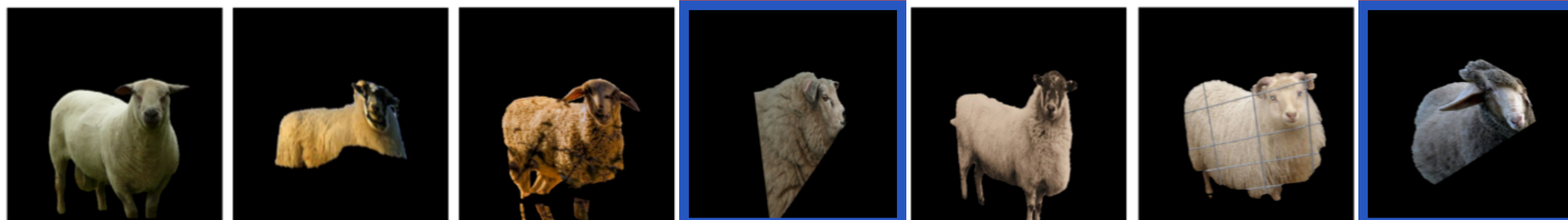
Sheep



SpaceJAM



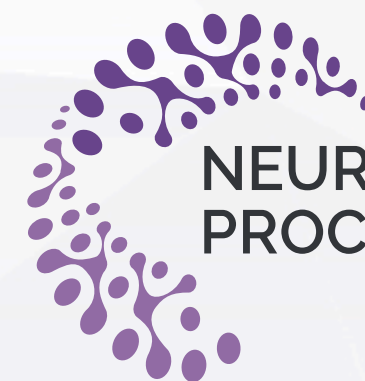
FastJAM (Ours)





BGU

Ben-Gurion University of the Negev



NEURAL INFORMATION
PROCESSING SYSTEMS

Thank You.

Thank you for your attention.
I welcome any questions or feedback.

omrihir@post.bgu.ac.il



FastJAM's Website

• <https://bgu-cs-vil.github.io/FastJAM/>