Robust Anomaly Detection in Videos using Multilevel Representations

Hung Vu[1], Tu Dinh Nguyen[2], Trung Le[2], Wei Luo[1] and Dinh Phung[2]

[1]Centre for Pattern Recognition and Data Analytics (PRaDA), Deakin University, Australia
[2]Monash University Clayton, VIC 3800, Australia

Outline

- Video Anomaly Detection (VAD)
- Related work
- Problem statement
- Proposed framework
- Experiments
- Conclusion

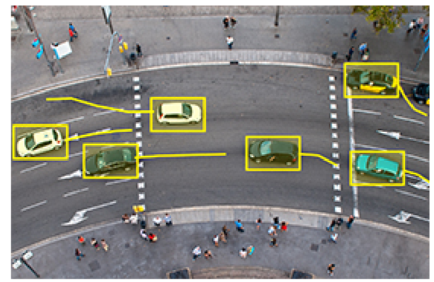
Video Anomaly Detection (VAD)

- **Video Anomaly Detection** = detect anomaly events in video data
- Anomaly events = events occur infrequently in comparison to normal events[1]

Example

A cyclist on a pedestrian footpath

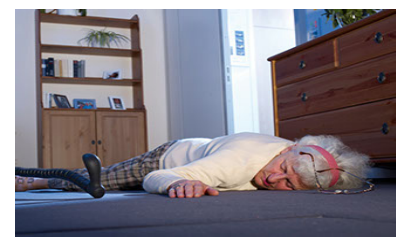
Applications



Traffic monitoring

Access detection

Fighting detection



Falling detection

[1] *(Sodemann et al., 2012)*

Related work

- **Deep detectors**
  - Convolutional Autoencoders (CAEs) [1]
  - CAEs + Long Short Term Memories (LSTMs)[2]
  - Conditional Generative Adversarial Networks (cGANs)[3]
  - Adversarial AEs[4]
  - ...

➔ work on low-level features (pixels/edges/motions)

[1] *(Hasan et al., 2016; Ribero, Lazzaretti , and Lopes, 2017);* [2] *(Chong and Tay, 2017; Luo, Liu, and Gao, 2017)*
3 (Ravanbakhsh et al., 2017a; 2017b); 4 (Sabokrou et al., 2018)

Problem statement

- **Low-level features → two issues:**
  - <u>Issue 1</u>: fragmented and interrupted detection

  - <u>Issue 2</u>: false detection by noise and environment changes

→unreliable and ineffective features

ground-truth

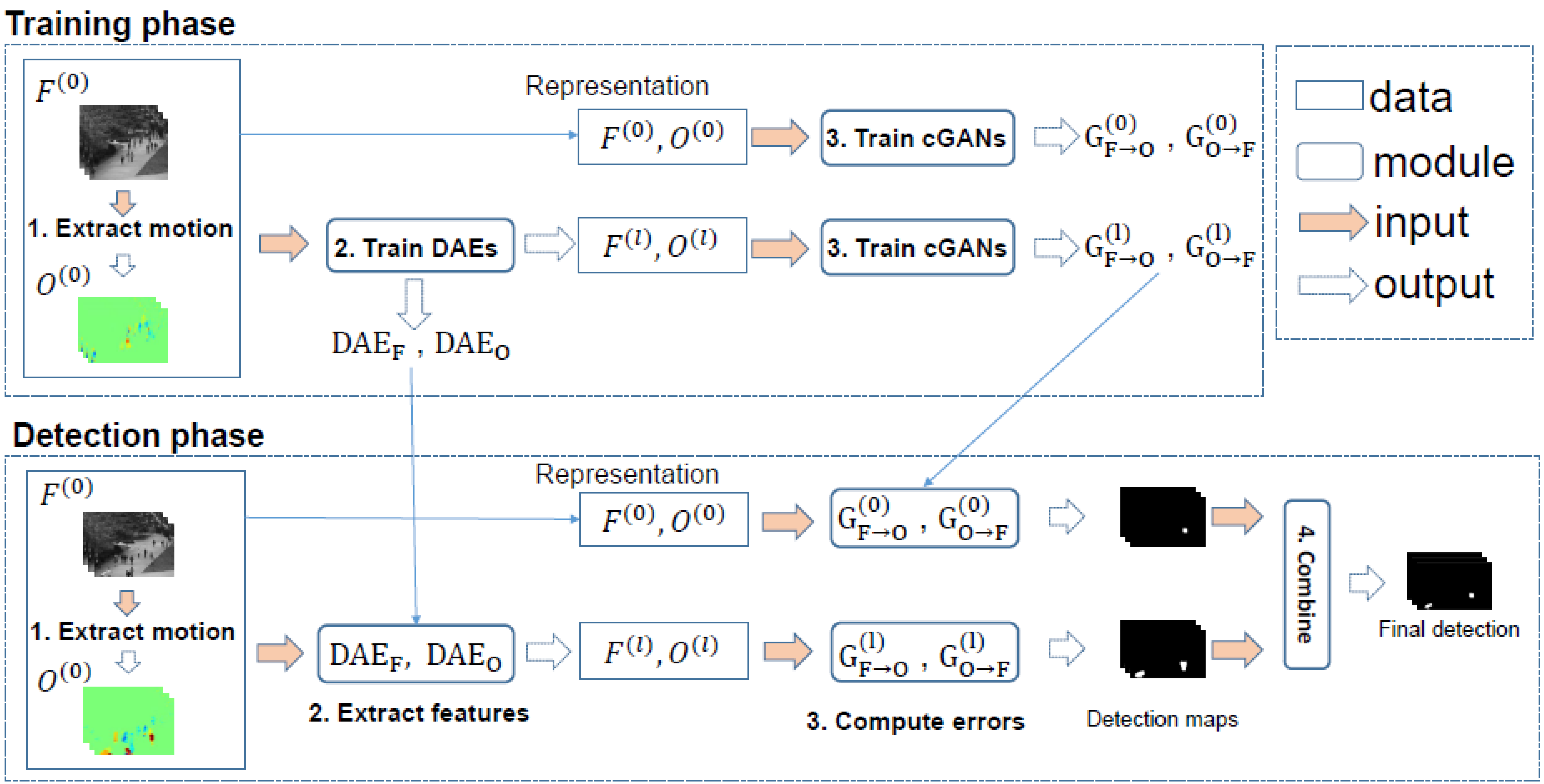
Idea

- **Detect abnormality at abstract-level features**
  - Abstractness extraction via deep networks[1]
    - Low layers: edges, corners, colors
    - High layers: objects and their relationship
  - ➔detect complete objects ➔ solve Issue 1 (fragments + interruption)
  - Combine low-level + abstract-level detections
    - Reason: true anomalies should appear at all level representations
  - ➔ Reduce false detections ➔ solve Issue 2 (many false detections)

[1](*Zeiler and Fergus 2014*)

Proposed framework

- Multilevel Anomaly Detector (MLAD)
- Two phases: training and detection



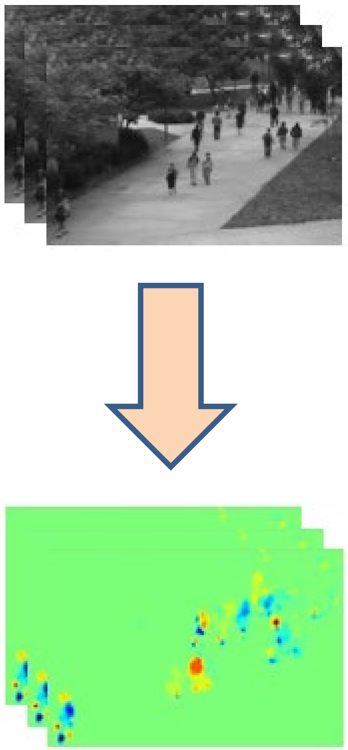
Proposed framework

- **Training phase:**
  - compute optical flow image for every frame
  - train Denoising Autoencoders (DAEs)
  - extract high-level features
  - train Conditional GANs (cGANs)

Proposed framework

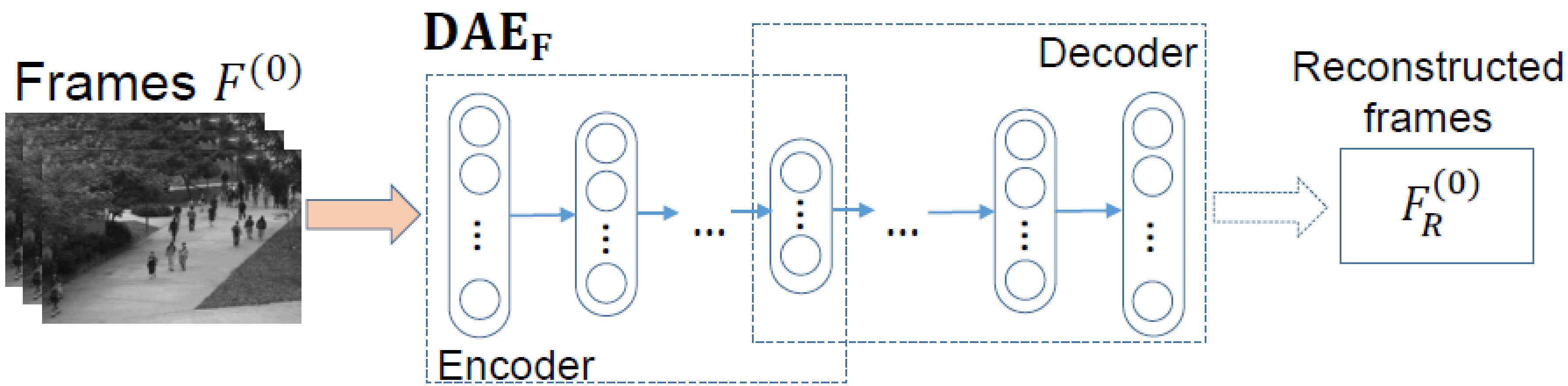
- **Training phase:**
  - compute optical flow image for every frame

Frames $F^{(0)}$



Proposed framework

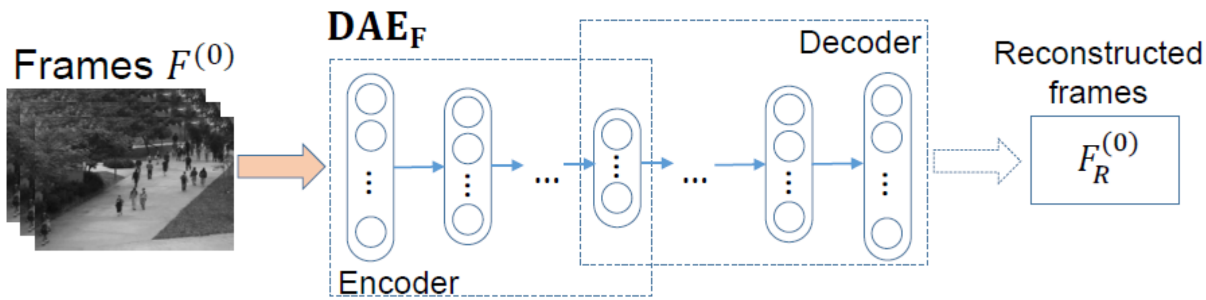
- **Training phase:**
  - compute optical flow image for every frame
  - train Denoising Autoencoders (DAEs)

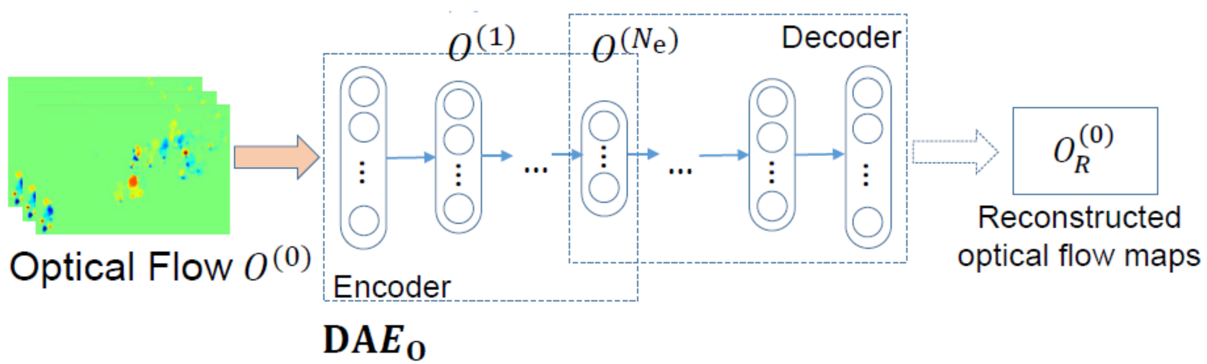


$$\min_{\theta,\phi} \mathcal{J}_{\mathrm{DAE}} = \min_{\theta,\phi} \frac{1}{|D|} \sum \underbrace{\left\| v_i - g_\phi(f_\theta(\tilde{v}_i)) \right\|_2^2}_{\text{reconstruction loss}} + \gamma \underbrace{\left(\Sigma_{l=1}^{N_e} \left\| W_e^{(l)} \right\|_2^2 + \Sigma_{l=1}^{N_e} \left\| W_d^{(l)} \right\|_2^2 \right)}_{\text{regularization term}}$$

Proposed framework

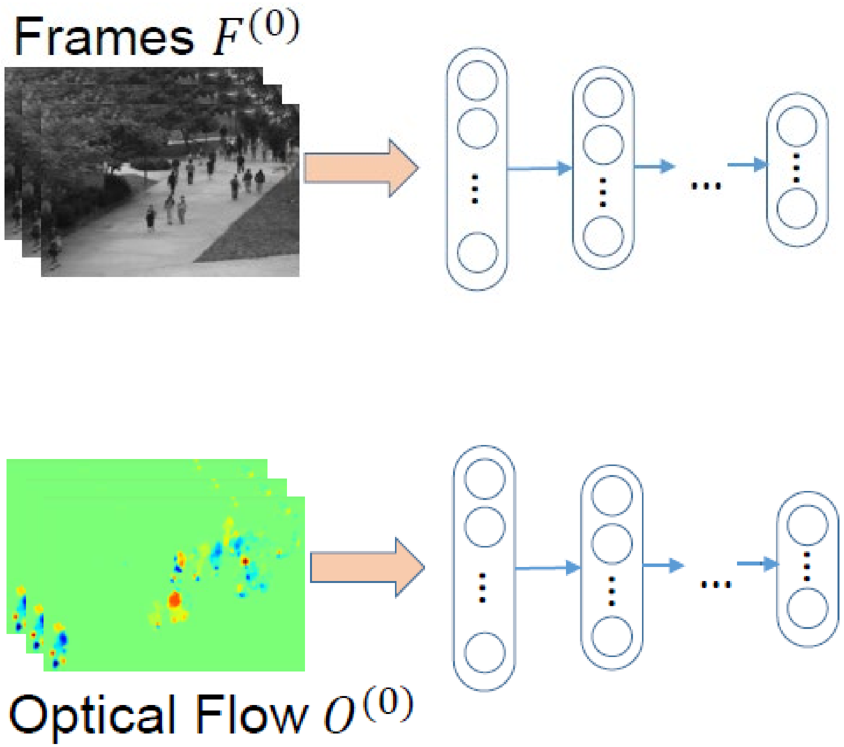
- Training phase:
  - compute optical flow image for every frame
  - train Denoising Autoencoders (DAEs)





Proposed framework

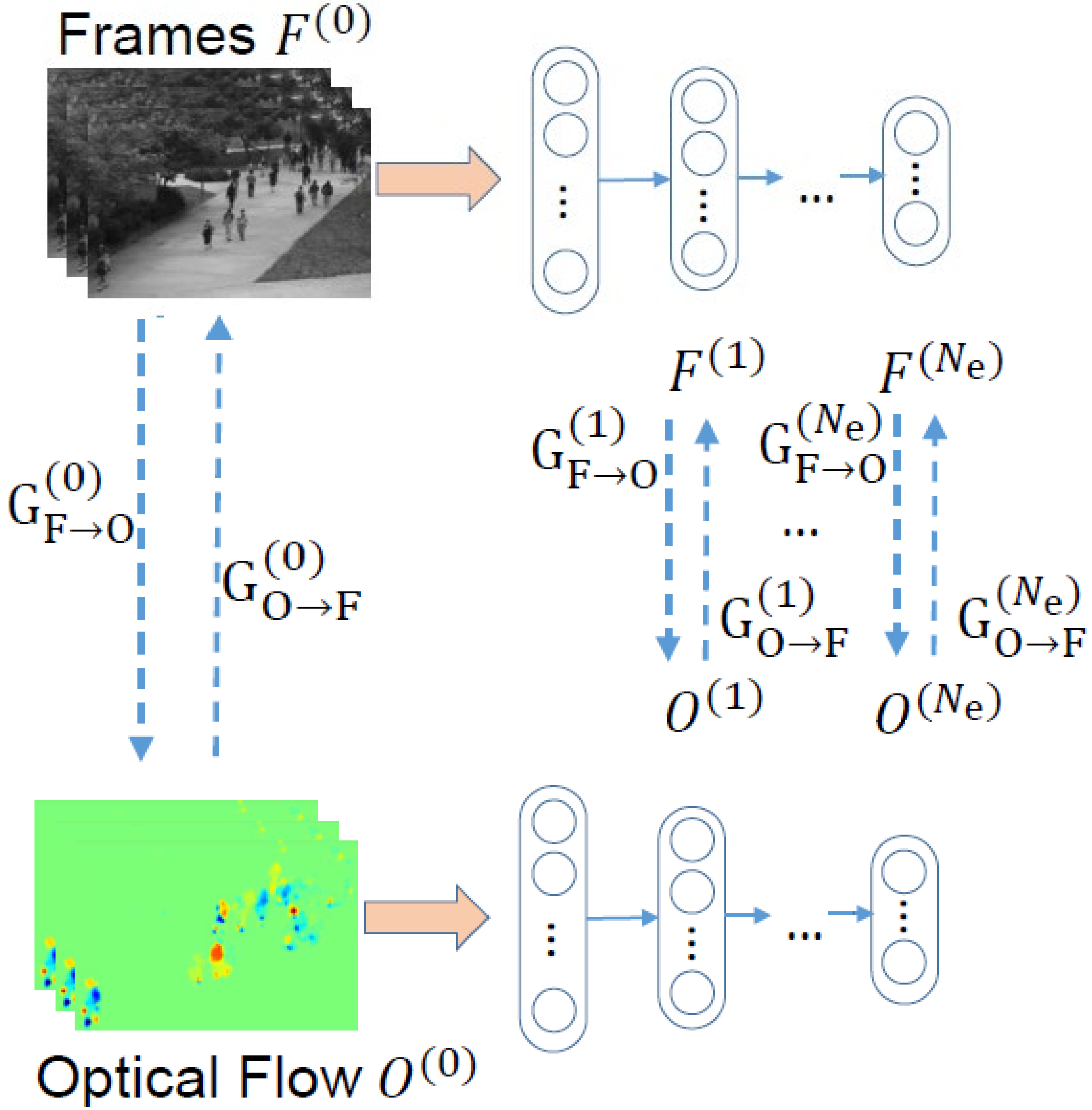
- Training phase:
  - compute optical flow image for every frame
  - train Denoising Autoencoders (DAEs)
  - extract high-level features



Proposed framework

- **Training phase:**
  - compute optical flow image for every frame
  - train Denoising Autoencoders (DAEs)
  - extract high-level features
  - train Conditional GANs (cGANs)[1]

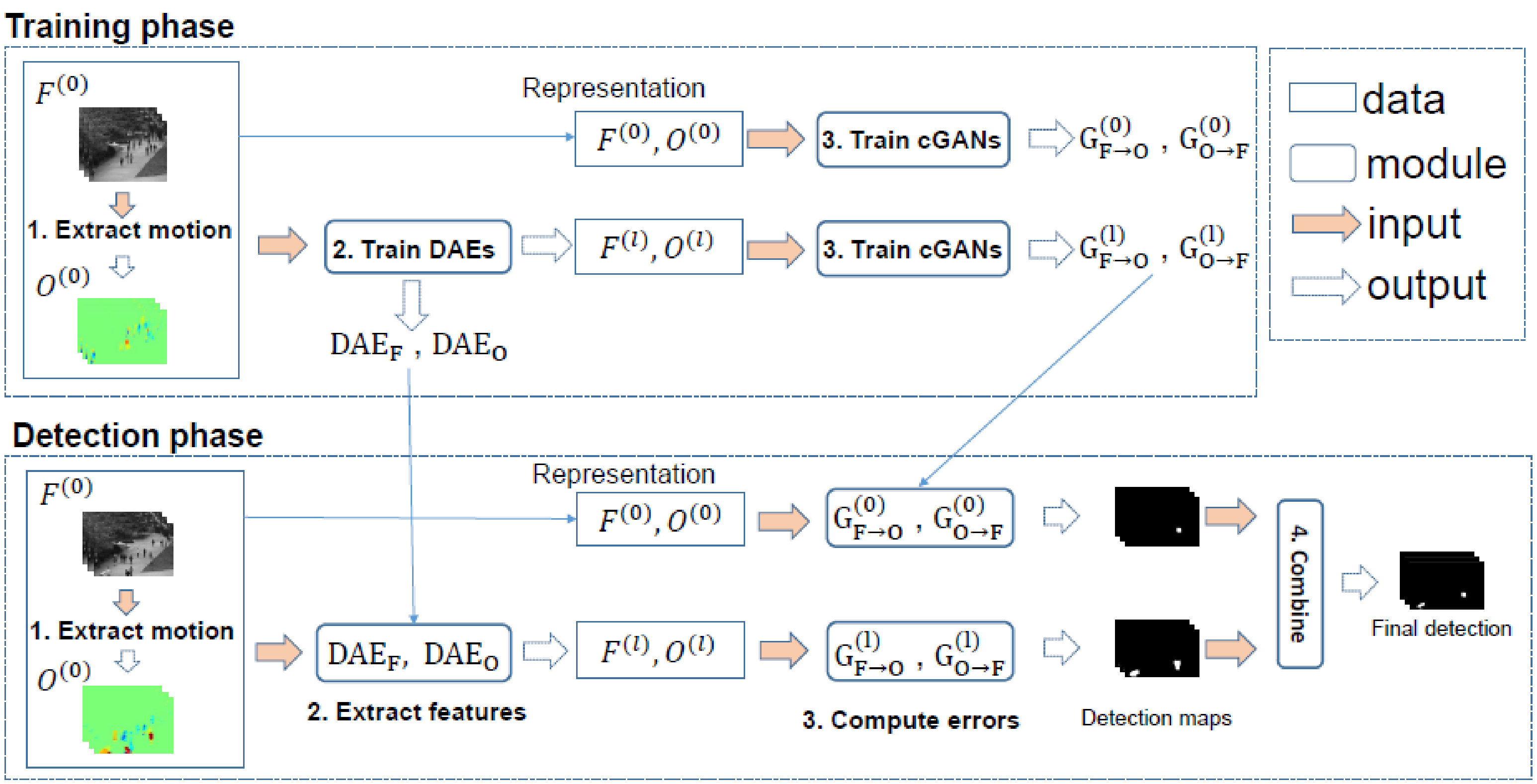
$$\mathcal{J}_{cGAN} = \mathbb{E}_{x,z}[\log(1 - D(x, G(x,z)))]$$
$$+\mathbb{E}_{x,z}[\log D(x,y)] + \lambda\|y - G(x,z)\|_1$$



[1](Isola et al., 2017)

Proposed framework

- **Detection phase:**
  - extract optical flow images for testing frames
  - compute high-level features
  - compute single level detections
  - consolidate detection maps

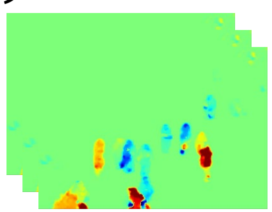


Proposed framework

- **Detection phase:**
  - extract optical flow images for testing frames

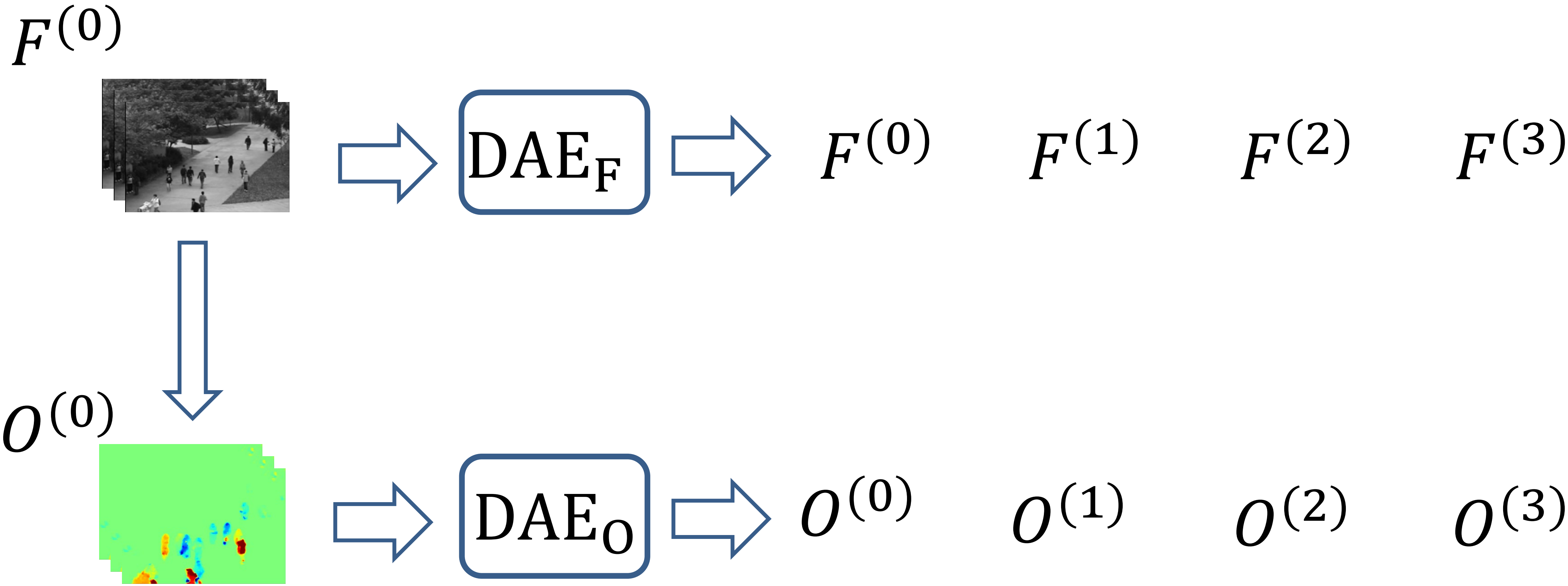
$F^{(0)}$

$O^{(0)}$



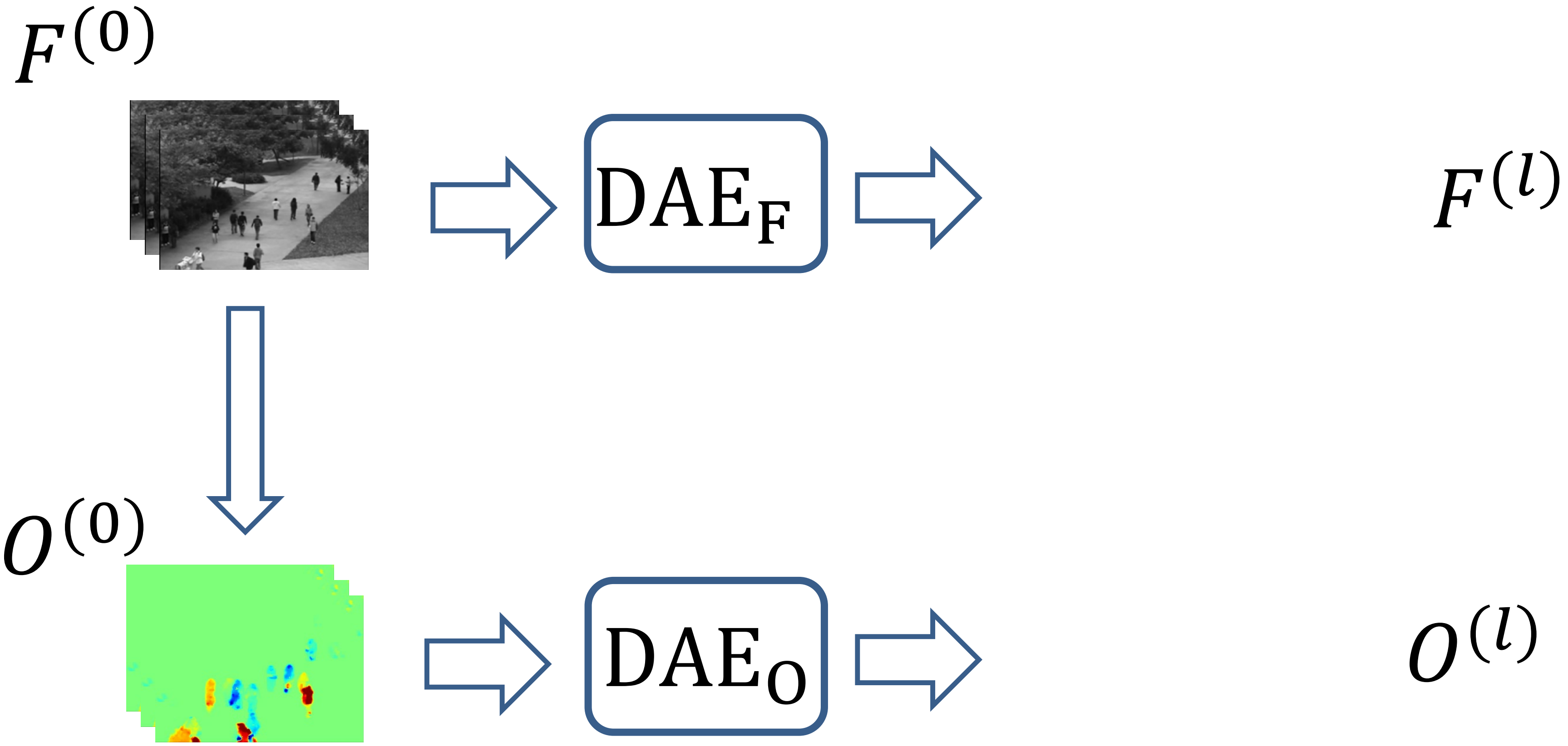
Proposed framework

- **Detection phase:**
  - extract optical flow images for testing frames
  - compute high-level features



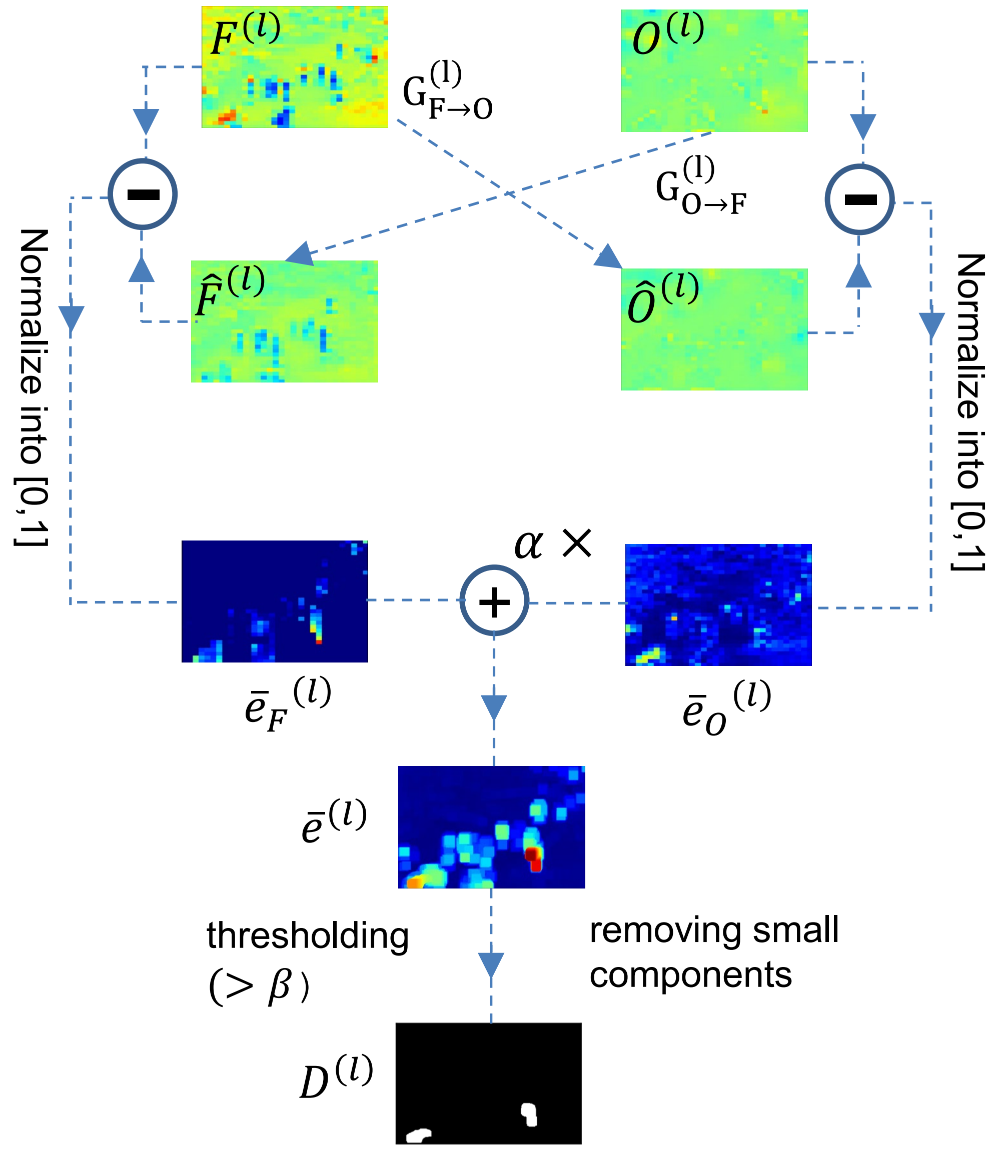
Proposed framework

- Detection phase:
  - extract optical flow images for testing frames
  - compute high-level features



Proposed framework

- **Detection phase:**
  - extract optical flow images for testing frames
  - compute high-level features
  - compute single level detections



Proposed framework

- **Detection phase:**
  - extract optical flow images for testing frames
  - compute high-level features
  - compute single level detections
  - consolidate detection maps

Algorithm 1 Combining multilevel detection maps

Input: Detection maps $\left\{D^{(l)}\right\}$, score maps $\left\{E^{(l)}\right\}$, object lists $\left\{C^{(l)}\right\}$, anomaly threshold β and overlapping threshold ρ

Output: Final detection D, E and C

$D \leftarrow D^{(0)}; \quad E \leftarrow E^{(0)}; \quad C \leftarrow C^{(0)}$

for $l \leftarrow 1, \ldots, N_e$ **do**

  for $c \in C$ and $c_l \in C^{(l)}$ **do**

    if ${}^{L(c \cap c_l)}/_{L(c)} \geq \rho$ **then**

      $D(c) \leftarrow D(c) \cup D^{(l)}(c_l)$

      $E(c_l \cup c) \leftarrow \max\left(E(c_l \cup c), E^{(l)}(c_l \cup c)\right)$

      $C(c) \leftarrow C(c) \cup C^{(l)}(c_l)$

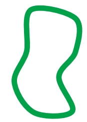
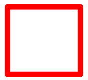
$E \leftarrow \min(E, 2\beta)$

$E \leftarrow \frac{E - \min(E)}{\max(E) - \min(E)}$

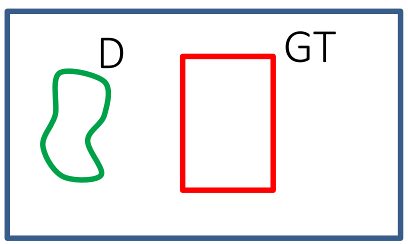
Experiments

- **Datasets:**
  - UCSD Ped 1, Ped 2[1] and Avenue[2]
  - resize into 256 x 256
- Experimental settings
  - $\alpha = 2, \beta = 0.8$ and $\rho = 0.75$ (best performance)
- Criteria: AUC (Area Under Curve) and EER (Equal Error Rate)
  - *frame-level*[1]
  - *pixel-level*[1]
  - *dual-pixel level*[3]

[1](Li, Mahadevan, and Vasconcelos, 2014),
[2](Lu, Shi, and Jia, 2013),
[3](Sabokrou et al., 2015)

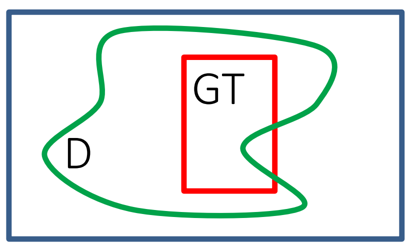
 detection (D)  ground-truth (GT)

frame-level



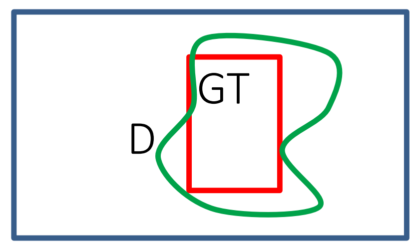
True Positive:
$|D| > 0$ and $|GT| > 0$

pixel-level



True Positive:
$|D \cap GT| / |GT| > 0.4$

dual-pixel level



True Positive:
$|D \cap GT| / |GT| > 0.4$ and $|D \cap GT| / |D| > 0.05$

Experiments

- Abstract feature representations

  - MLAD_0: low-level detector **only**

  - $\mathrm{MLAD}_{0+\mathrm{Alex}}$: low-level detector + **high-level** detector using Conv5 of **AlexNet**[1]

  - MLAD_{0+3}: low-level detector + **high-level** detector using the 3rd layer's **activation of our DAEs**

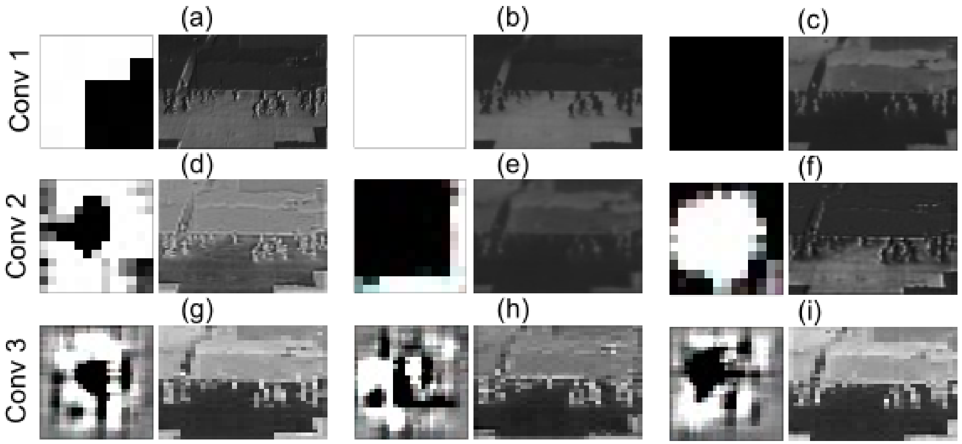
[1] *(Krizhevsky, Sutskever, and Hinton 2012)*

Experiments

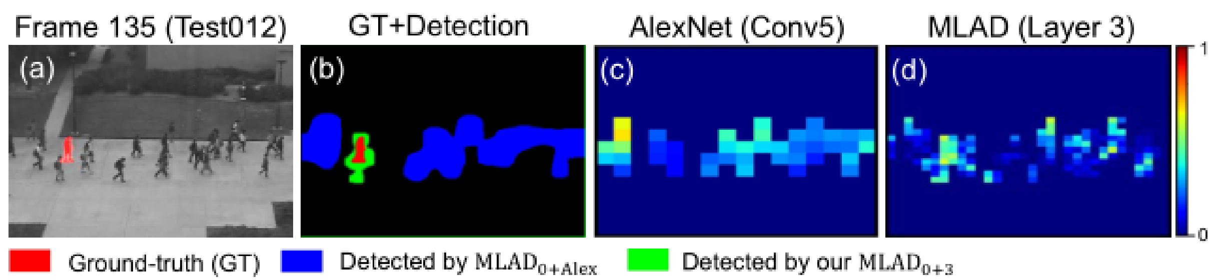
- Abstract feature representations

	UCSD Ped 1			UCSD Ped 1*			UCSD Ped 2			Avenue		
	Pixel		Dual	Pixel		Dual	Pixel		Dual	Pixel		Dual
	AUC↑	EER↓	AUC↑	AUC↑	EER↓	AUC↑	AUC↑	EER↓	AUC↑	AUC↑	EER↓	AUC↑
$MLAD_0$	66.07	**22.38**	59.74	64.41	22.32	56.79	92.96	5.47	92.39	47.07	43.90	46.05
$MLAD_{0+Alex}$	63.48	24.35	56.19	61.89	24.24	53.04	94.33	**4.43**	92.59	40.60	46.33	40.02
$MLAD_{0+3}$	**66.60**	22.65	**60.79**	**66.95**	**21.08**	**58.55**	**94.45**	4.58	**93.99**	**52.82**	**38.82**	**51.76**

Abstract-level representation vs low-level representation



Filters trained by $\mathrm{DAE_F}$



Example

➔ **our trained multilevel detector:**

- ❑ **improves performance**
- ❑ **better than AlexNet-based detector**

Experiments

- **Combined detections**
  - different networks
    - (A) 32/16/8
    - (B) 32/64/128
    - (C) 32/64/128/256
    - (D) 64/128/256/512/1024
  - consolidation strategy
    - (I) low-level + one abstract-level detector
    - (II) low-level + all abstract-level detectors
    - (III) low-level + highest-level ($\geq$ 3) detectors

Layers		1	2	3	4	5	0+1	0+2	0+3	0+4	0+5	0+all	0+≥3
Ped 1*	(A)	34.89	37.02	57.01	_	_	64.28	60.69	***68.36***	_	_	63.14	_
	(B)	33.95	38.68	42.60	_	_	62.01	59.40	62.20	_	_	57.76	_
	(C)	26.56	37.25	36.87	53.98	_	60.21	60.61	63.23	*64.61*	_	56.89	64.13
	(D)	36.21	30.42	38.14	46.88	33.49	63.09	62.51	64.24	63.46	*64.67*	62.60	63.47
	$MLAD_0$: 64.41												
Ped 2	(A)	45.52	47.61	59.83	_	_	*93.11*	92.51	*94.45*	_	_	92.51	_
	(B)	45.68	54.22	47.04	_	_	92.44	92.68	*93.06*	_	_	92.78	_
	(C)	55.86	56.40	63.25	66.20	_	92.85	*95.34*	*96.12*	*93.69*	_	*96.87*	*96.98*
	(D)	51.20	53.39	58.68	78.65	64.73	*93.13*	*96.25*	*97.22*	*96.67*	*97.36*	*97.61*	***98.28***
	$MLAD_0$: 92.96												
Avenue	(A)	43.68	46.52	*52.33*	_	_	*49.31*	*50.73*	*52.82*	_	_	*48.66*	_
	(B)	41.03	36.65	*51.82*	_	_	*48.35*	46.88	*49.98*	_	_	*49.69*	_
	(C)	37.88	41.54	*50.04*	*47.19*	_	*47.39*	*48.38*	*50.31*	*48.43*	_	*50.93*	*51.59*
	(D)	36.83	40.40	46.35	*52.45*	*52.43*		*49.28*	*50.1*	*50.21*	*48.74*		
	$MLAD_0$: 47.07												

(A) 32/16/8 (B) 32/64/128 (C) 32/64/128/256 (D) 64/128/256/512/1024

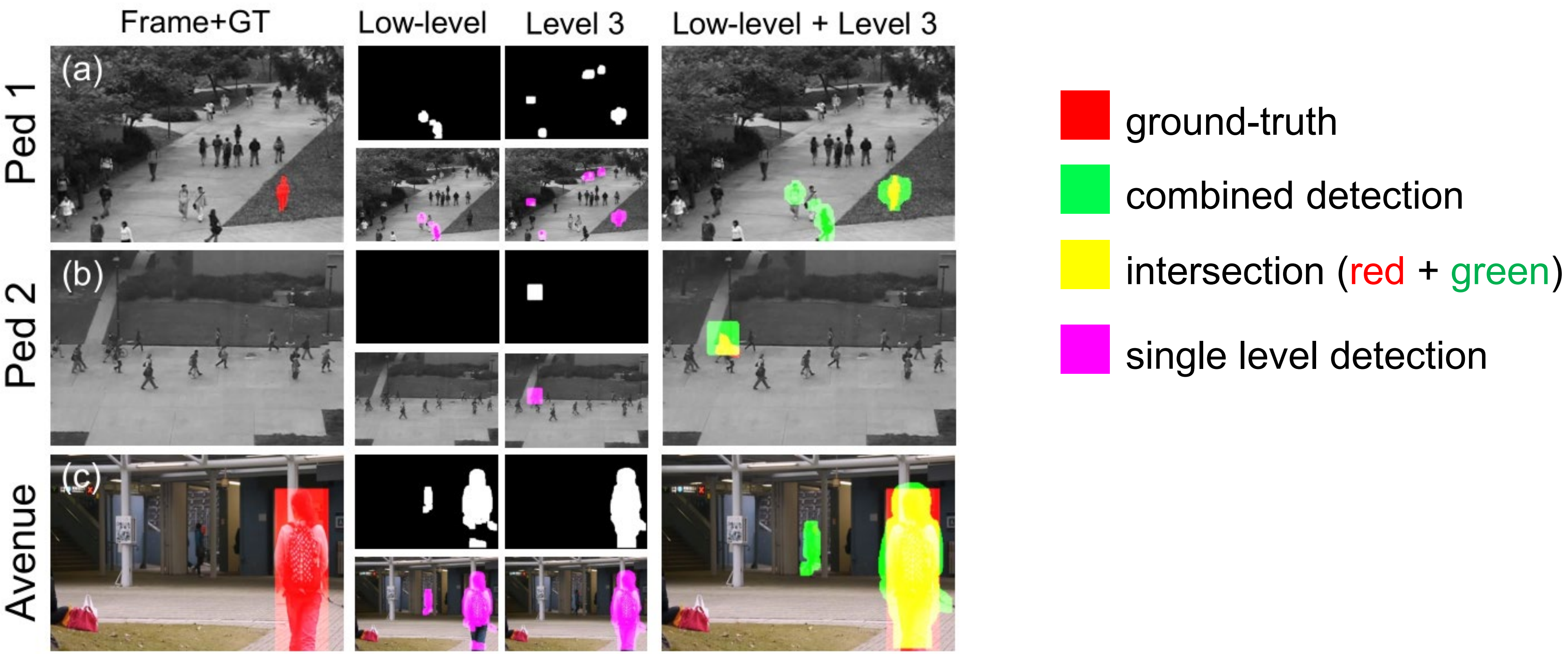
➔ **Best network and strategy combination: (A) 32/16/8 and (I) low-level + one abstract-level**

Experiments

- Combined detections
  - Some results

Layers		1	2	3	4	5	0+1	0+2	0+3	0+4	0+5	0+all	0+≥3
Ped 1*	(A)	34.89	37.02	57.01	_	_	64.28	60.69	***68.36***	_	_	63.14	_
	(B)	33.95	38.68	42.60	_	_	62.01	59.40	62.20	_	_	57.76	_
	(C)	26.56	37.25	36.87	53.98	_	60.21	60.61	63.23	*64.61*	_	56.89	64.13
	(D)	36.21	30.42	38.14	46.88	33.49	63.09	62.51	64.24	63.46	*64.67*	62.60	63.47
	$MLAD_0$: 64.41												
Ped 2	(A)	45.52	47.61	59.83	_	_	*93.11*	92.51	*94.45*	_	_	92.51	_
	(B)	45.68	54.22	47.04	_	_	92.44	92.68	*93.06*	_	_	92.78	_
	(C)	55.86	56.40	63.25	66.20	_	92.85	*95.34*	*96.12*	*93.69*	_	*96.87*	*96.98*
	(D)	51.20	53.39	58.68	78.65	64.73	*93.13*	*96.25*	*97.22*	*96.67*	*97.36*	*97.61*	***98.28***
	$MLAD_0$: 92.96												
Avenue	(A)	43.68	46.52	*52.33*	_	_	*49.31*	*50.73*	*52.82*	_	_	*48.66*	_
	(B)	41.03	36.65	*51.82*	_	_	*48.35*	46.88	*49.98*	_	_	*49.69*	_
	(C)	37.88	41.54	*50.04*	*47.19*	_	*47.39*	*48.38*	*50.31*	*48.43*	_	*50.93*	*51.59*
	(D)	36.83	40.40	46.35	*52.45*	*52.43*		*49.28*	*50.1*	*50.21*	*48.74*		
	$MLAD_0$: 47.07												

(A) 32/16/8 (B) 32/64/128 (C) 32/64/128/256 (D) 64/128/256/512/1024



Experiments

- Video anomaly detection

	Ped 1					Ped 2					Avenue				
	Frame		Pixel		Dual	Frame		Pixel		Dual	Frame		Pixel		Dual
	AUC↑	EER↓	AUC↑	EER↓	AUC↑	AUC↑	EER↓	AUC↑	EER↓	AUC↑	AUC↑	EER↓	AUC↑	EER↓	AUC↑
Machine learning methods															
OC-SVM(Vu et al. 2017)	59.06	42.97	21.78	37.47	11.72	61.01	44.43	26.27	26.47	19.23	71.66	33.87	33.16	47.55	33.15
GMM(Vu et al. 2017)	60.33	38.88	36.64	35.07	13.60	75.20	30.95	51.93	18.46	40.33	67.27	35.84	43.06	43.13	41.64
MCOV (Wang et al. 2017)	-	26.0	65.8	-	-	-	-	-	-	-	-	-	-	-	-
MDT (Mahadevan et al. 2010)	81.8	25.0	44.0	55.0	-	85.0	25.0	-	55.0	-	-	-	-	-	-
Deep models															
CAE (FR+OF)(Ribeiro, Lazzaretti, and Lopes 2017)	58.50	43.10	-	-	-	82.10	26.90	-	-	-	62.0	41.8	-	-	-
ConvAE(Hasan et al. 2016)	81.00	27.90	-	-	-	90.00	21.70	-	-	-	70.20	<u>25.10</u>	-	-	-
Adversarial AE(Sabokrou et al. 2018)	-	-	-	-	-	-	13.00	-	-	-	-	-	-	-	-
Conv-WTA+SVM[1x1](Tran and Hogg 2017)	81.3	27.9	56	46.8	-	96.6	8.9	89.3	16.9	-	-	-	-	-	-
AMDN(Xu et al. 2015)	92.1	16.0	67.2	40.1	-	-	-	90.8	17.0	-	-	-	-	-	-
DeepGMM(Feng, Yuan, and Lu 2017)	92.5	15.1	69.9	64.9	-	-	-	-	-	-	-	-	-	-	-
Plug-and-Play CNN(Ravanbakhsh et al. 2018)	95.7	<u>8.0</u>	64.5	40.8	-	88.4	18.0	-	-	-	-	-	-	-	-
GAN/generator(Ravanbakhsh et al. 2017a)	**97.40**	<u>8.0</u>	<u>70.30</u>	35.00	-	93.50	14.00	-	-	-	-	-	-	-	-
GAN/discriminator(Ravanbakhsh et al. 2017b)	<u>96.80</u>	**7.0**	**70.80**	<u>34.00</u>	-	95.50	11.00	-	-	-	-	-	-	-	-
Proposed system															
MLAD_{0+3}(A)	82.34	23.50	66.60	**22.65**	**60.79**	<u>97.52</u>	<u>4.68</u>	<u>94.45</u>	<u>4.58</u>	<u>93.99</u>	71.54	36.38	52.82	38.82	51.76
MLAD (best for each dataset)	82.34	23.50	66.60	**22.65**	**60.79**	**99.21**	**2.49**	**97.22**	**1.74**	**96.75**					
	MLAD_{0+3}(A)					MLAD_{0+3}(D)					MLAD_{5}(D)				

➔ Significantly improve over the state-of-the-art deep detectors

Pixel-level EER improvement:

- ❑ **Ped 1: 11.35%**
- ❑ **Ped 2: 12.32%**
- ❑ **Avenue: 4.31%**

Conclusion

- **Low-level feature based detectors**
  - fragmented and interrupted detection regions
  - false detections by noise and environment changes
- **Proposed detector (MLAD)**
  - combine low-level and abstract-level detections
    - increase reliability and reduce false detections
- **Experiments**
  - three standard benchmarks
  - improve at least 4% in pixel-level EER in VAD task

References

Sodemann, A.A., Ross, M.P., Borghetti, B.J.: A Review of anomaly detection in automated surveillance. IEEE Trans. Syst. Man Cybern., Part C, Appl. Rev. 42(6), 1257-1272 (2012).

Hasan, M.; Choi, J.; Neumann, J.; Roy-Chowdhury, A. K.; and Davis, L. S. 2016. Learning Temporal Regularity in Video Sequences. In CVPR, volume abs/1604.04574.

Ribeiro, M.; Lazzaretti, A. E.; and Lopes, H. S. 2017. A Study of Deep Convolutional Auto-Encoders for Anomaly Detection in Videos. Pattern Recognition Letters.

Chong, Y. S., and Tay, Y. H. 2017. Abnormal Event Detection in Videos Using Spatiotemporal Autoencoder. In Advances in Neural Networks - ISNN 2017: 14th International Symposium, Part II, 189–196.

Luo, W.; Liu, W.; and Gao, S. 2017. Remembering history with convolutional lstm for anomaly detection. In ICME, 439–444.

Ravanbakhsh, M.; Nabi, M.; Sangineto, E.; Marcenaro, L.; Regazzoni, C. S.; and Sebe, N. 2017a. Abnormal Event Detection in Videos using Generative Adversarial Nets. In International Conference on Image Processing (ICIP), 1577–1581.

Ravanbakhsh, M.; Sangineto, E.; Nabi, M.; and Sebe, N. 2017b. Training Adversarial Discriminators for crosschannel Abnormal Event Detection in Crowds. CoRR.

Sabokrou, M.; Khalooei, M.; Fathy, M.; and Adeli, E. 2018. Adversarially learned one-class classifier for novelty detection. In CVPR.

Isola, P.; Zhu, J.; Zhou, T.; and Efros, A. A. 2017. Image-to-image translation with conditional adversarial networks. In CVPR, 5967–5976.

Li, W.-X.; Mahadevan, V.; and Vasconcelos, N. 2014. Anomaly Detection and Localization in Crowded Scenes. In TPAMI, volume 36, 18–32.

Lu, C.; Shi, J.; and Jia, J. 2013. Abnormal Event Detection at 150 FPS in MATLAB. In ICCV.

Sabokrou, M.; Fathy, M.; Hosseini, M.; and Klette, R. 2015. Real-Time Anomaly Detection and Localization in Crowded Scenes. CVPRW.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. ImageNet Classification with Deep Convolutional Neural Networks. In NIPS, 1106–1114.

Question

THANK YOU