Time-Contrastive Networks: Self-Supervised Learning from Video

CAP6412 Advanced Computer Vision

Furkan Çimen

(slides are modified from https://sermanet.github.io/imitate/)

Motivation

- To learn a deep learning embedding for an image
  - Map an image to 32 dimensional vector
- Which is invariant to:
  - View
  - Agent
  - Scale
  - Background
  - Orientation

Multi-view capture



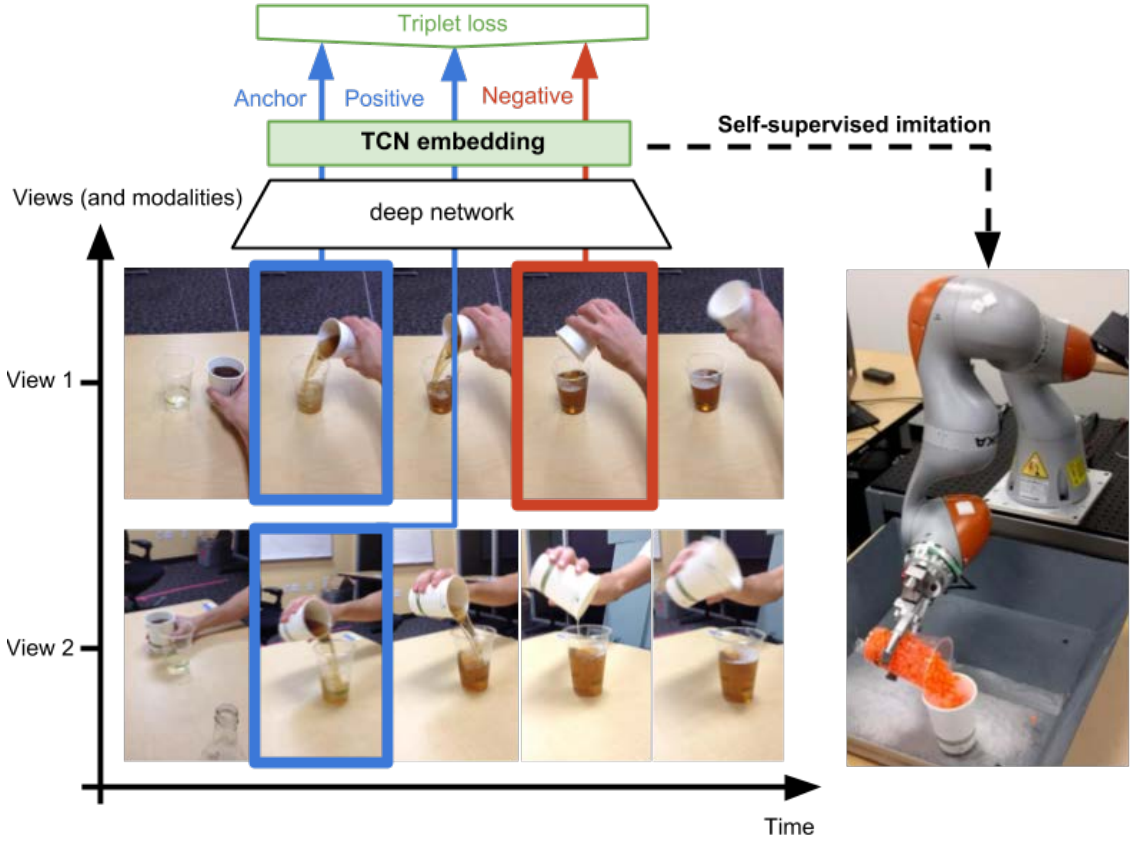


1st-person view

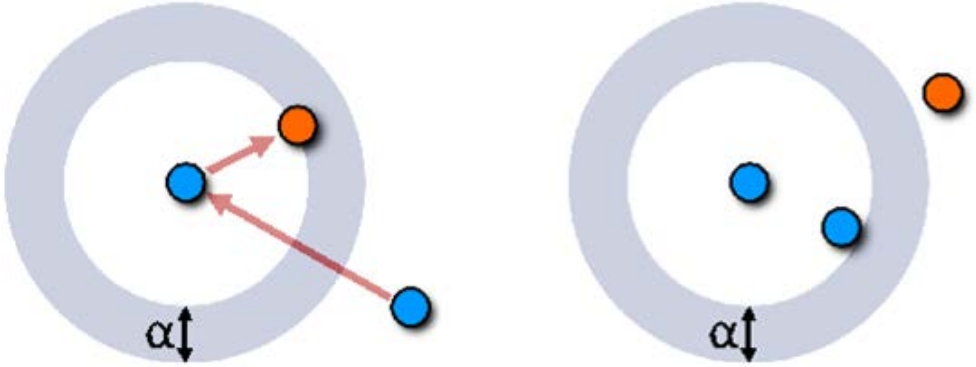


3rd-person view

Time-Contrastive Networks (TCN)



Time-Contrastive Networks (TCN)

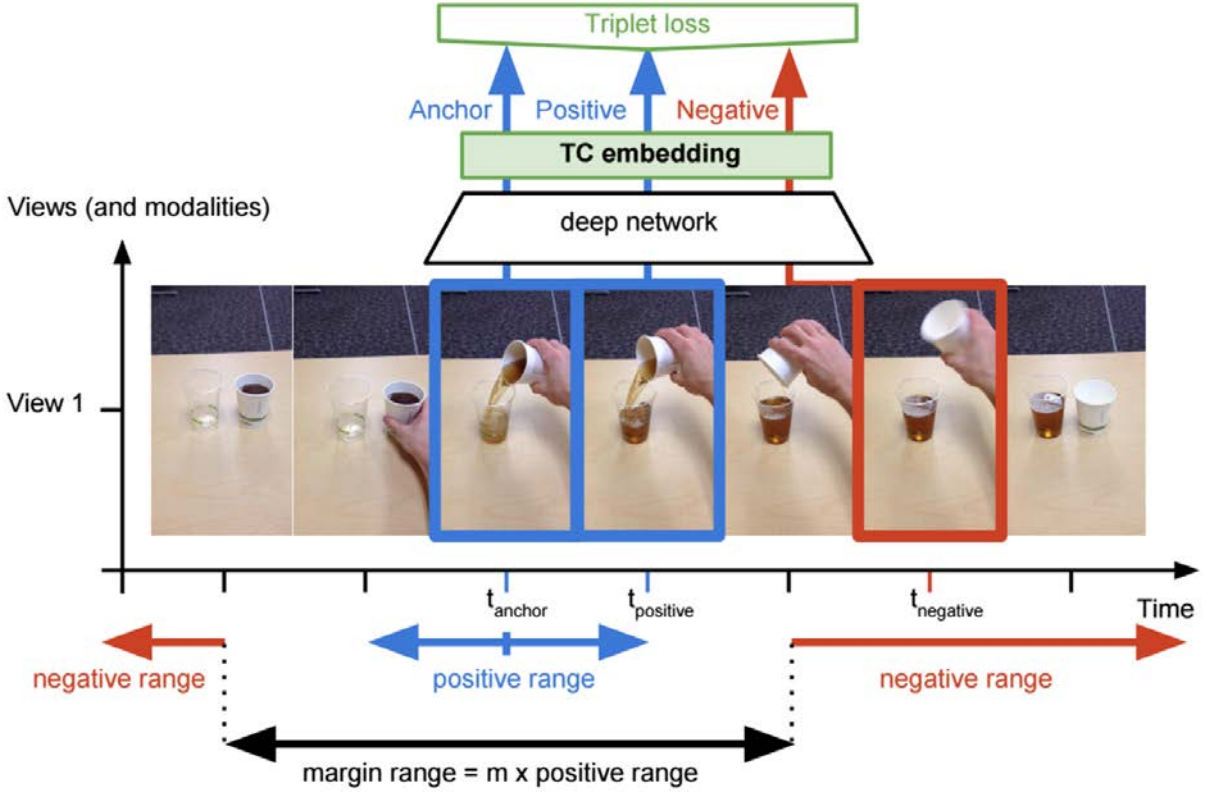


(a) Triplet: before. (b) Triplet: after.

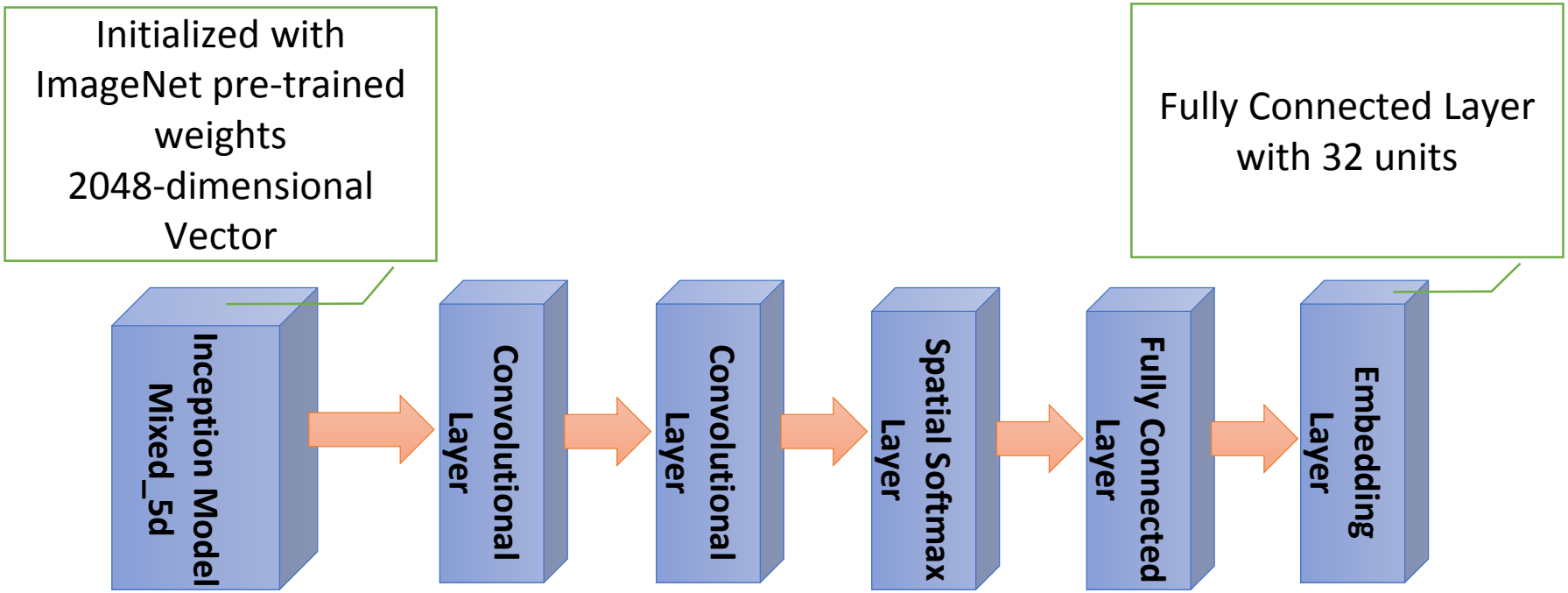
$$\|f(x_i^a) - f(x_i^p)\|_2^2 + \alpha < \|f(x_i^a) - f(x_i^n)\|_2^2 \quad \forall (f(x_i^a), f(x_i^p), f(x_i^n)) \in \mathcal{T}$$

$$\sum_i^N \left[\|f(x_i^a) - f(x_i^p)\|_2^2 - \|f(x_i^a) - f(x_i^n)\|_2^2 + \alpha \right]_+$$

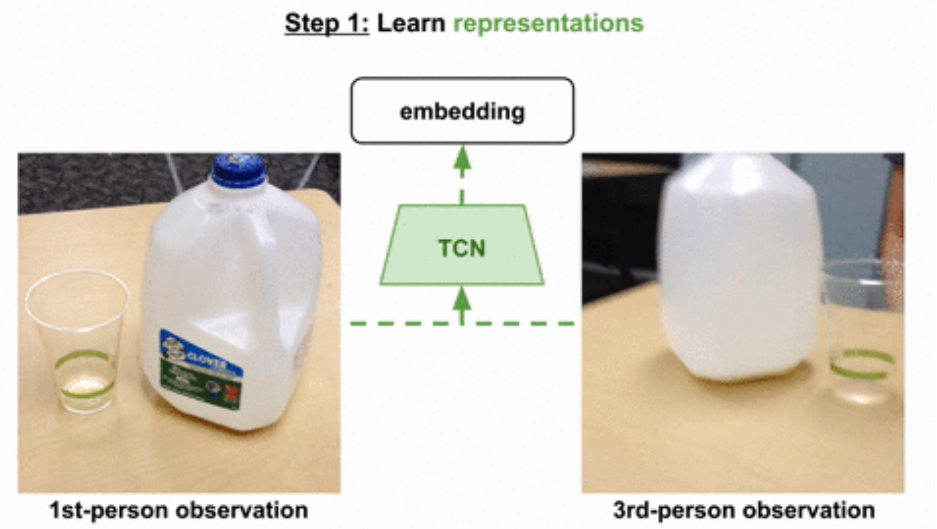
Single-View TCN

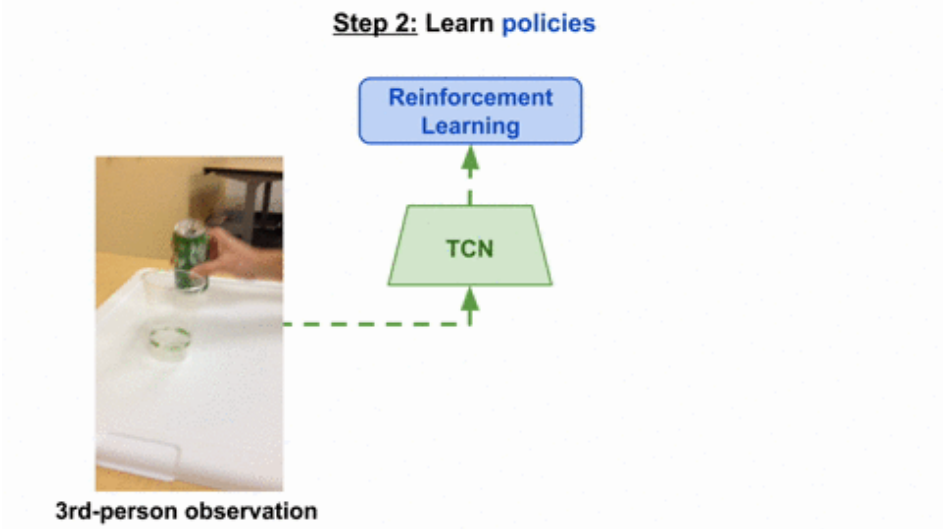


TCN Structure



Approach (pouring, real)





Attribute Classifiers for Pouring

- Is hand in contact with the container? (yes or no)
- Is the Container within pouring distance of the recipient? (yes or no)
- What is the tilt angle of the pouring container? (values 90 45 0 and -45 degrees)
- Is the liquid flowing? (yes or no)
- Does the recipient contain liquid? (yes or no)

Quantitative analysis: Individual Attributes

Method	hand contact with container	within pouring distance	container angle	liquid is flowing	recipient has liquid
Random	49.9%	48.9%	74.5%	49.2%	48.4%
Imagenet Inception	47.4%	45.2%	71.8%	48.8%	49.2%
Shuffle & Learn	17.2%	17.8%	46.3%	25.7%	28.0%
single-view TCN (triplet)	12.6%	14.4%	41.2%	21.6%	31.9%
multi-view TCN (npairs)	8.0%	9.0%	35.9%	24.7%	35.5%
multi-view TCN (triplet)	7.8%	10.0%	34.8%	22.7%	31.5%
multi-view TCN (lifted)	7.8%	9.0%	35.4%	17.9%	27.7%

Quantitative Analysis: Video Alignment

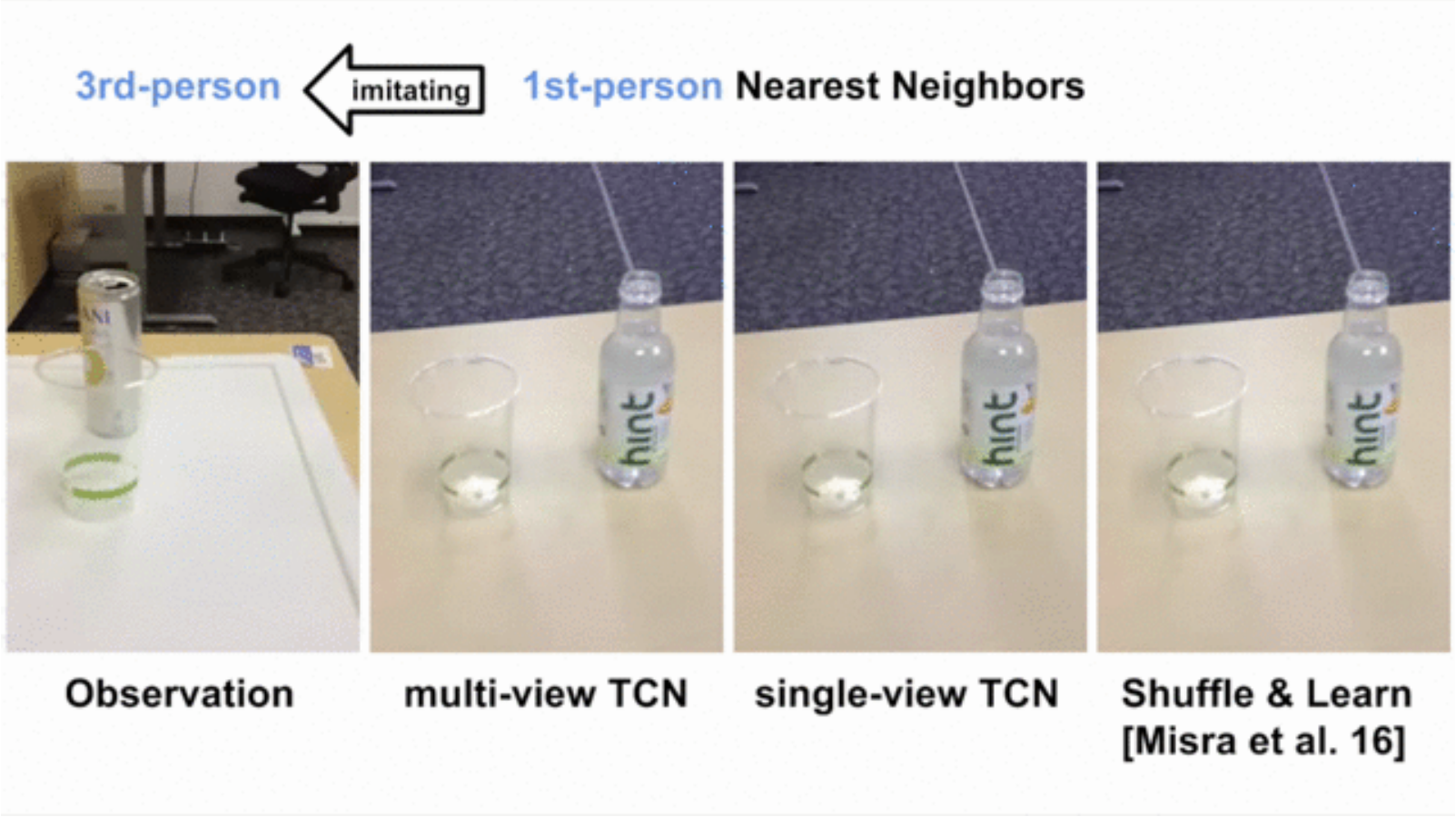
Method	alignment error	classif. error	training iteration
Random	28.1%	54.2%	-
Inception-ImageNet	29.8%	51.9%	-
Shuffle & Learn[31]	22.8%	27.0%	575k
single-view TCN (triplet)	25.8%	24.3%	266k
multi-view TCN (npairs)	18.1%	22.2%	938k
multi-view TCN (triplet)	18.8%	21.4%	397k
multi-view TCN (lifted)	18.0%	19.6%	119k

Nearest Neighbor Imitation



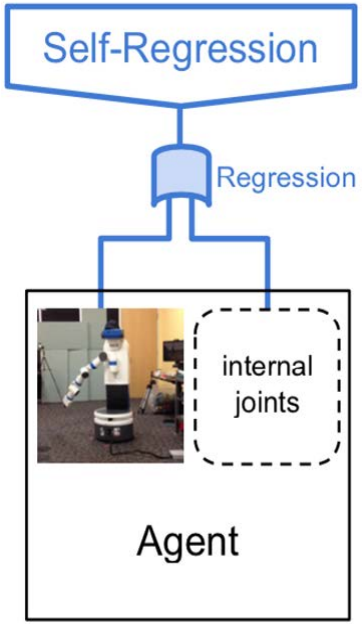
Observation **multi-view TCN**

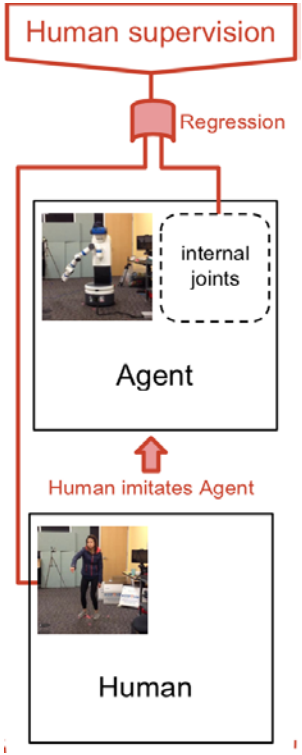
Nearest Neighbor Imitation



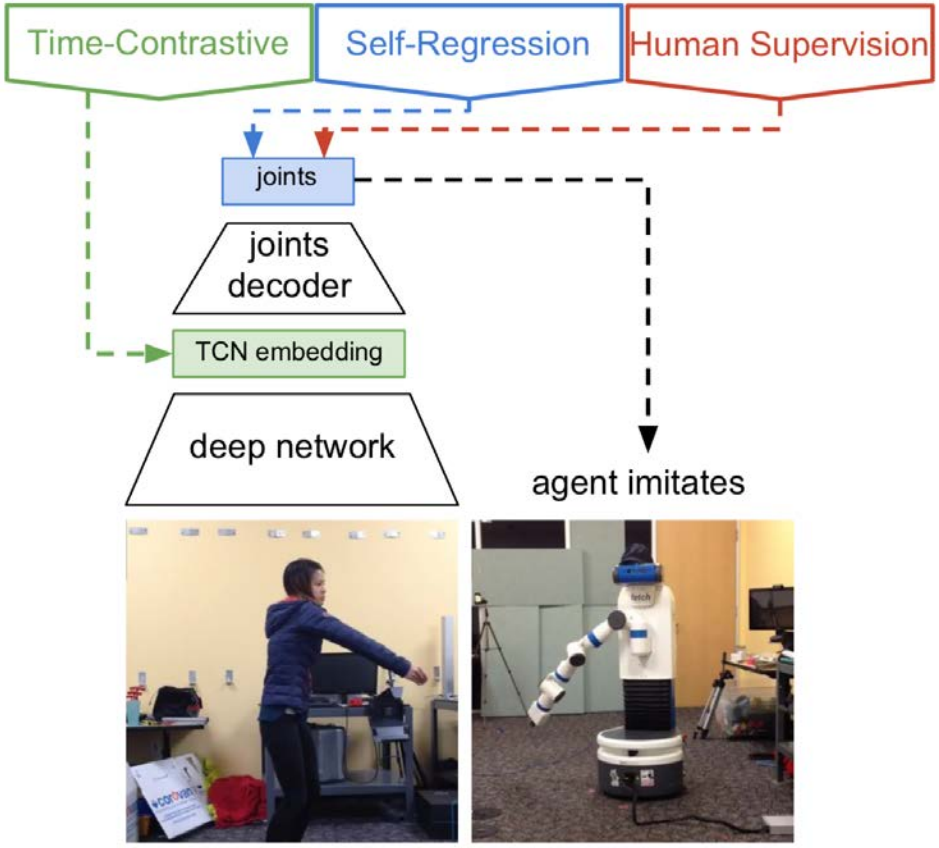
Robot Joint Angles Regression



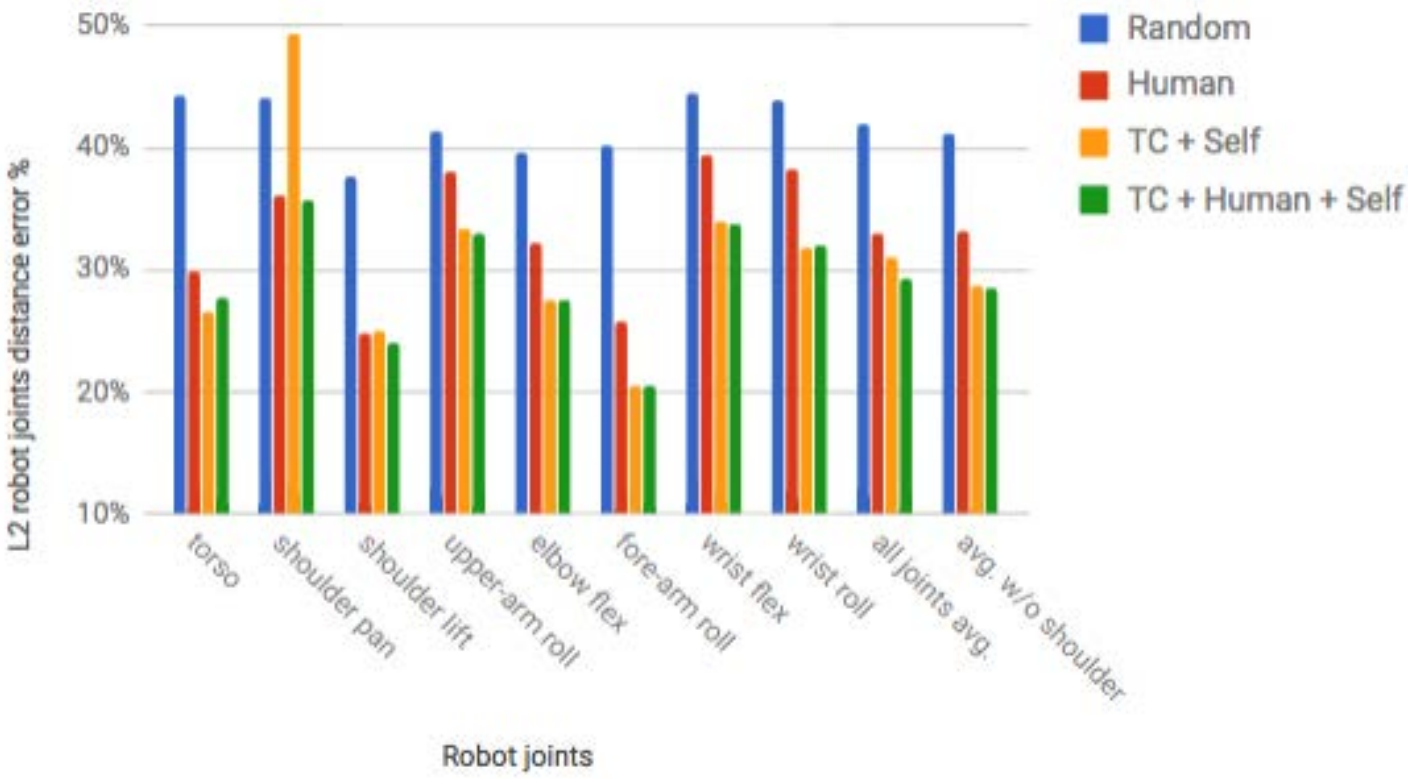




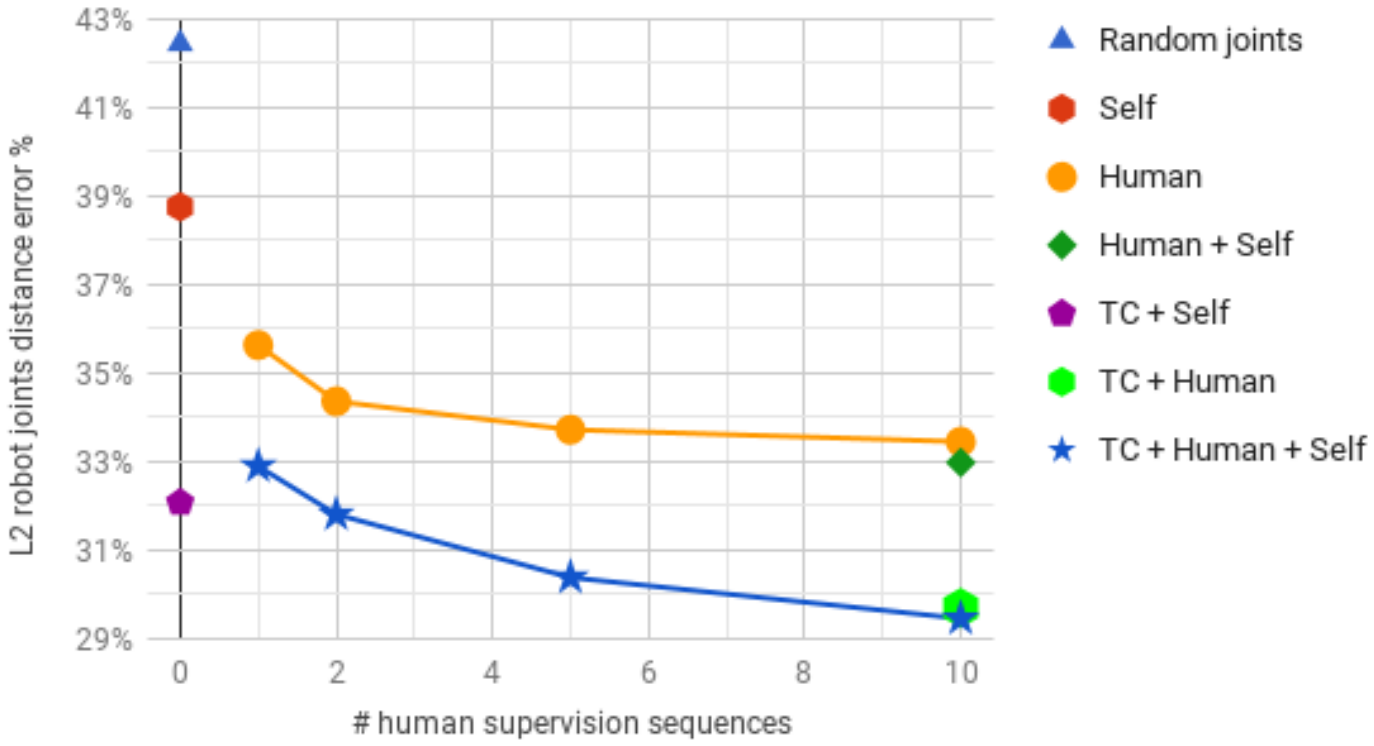
Self-supervision signals



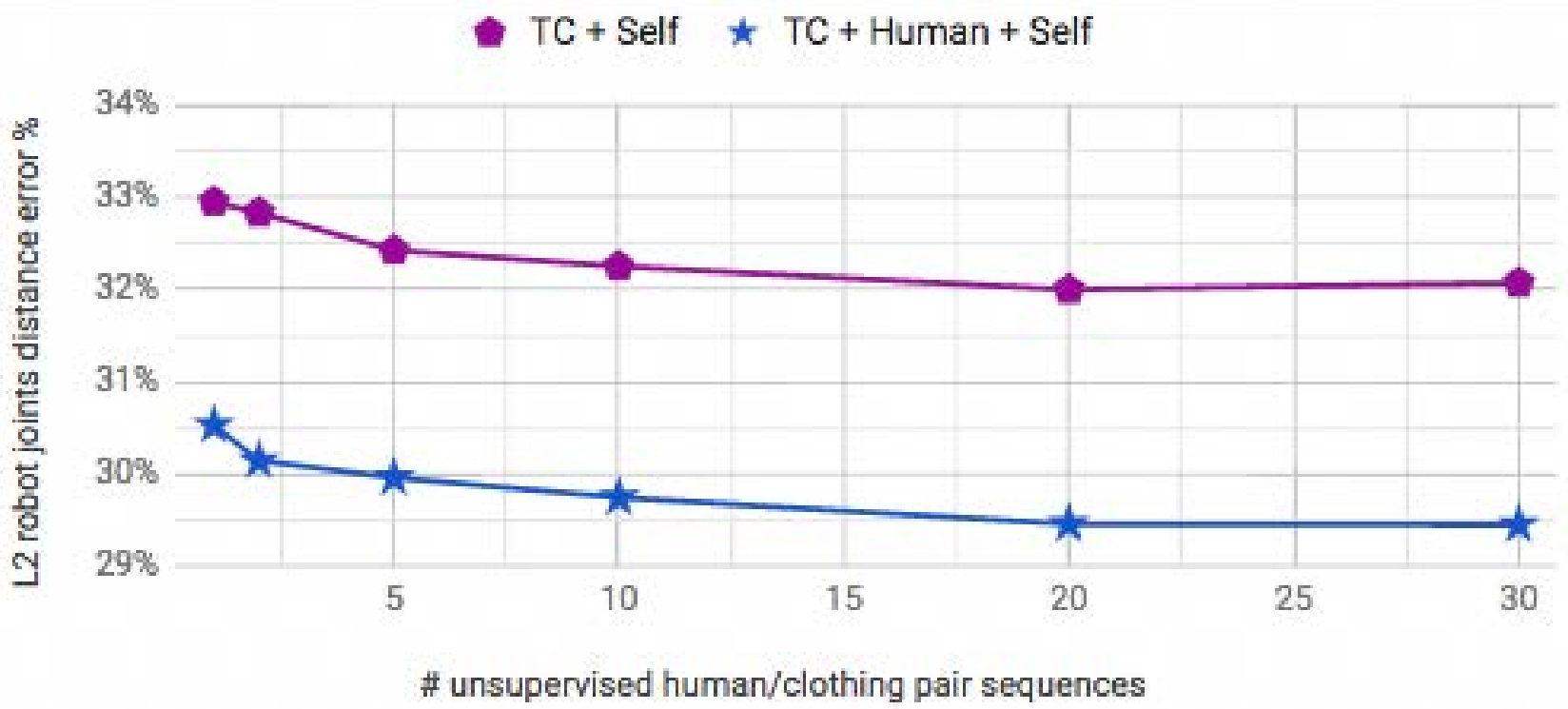
Robot Joint Angles - L2 Error



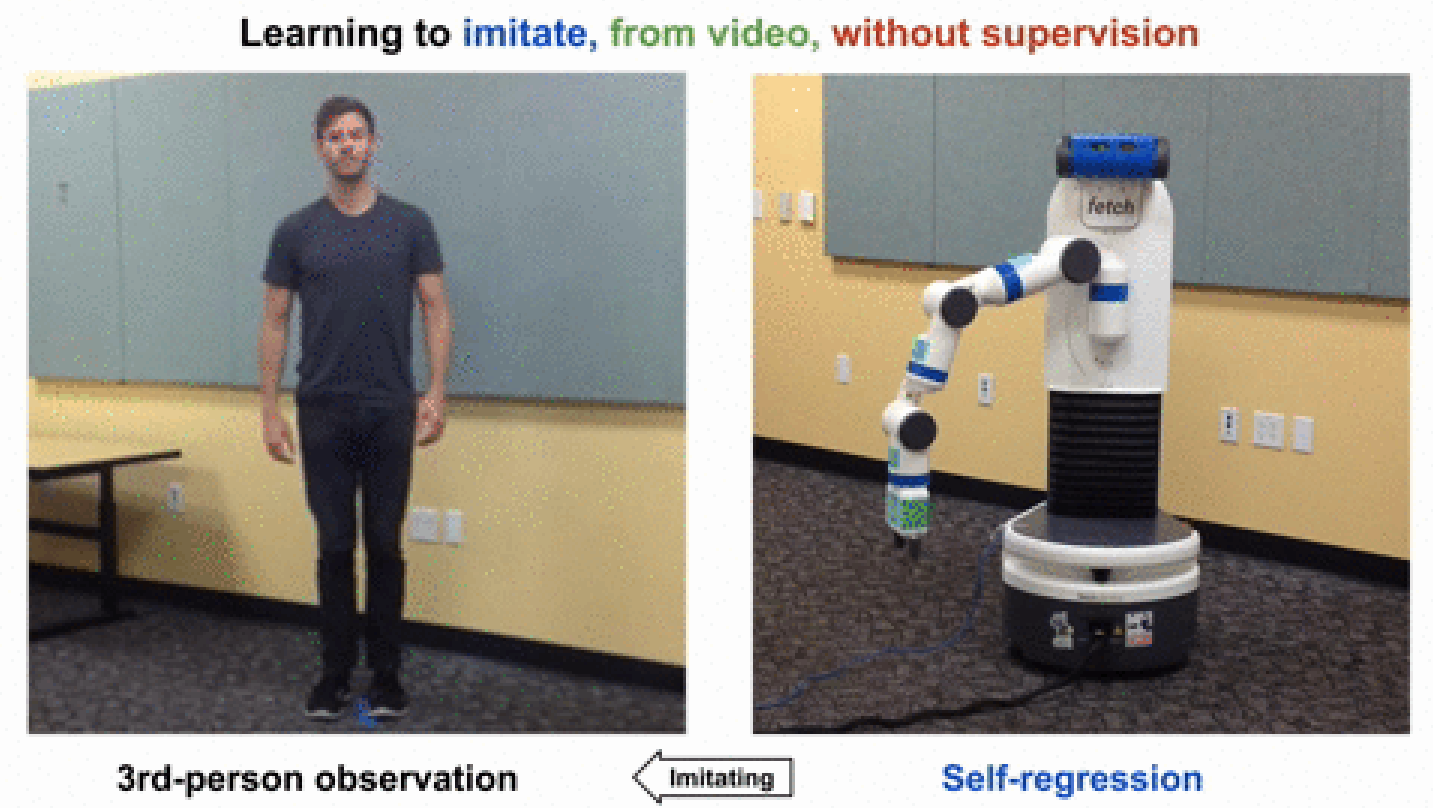
Robot Joint Angles - L2 Error



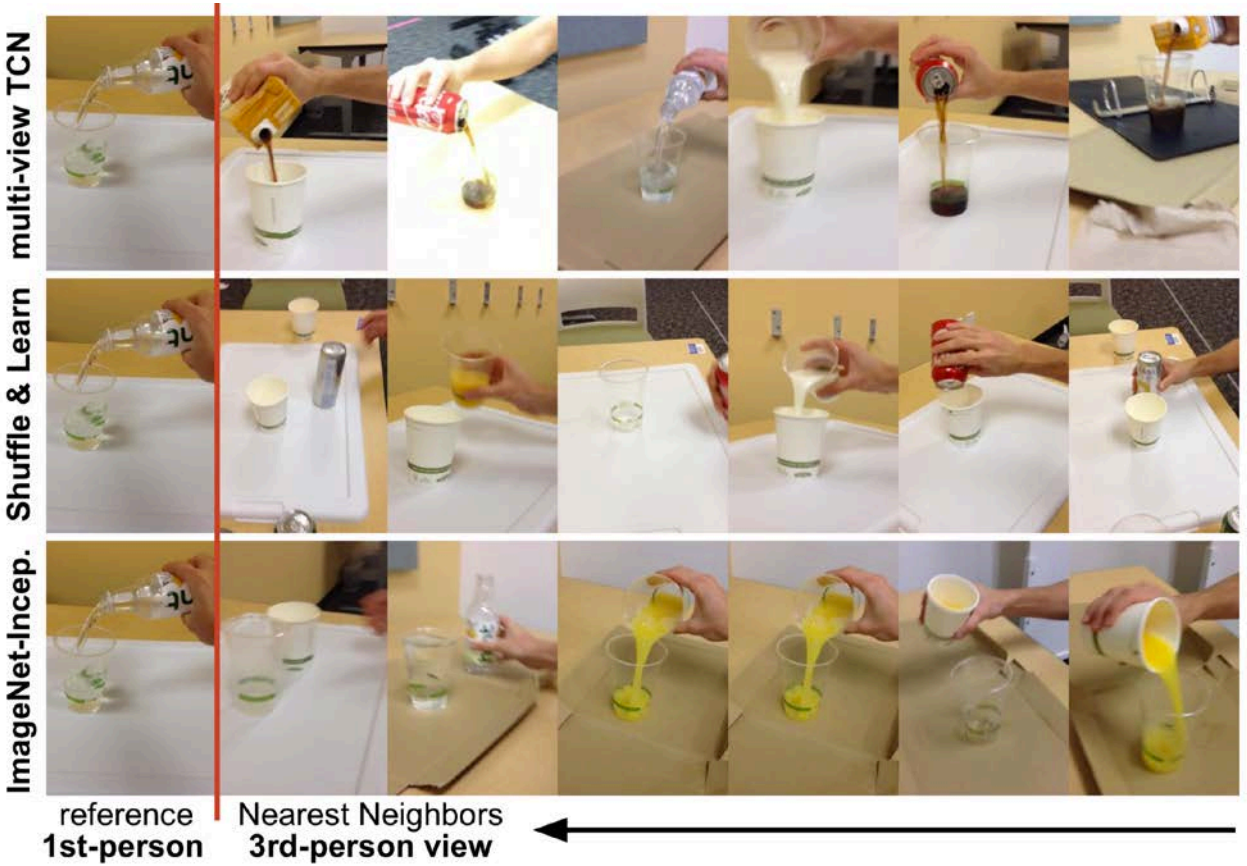
Robot Joint Angles - L2 Error



Pose imitation (real robot)



Label-free pouring imitation



Label-free pose imitation

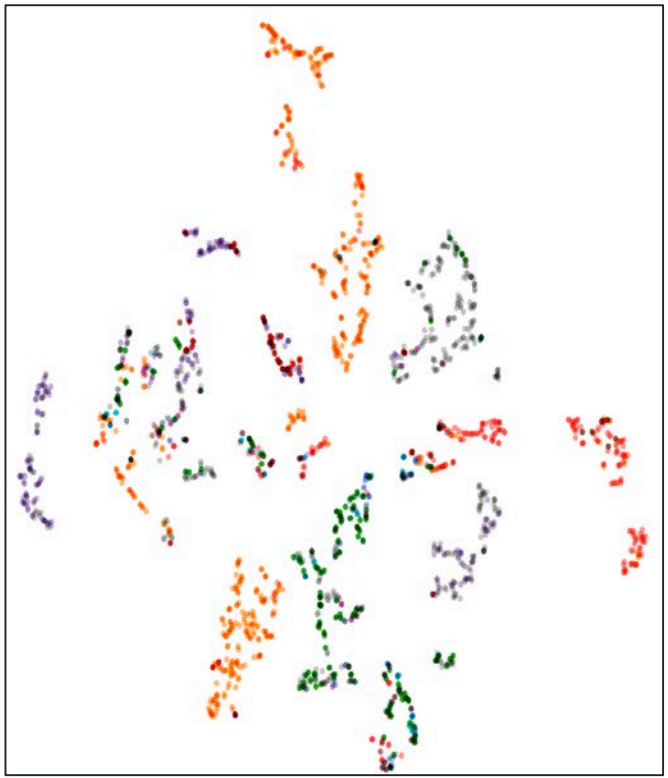


t-SNE colored by attribute combinations



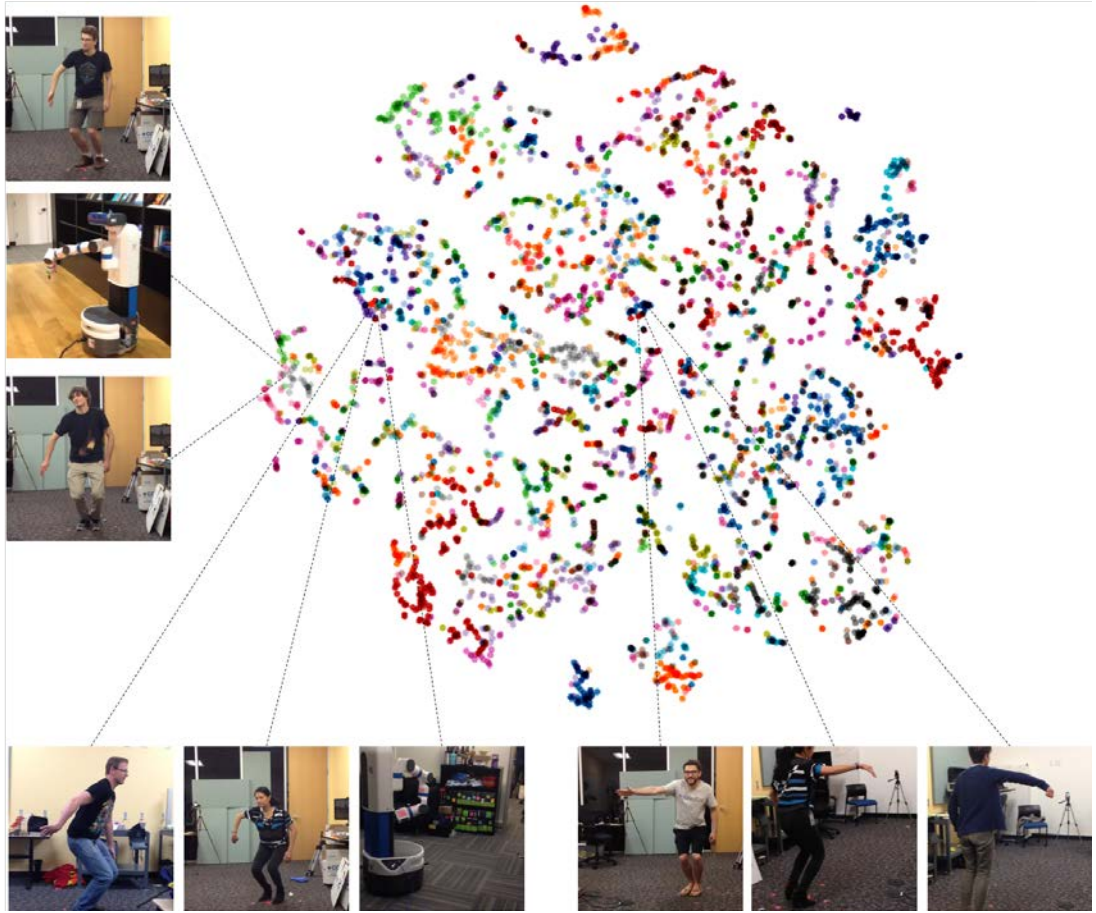
before training

- Hand contact with container?
- Within pouring distance?
- Container angle
- Liquid is flowing?
- Recipient has liquid?

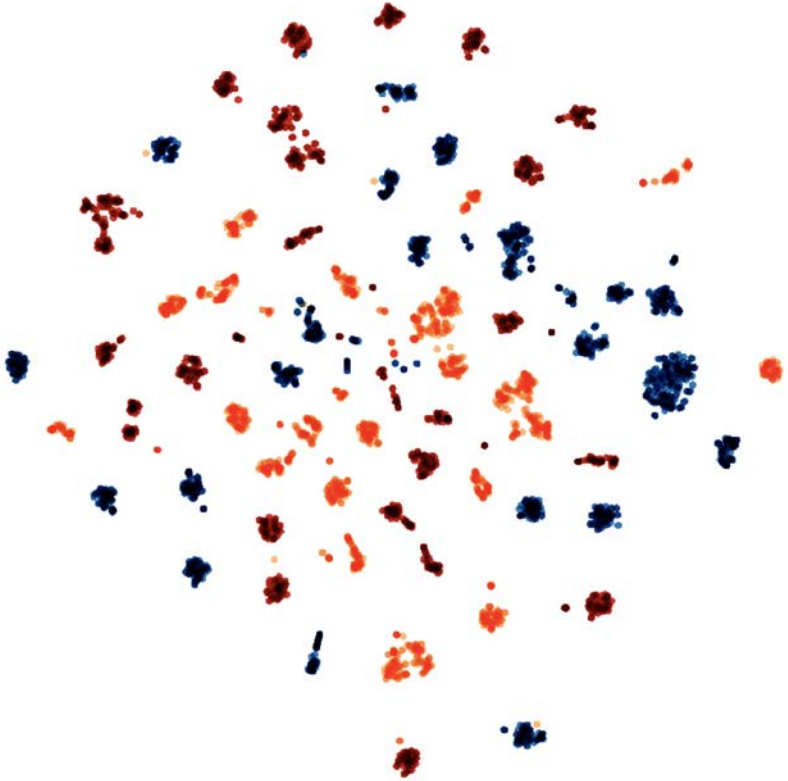


after training

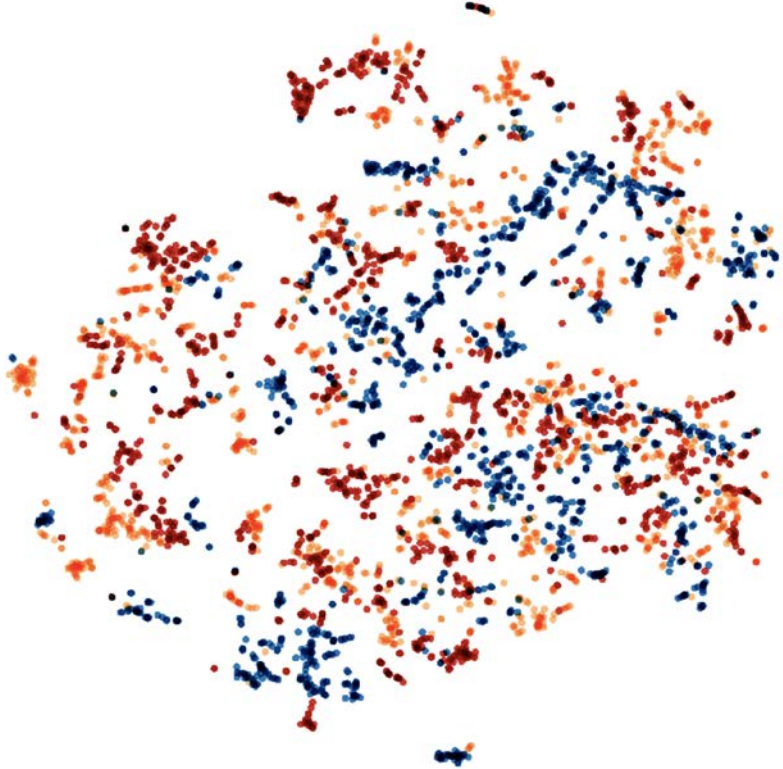
t-SNE embedding colored by agent for model ”TC+Self”



t-SNE colored by view



before training



after training

t-SNE colored by agent

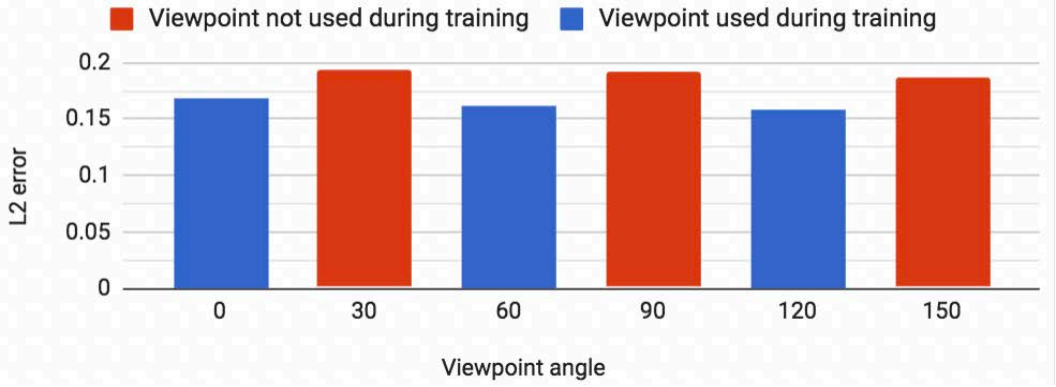


before training



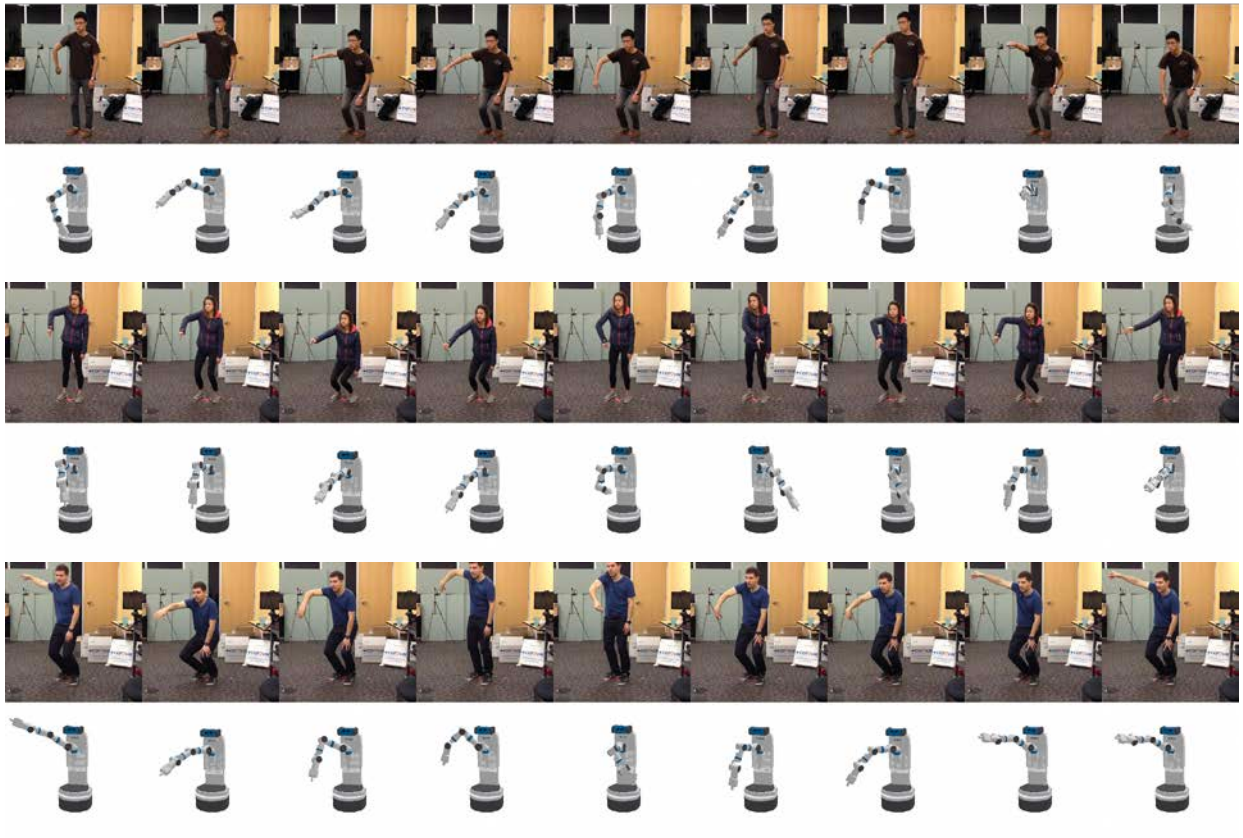
after training

Testing for orientation and scale invariance





Self-supervised imitation examples



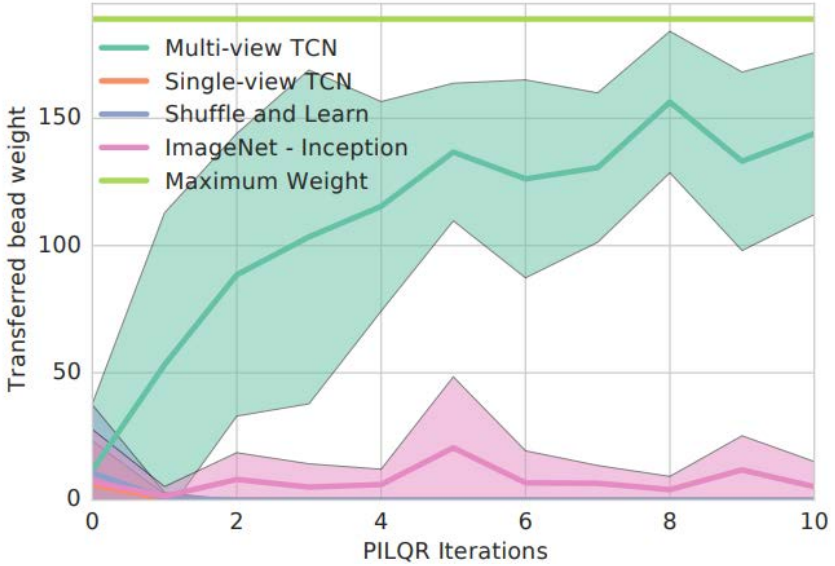
Learning Robotic Behaviors with Reinforcement Learning

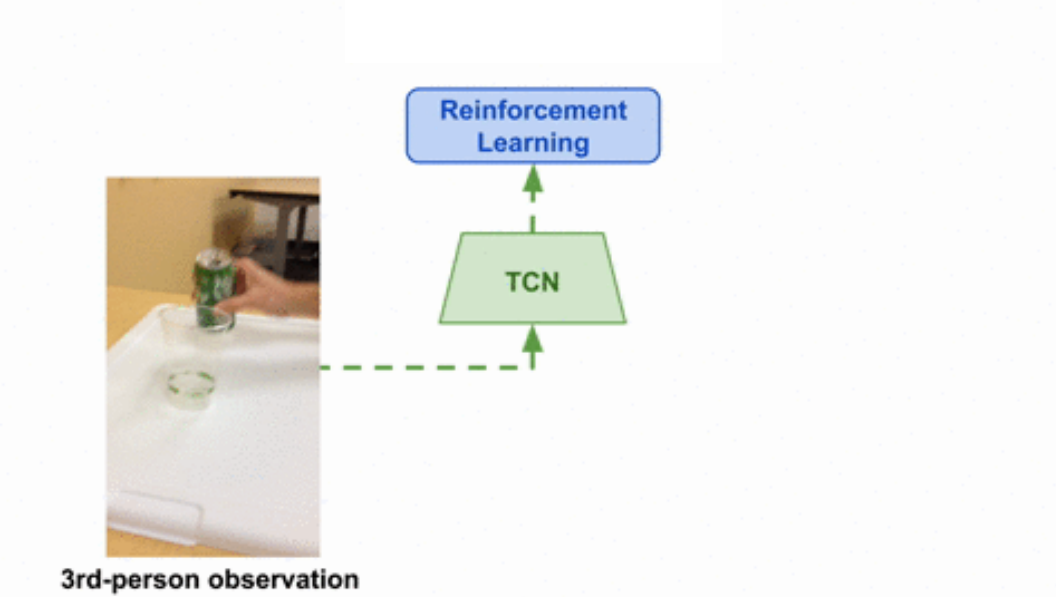
- Squared Euclidean distance and a Huber-style loss:
- V=(v1,...v_T)⟶ embeddings of frames in video demonstration
- W = (w1, ...w_T) ⟶ embeddings of camera images of Robot

$$R(\mathbf{v}_t, \mathbf{w}_t) = -\alpha \|\mathbf{w}_t - \mathbf{v}_t\|_2^2 - \beta\sqrt{\gamma + \|\mathbf{w}_t - \mathbf{v}_t\|_2^2}$$

Quantitative analysis: Pouring

- Beads weight in target recipient





Conclusion

With Time-Contrastive Networks, we showed we can:

- **Learn a self-supervised understanding of the world**
- **Use it to learn real-world tasks, entirely from 3rd person videos of humans** (addressing correspondence problem, no labels, no reward function design, providing sample-efficiency for RL, quickly learning tasks, no kinesthetic demonstrations)

Thank you!

Q & A