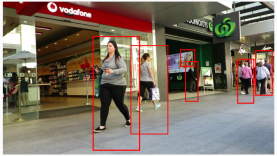
ODANet: Online Deep Appearance Network for Identity-Consistent Multi-Person Tracking

Guillaume Delorme[1], Yutong Ban[2], Guillaume Sarrazin[1], Xavier Alameda-Pineda[1]
[1]Inria, LJK, Univ. Grenoble Alpes, France [2] MIT CSAIL Distributed Robotics Lab

10/01/2021, MPRSS 2020, Milano, Italy

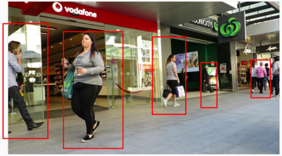
Introduction

Tracking by detection



t

t + 1

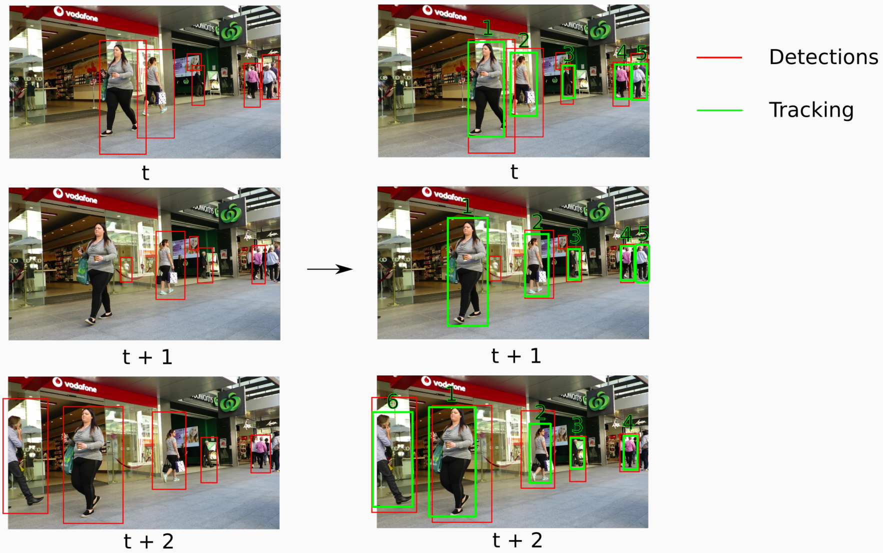


t + 2

Detections

Tracking

Tracking by detection



Multiple Object Tracking

- Our tracker can be seen as a generalization of a **Kalman Filter** dealing with **multiple** objects at once.

Multiple Object Tracking

- Our tracker can be seen as a generalization of a **Kalman Filter** dealing with **multiple** objects at once.
- Dealing with multiple objects/targets introduces a detection-to-target **assignment problem**.

Multiple Object Tracking

- Our tracker can be seen as a generalization of a **Kalman Filter** dealing with **multiple** objects at once.
- Dealing with multiple objects/targets introduces a detection-to-target **assignment problem**.It alternates between a **GMM responsibility** computation and a weighted **Kalman forward pass**.

Multiple Object Tracking

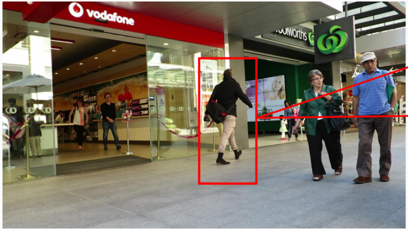
- Our tracker can be seen as a generalization of a **Kalman Filter** dealing with **multiple** objects at once.
- Dealing with multiple objects/targets introduces a detection-to-target **assignment problem**.It alternates between a **GMM responsibility** computation and a weighted **Kalman forward pass**.
- To fully **disambiguate** the assignment problem, we need a **discriminative** appearance model, which **adapts** to the situation at hand.

Appearance modeling

The observation model

- A key component of our model is the definition of the Observation Model:

$$\overbrace{p(\mathbf{o}_t|\mathbf{x}_t, Z_t = n)}^{\text{Which track to associate } \mathbf{o}_t \text{ with?}} = \underbrace{p(\mathbf{y}_t|\mathbf{x}_t, Z_t = n)}_{\text{Geometric Model, the closest one}} \times \underbrace{p(\mathbf{u}_t|Z_t = n)}_{\text{Appearance Model, the most similar one}} \tag{1}$$



$y_{t,k}$: Position, and size of $o_{t,k}$

u_{tk}: photometric/appearance information:

Histogram-based appearance model

- Previous strategy: use of **hand-crafted** descriptors and metrics to compute appearance similarity.

Histogram-based appearance model

- Previous strategy: use of **hand-crafted** descriptors and metrics to compute appearance similarity.
- Lack **discriminative** power and **robustness**, due to appearance variations (Illumination, pose, background, occlusions...)

Deep Appearance Model

Inspired by model-based tracking methods[1], our strategy is to **learn** a descriptor.

- Deep Appearance Models are generally trained **offline**, on large manually annotated datasets[2].

[1] Junlin Hu, Jiwen Lu, and Yap-Peng Tan. "Deep metric learning for visual tracking". In: *IEEE Transactions on Circuits and Systems for Video Technology* 26.11 (2016), pp. 2056–2068.

[2] Siyu Tang et al. "Multiple People Tracking by Lifted Multicut and Person Re-identification". In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Washington, DC, USA: IEEE Computer Society, July 2017, pp. 3701–3710.

Deep Appearance Model

Inspired by model-based tracking methods[1], our strategy is to **learn** a descriptor.

- Deep Appearance Models are generally trained **offline**, on large manually annotated datasets[2].
- While they seek generality, they **lack discriminative power**.

[1] Junlin Hu, Jiwen Lu, and Yap-Peng Tan. "Deep metric learning for visual tracking". In: *IEEE Transactions on Circuits and Systems for Video Technology* 26.11 (2016), pp. 2056–2068.

[2] Siyu Tang et al. "Multiple People Tracking by Lifted Multicut and Person Re-identification". In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Washington, DC, USA: IEEE Computer Society, July 2017, pp. 3701–3710.

Deep Appearance Model

Inspired by model-based tracking methods[1], our strategy is to **learn** a descriptor.

- Deep Appearance Models are generally trained **offline**, on large manually annotated datasets[2].
- While they seek generality, they **lack discriminative power**.
- We want to train a NN ψ_ω using past detections annotated by the tracker, and update it **every few frames**.

[1] Junlin Hu, Jiwen Lu, and Yap-Peng Tan. "Deep metric learning for visual tracking". In: *IEEE Transactions on Circuits and Systems for Video Technology* 26.11 (2016), pp. 2056–2068.

[2] Siyu Tang et al. "Multiple People Tracking by Lifted Multicut and Person Re-identification". In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Washington, DC, USA: IEEE Computer Society, July 2017, pp. 3701–3710.

Appearance model update

We update only the top layers of ψ_ω in a siamese setting using the contrastive loss

$$\mathcal{J}(\omega) = \frac{1}{2}\sum_{i,j=1} \max(0, 1 - l_{ij}(\tau - \|\psi_\omega(\mathbf{u}_i) - \psi_\omega(\mathbf{u}_j)\|^2)),$$

Appearance model update

We update only the top layers of ψ_ω in a siamese setting using the contrastive loss

$$\mathcal{J}(\omega) = \frac{1}{2} \sum_{i,j=1} \max(0, 1 - l_{ij}(\tau - \|\psi_\omega(\mathbf{u}_i) - \psi_\omega(\mathbf{u}_j)\|^2)),$$

It needs to be supervised with **binary** (+/-) labels. We have access to past **posterior** estimation $q(\mathbf{z}_t)$,thus we use **soft labelisation** instead:

Appearance model update

We update only the top layers of ψ_ω in a siamese setting using the contrastive loss

$$\mathcal{J}(\omega) = \frac{1}{2}\sum_{i,j=1} \max(0, 1 - l_{ij}(\tau - \|\psi_\omega(\mathbf{u}_i) - \psi_\omega(\mathbf{u}_j)\|^2)),$$

It needs to be supervised with **binary** (+/-) labels. We have access to past **posterior** estimation $q(\mathbf{z}_t)$,thus we use **soft labelisation** instead:

$$\gamma_{ij} = p(Z_{t_i k_i} = Z_{t_j k_j} | \mathbf{o}_{1:t-1}) \approx \sum_{n=1}^{N} q(Z_{t_i k_i} = n) q(Z_{t_j k_j} = n).$$

We label positive pairs with $l_{ij} = \gamma_{ij}$ and negative pairs with $l_{ij} = -(1 - \gamma_{ij})$.

Implementation details

- To make it work smoothly, we use 2 models in parallel, one for training and the other for inference. Our implementation reaches 10 FPS in our framework.

[3] Ergys Ristani et al. "Performance Measures and a Data Set for Multi-Target, Multi-Camera Tracking". In: *ECCV Workshops*. 2016.

Implementation details

- To make it work smoothly, we use 2 models in parallel, one for training and the other for inference. Our implementation reaches 10 FPS in our framework.
- The convolutional layers of ψ are pretrained using external Re-ID dataset, using a standard training framework[3].

[3] Ergys Ristani et al. “Performance Measures and a Data Set for Multi-Target, Multi-Camera Tracking”. In: *ECCV Workshops*. 2016.

Results

Quantitative results: evaluation settings

We evaluate our tracker in a **multi-party conversation**: the robot has a **low fov** and often change position, increasing **identity switches**.

[4] A. Milan et al. "MOT16: A Benchmark for Multi-Object Tracking". In: *arXiv:1603.00831 [cs]* (Mar. 2016). arXiv: 1603.00831. URL: http://arxiv.org/abs/1603.00831.

[5] Keni Bernardin and Rainer Stiefelhagen. "Evaluating Multiple Object Tracking Performance: The CLEAR MOT Metrics". In: *EURASIP Journal on Image and Video Processing* (2008).

Quantitative results: evaluation settings

We evaluate our tracker in a **multi-party conversation**: the robot has a **low fov** and often change position, increasing **identity switches**.
We also want to use a **standard** dataset, thus we use of the MOT16 dataset[4] that we divide in 2 evaluation settings:

- *moving surveillance camera* for the sequences with camera **fixed**: we **simulate** the **camera movement** to increase identity switches.

[4] A. Milan et al. "MOT16: A Benchmark for Multi-Object Tracking". In: *arXiv:1603.00831 [cs]* (Mar. 2016). arXiv: 1603.00831. URL: `http://arxiv.org/abs/1603.00831`.

[5] Keni Bernardin and Rainer Stiefelhagen. "Evaluating Multiple Object Tracking Performance: The CLEAR MOT Metrics". In: *EURASIP Journal on Image and Video Processing* (2008).

Quantitative results: evaluation settings

We evaluate our tracker in a **multi-party conversation**: the robot has a **low fov** and often change position, increasing **identity switches**.
We also want to use a **standard** dataset, thus we use of the MOT16 dataset[4] that we divide in 2 evaluation settings:

- *moving surveillance camera* for the sequences with camera **fixed**: we **simulate** the **camera movement** to increase identity switches.
- *robot navigating in the crowd* for the sequences where the camera is **moving**.

We use the CLEAR metrics[5] to evaluate the quality of the tracker results.

---

[4] A. Milan et al. "MOT16: A Benchmark for Multi-Object Tracking". In: *arXiv:1603.00831 [cs]* (Mar. 2016). arXiv: 1603.00831. URL: `http://arxiv.org/abs/1603.00831`.
[5] Keni Bernardin and Rainer Stiefelhagen. "Evaluating Multiple Object Tracking Performance: The CLEAR MOT Metrics". In: *EURASIP Journal on Image and Video Processing* (2008).

Quantitative results: moving surveillance setting

Model	Rcll (↑)	Prcn(↑)	IDs(↓)	FM (↓)	MOTA(↑)
CH[6]	49.41	88.20	266	759	42.49
ODA-FR	49.53	88.66	195	702	42.97
ODA-UP (Ours)	54.72	86.68	591	976	**45.63**

Table 1: Results on the *moving surveillance camera* setting.

- ODA-UP stands for our *online deep appearance update*.
- ODA-FR refers to the same appearance model architecture, but frozen (FR), trained on an external person Re-ID dataset.
- CH stands for Color Histogram based appearance model.

[6] Yutong Ban et al. "Tracking a varying number of people with a visually-controlled robotic head". In: *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2017, pp. 4144–4151.

Quantitative results: robot navigating in the crowd

Model	Rcll (↑)	Prcn(↑)	IDs(↓)	FM (↓)	MOTA(↑)
CH[7]	45.81	91.80	698	1704	41.15
ODA-FR	45.78	93.12	516	1524	41.97
ODA-UP (Ours)	52.29	90.48	782	1499	**46.15**

Table 2: Results on the *robot navigating in the crowd* setting.

- ODA-UP stands for our *online deep appearance update*.
- ODA-FR refers to the same appearance model architecture, but frozen (FR), trained on an external person Re-ID dataset.
- CH stands for Color Histogram based appearance model.

[7] Yutong Ban et al. "Tracking a varying number of people with a visually-controlled robotic head". In: *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2017, pp. 4144–4151.

Conclusion

Thank you for your attention.