DeepRob

Seminar 5
Object Tracking
University of Michigan and University of Minnesota

This Week: Object Tracking

- Seminar 5: Recurrent Networks and Object Tracking
  1. DeepIM: Deep Iterative Matching for 6D Pose Estimation, Li et al., 2018
  2. PoseRBPF: A Rao-Blackwellized Particle Filter for 6D Object Pose Tracking, Deng et al., 2019
  3. 6-PACK: Category-level 6D Pose Tracker with Anchor-Based Keypoints, Wang et al., 2020
  4. XMem: Long-Term Video Object Segmentation with an Atkinson-Shiffrin Memory Model, Cheng and Schwing, 2022
- Seminar 6: Visual Odometry and Localization
  1. Backprop KF: Learning Discriminative Deterministic State Estimators, Haarnoja et al., 2016
  2. Differentiable Particle Filters: End-to-End Learning with Algorithmic Priors, Jonschkowski et al., 2018
  3. Multimodal Sensor Fusion with Differentiable Filters, Lee et al., 2020
  4. Differentiable SLAM-net: Learning Particle SLAM for Visual Navigation, Karkus et al., 2021

Today: Object Tracking

- Seminar 5: Recurrent Networks and Object Tracking
  1. DeepIM: Deep Iterative Matching for 6D Pose Estimation, Li et al., 2018
  2. PoseRBPF: A Rao-Blackwellized Particle Filter for 6D Object Pose Tracking, Deng et al., 2019
  3. 6-PACK: Category-level 6D Pose Tracker with Anchor-Based Keypoints, Wang et al., 2020
  4. XMem: Long-Term Video Object Segmentation with an Atkinson-Shiffrin Memory Model, Cheng and Schwing, 2022

- Seminar 6: Visual Odometry and Localization
  1. Backprop KF: Learning Discriminative Deterministic State Estimators, Haarnoja et al., 2016
  2. Differentiable Particle Filters: End-to-End Learning with Algorithmic Priors, Jonschkowski et al., 2018
  3. Multimodal Sensor Fusion with Differentiable Filters, Lee et al., 2020
  4. Differentiable SLAM-net: Learning Particle SLAM for Visual Navigation, Karkus et al., 2021

DeepIM

Deep Iterative Matching for 6D Pose Estimation

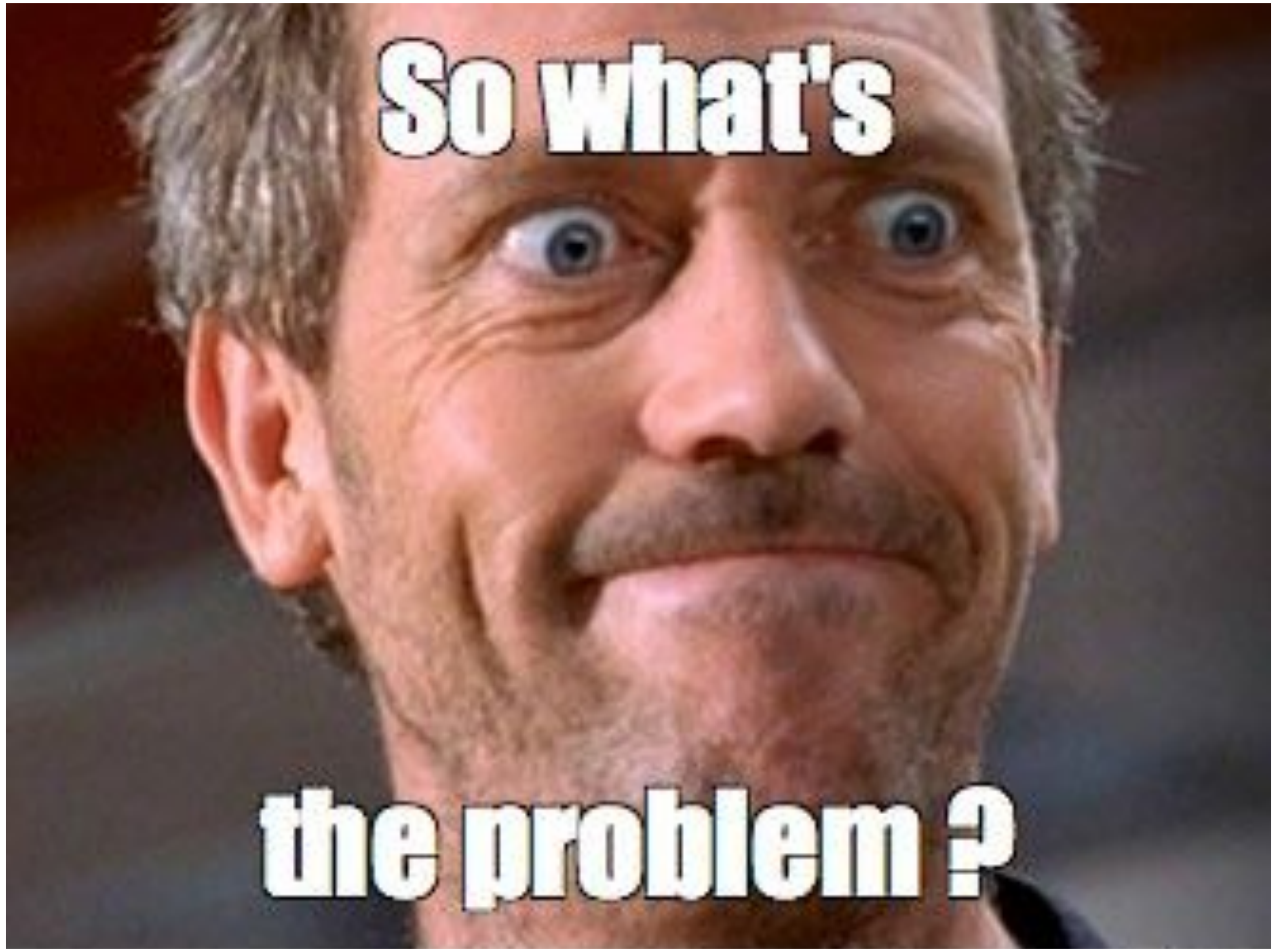
By: Yi Li, Gu Wang, Xiangyang Ji, Yu Xiang, Dieter Fox

Presented by: Saurav Telge, Rutwik Patel

The torch bearers of this research

- **Yi Li**
  - PhD student at University of Washington.
  - Advised by: Professor **Dieter Fox**.

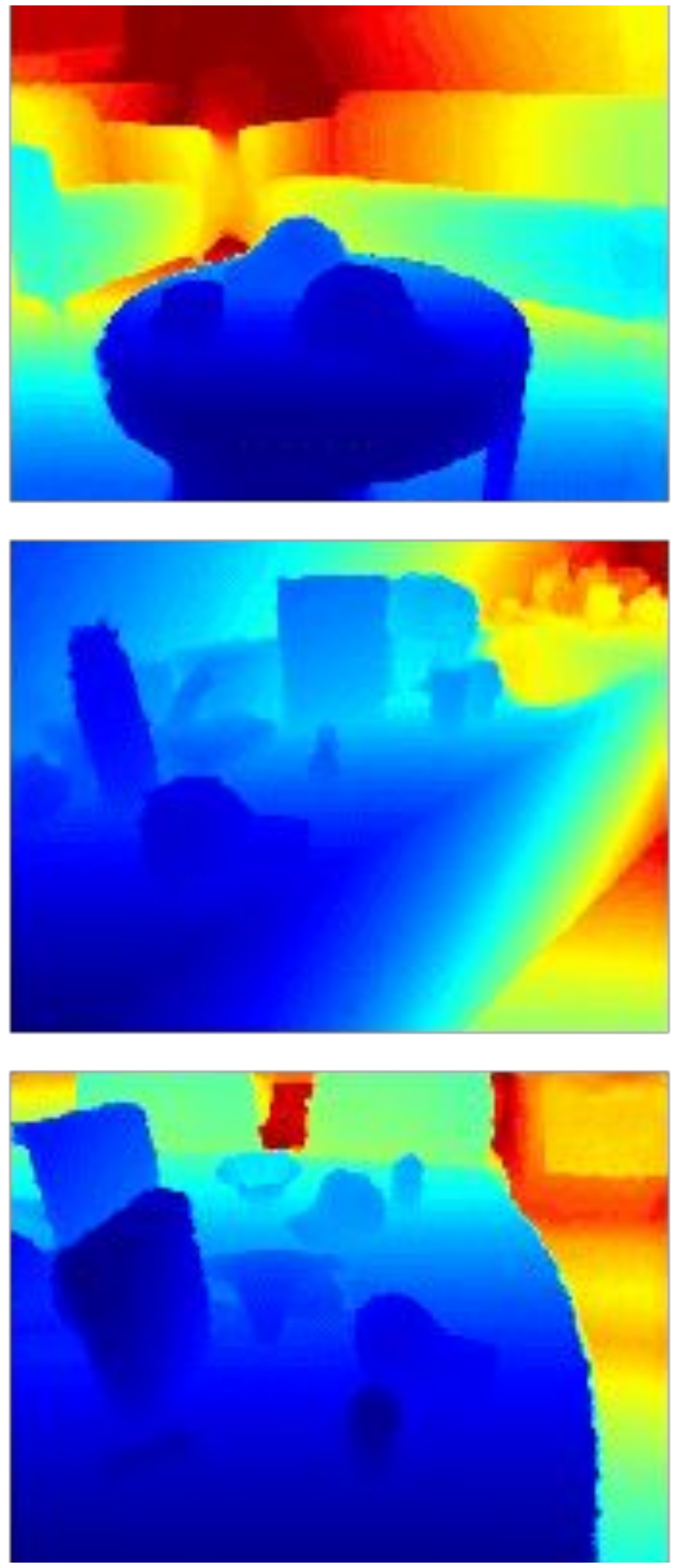
- **Gu Wang**
  - PhD student at Tsinghua University.
  - Advised by: Professor **Xiangyang Ji**.


So what's
the problem ?

Problem

Pose estimation

RGB Images



Depth map

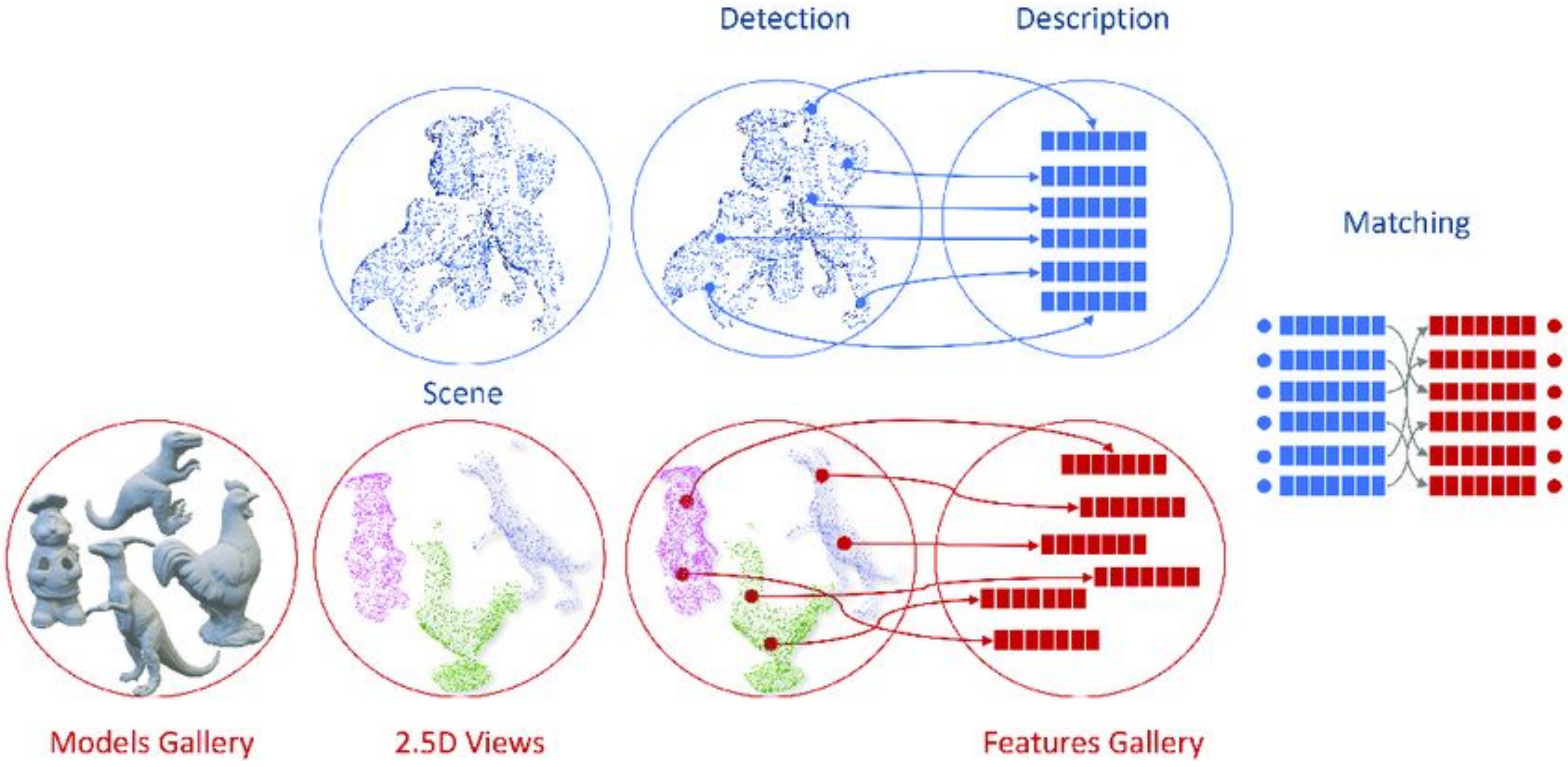

HOW FASCINATING
PLEASE, TELL ME MORE

Contributions

1. A framework for iterative pose matching.
2. An untangled representation of rotation and translation of 3D objects.
3. A new loss function for estimating difference between predicted pose and target pose.

Background

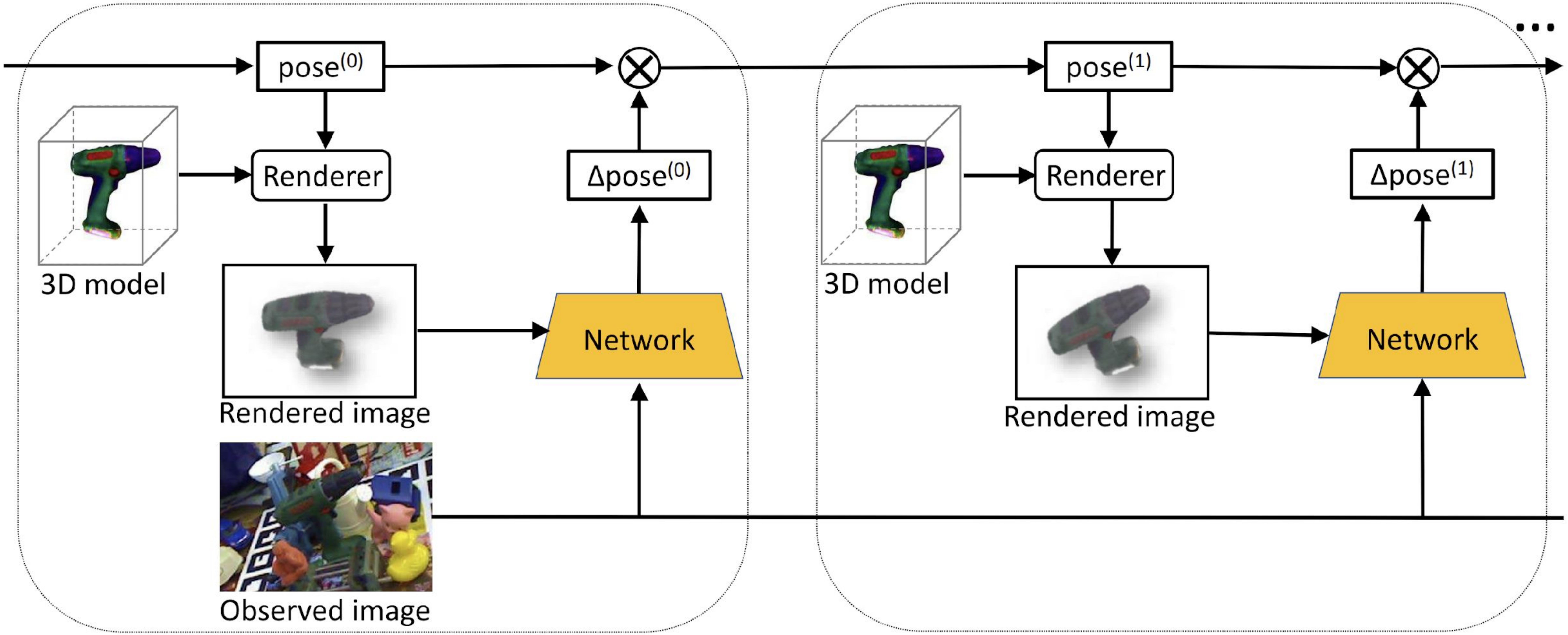
Matching 2D-3D correspondences



Texture-less object

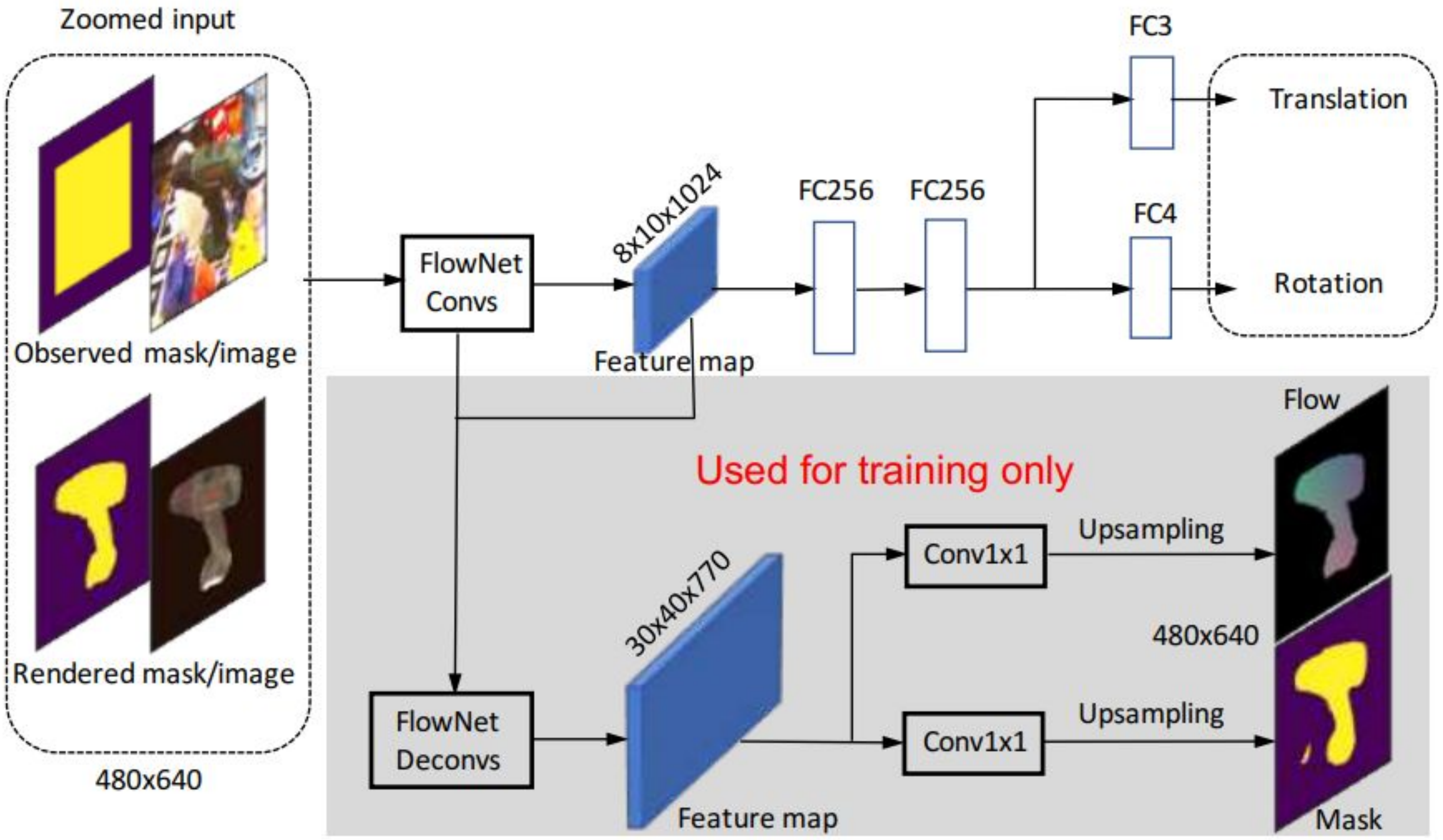
Textured object

Approach



pose$^{(0)}$
Renderer
3D model
Rendered image
Observed image
Δpose$^{(0)}$
Network
pose$^{(1)}$
Renderer
3D model
Rendered image
Δpose$^{(1)}$
Network
...

DeepIM network architecture



Methods

High-resolution Zoom In

Untangled Transformation Representation

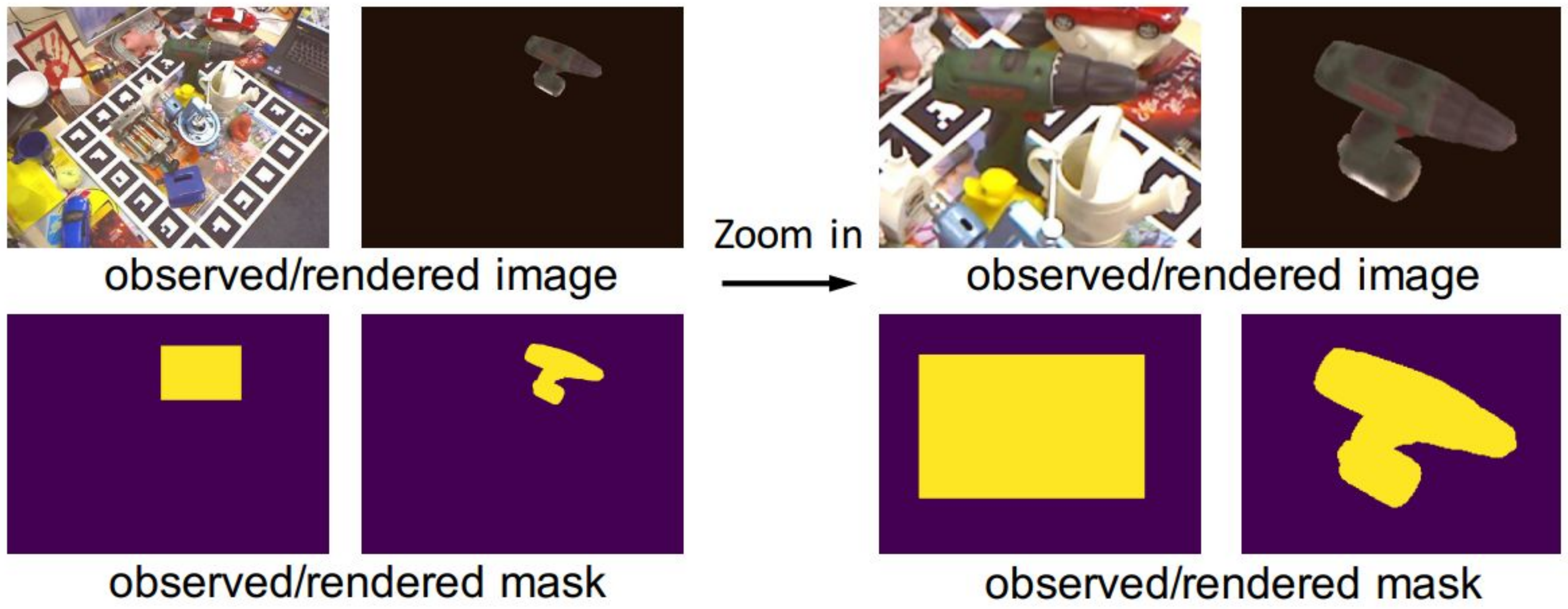
Matching Loss

Methods

High-resolution Zoom In

Untangled Transformation Representation

Matching Loss

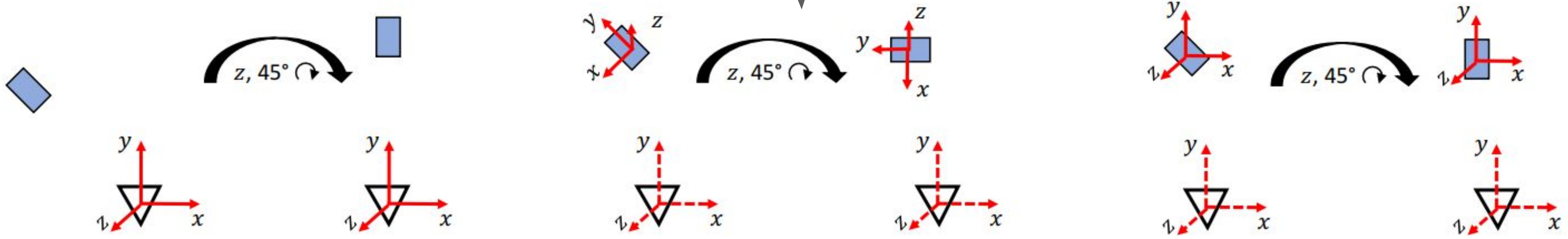


Methods

High-resolution Zoom In

Untangled Transformation Representation

Matching Loss



(a) Naïve Coordinate (b) Model Coordinate (c) Camera Coordinate

$$\mathbf{t}_\Delta = (v_x, v_y, v_z)$$

$$\begin{aligned} v_x &= f_x(x_{\text{tgt}}/z_{\text{tgt}} - x_{\text{src}}/z_{\text{src}}), \\ v_y &= f_y(y_{\text{tgt}}/z_{\text{tgt}} - y_{\text{src}}/z_{\text{src}}), \\ v_z &= \log(z_{\text{src}}/z_{\text{tgt}}), \end{aligned}$$

Methods

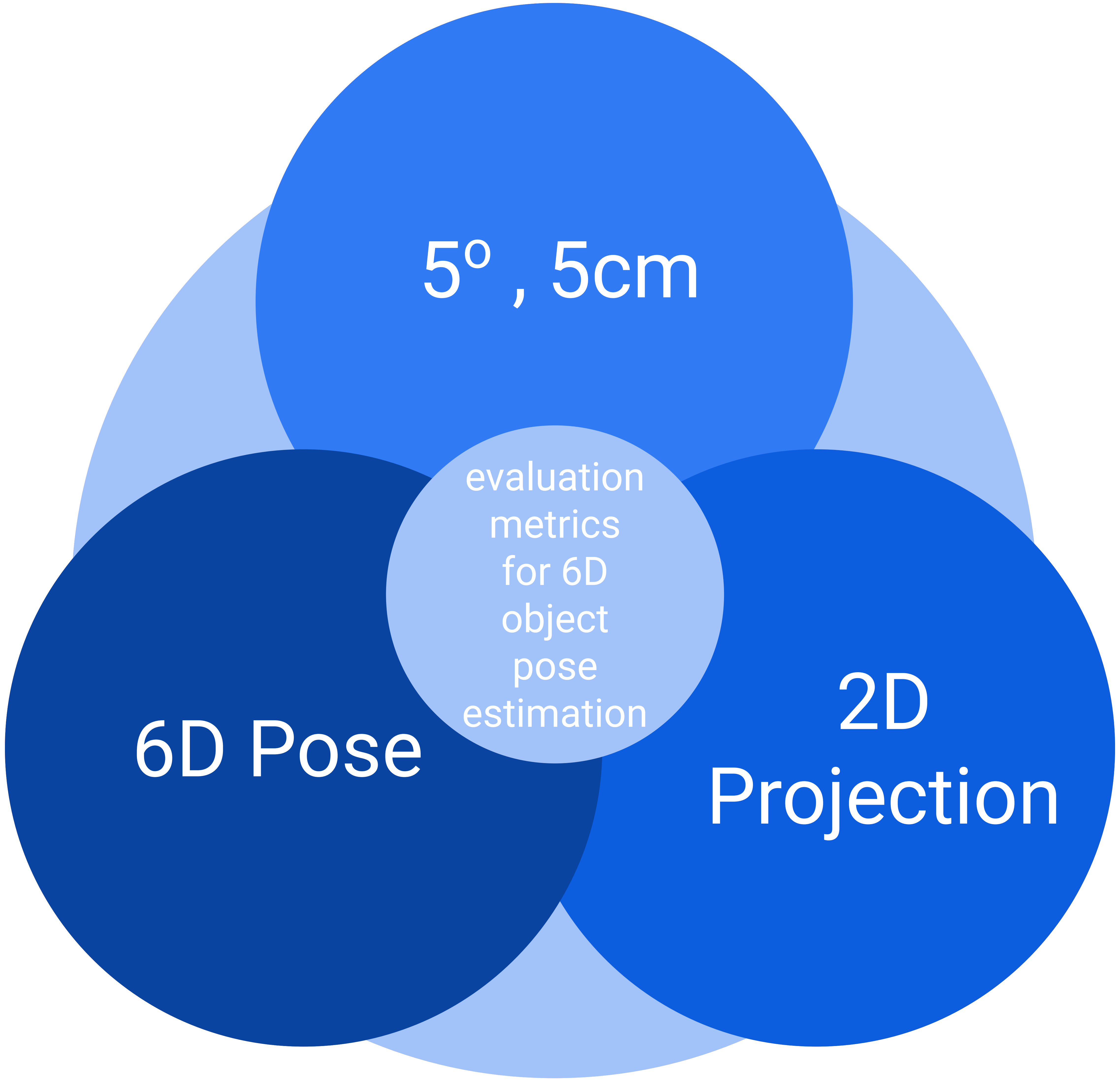
High-resolution Zoom In

Untangled Transformation Representation

Matching Loss

$$L_{\text{pose}}(\mathbf{p}, \hat{\mathbf{p}}) = \frac{1}{n} \sum_{i=1}^{n} L_1\big((\mathbf{R}\mathbf{x}_i + \mathbf{t}) - (\hat{\mathbf{R}}\mathbf{x}_i + \hat{\mathbf{t}})\big)$$

Evaluation metrics



method	PoseCNN	PoseCNN +OURS	Faster R-CNN	Faster R-CNN +OURS
5cm 5°	19.4	85.2	11.9	83.4
6D Pose	62.7	88.6	33.1	86.9
Proj. 2D	70.2	97.5	20.9	95.7

Models for generating initials poses & improvement using the DeepIM network

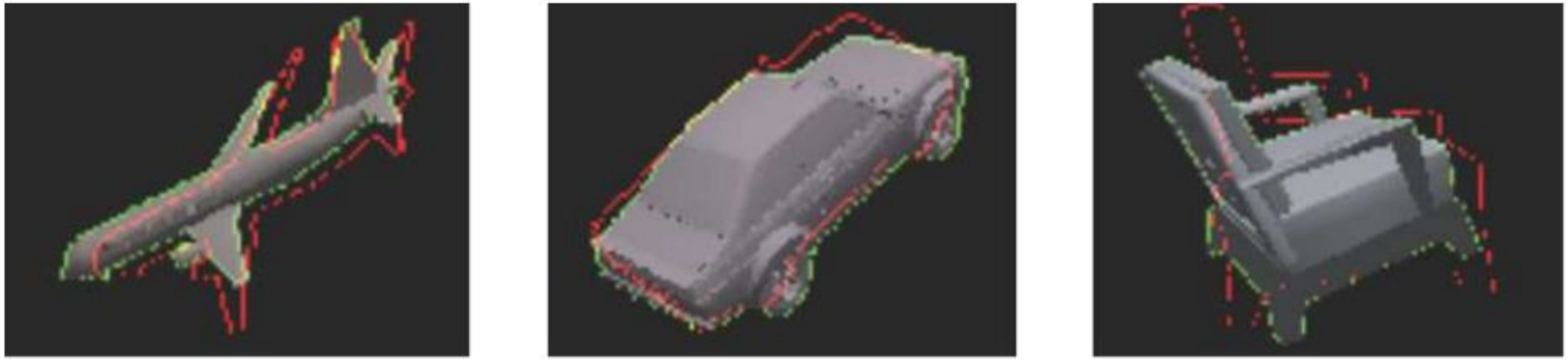
methods	[2]	BB8 w ref. [20]	SSD-6D w ref. [11]	Tekin et al. [26]	PoseCNN [29]	PoseCNN [29] +OURS
5cm 5°	40.6	69.0	-	-	19.4	**85.2**
6D Pose	50.2	62.7	79	55.95	62.7	**88.6**
Proj. 2D	73.7	89.3	-	90.37	70.2	**97.5**

Comparison with state-of-the-art methods on the LINEMOD dataset


WAITING FOR RESULTS LIKE

Results

Examples of refined poses on the Occlusion LINEMOD dataset using the results from PoseCNN as initial poses



pose refinement of unseen 3D models from the ModelNet dataset

Conclusions

- Accurate and efficient estimation of the 6D pose of an object from a single RGB image.
- The 6D pose estimation has a wide range of applications in robotics, augmented reality, and object recognition, among others.
- Limitations

  Computationally expensive, Limited applicability, Sensitivity to initialization.

- Future directions

  The iterative refinement process can also be extended to other tasks, such as object detection, segmentation, and tracking.

Thank you


THE PERSON LOOKING AT
THIS MEME RIGHT NOW IS AWESOME
GO HAVE A GOOD DAY NOW
imgflip.com

PoseRBPF

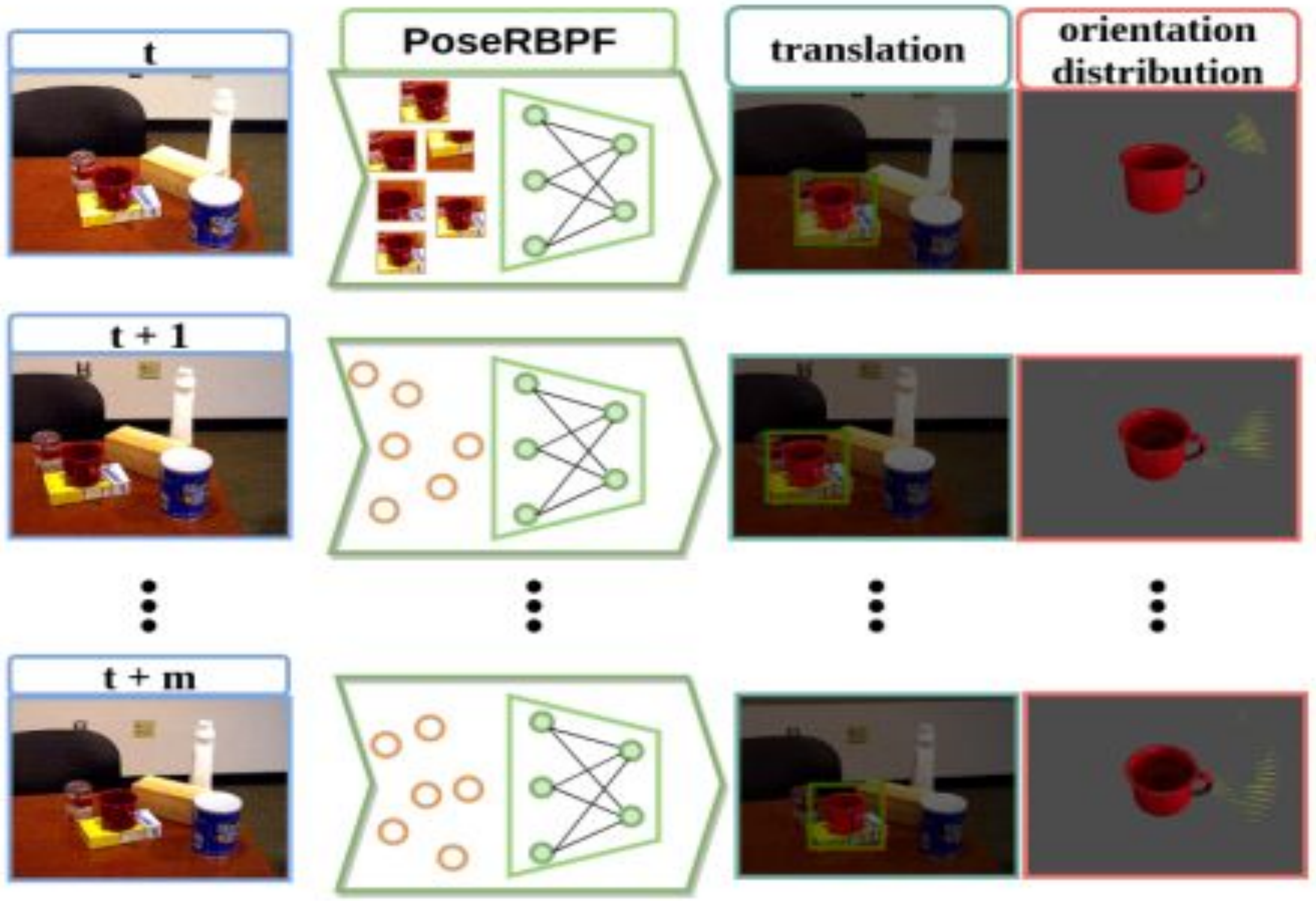
A Rao-Blackwellized Particle Filter for 6D Object Pose Tracking

By: Xinke Deng, Arsalan Mousavian, Yu Xiang, Fei Xia, Timothy Bretl, Dieter Fox

Presented by: Siddharth Rao Appala, Rishitha Gollamudi

Motivation

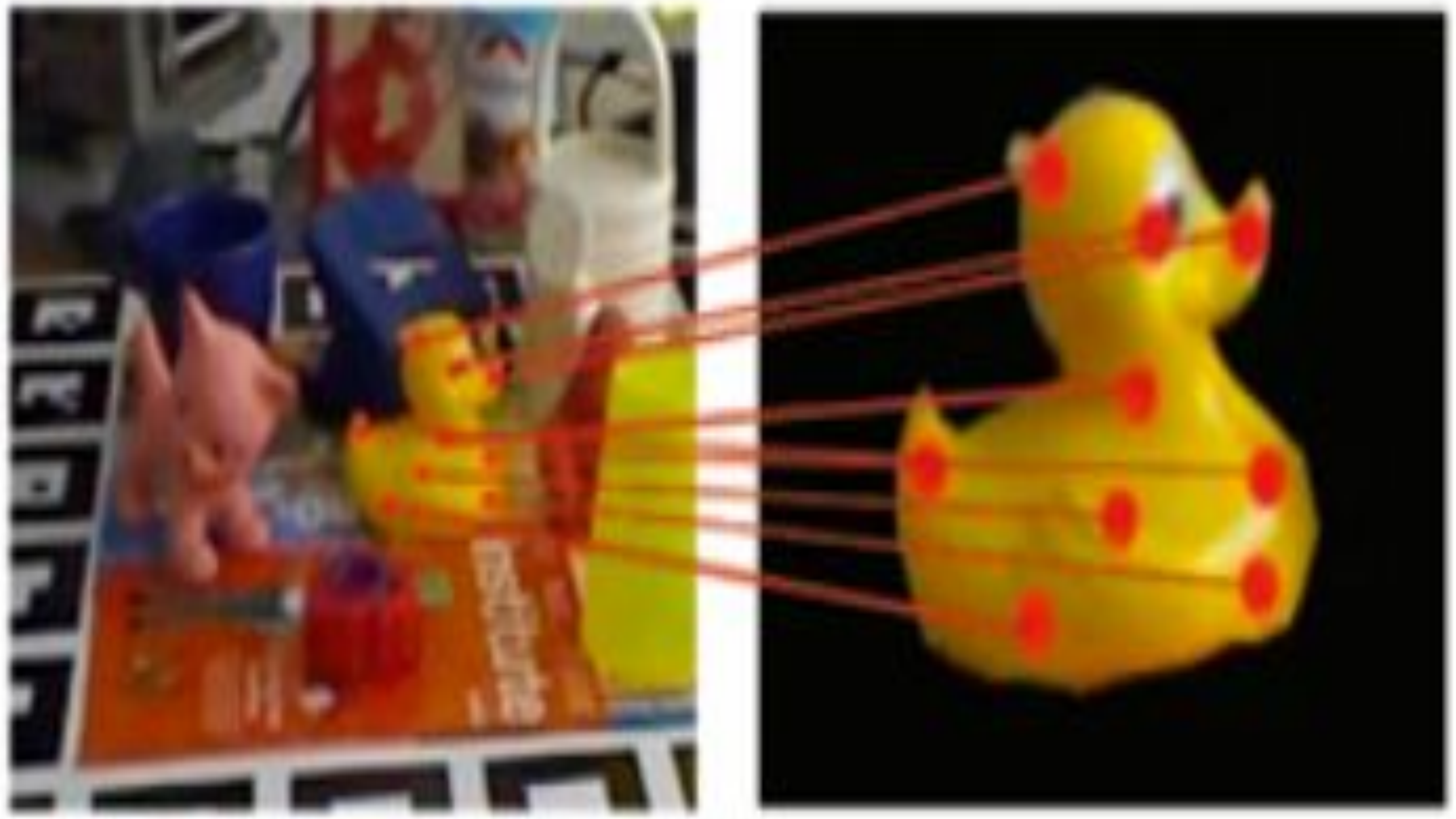
- The paper aims to develop a novel 6D pose tracking framework that tracks objects with 6 degrees of freedom over a video sequence.
- Tasks like robot manipulation and grasp planning require accurate 6D pose tracking with uncertainty estimates and robustness to object symmetries.
- This can be achieved by accounting for the temporal information.



Contributions

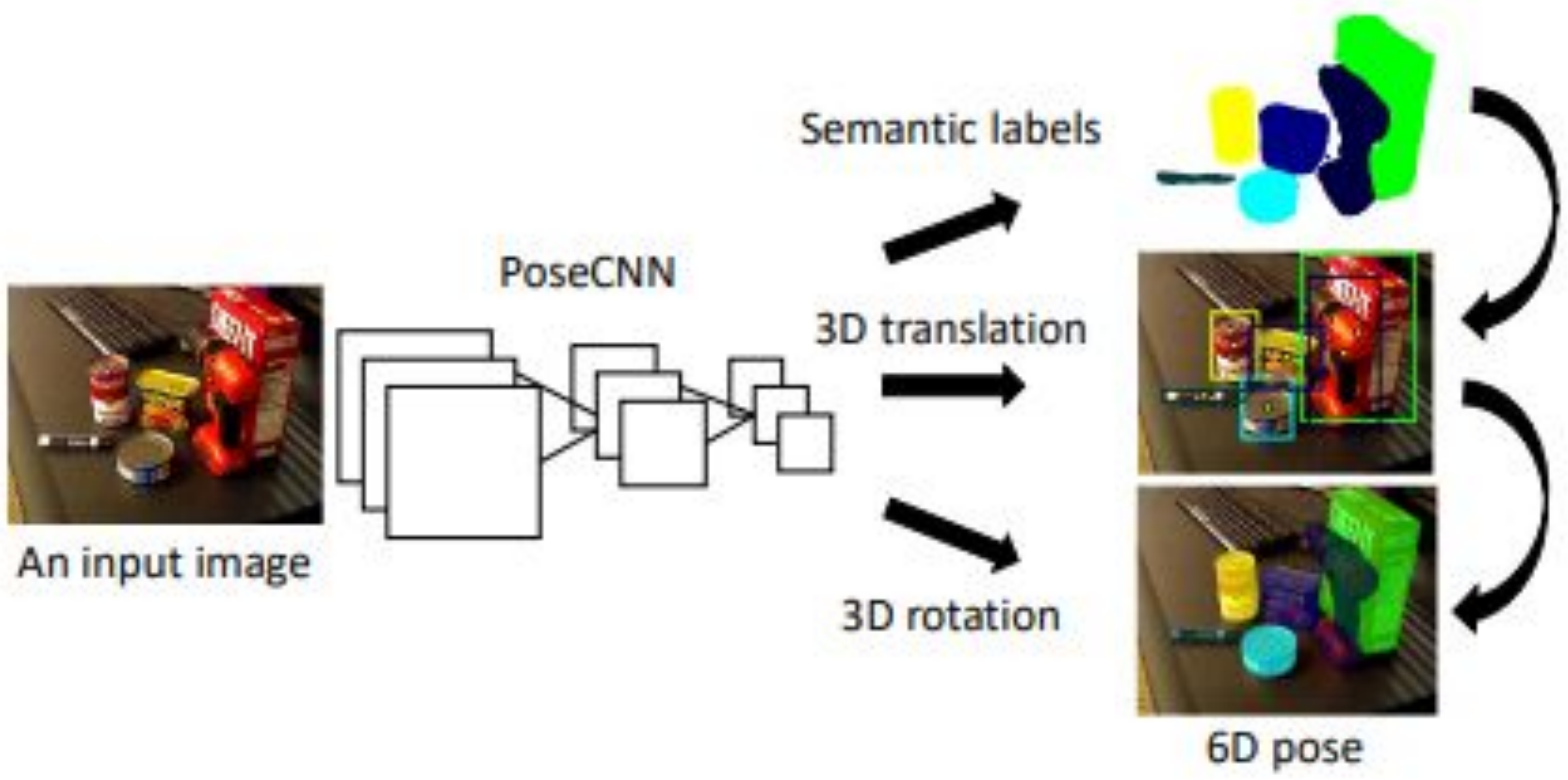
1. Introduced a novel 6D object pose estimation pipeline that combines Rao-Blackwellized particle filtering with a learned autoencoder to generate full distribution over 6D poses
2. The proposed framework can track full distributions over 6D object poses for objects with arbitrary kinds of symmetries, without the need for any manual symmetry labeling.

Related Work & Short comings

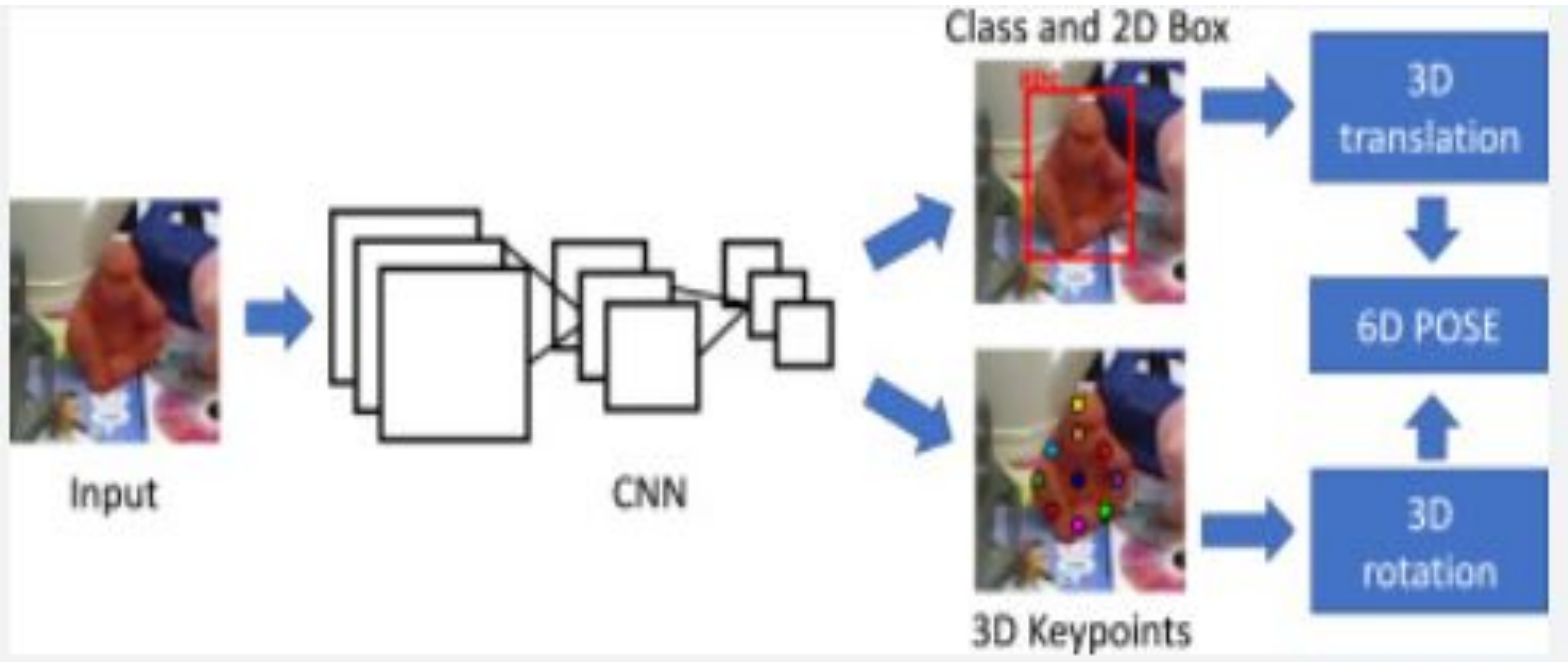


Traditional approaches - key point detection and local feature matching

Related Work & Short comings



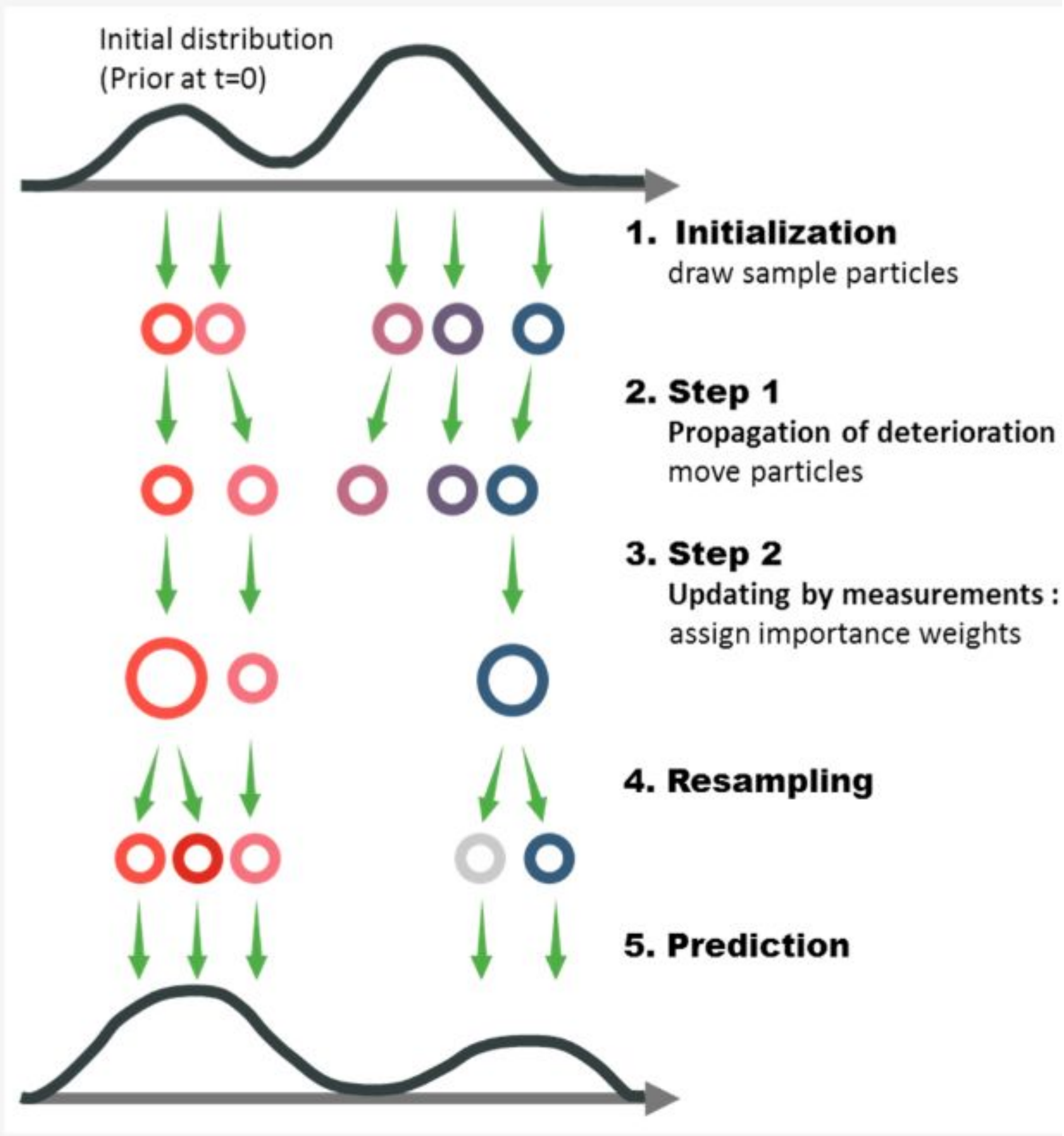
PoseCNN



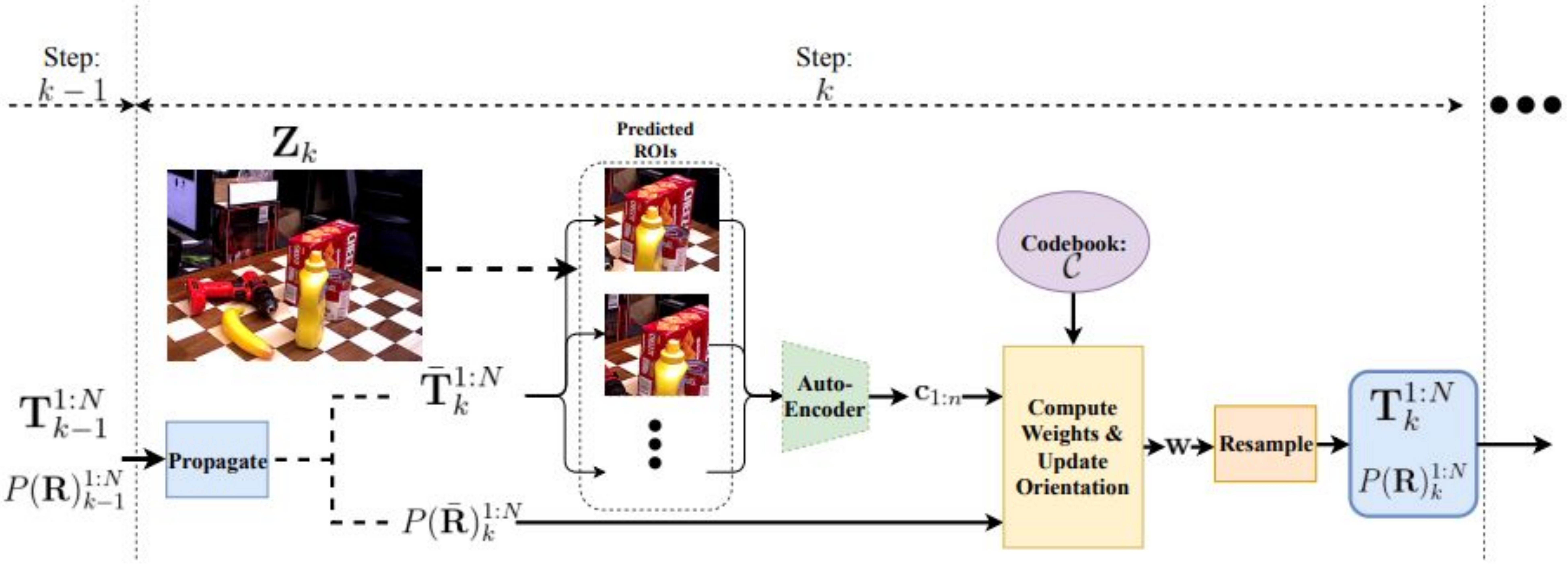
Object detection based approaches

Particle Filtering

- A particle filter is a statistical algorithm which express the distribution of a state space model by extracting random state particles from the posterior probability.
- RBPF - decreases number of particles necessary to achieve same accuracy with regular PF
- Divide the state vector into two parts: one part that can be updated efficiently using a closed-form equation, and another part is updated using particle filtering.



Approach



Step:
$k-1$
Step:
k
$\mathbf{Z}_k$
Predicted
ROIs
Codebook:
$\mathcal{C}$
$\mathbf{T}_{k-1}^{1:N}$
$P(\mathbf{R})_{k-1}^{1:N}$
Propagate
$\bar{\mathbf{T}}_k^{1:N}$
$P(\bar{\mathbf{R}})_k^{1:N}$
Auto-
Encoder
$\mathbf{c}_{1:n}$
Compute
Weights &
Update
Orientation
$\mathbf{w}$
Resample
$\mathbf{T}_k^{1:N}$
$P(\mathbf{R})_k^{1:N}$

Approach - Motion Priors

Based on the Rao Blackwellized Particle filter approach,

- The translation distribution is propagated using

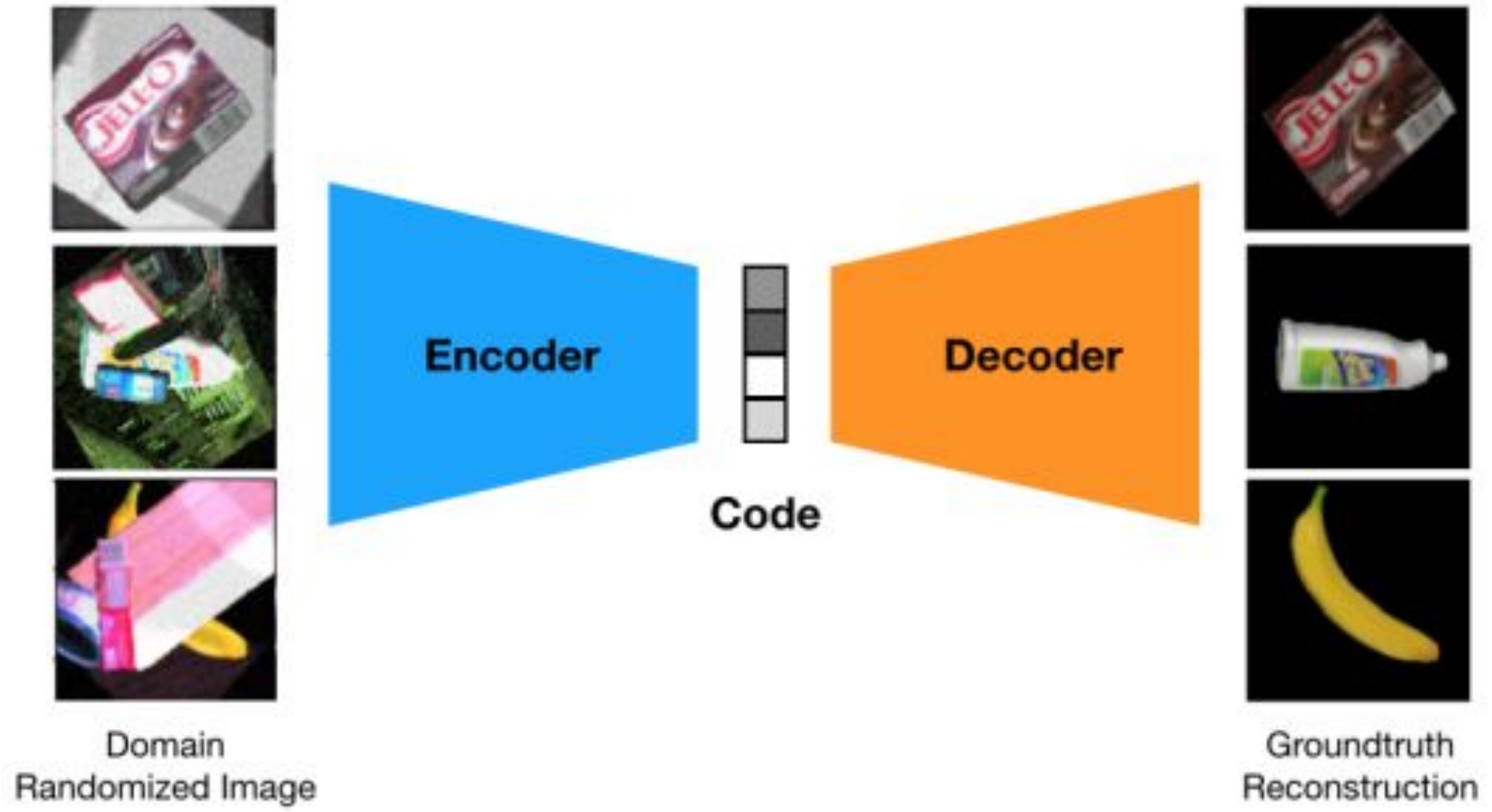
$$P(\mathbf{T}_k|\mathbf{T}_{k-1}, \mathbf{T}_{k-2}) = \mathcal{N}\left(\mathbf{T}_{k-1} + \alpha(\mathbf{T}_{k-1} - \mathbf{T}_{k-2}), \mathbf{\Sigma_T}\right)$$

- The rotation distribution is propagated using

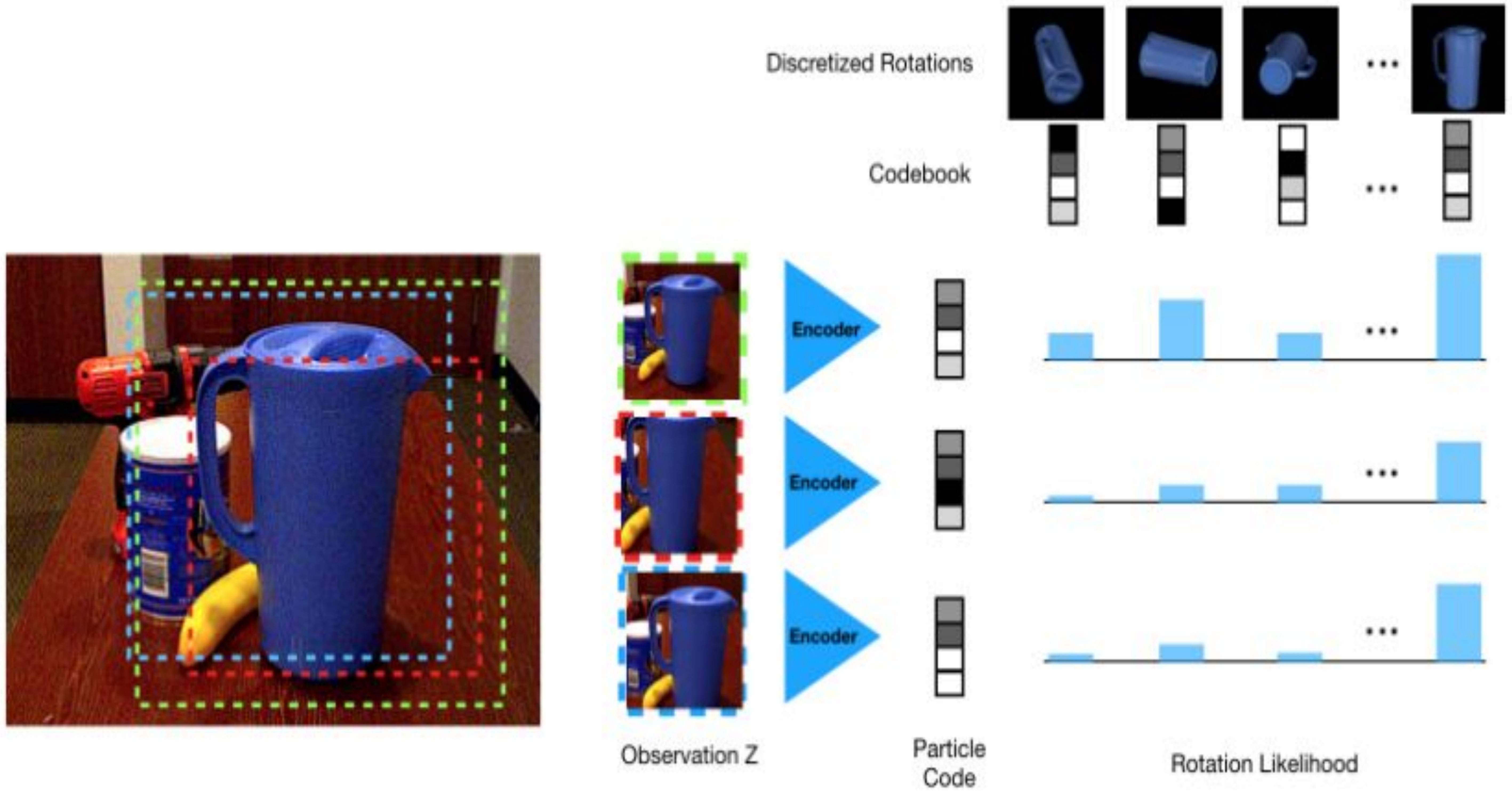
$$P(\mathbf{R}_k|\mathbf{R}_{k-1}) = \mathcal{N}\left(\mathbf{R}_{k-1}, \mathbf{\Sigma_R}\right)$$

$$P(\mathbf{R}_k|\mathbf{T}^i_k, \mathbf{Z}_{1:k}) \propto P(\mathbf{R}_k|\mathbf{T}^i_k, \mathbf{Z}_k)P(\mathbf{R}_k|\mathbf{R}_{k-1}),$$

Approach - Autoencoder

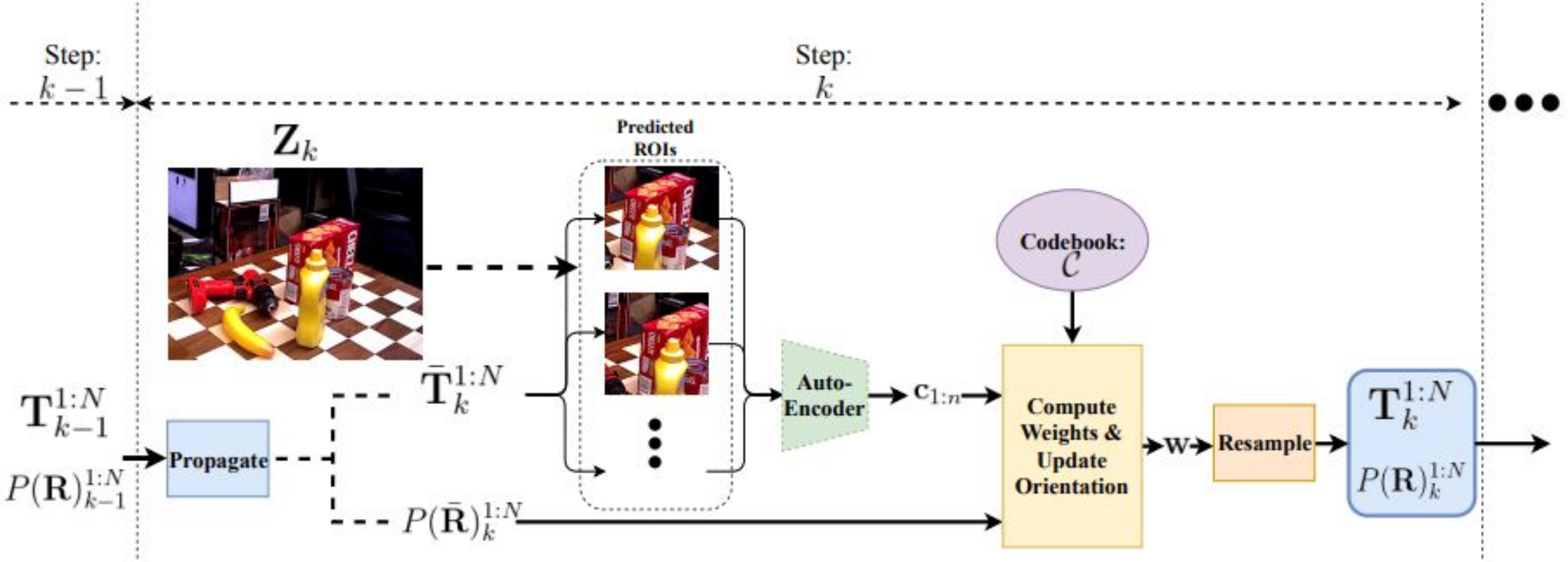


Autoencoder



Codebook matching

Approach - Weight update and resampling



Weight update: $P(\mathbf{T}_k^i|\mathbf{Z}_{1:k}) \propto \sum_{\mathbf{R}_k} P(\mathbf{Z}_k|\mathbf{T}_k^i, \mathbf{R}_k)P(\mathbf{R}_k|\mathbf{T}_{1:k-1}^i, \mathbf{Z}_{1:k-1}),$

Approach - Summary

input : $\mathbf{Z}_k$, $(\mathbf{T}_{k-1}^{1:N}, P(\mathbf{R})_{k-1}^{1:N})$

output: $(\mathbf{T}_{k}^{1:N}, P(\mathbf{R})_{k}^{1:N})$

begin

$\{w^i\}_{i=1}^{N} \leftarrow \emptyset$;

$(\bar{\mathbf{T}}_{k}^{1:N}, P(\bar{\mathbf{R}})_{k}^{1:N}) \leftarrow Propagate(\mathbf{T}_{k-1}^{1:N}, P(\mathbf{R})_{k-1}^{1:N})$;

for $(\bar{\mathbf{T}}_{k}^{i}, P(\bar{\mathbf{R}})_{k}^{i}) \in (\bar{\mathbf{T}}_{k}^{1:N}, P(\bar{\mathbf{R}})_{k}^{1:N})$ **do**

$P(\bar{\mathbf{R}})_{k}^{i} \leftarrow Codebook\_Match(\mathbf{Z}_k, \bar{\mathbf{T}}_{k}^{i}) * P(\bar{\mathbf{R}})_{k}^{i}$;

$w^i \leftarrow Evaluate(\mathbf{Z}_k, \bar{\mathbf{T}}_{k}^{i}, P(\bar{\mathbf{R}}_{k}^{i}))$;

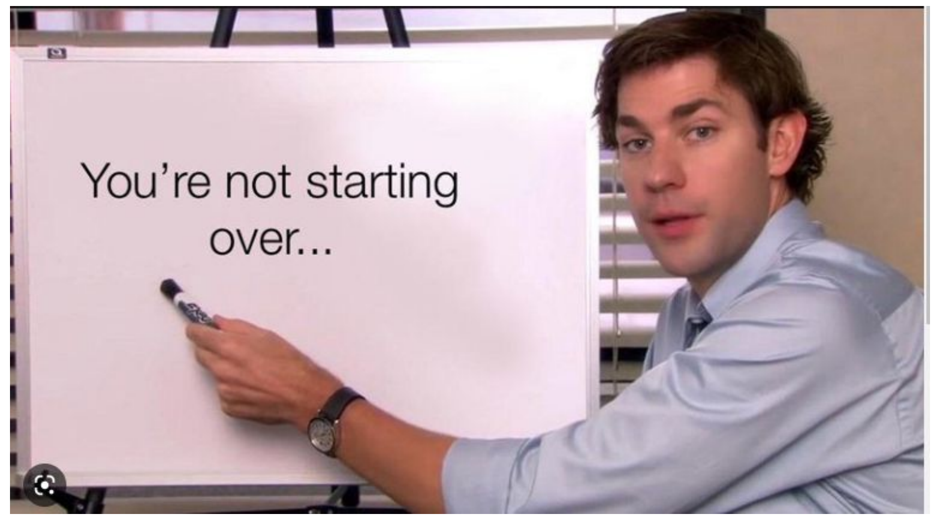
end

$(\mathbf{T}_{k}^{1:N}, P(\mathbf{R})_{k}^{1:N}) \leftarrow$

$Resample(\bar{\mathbf{T}}_{k}^{1:N}, P(\bar{\mathbf{R}})_{k}^{1:N}, \{w^i\}_{i=1}^{N})$;

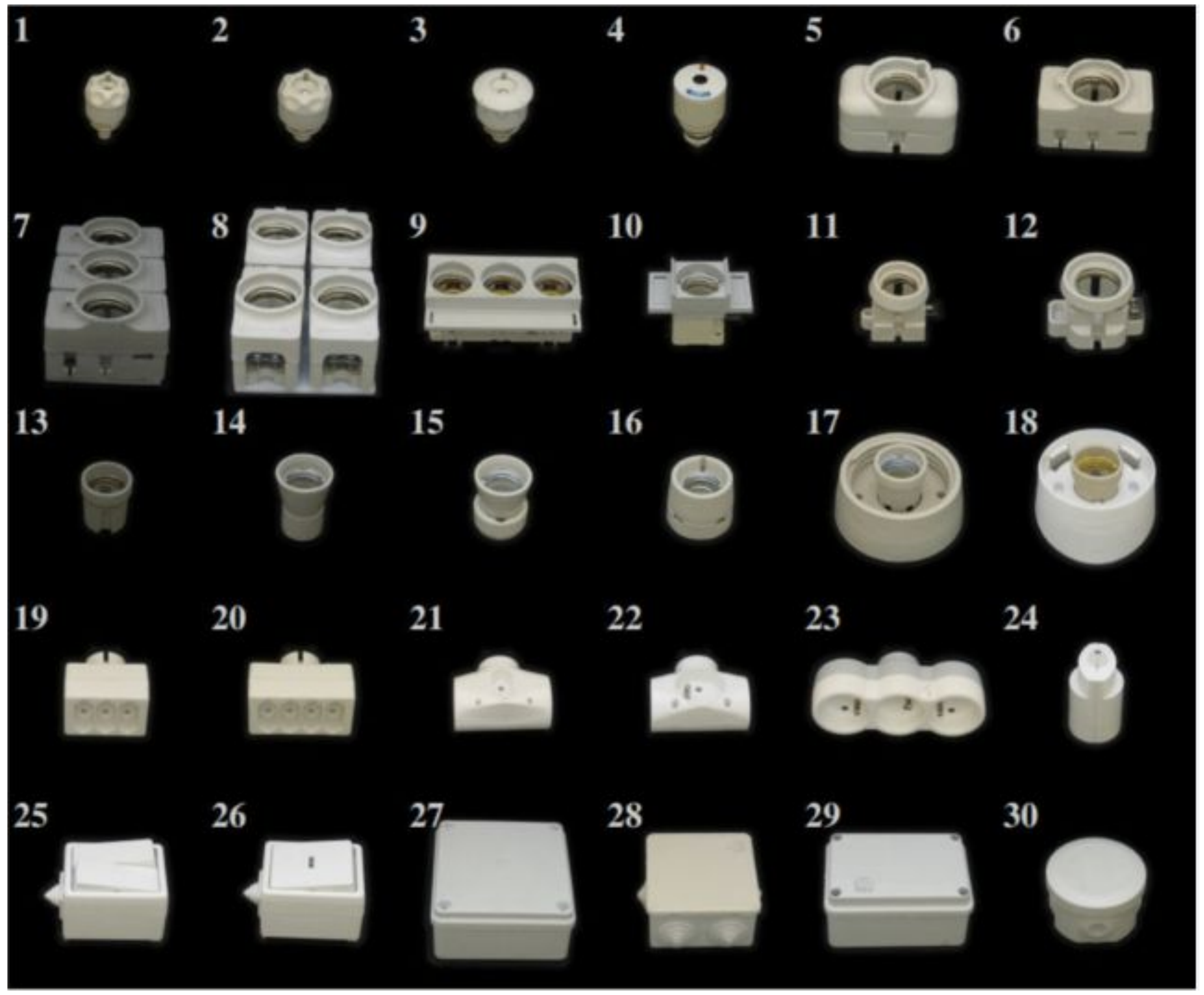
end

Algorithm 1: 6D Object Pose Tracking with PoseRBPF



Evaluation

YCB Video dataset
RGBD video sequences of 21 objects
Metrics: ADD, ADD-S



T-LESS dataset
RGB-D sequences of 30 non textured industrial objects
Metrics: Visual Surface Discrepancy

Results - YCB Video dataset

	RGB									
	PoseCNN [43]		DOPE [40]		PoseRBPF 50 particles		PoseRBPF 200 particles		PoseRBPF++ 200 particles	
objects	ADD	ADD-S	ADD	ADD-S	ADD	ADD-S	ADD	ADD-S	ADD	ADD-S
002_master_chef_can	50.9	84.0	-	-	56.1	75.6	58.0	77.1	**63.3**	**87.5**
003_cracker_box	51.7	76.9	55.9	69.8	73.4	85.2	76.8	87.0	**77.8**	**87.6**
004_sugar_box	68.6	84.3	75.7	87.1	73.9	86.5	75.9	87.6	**79.6**	**89.4**
005_tomato_soup_can	66.0	80.9	**76.1**	**85.1**	71.1	82.0	74.9	84.5	73.0	83.6
006_mustard_bottle	79.9	90.2	81.9	90.9	80.0	90.1	82.5	91.0	**84.7**	**92.0**
007_tuna_fish_can	**70.4**	**87.9**	-	-	56.1	73.8	59.0	79.0	64.2	82.7
008_pudding_box	**62.9**	**79.0**	-	-	54.8	69.2	57.2	72.1	64.5	77.2
009_gelatin_box	75.2	87.1	-	-	83.1	89.7	**88.8**	**93.1**	83.0	90.8
010_potted_meat_can	**59.6**	**78.5**	39.4	52.4	47.0	61.3	49.3	62.0	51.8	66.9
011_banana	**72.3**	**85.9**	-	-	22.8	64.1	24.8	61.5	18.4	66.9
019_pitcher_base	52.5	76.8	-	-	74.0	87.5	**75.3**	**88.4**	63.7	82.1
021_bleach_cleanser	50.5	71.9	-	-	51.6	66.7	54.5	69.3	**60.5**	**74.2**
024_bowl	6.5	69.7	-	-	26.4	**88.2**	**36.1**	86.0	28.4	85.6
025_mug	57.7	78.0	-	-	67.3	83.7	70.9	85.4	**77.9**	**89.0**
035_power_drill	55.1	72.8	-	-	64.4	80.6	70.9	**85.0**	**71.8**	84.3
036_wood_block	**31.8**	**65.8**	-	-	0.0	0.0	2.8	33.3	2.3	31.4
037_scissors	35.8	56.2	-	-	20.6	30.9	21.7	33.0	**38.7**	**59.1**
040_large_marker	58.0	71.4	-	-	45.7	54.1	48.7	59.3	**67.1**	**76.4**
051_large_clamp	25.0	49.9	-	-	27.0	73.2	**47.3**	**76.9**	38.3	59.3
052_extra_large_clamp	15.8	47.0	-	-	50.4	68.7	**26.5**	**69.5**	32.3	44.3
061_foam_brick	40.4	87.8	-	-	75.8	88.4	78.2	89.7	**84.1**	**92.6**
ALL	53.7	75.9	-	-	57.1	74.8	59.9	77.5	**62.1**	**78.4**

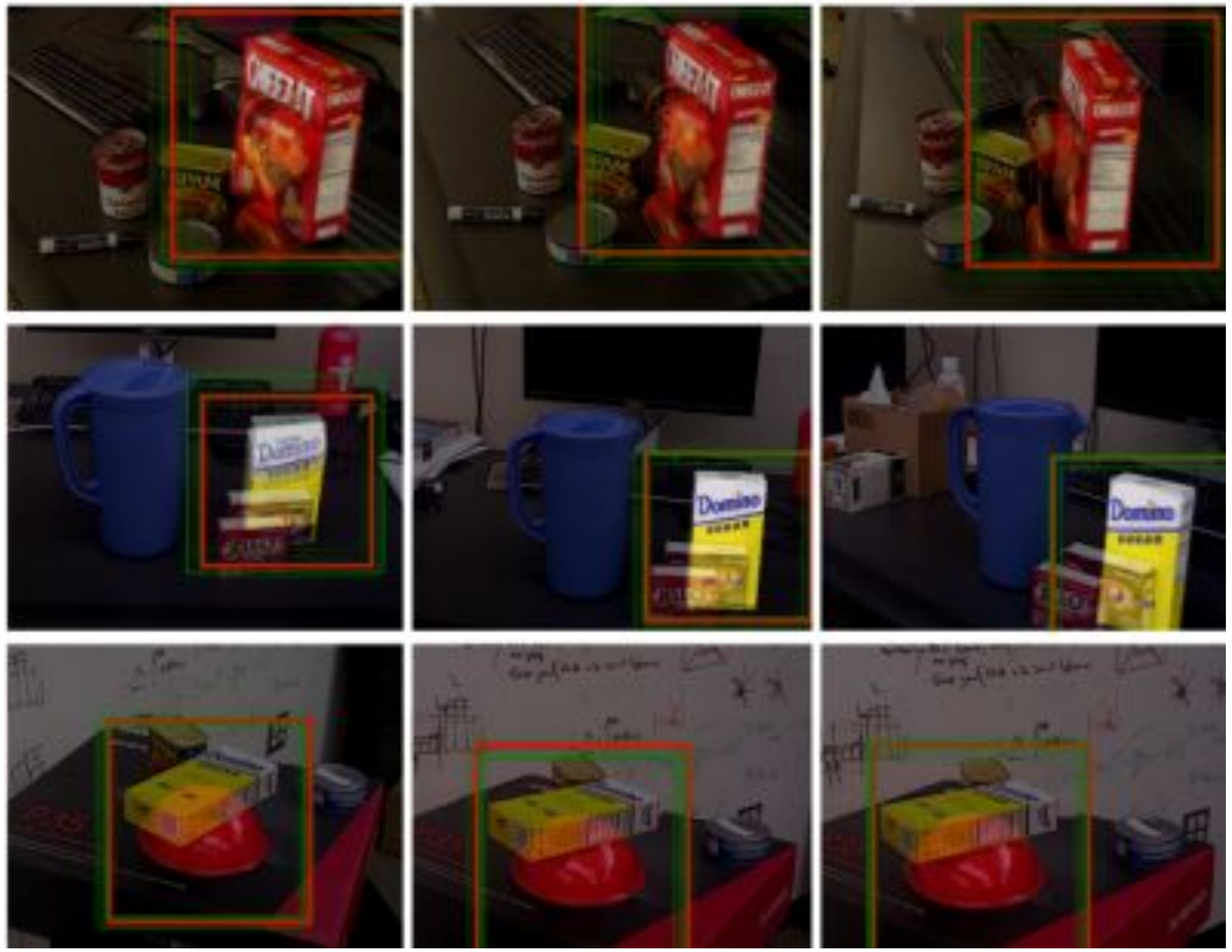
PoseRBPF++ - 50% of the particles around PoseCNN predictions and the other 50% from the particles of the previous time step

Results - TLESS dataset

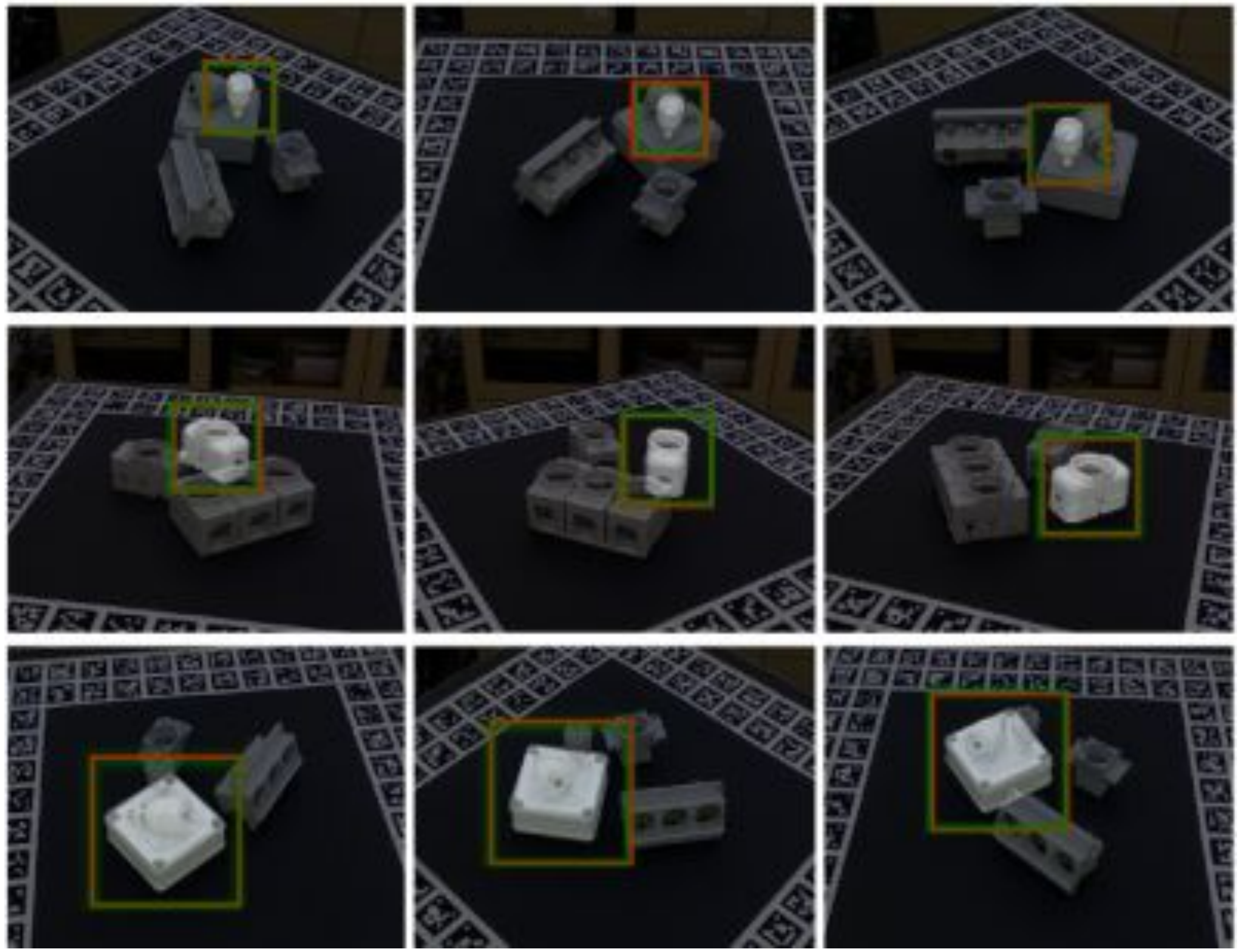
Object	Without GT 2D BBs					With GT 2D BBs	
	RGB			RGB-D			
	SSD [37]	RetinaNet [37]	RetinaNet PoseRBPF	RetinaNet [37] + ICP	RetinaNet PoseRBPF	[37]	PoseRBPF
1	5.65	8.87	**27.60**	22.32	**61.30**	12.33	**80.90**
2	5.46	13.22	**26.60**	29.49	**63.10**	11.23	**85.80**
3	7.05	12.47	**37.70**	38.26	**74.30**	13.11	**85.60**
4	4.61	6.56	**23.90**	23.07	**64.50**	12.71	**62.00**
5	36.45	34.80	**54.40**	76.10	**86.70**	66.70	**89.80**
6	23.15	20.24	**73.00**	67.64	**71.50**	52.30	**97.80**
7	15.97	16.21	**51.60**	73.88	**88.00**	36.58	**91.20**
8	10.86	19.74	**37.90**	67.02	**84.00**	22.05	**95.60**
9	19.59	36.21	**41.60**	78.24	**86.00**	46.49	**77.10**
10	10.47	11.55	**41.50**	**77.65**	74.30	14.31	**85.30**
11	4.35	6.31	**38.30**	35.89	**62.60**	15.01	**89.50**
12	7.80	8.15	**39.60**	49.30	**71.00**	31.34	**91.20**
13	3.30	4.91	**20.40**	42.50	**42.10**	13.60	**89.30**
14	2.85	4.61	**32.00**	30.53	**50.10**	45.32	**70.20**
15	7.90	26.71	**41.60**	**83.73**	76.60	50.00	**96.60**
16	13.06	21.73	**39.10**	67.42	**83.80**	36.09	**97.00**
17	41.70	**64.84**	40.00	**86.17**	78.40	81.11	**87.00**
18	47.17	14.30	**47.90**	**84.34**	81.10	52.62	**89.70**
19	15.95	22.46	**40.60**	50.54	**61.80**	50.75	**83.20**
20	2.17	5.27	**29.60**	14.75	**55.00**	37.75	**70.00**
21	19.77	17.93	**47.20**	40.31	**72.70**	50.89	**84.40**
22	11.01	18.63	**36.60**	35.23	**63.80**	47.60	**77.70**
23	7.98	18.63	**42.00**	42.52	**82.40**	35.18	**85.90**
24	4.74	4.23	**48.20**	59.54	**83.20**	11.24	**91.80**
25	21.91	18.76	**39.50**	70.89	**77.70**	37.12	**88.70**
26	10.04	12.62	**47.80**	66.20	**85.00**	28.33	**90.90**
27	7.42	21.13	**41.30**	**73.51**	68.00	21.86	**79.10**
28	21.78	23.07	**49.50**	61.20	**79.30**	42.58	**72.10**
29	15.33	26.65	**60.50**	73.04	**86.30**	57.01	**96.00**
30	34.63	29.58	**52.70**	**92.90**	80.10	70.42	**77.00**
Mean	14.67	18.35	**41.67**	57.14	**73.16**	36.79	**85.28**

PoseRBPF outperforms other methods in most object classes

Qualitative Results



YCB Video dataset



TLESS dataset

Conclusions

- In conclusion, PoseRBPF is a 6D pose tracking framework that uses a particle filtering approach with a learned autoencoder to estimate full distributions over object poses.
- The proposed method overcomes the shortcomings of existing approaches by estimating uncertainties and providing robustness against symmetry and occlusion.
- PoseRBPF achieves state-of-the-art results on two benchmarks.

Limitations and Future Work

Limitations

- PoseRBPF does not generalize well to unseen objects as codebooks are generated only for objects in the training set.
- Each object requires a codebook entry for each of the 191,808 possible orientations, making it highly inefficient to store.

Future work

Methods to generate object independent codebook entries can be explored.

Thank You!

6-PACK

Category-level **6**D **P**ose Tracker with **A**nchor-Based **K**eypoints

By: Chen Wang, Roberto Martín-Martín, Danfei Xu, Jun Lv
Cewu Lu, Li Fei-Fei, Silvio Savarese, Yuke Zhu

Presented by: Abigail Rafter, Joshua Friesen

The Authors

- **Chen Wang** - PhD student at Stanford
- **Roberto Martın-Martın** - PhD student at Stanford
- **Danfei Xu** - PhD student at Stanford
- **Jun Lv** - PhD student at Shanghai Jiao Tong University
- **Cewu Lu** - Professor at Shanghai Jiao Tong University
- **Fei-Fei Li** - Professor at Stanford University
- **Silvio Savarese** - Professor at Stanford University
- **Yuke Zhu** - Professor at UT Austin


Robot View
x4

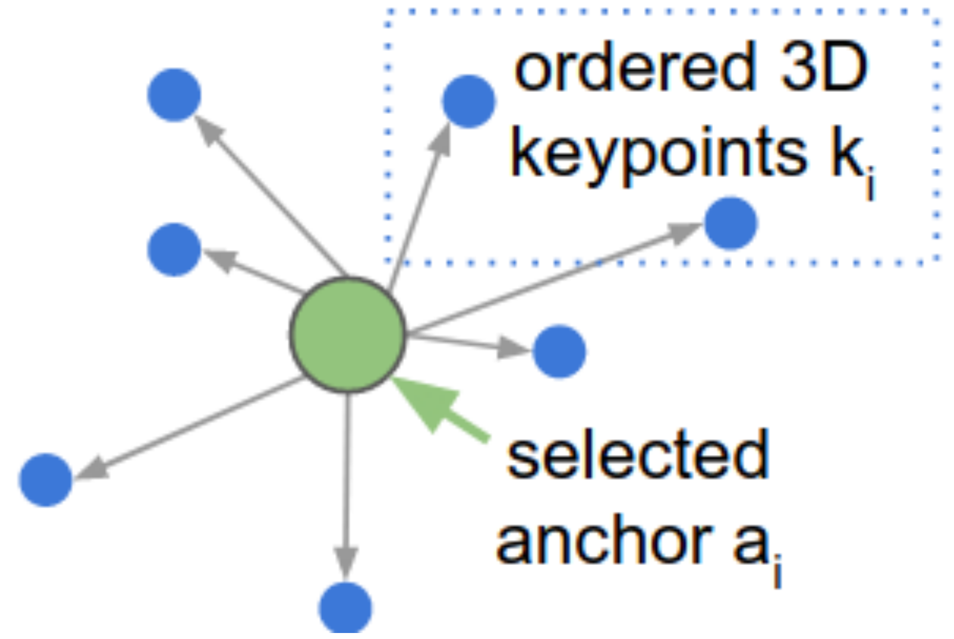
6D Pose Tracking

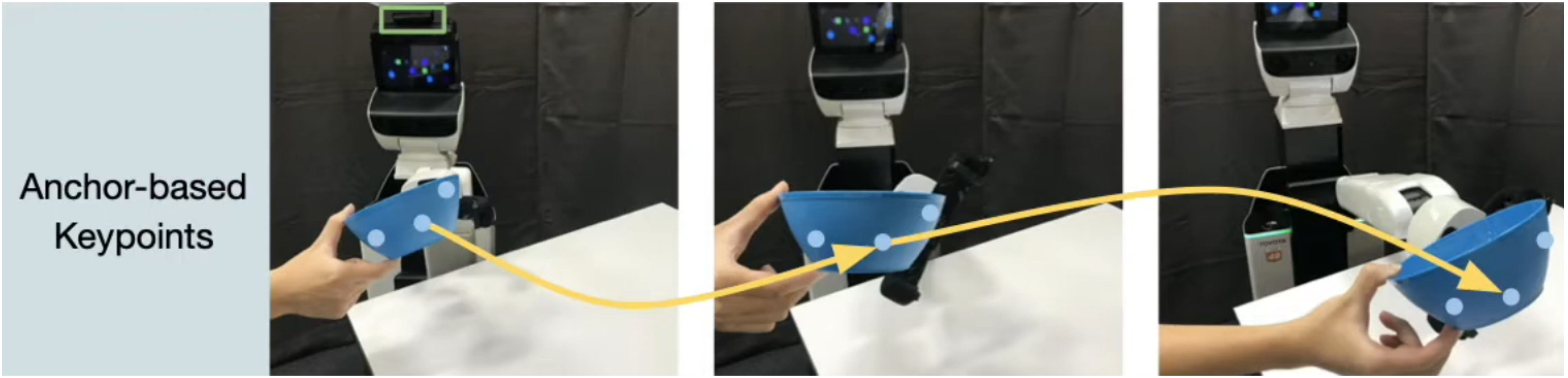
- Common form of state representation for robotics
- Pose tracking in real-time allows for fast feedback control

What Exists	Proposed
• Requires known 3D models	• Category-level 6D tracking • Anchor-based keypoints

Contributions

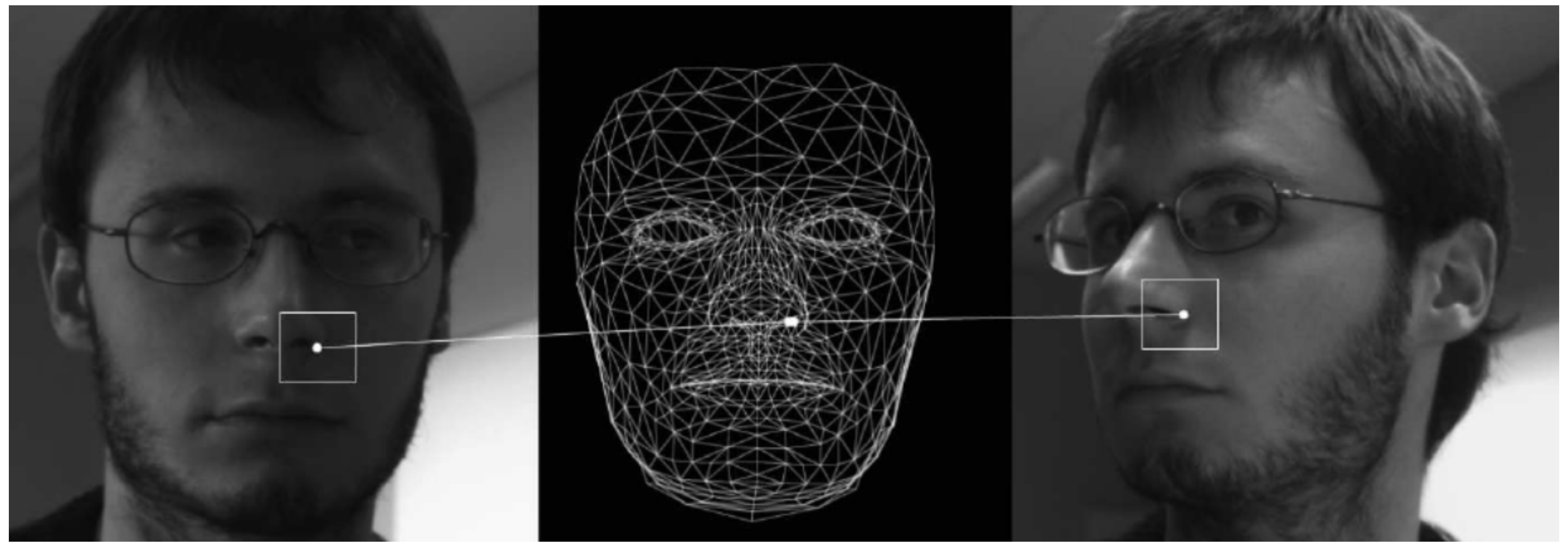
1. Anchor-Based Keypoints
2. Temporal 6D Category-Level Pose Tracking
3. State Of The Art Accuracy & Real Time Performance



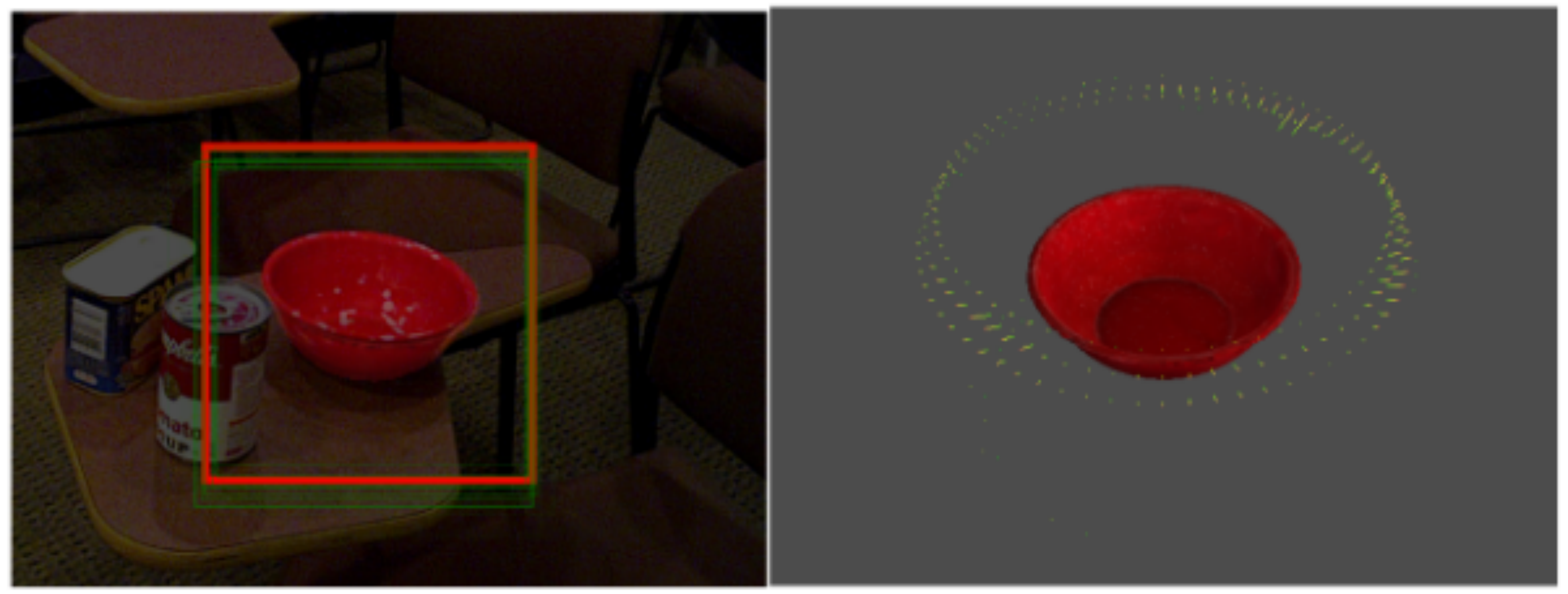


Background

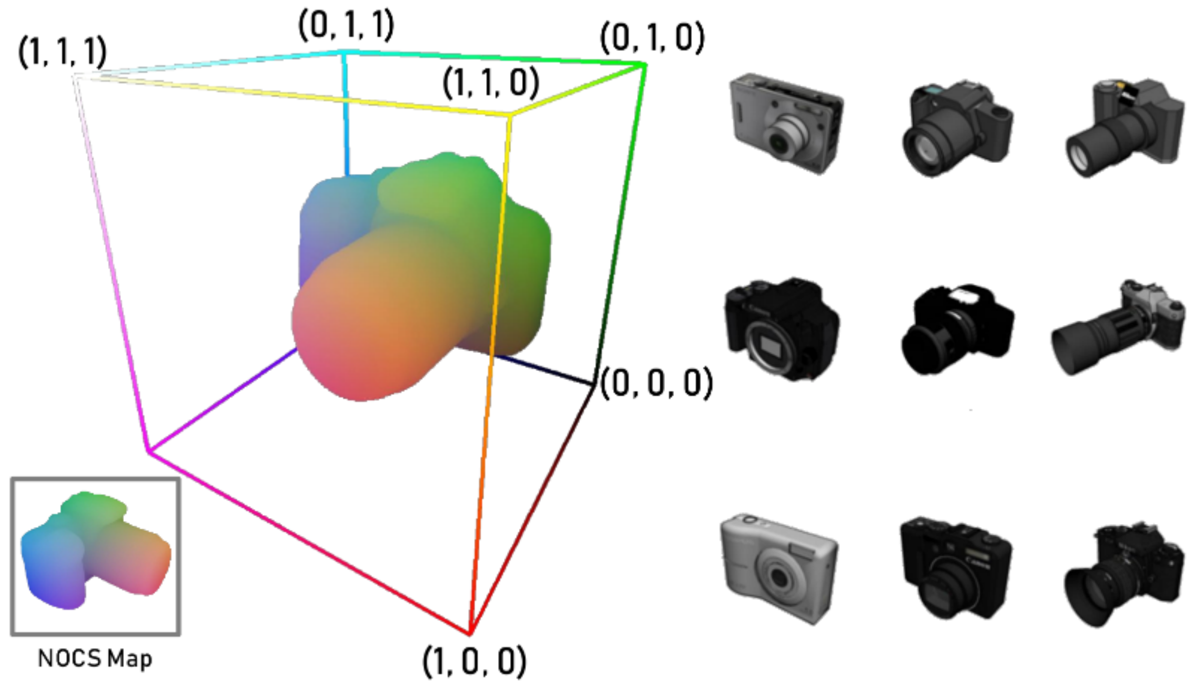
1. Matching View to Template



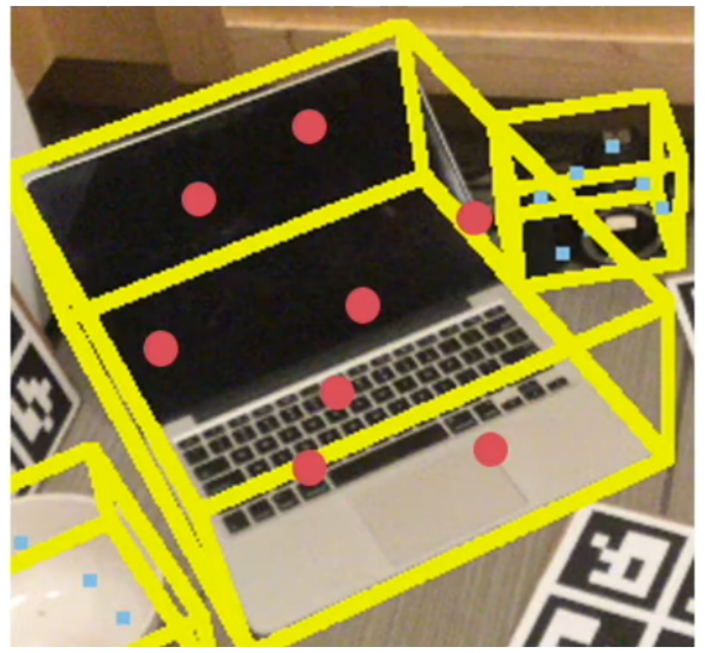
2. Matching View with Render



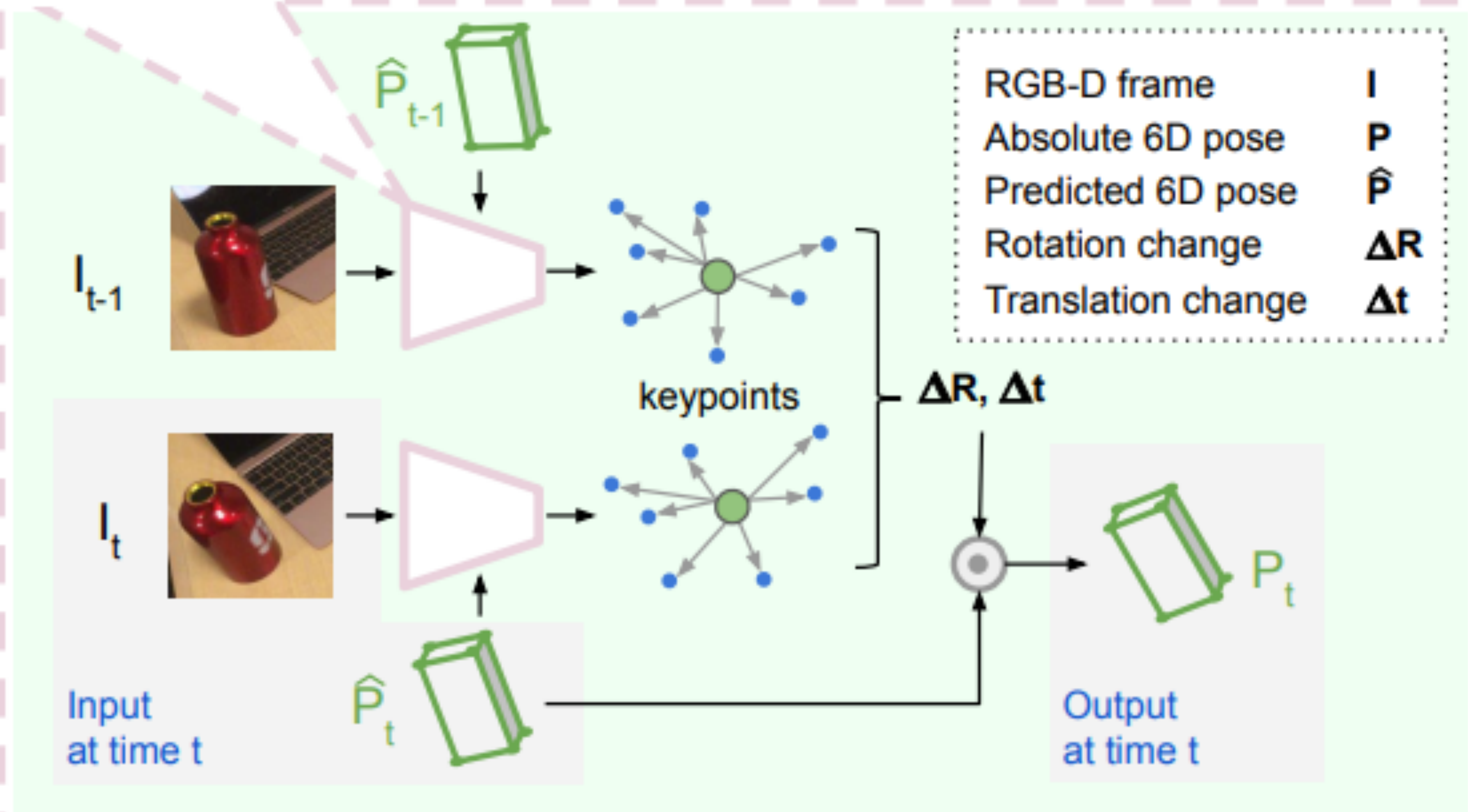
3. Category Level Estimation



4. Anchor-based Keypoints

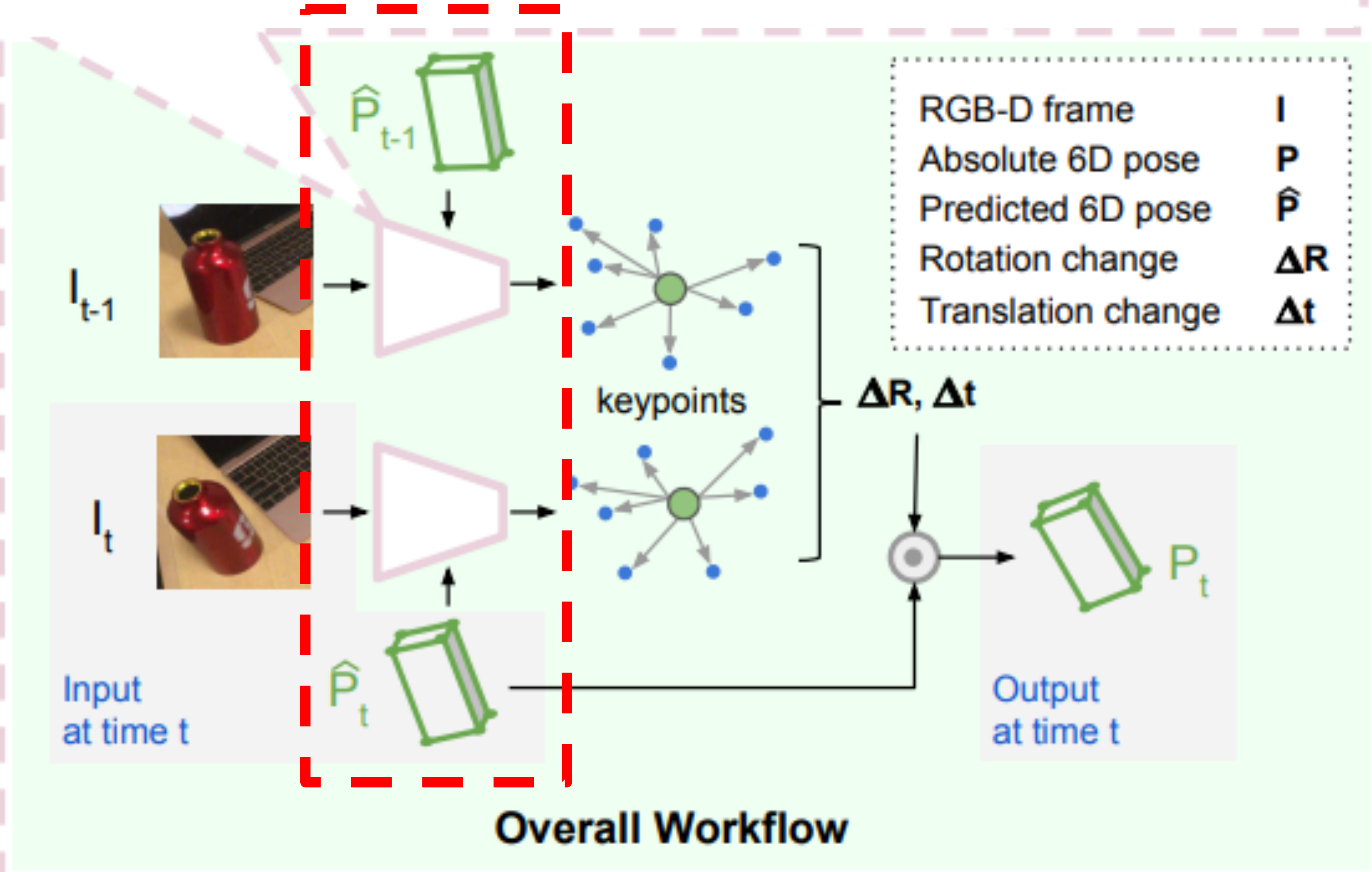


Anchor-Based Keypoint Generation

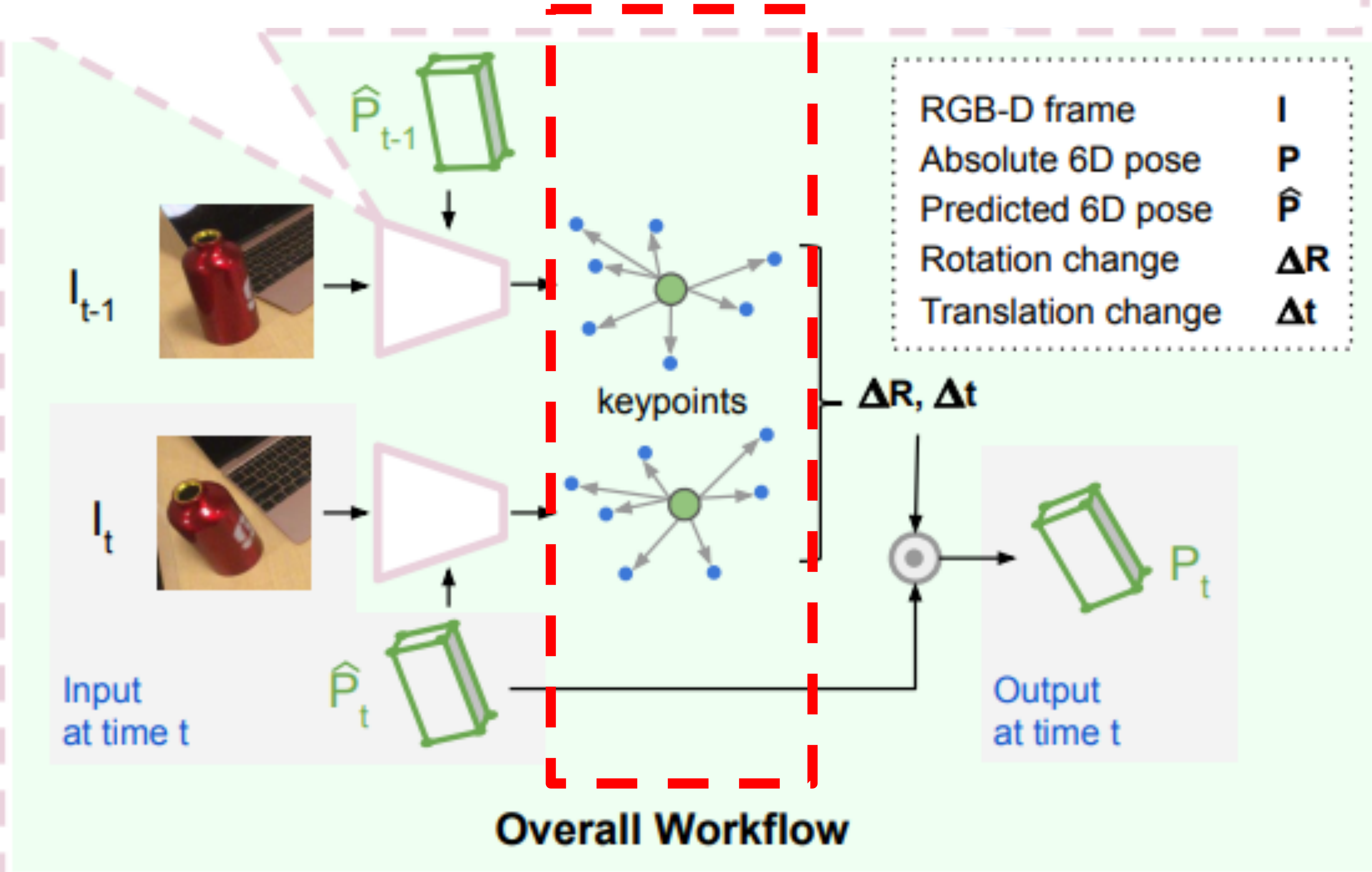


Overall Workflow

Anchor-Based Keypoint Generation



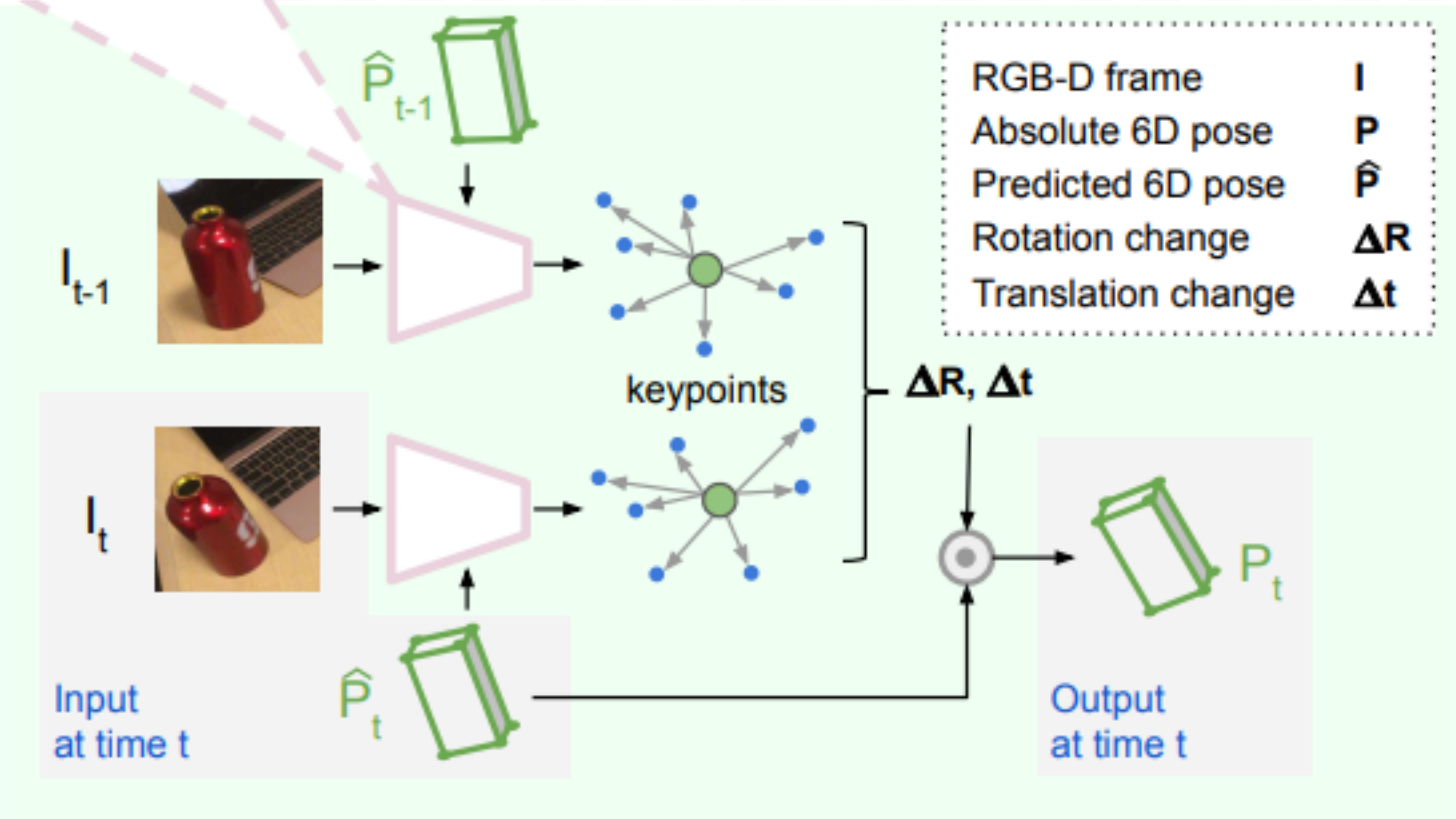
Anchor-Based Keypoint Generation



Anchor-Based Keypoint Generation

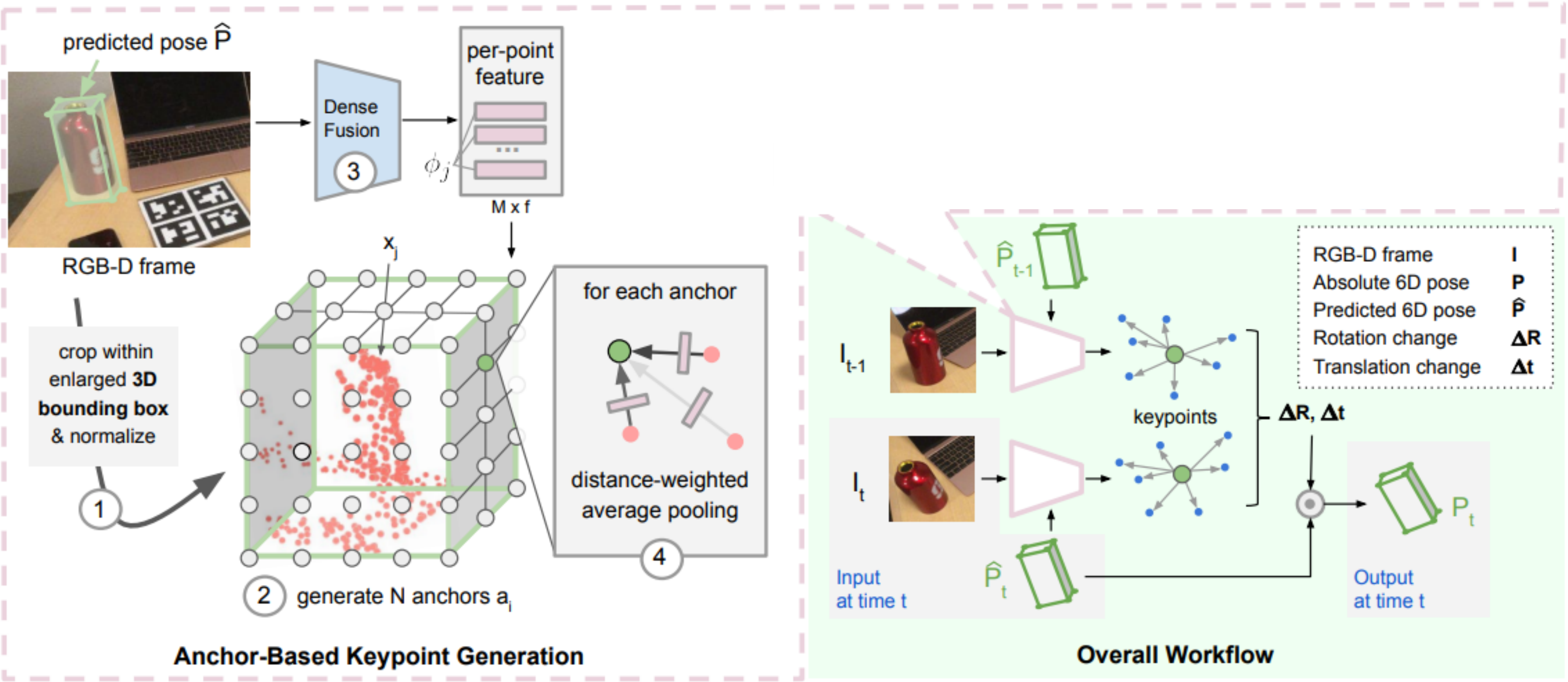
predicted pose $\hat{P}$

RGB-D frame

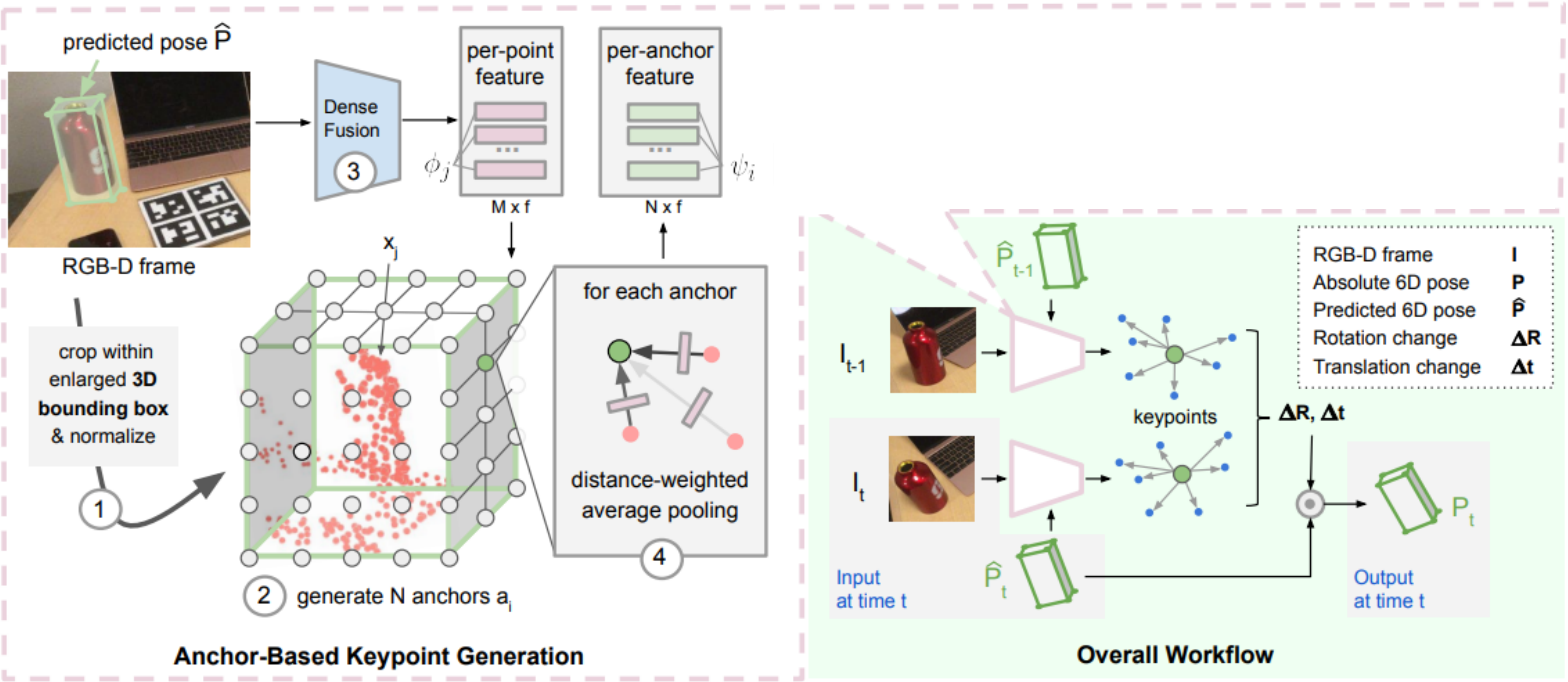


Overall Workflow

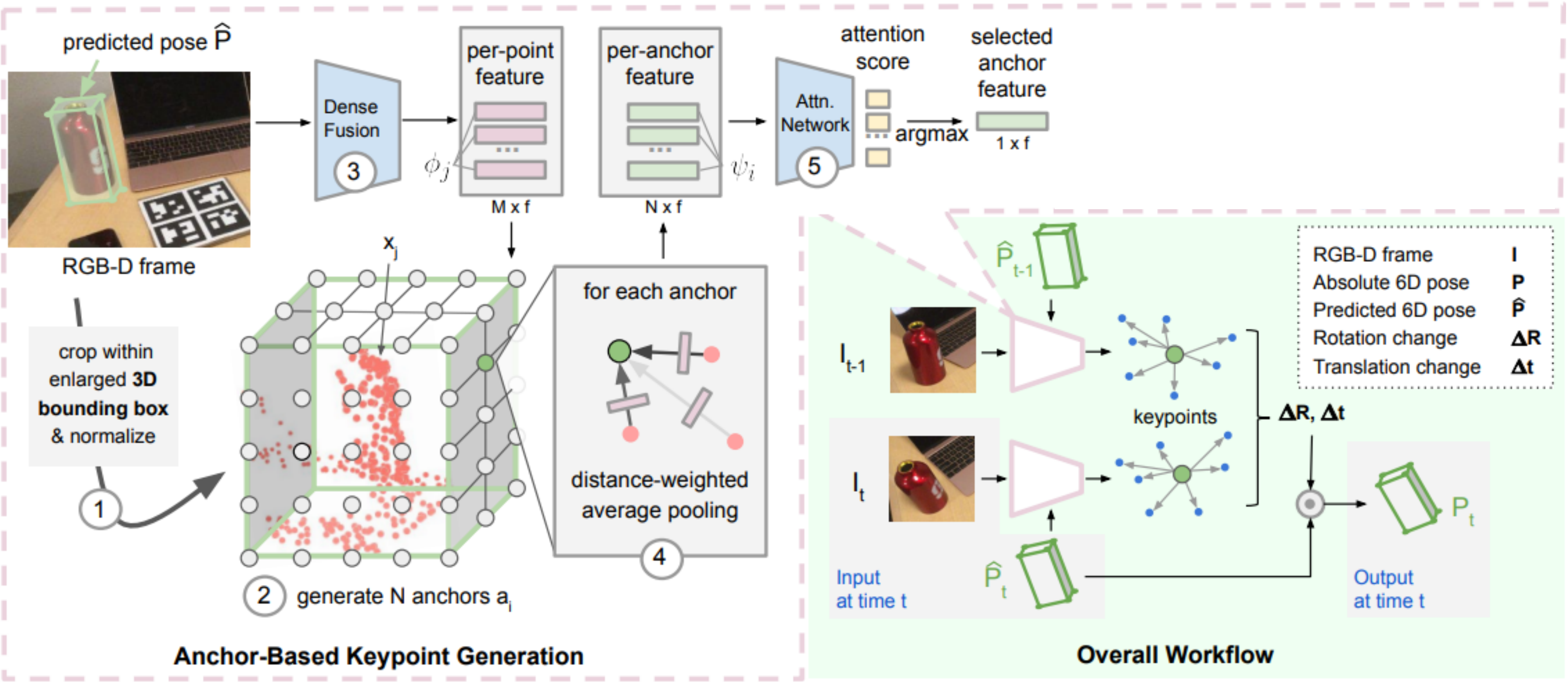
Anchor-Based Keypoint Generation



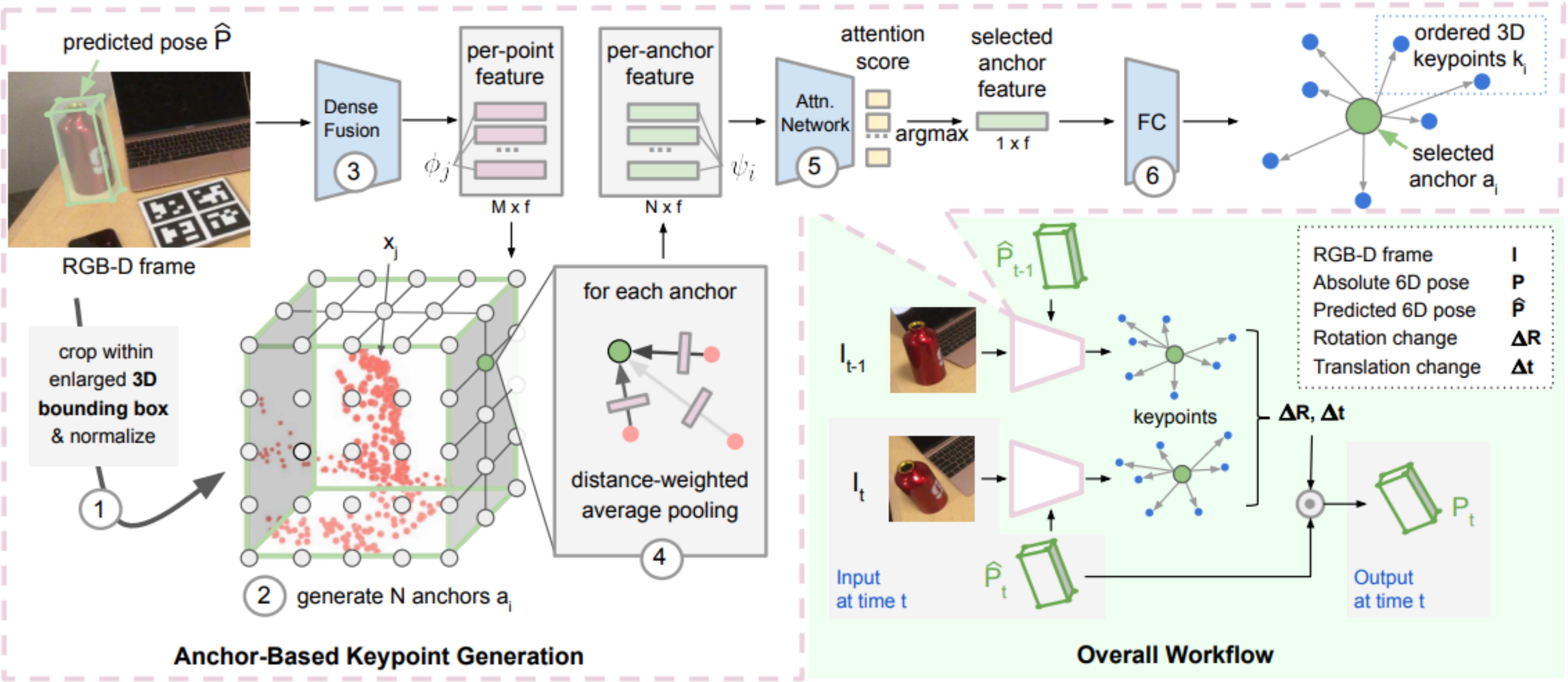
Anchor-Based Keypoint Generation



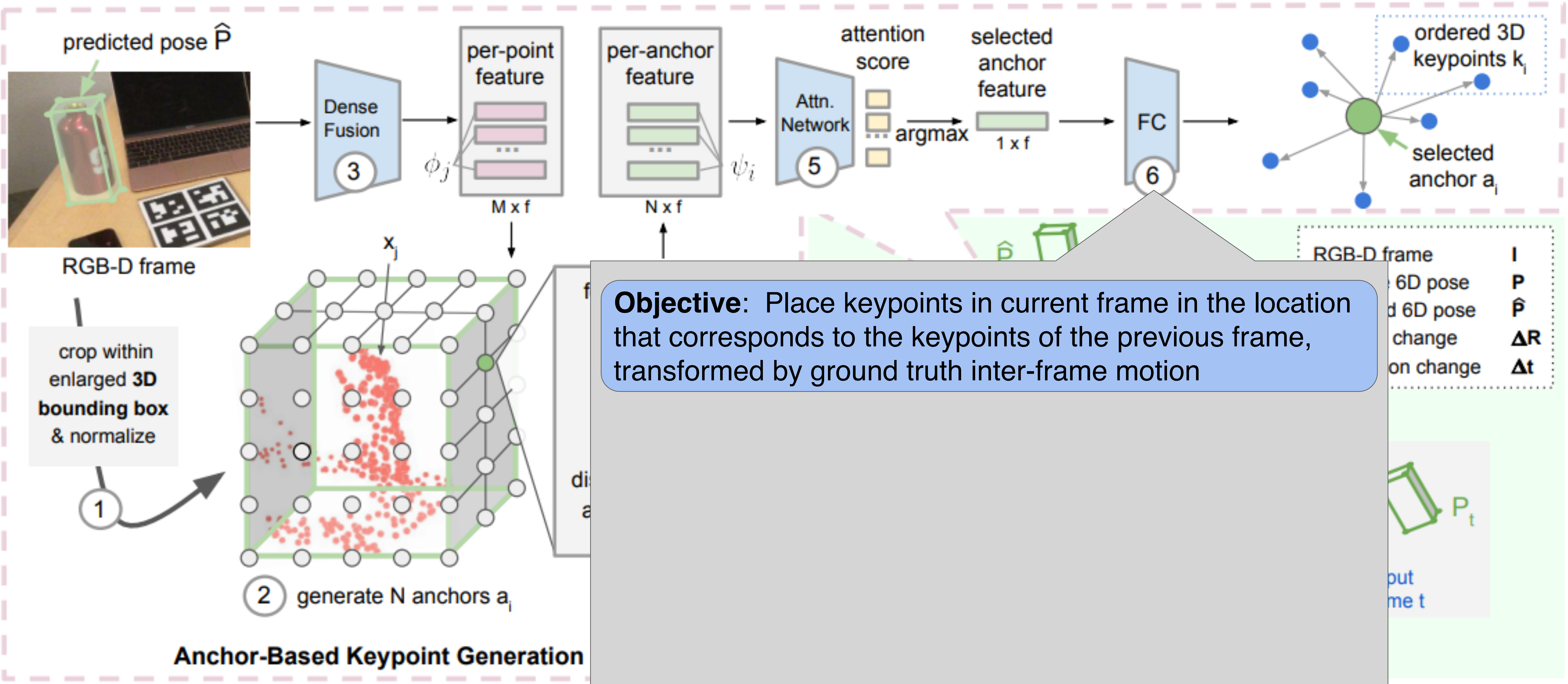
Anchor-Based Keypoint Generation



Anchor-Based Keypoint Generation

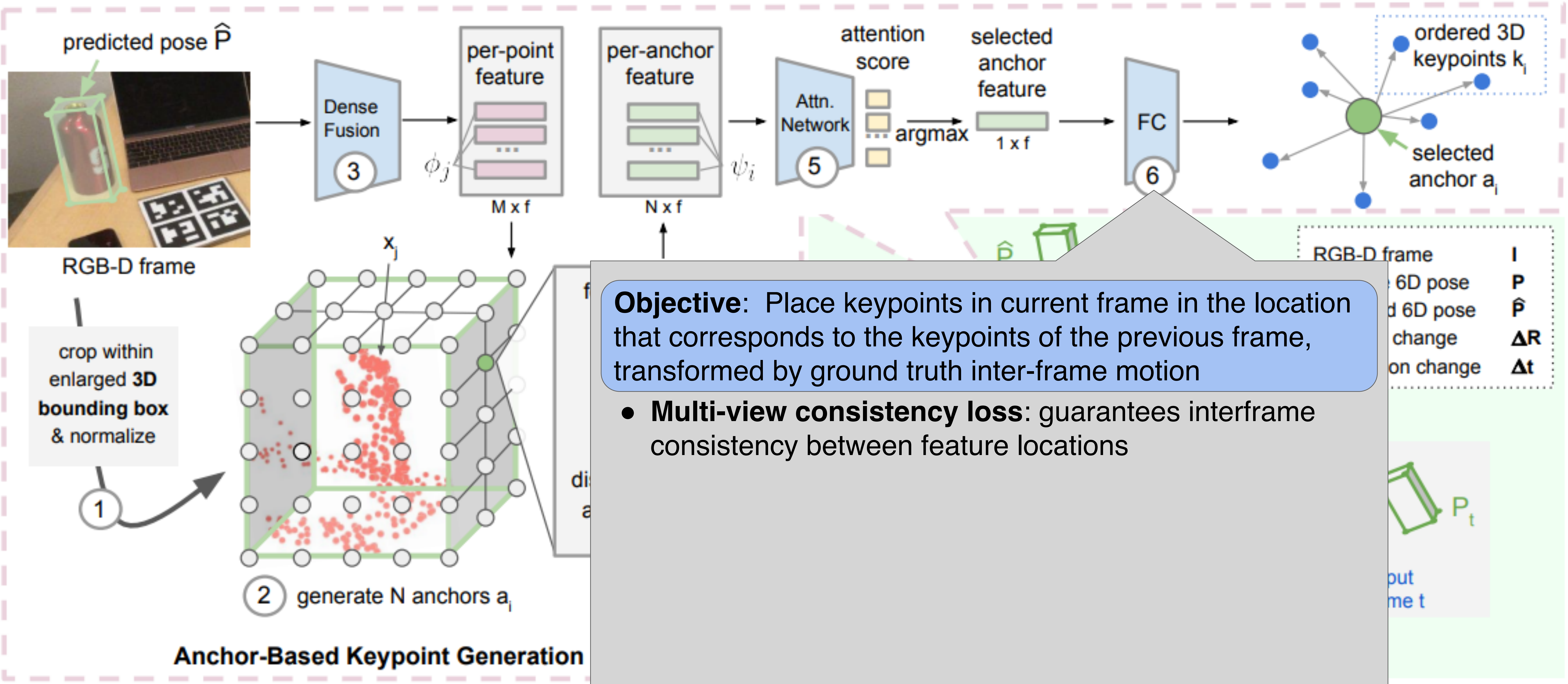


Anchor-Based Keypoint Generation



Objective: Place keypoints in current frame in the location that corresponds to the keypoints of the previous frame, transformed by ground truth inter-frame motion

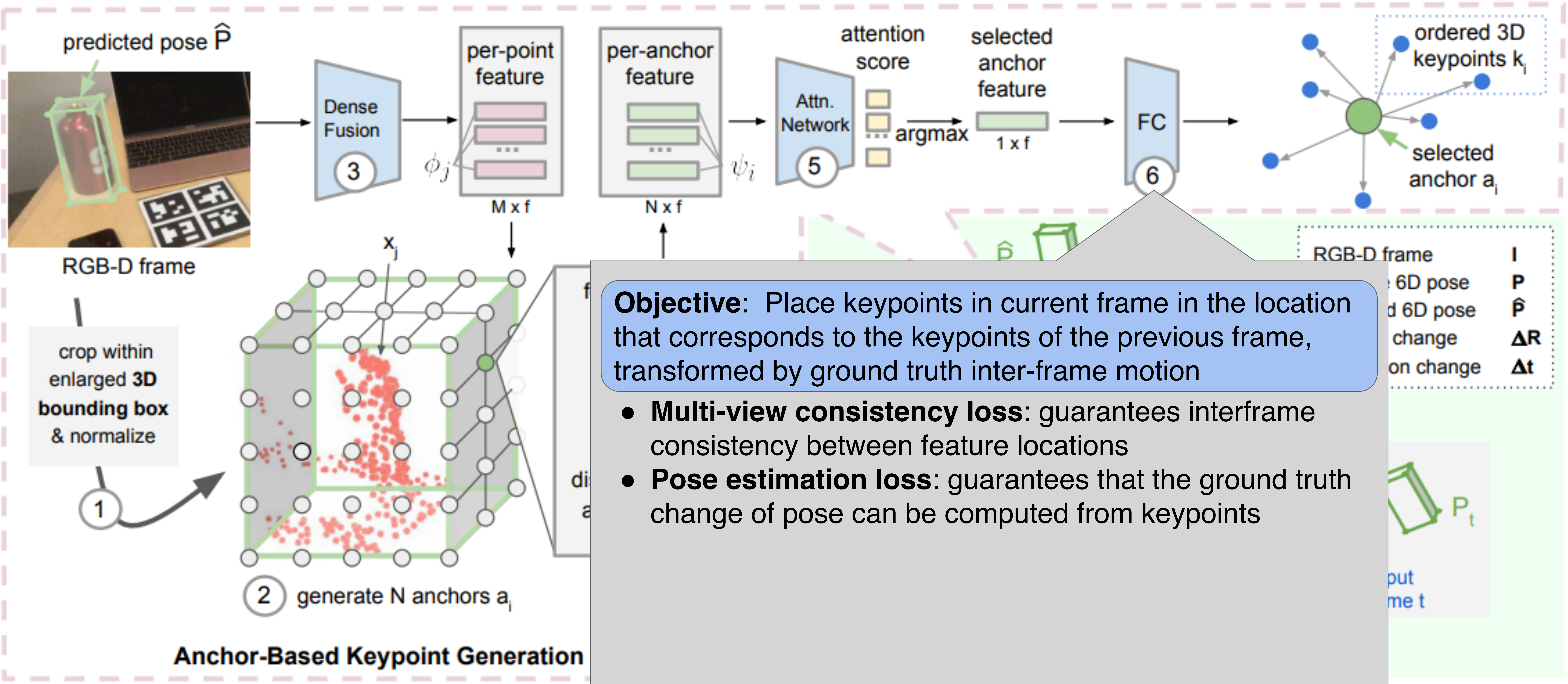
Anchor-Based Keypoint Generation



Objective: Place keypoints in current frame in the location that corresponds to the keypoints of the previous frame, transformed by ground truth inter-frame motion

- **Multi-view consistency loss**: guarantees interframe consistency between feature locations

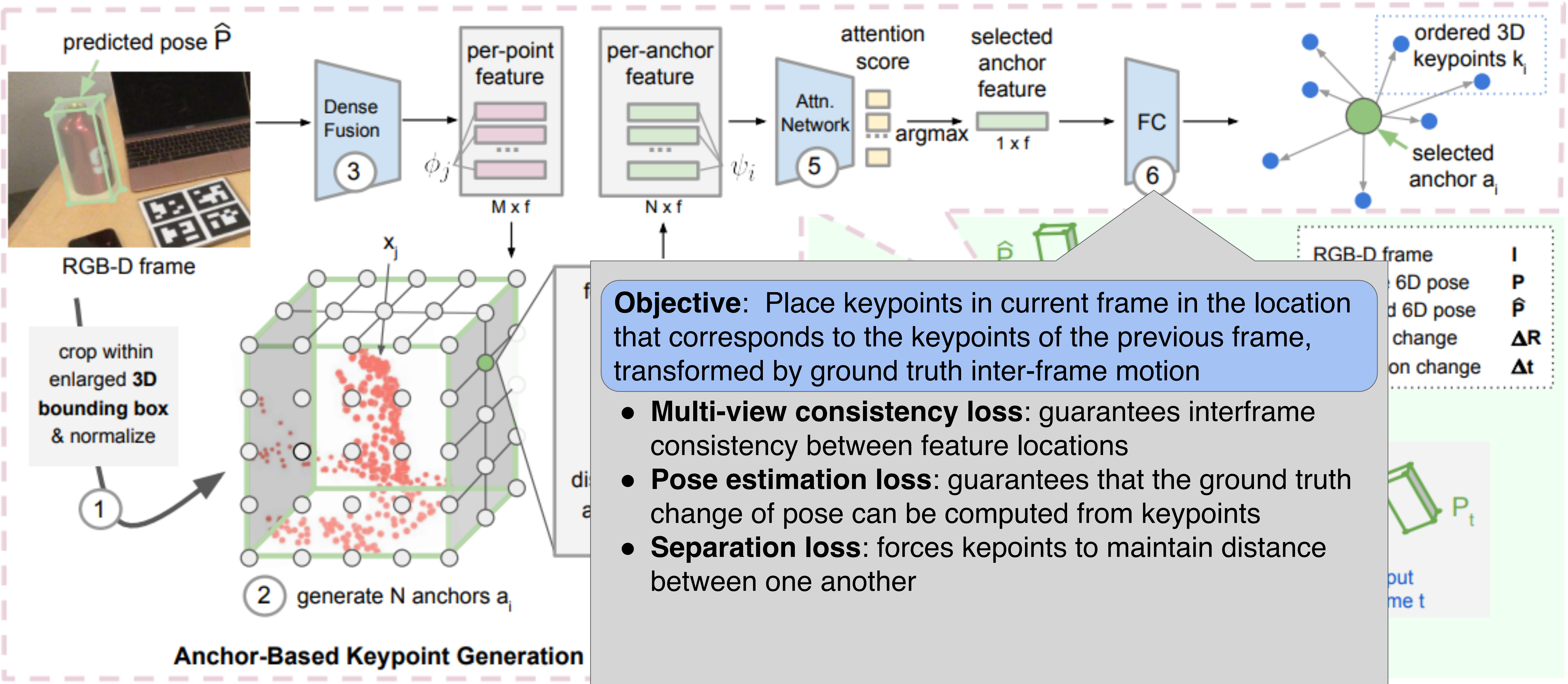
Anchor-Based Keypoint Generation



Objective: Place keypoints in current frame in the location that corresponds to the keypoints of the previous frame, transformed by ground truth inter-frame motion

- **Multi-view consistency loss**: guarantees interframe consistency between feature locations
- **Pose estimation loss**: guarantees that the ground truth change of pose can be computed from keypoints

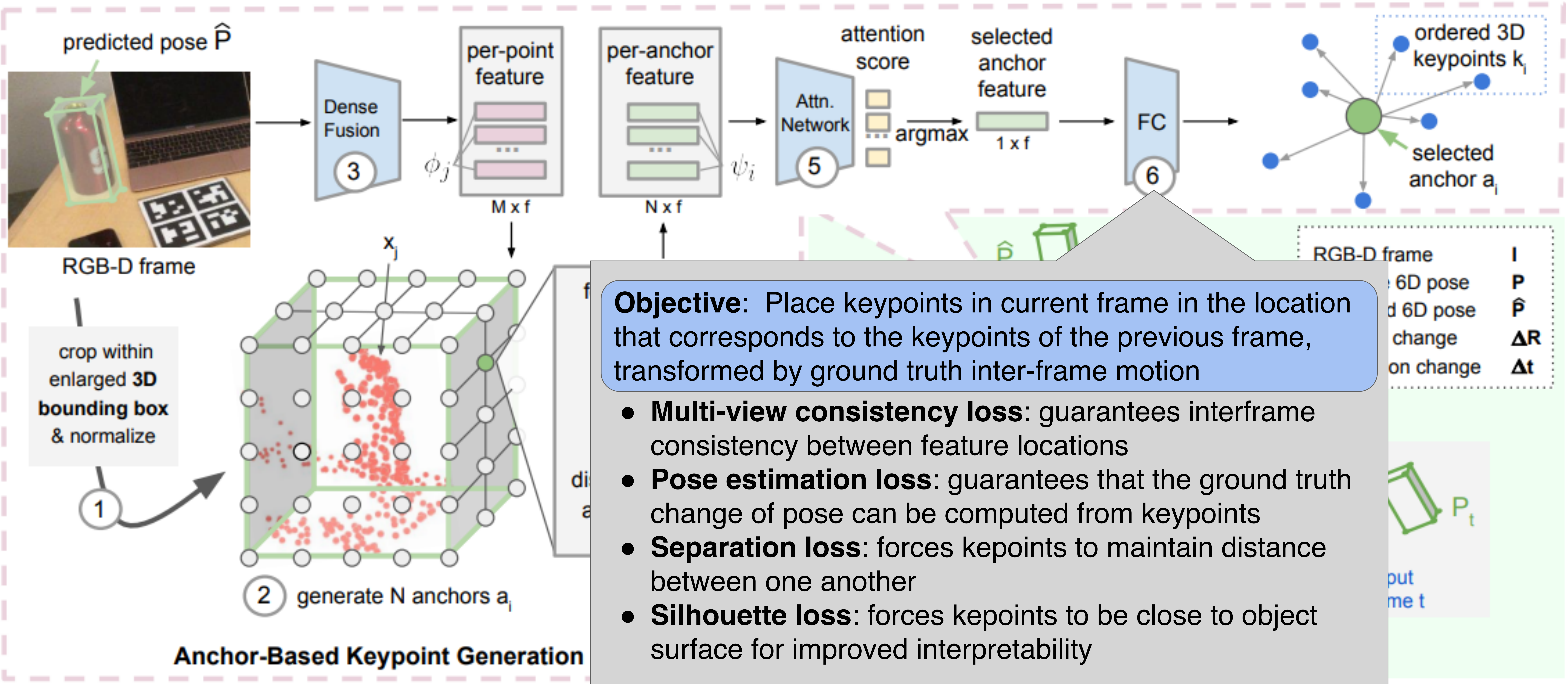
Anchor-Based Keypoint Generation



Objective: Place keypoints in current frame in the location that corresponds to the keypoints of the previous frame, transformed by ground truth inter-frame motion

- **Multi-view consistency loss**: guarantees interframe consistency between feature locations
- **Pose estimation loss**: guarantees that the ground truth change of pose can be computed from keypoints
- **Separation loss**: forces kepoints to maintain distance between one another

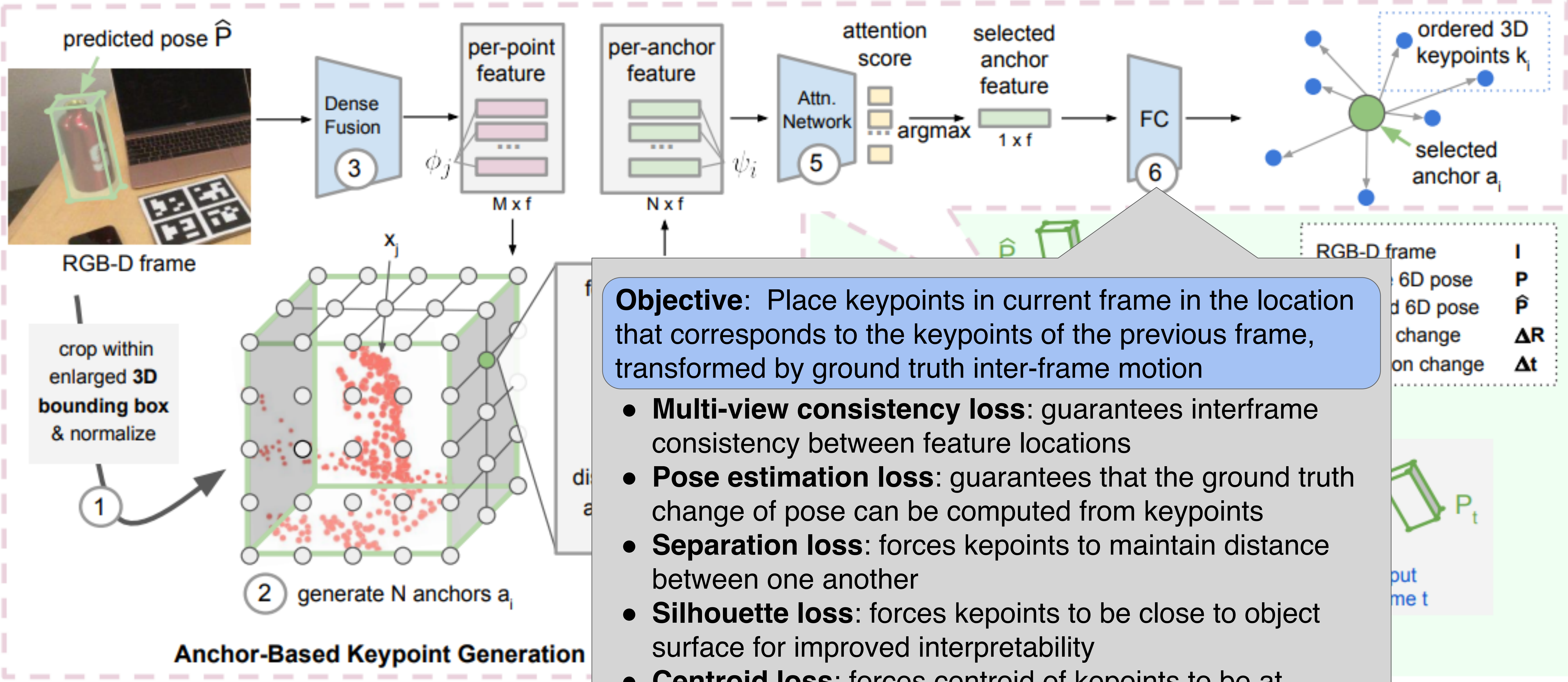
Anchor-Based Keypoint Generation



Objective: Place keypoints in current frame in the location that corresponds to the keypoints of the previous frame, transformed by ground truth inter-frame motion

- **Multi-view consistency loss**: guarantees interframe consistency between feature locations
- **Pose estimation loss**: guarantees that the ground truth change of pose can be computed from keypoints
- **Separation loss**: forces kepoints to maintain distance between one another
- **Silhouette loss**: forces kepoints to be close to object surface for improved interpretability

Anchor-Based Keypoint Generation



Objective: Place keypoints in current frame in the location that corresponds to the keypoints of the previous frame, transformed by ground truth inter-frame motion

- **Multi-view consistency loss**: guarantees interframe consistency between feature locations
- **Pose estimation loss**: guarantees that the ground truth change of pose can be computed from keypoints
- **Separation loss**: forces kepoints to maintain distance between one another
- **Silhouette loss**: forces kepoints to be close to object surface for improved interpretability
- **Centroid loss**: forces centroid of kepoints to be at centroid of object

Experimental Design

Experimental Design

- **Dataset**: NOCS-REAL275

Experimental Design

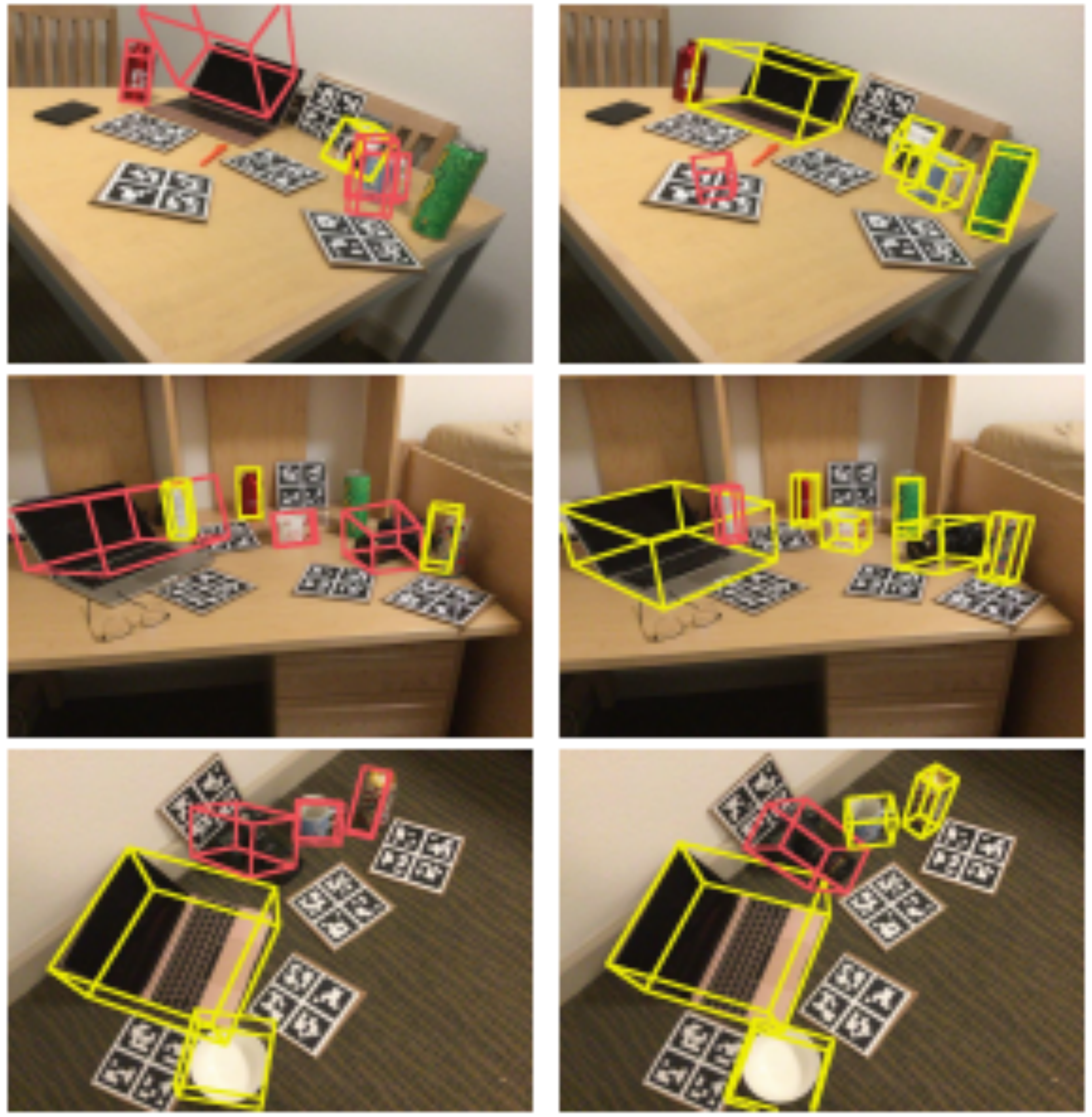
- **Dataset**: NOCS-REAL275
- **Evaluation metrics**:
  - **5°5 cm**: percentage of tracking results with orientation error < 5° and translation error < 5 cm
  - **IoU25**: percentage of volume overlap between the prediction and ground-truth 3D bounding box that is larger than 25%
  - $\mathbf{R_{err}}$: mean of orientation error in degrees
  - $\mathbf{T_{err}}$: mean of translation error in centimeters

Experimental Design

Experimental Design

- **Baselines**:
  - **NOCS** [46]: State-of-the-art category-level 6D pose estimation method that uses per-pixel prediction
  - **ICP** [50]: Standard point-to-plane ICP algorithm implemented in Open3D
  - **KeypointNet** [41]: Implementation of proposed model without the anchor-based attention mechanism
  - **6-PACK without temporal prediction**: Predicted pose in the next frame is the previous estimated pose
  - **6-PACK**: predicted pose in the next frame extrapolates from the last estimated inter-frame change of pose (constant velocity model)

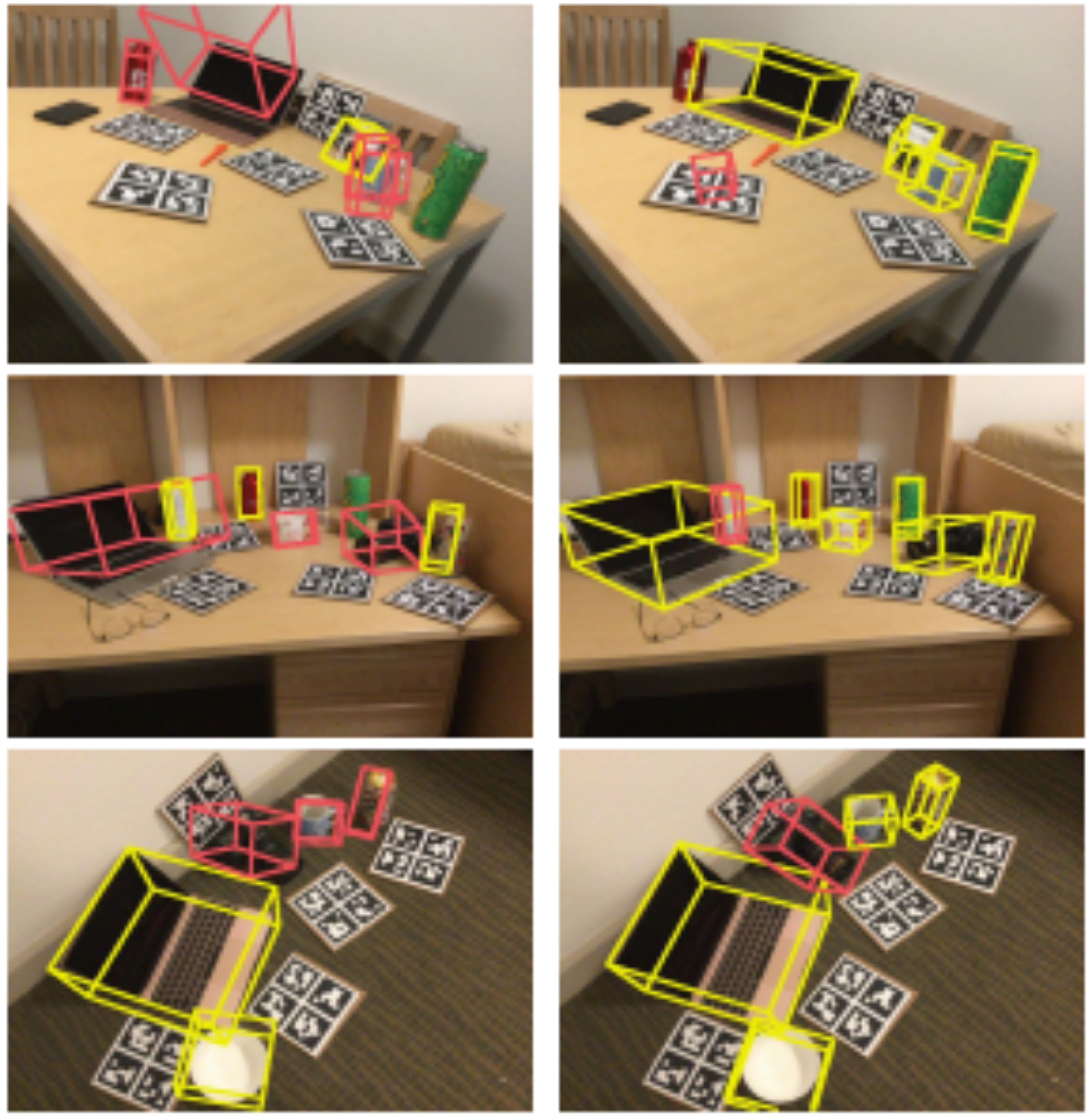
Results



NOCS 6-Pack

		NOCS [46]	ICP [50]	Keypoint Net [41]	Ours w/o temporal	Ours
bottle	5°5cm	5.5	10.1	5.9	23.7	**24.5**
	IoU25	48.7	29.9	23.1	**92.0**	91.1
	R_{err}	25.6	48.0	28.5	15.7	**15.6**
	T_{err}	14.4	15.7	9.5	4.2	**4.0**
bowl	5°5cm	**62.2**	40.3	16.8	53.0	55.0
	IoU25	99.6	79.7	74.7	**100.0**	**100.0**
	R_{err}	**4.7**	19.0	9.8	5.3	5.2
	T_{err}	**1.2**	4.7	8.2	1.6	1.7
camera	5°5cm	0.6	**12.6**	1.8	8.4	10.1
	IoU25	90.6	53.1	30.9	**91.0**	87.6
	R_{err}	**33.8**	80.5	45.2	43.9	35.7
	T_{err}	**3.1**	12.2	8.5	5.5	5.6
can	5°5cm	7.1	17.2	4.3	**25.0**	22.6
	IoU25	77.0	40.5	42.6	89.9	**92.6**
	R_{err}	16.9	47.1	28.8	**12.5**	13.9
	T_{err}	**4.0**	9.4	13.1	5.0	4.8
laptop	5°5cm	25.5	14.8	49.2	62.4	**63.5**
	IoU25	94.7	50.9	94.6	97.8	**98.1**
	R_{err}	8.6	37.7	6.5	4.9	**4.7**
	T_{err}	**2.4**	9.2	4.4	2.5	2.5
mug	5°5cm	0.9	6.2	3.1	22.4	**24.1**
	IoU25	82.8	27.7	52.0	**100.0**	95.2
	R_{err}	31.5	56.3	61.2	**20.3**	21.3
	T_{err}	4.0	9.2	6.7	**1.8**	2.3
Overall	5°5cm	17.0	16.9	13.5	32.5	**33.3**
	IoU25	82.2	47.0	53.0	**95.1**	94.2
	R_{err}	20.2	48.1	30.0	17.1	**16.0**
	T_{err}	4.9	10.5	8.4	**3.4**	3.5

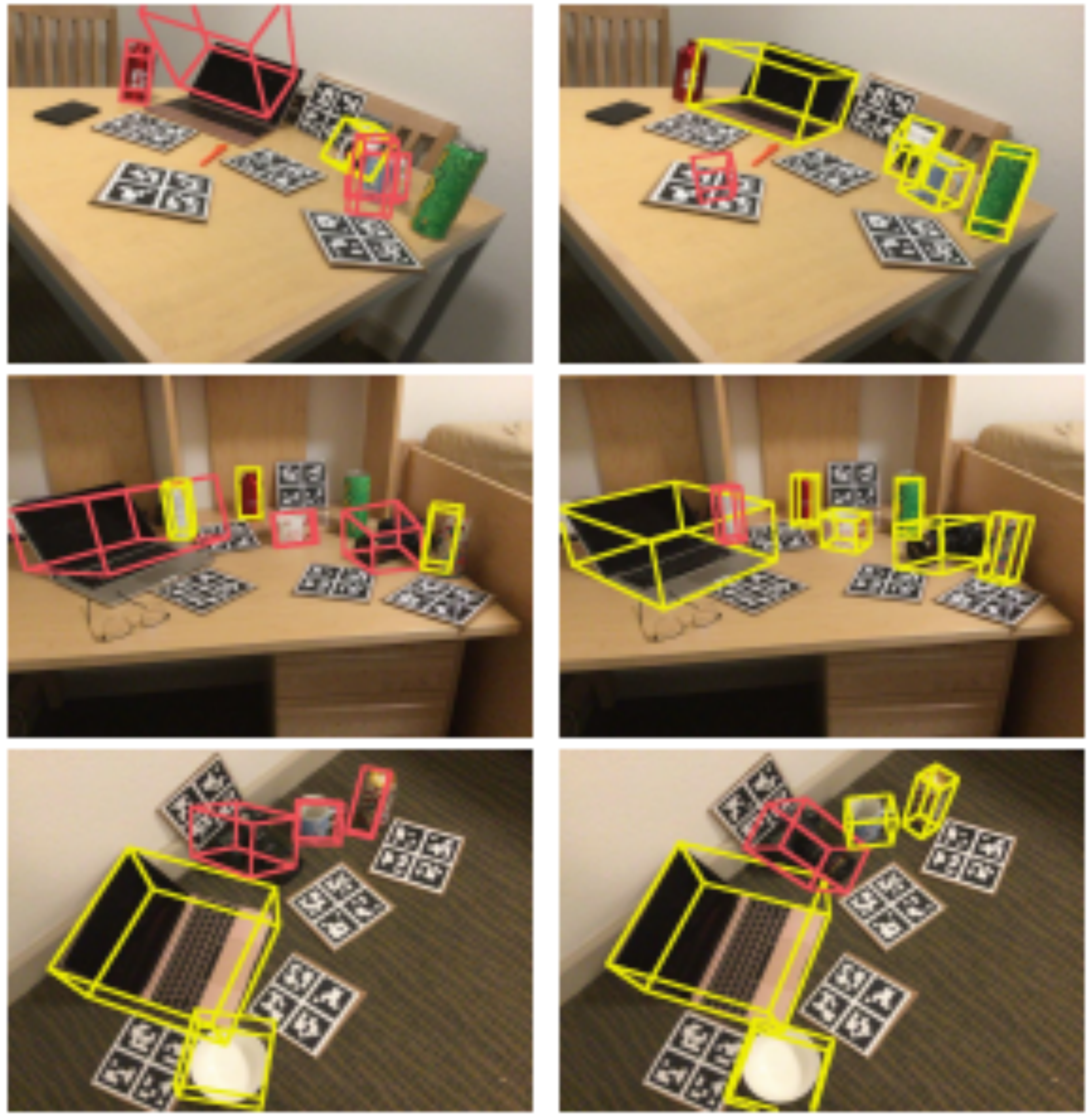
Results



NOCS 6-Pack

		NOCS [46]	ICP [50]	Keypoint Net [41]	Ours w/o temporal	Ours
bottle	5°5cm	5.5	10.1	5.9	23.7	**24.5**
	IoU25	48.7	29.9	23.1	**92.0**	91.1
	R_{err}	25.6	48.0	28.5	15.7	**15.6**
	T_{err}	14.4	15.7	9.5	4.2	**4.0**
bowl	5°5cm	**62.2**	40.3	16.8	53.0	55.0
	IoU25	99.6	79.7	74.7	**100.0**	**100.0**
	R_{err}	**4.7**	19.0	9.8	5.3	5.2
	T_{err}	**1.2**	4.7	8.2	1.6	1.7
camera	5°5cm	0.6	**12.6**	1.8	8.4	10.1
	IoU25	90.6	53.1	30.9	**91.0**	87.6
	R_{err}	**33.8**	80.5	45.2	43.9	35.7
	T_{err}	**3.1**	12.2	8.5	5.5	5.6
can	5°5cm	7.1	17.2	4.3	**25.0**	22.6
	IoU25	77.0	40.5	42.6	89.9	**92.6**
	R_{err}	16.9	47.1	28.8	**12.5**	13.9
	T_{err}	**4.0**	9.4	13.1	5.0	4.8
laptop	5°5cm	25.5	14.8	49.2	62.4	**63.5**
	IoU25	94.7	50.9	94.6	97.8	**98.1**
	R_{err}	8.6	37.7	6.5	4.9	**4.7**
	T_{err}	**2.4**	9.2	4.4	2.5	2.5
mug	5°5cm	0.9	6.2	3.1	22.4	**24.1**
	IoU25	82.8	27.7	52.0	**100.0**	95.2
	R_{err}	31.5	56.3	61.2	**20.3**	21.3
	T_{err}	4.0	9.2	6.7	**1.8**	2.3
Overall	5°5cm	17.0	16.9	13.5	32.5	**33.3**
	IoU25	82.2	47.0	53.0	**95.1**	94.2
	R_{err}	20.2	48.1	30.0	17.1	**16.0**
	T_{err}	4.9	10.5	8.4	**3.4**	3.5

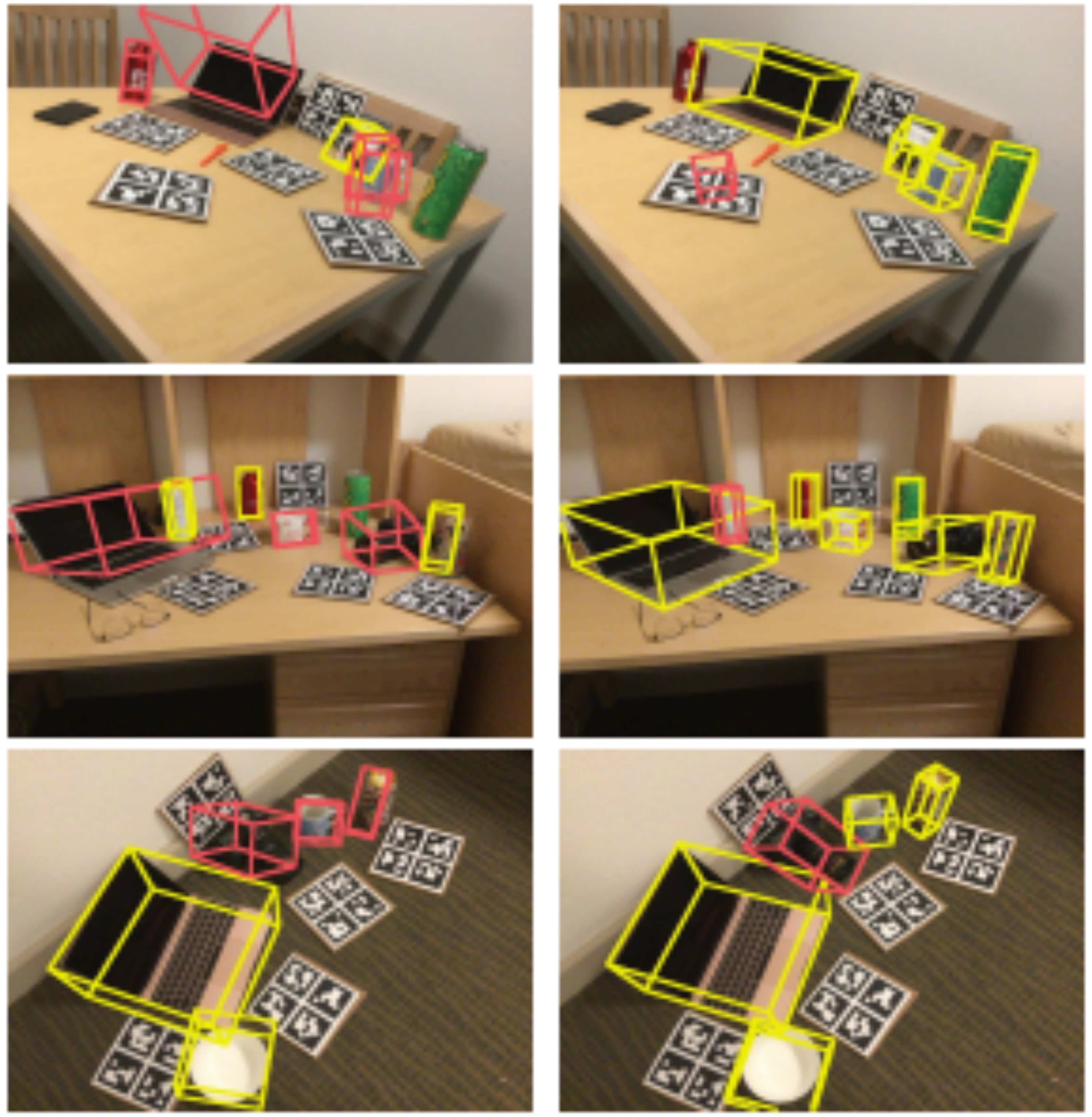
Results



NOCS 6-Pack

		NOCS [46]	ICP [50]	Keypoint Net [41]	Ours w/o temporal	Ours
bottle	5°5cm	5.5	10.1	5.9	23.7	**24.5**
	IoU25	48.7	29.9	23.1	**92.0**	91.1
	R_{err}	25.6	48.0	28.5	15.7	**15.6**
	T_{err}	14.4	15.7	9.5	4.2	**4.0**
bowl	5°5cm	**62.2**	40.3	16.8	53.0	55.0
	IoU25	99.6	79.7	74.7	**100.0**	**100.0**
	R_{err}	**4.7**	19.0	9.8	5.3	5.2
	T_{err}	**1.2**	4.7	8.2	1.6	1.7
camera	5°5cm	0.6	**12.6**	1.8	8.4	10.1
	IoU25	90.6	53.1	30.9	**91.0**	87.6
	R_{err}	**33.8**	80.5	45.2	43.9	35.7
	T_{err}	**3.1**	12.2	8.5	5.5	5.6
can	5°5cm	7.1	17.2	4.3	**25.0**	22.6
	IoU25	77.0	40.5	42.6	89.9	**92.6**
	R_{err}	16.9	47.1	28.8	**12.5**	13.9
	T_{err}	**4.0**	9.4	13.1	5.0	4.8
laptop	5°5cm	25.5	14.8	49.2	62.4	**63.5**
	IoU25	94.7	50.9	94.6	97.8	**98.1**
	R_{err}	8.6	37.7	6.5	4.9	**4.7**
	T_{err}	**2.4**	9.2	4.4	2.5	2.5
mug	5°5cm	0.9	6.2	3.1	22.4	**24.1**
	IoU25	82.8	27.7	52.0	**100.0**	95.2
	R_{err}	31.5	56.3	61.2	**20.3**	21.3
	T_{err}	4.0	9.2	6.7	**1.8**	2.3
Overall	5°5cm	17.0	16.9	13.5	32.5	**33.3**
	IoU25	82.2	47.0	53.0	**95.1**	94.2
	R_{err}	20.2	48.1	30.0	17.1	**16.0**
	T_{err}	4.9	10.5	8.4	**3.4**	3.5

Results



		NOCS [46]	ICP [50]	Keypoint Net [41]	Ours w/o temporal	Ours
bottle	5°5cm	5.5	10.1	5.9	23.7	**24.5**
	IoU25	48.7	29.9	23.1	**92.0**	91.1
	R_{err}	25.6	48.0	28.5	15.7	**15.6**
	T_{err}	14.4	15.7	9.5	4.2	**4.0**
bowl	5°5cm	**62.2**	40.3	16.8	53.0	55.0
	IoU25	99.6	79.7	74.7	**100.0**	**100.0**
	R_{err}	**4.7**	19.0	9.8	5.3	5.2
	T_{err}	**1.2**	4.7	8.2	1.6	1.7
camera	5°5cm	0.6	**12.6**	1.8	8.4	10.1
	IoU25	90.6	53.1	30.9	**91.0**	87.6
	R_{err}	**33.8**	80.5	45.2	43.9	35.7
	T_{err}	**3.1**	12.2	8.5	5.5	5.6
can	5°5cm	7.1	17.2	4.3	**25.0**	22.6
	IoU25	77.0	40.5	42.6	89.9	**92.6**
	R_{err}	16.9	47.1	28.8	**12.5**	13.9
	T_{err}	**4.0**	9.4	13.1	5.0	4.8
laptop	5°5cm	25.5	14.8	49.2	62.4	**63.5**
	IoU25	94.7	50.9	94.6	97.8	**98.1**
	R_{err}	8.6	37.7	6.5	4.9	**4.7**
	T_{err}	**2.4**	9.2	4.4	2.5	2.5
mug	5°5cm	0.9	6.2	3.1	22.4	**24.1**
	IoU25	82.8	27.7	52.0	**100.0**	95.2
	R_{err}	31.5	56.3	61.2	**20.3**	21.3
	T_{err}	4.0	9.2	6.7	**1.8**	2.3
Overall	5°5cm	17.0	16.9	13.5	32.5	**33.3**
	IoU25	82.2	47.0	53.0	**95.1**	94.2
	R_{err}	20.2	48.1	30.0	17.1	**16.0**
	T_{err}	4.9	10.5	8.4	**3.4**	3.5

Conclusions

- **Summary**: Anchor-based keypoint generation for 6D pose tracking
- 6-PACK demonstrates state-of-the-art performance on a challenging category-based 6D object pose tracking problem
- 6-PACK enables real-time tracking and robot interaction

Limitations and Future Work

- Only works on RGB-D data
- 10 Hz pose tracking on robot
- Only trained on 6 categories of objects

References

[41] S. Suwajanakorn, N. Snavely, J. Tompson, and M. Norouzi, "Discovery of latent 3d keypoints via end-to-end geometric reasoning," arXiv preprint arXiv:1807.03146, 2018.

[46] H. Wang, S. Sridhar, J. Huang, J. Valentin, S. Song, and L. J. Guibas, "Normalized object coordinate space for category level 6d object pose and size estimation," arXiv preprint arXiv:1901.02970, 2019.

[50] Q.-Y. Zhou, J. Park, and V. Koltun, "Open3D: A modern library for 3D data processing," arXiv:1801.09847, 2018.

Thank you

DR

Next Time: Visual Odometry and Localization

- Seminar 5: Recurrent Networks and Object Tracking
  1. DeepIM: Deep Iterative Matching for 6D Pose Estimation, Li et al., 2018
  2. PoseRBPF: A Rao-Blackwellized Particle Filter for 6D Object Pose Tracking, Deng et al., 2019
  3. 6-PACK: Category-level 6D Pose Tracker with Anchor-Based Keypoints, Wang et al., 2020
  4. XMem: Long-Term Video Object Segmentation with an Atkinson-Shiffrin Memory Model, Cheng and Schwing, 2022

- Seminar 6: Visual Odometry and Localization
  1. Backprop KF: Learning Discriminative Deterministic State Estimators, Haarnoja et al., 2016
  2. Differentiable Particle Filters: End-to-End Learning with Algorithmic Priors, Jonschkowski et al., 2018
  3. Multimodal Sensor Fusion with Differentiable Filters, Lee et al., 2020
  4. Differentiable SLAM-net: Learning Particle SLAM for Visual Navigation, Karkus et al., 2021

DR

DeepRob

Seminar 5
Object Tracking
University of Michigan and University of Minnesota