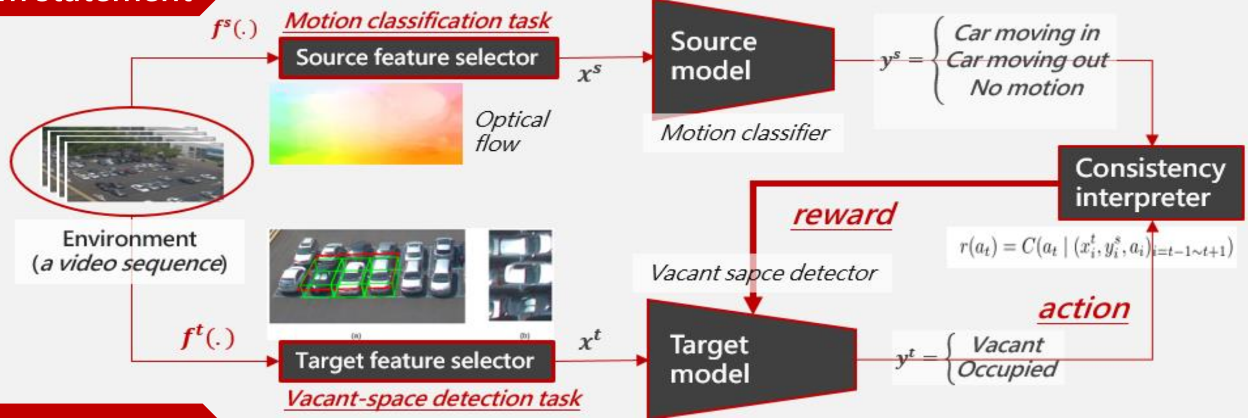
Vacant Parking Space Detection based on Task Consistency and Reinforcement Learning

Manh-Hung Nguyen
Faculty of Electrical Electronic Engineering, HCM-UTE, Vietnam
hungnm@hcm.ute.edu.vn

Tzu-Yin Chao
Department of Computer Science, National Chiao Tung University, Taiwan
chaoziyin@gmail.com

Ching-Chun Huang
Department of Computer Science, National Chiao Tung University, Taiwan
chingchun@cs.nctu.edu.tw

Problem statement



Contributions

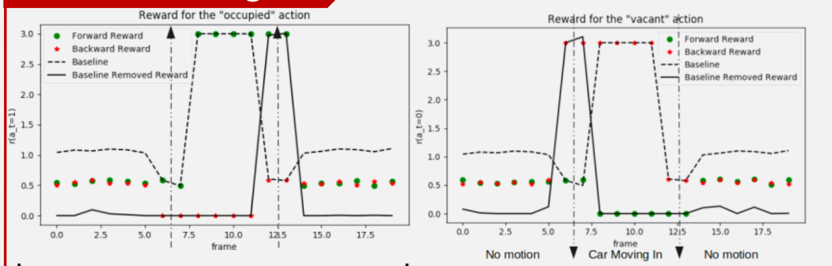
- We proposed a novel framework that allows the system to train a target model (e.g., a vacant-space detector) via the task consistency with a source model (e.g., a car motion classifier).
- Unlike transfer learning, the source model and the target model in our framework are not restricted to deal with the same type of task.
- The proposed framework is suitable for online learning, which is lable-free (unsupervised rewards).
- We test the method on a parking lot scenario and corrupted rewards are filtered out automatically

Algorithm

Algorithm 1 Task Consistency learning (TCL)

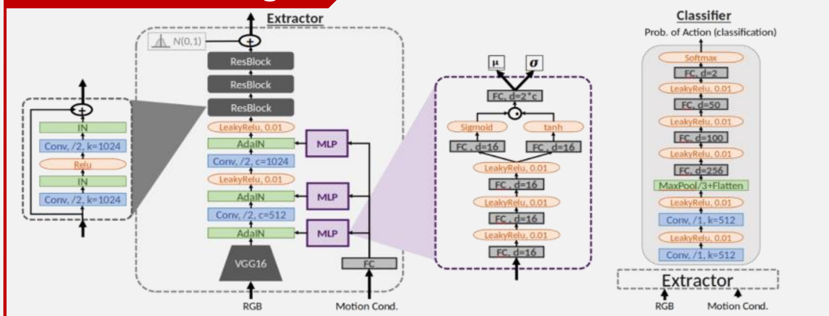
1: **procedure** TCL(Require: $M_\phi^s(x_i^s)$, Environment, $f^s(s_t)$, $f^t(s_t)$; Output: target network θ)
2: $D_{online} = \varnothing$
3: Random initialize θ
4: **while** collect training trajectories **do** ▷ Data collection
5: Query s_t from the environment
6: $x_t^s = f^s(s_t)$, $x_t^t = f^t(s_t)$ ▷ Select source and target input
7: $\hat{y}_t^s = M_\phi^s(x_t^s)$ ▷ Predict source decision
8: **if** $P(y_t^s) > \delta$ **then**
9: Extract $x_i^t = f_t(s_i) \mid i = 1 \sim T$ from the segment trajectory around the frame t^{th}.
10: **end if**
11: Add the tuple $(x_i^s, \hat{y}_i^s, x_i^t)_{i=1:T}$ into D_{online}
12: **end while**
13: **while** Training **do**
14: **for** each training trajectory **do**
15: Run the target / policy model $\pi_\theta(a_t \mid x_t)$ to get a_t or y_t
16: Estimate $r(a_t)$ by equation (5)
17: Estimate baseline value by equation (6)
18: Estimate the gradient in equation (4)
19: Update the policy $\theta \leftarrow \theta + \alpha \nabla_\theta J(\theta)$
20: **end for**
21: **end while**
22: Repeat the data collection process in the 4th line
23: Repeat the training process in the 13th line
24: **end procedure**

Reward design



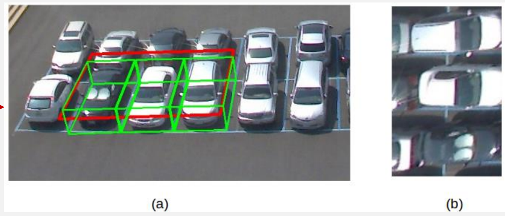
	Forward Reward $r_F(a_t \mid a_{t-1}, y_{t-1}^s)$		Backward Reward $r_B(a_t \mid a_{t+1}, y_{t+1}^s)$		
y_{t-1}^s	occupy $(a_t = 1)$	vacant $(a_t = 0)$	occupy $(a_t = 1)$	vacant $(a_t = 0)$	y_{t+1}^s
CI_{t-1}	$1 * \lambda$	0	0	$1 * \lambda$	CI_{t+1}
NM_{t-1}	P_{t-1}^O	P_{t-1}^V	P_{t+1}^O	P_{t+1}^V	NM_{t+1}
CO_{t-1}	0	$1 * \lambda$	$1 * \lambda$	0	CO_{t+1}

Network design

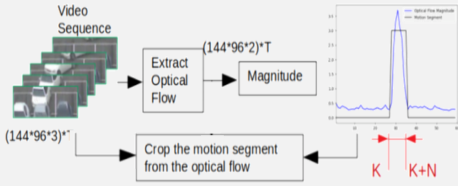


Experimental settings

- 120 videos from a 90-degree view camera
- Each video includes 500 frames

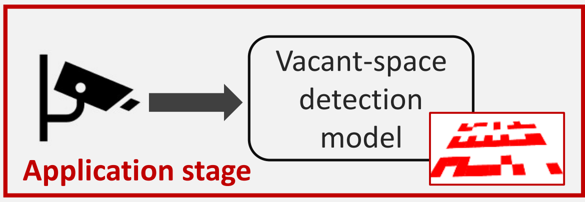


Local slot normalization



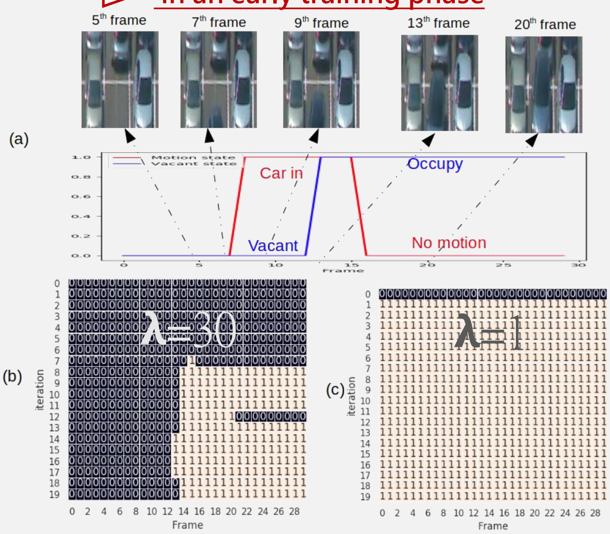
Extract 1530 training trajectories

Training process

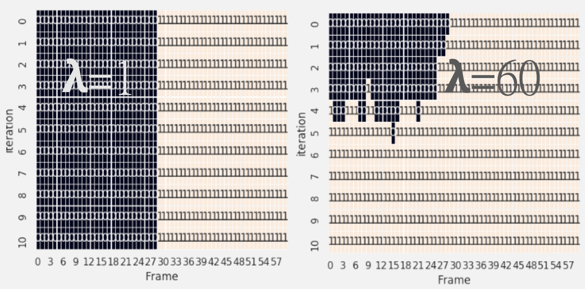


Lambda selection

▷ In an early training phase



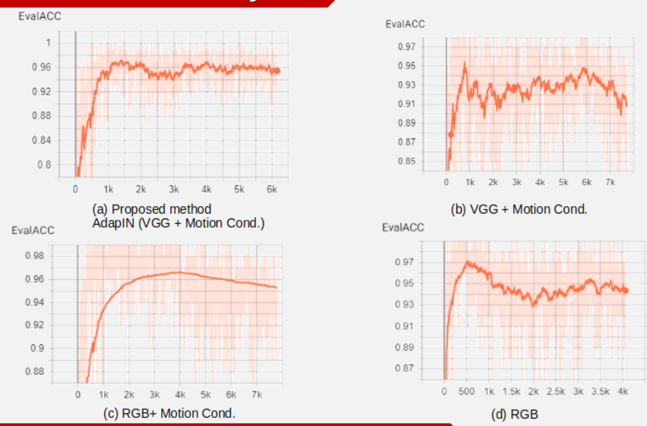
▷ Learning from corrupted trajectories



(a) $\lambda=1$ (b) $\lambda=60$

· "0" means vacant state "1" means occupy state.

Ablation study



Learning on a new parking lot

Supervised learning (Fine tune)	97.93 (M=600)	98.18 (M=1000)	98.57 (M=1400)	98.76 (M=1700)	99.21 (M=2000)
Task Consistency learning (Fine tune)	98.84 * (N=300)	99.15 * (N=500)	99.37 * (N=700)	99.57 * (N=850)	99.69 * (N=1000)
Task consistency learning (Train from scratch)	98.15 (N=300)	98.38 (N=500)	98.45 (N=700)	99.48 (N=850)	99.54 (N=1000)

· M means the number of training samples for supervised learning.
· N means the number of training trajectories for task consistency learning.

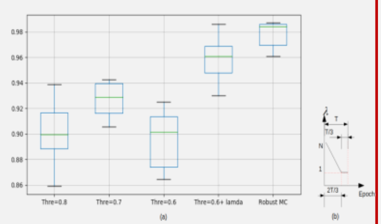
45-degree view(as new parking lot scenario) 90-degree view (as original parking lot scenario)

Learning from an imperfect motion classifier

The number of clean/noisy trajectories under different thresholds identified by the baseline motion classifier and robust motion classifier. (Evaluate on 1530 trajectories)

· Cl: Clean, No: Noisy, MC: Motion Classifier

	Basedline MC		Robust MC	
δ	Cl	No	Cl	No
0.6	950	142	1406	27
0.7	666	82	1292	11
0.8	393	33	1090	0



Conclusions

- We proposed a task consistency framework, which enables the system to learn a target task from a source task in a reinforcement learning manner.
- The framework has two benefits:
  - The source model and target model are not restricted to deal with the same type of task.
  - By applying reinforcement learning approach with unsupervised rewards, our framework is label-free.
- The framwork is applied to learn a vacant space detector based on a motion classifier:
  - The reward design is capable of filtering out some easy corrupted rewards automatically.