ISSN 2616-7433 Vol. 2, Issue 9: 37-39, DOI: 10.25236/FSST.2020.020909

Multi Instance Deep Learning Target Tracking

Cheng Huaihui[1], Zhang Shengbo[2]

1 Nanyang Technological University, Singapore 639798
2 Carnegie Mellon University, Pittsburgh, PA 15213, United States

ABSTRACT. *In order to solve the problem of low tracking accuracy caused by the lack of appearance model and motion model in multi instance tracking algorithm, this paper proposes a multi instance deep learning target tracking algorithm. In the original multi example tracking algorithm, the motion model only assumes that the motion of the object between frames will not exceed a certain range, which can not effectively reflect the motion state of the target. Particle filter algorithm is introduced to predict the target and improve the accuracy of tracking. The experimental results of different image sequences in complex environment show that the proposed algorithm has higher tracking accuracy and better robustness than other tracking algorithms.*

KEYWORDS: *Target tracking, Multi example learning, Deep learning*

1. Introduction

Multi-instance (MI) learning is a variant of inductive machine learning where each learning example contains a bag of instances instead of a single feature vector. The training data of MI learning is $\left\{(X_1, y_1), (X_2, y_2)\cdots(Xn, yn)\right\}$,wher X_i is a package containing M instances $\left\{x_{i1}, x_{i2}\cdots x_{im}\right\}$, $yi \in \left\{0, 1\right\}$ Is the label of the corresponding sample package. The sample package label is defined as

$$y_i = \max(y_{ij}) \tag{1}$$

y_{ij} is the instance label of package x_i According to formula (1), a package containing at least one positive instance is labelled as positive and vice versa. In the process of target tracking, the enhanced learning method is used to train the classifier, that is, to maximize the log likelihood function below.

$$L = \sum_i (\lg p(y_i | X_i)) \tag{2}$$

In formula 2, $p(y_i | Xi)$ is the probability of the packet x_i being of class y_i. $p(y_i | Xi)$ can be calculated with the probability of instance, $p(y_i | x_{ij})$. Calculation with noisy or model,The formula is as follows

$$p(y_i | X_i) = 1 - \prod (1 - p(y_i | x_{ij})) \tag{3}$$

The probability function of each instance is:

$$p(y | x) = \sigma(H(x)) \tag{4}$$

In equation 4, $\sigma(x) = 1/(1 + e^{-x})$ and H(x)Is a strong classifier which is composed of several weak classifiers.

In the process of target tracking, it is very difficult to obtain all weak features at one time, and it’s also difficult to meet the real-time requirements of tracking. Therefore, the original multi-instance learning tracking algorithm randomly generates m weak features from all weak features to form a weak feature pool. Then a sample set is selected as a set of follow-up features. The algorithm keeps m candidate weak classifiers $(m > k)$ and selects k weak classifiers by using the framework of reinforcement learning:

ISSN 2616-7433 Vol. 2, Issue 9: 37-39, DOI: 10.25236/FSST.2020.020909

$$h_k = \arg\max L(H_{k-1} + h) \tag{5}$$

In formula 5, L is the same as L in formula 2. As shown in formula 2, the essence of selecting weak classifiers is that after cascading the weak classifiers the strong classifier can maximize the difference between the positive and negative packet responses. In the whole weak classifier set, the weak classifier h with the largest L can be selected.

Weak classifier is estimated on line by Haar-like feature $f_k(x)$ and four parameters. The calculation formula of weak classifier is $(\mu_1, \sigma_1, \mu_0, \sigma_0)$ On line estimation, The calculation formula of weak classifier is as follows.

$$h_k(x) = \lg\left[\frac{p_t(y = 1 | f_k(x))}{p_t(y = 0 | f_k(x))}\right] \tag{6}$$

In equation 6, $p_t(y = 1 | f_k(x)), p_t(y = 0 | f_k(x))$ is the probability function of the instance $p(y=1) = p(y=0)$. The Bayes formula can be used to deduce:

$$h_k(x) = \lg\left[\frac{p_t(f_k(x) | y = 1)}{p_t(f_k(x) | y = 0)}\right] \tag{7}$$

$p_t(f_k(x) | y = 1) \sim N(\mu_1, \sigma_1)$ $p_t(f_k(x) | y = 0) \sim N(\mu_0, \sigma_0)$. μ_1 and σ_1 are the mean and standard deviation of Gaussian distribution of positive instances; μ_0 and σ_0 are the mean and standard deviation of the Gaussian distribution of the negative instances. Every time the weak classifier receives new data $\{(x_1, y_1), (x_2, y_2) \ldots (x_n, y_n)\}$,the classifier need to update its parameters:

$$\mu_1 \leftarrow \rho\mu_1 + (1 - \rho)\frac{1}{n}\sum_{i|yi=1} f_k(x) \tag{8}$$

$$\sigma \leftarrow \rho\sigma + (1 - \rho)\sqrt{\frac{1}{n}\sum_{i|yi=1}(f_k(x_i) - \mu_1)^2} \tag{9}$$

In equation 9, ρis the learning rate and $0 < \rho < 1$. The formulas to update $\mu 0$, $\sigma 0$ are similar to formula 8 and formula 9.

2. Multi-Instance Deep Learning Target Tracking

The flow of multi example deep learning tracking algorithm is as follows:

(1)For $t-1$ frame, the target position is l_{t-1},Particle filter is used to estimate a set of positions of the target in the next image $l(x) = \{l_1(x), l_2(x) \cdots l_s(x)\}$,Get a set of image blocks from particle set: $X^8 = \{x \in l(x)\}$,Then the feature vector of each image block is extracted by the encoder of sdae.

(2) Using the extracted feature vector to build a weak classifier, and then using the enhanced learning algorithm to select the most discriminative weak classifier to build a strong classifier online, at the same time using the newly generated feature vector to replace some of the weakest features.

(3)Use strong classifier for all $x \in X^8$ estimate $p(y=1|x)$.

(4)Update target's best location $l_t = l[\arg\max p(y|x)]$

(5)Collect positive and negative sample packages $X^r = \{x \| l(x) - l_t \| \langle r\}$, $X^{r,\beta} = \{x | \gamma \langle \| l(x) - l_t \| < \beta\}$

ISSN 2616-7433 Vol. 2, Issue 9: 37-39, DOI: 10.25236/FSST.2020.020909

(6)Use positive sample package X^{γ} And negative sample package $X^{\gamma,\beta}$ Update appearance model.

3. Results

The center error represents the Euclidean distance between the tracking result and the center of the real target. The smaller the distance is, the more effective and accurate the tracking result is. As shown in Table 1, the center error of the algorithm in this paper is generally smaller than that of other algorithms, indicating that the tracking results have high accuracy.

Table 1 Mean Center Error

Picture sequence	ML	WMIL	TLD	CT	CXT	MTT	Algorithm in this paper
david	24	*9*	11	11	*9*	20	8
faceOcc1	28	*15*	32	41	28	30	12
faceOcc2	21	12	19	24	10	14	*11*
girl	33	*8*	11	19	11	12	9
sylvester	14	11	12	14	20	*10*	7
tiger2	18	*17*	41	71	45	57	14
singer1	23	12	7	35	18	*8*	11
Average	23	*13*	19	31	20	22	10

4. Conclusion

In this paper, a tracking algorithm based on multi instance deep learning is proposed, which can effectively express the image by using reinforcement learning and apply the information from offline training to online tracking process. By replacing the feature vectors extracted from the deep network in real time, new feature representation is introduced into the target representation to reflect the appearance of changes of the target in the tracking process, which improves the adaptability and robustness of the model. Particle filter is also introduced to improve the motion model of multi instance learning and improve the tracking accuracy.

References

[1] Wu Y, Lim J, and Yang M H. Online object tracking: a benchmark[C]. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Portland, USA, 2013: 2411-2418.
[2] Ross D A, Lim J, Lin R S, et al (2008). Incremental learning forrobust visual tracking [J]. International Journal of Computer Vision, vol.77, no.3, pp.125-141.
[3] Zhang K, Zhang L, Yang M H (2014). Fast compressive tracking [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.36, no.10, pp.2002-2015.
[4] Zhong W, Lu H C, Yang M H (2014). Robust object tracking via sparse collaborative appearance model [J]. IEEE Transactions on Image Processing, vol.23, no.5, pp.2356-2368.
[5] Zhang K, Zhang L, Liu Q, et al (2014). Fast tracking via densespatio-temporal context learning[C]. Proceedings of European Conference on Computer Vision, Zurich, Switzerland, pp.127-141.
[6] Kalal Z, Mikolajczyk K, Matas J (2012). Tracking-learning-detection [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.34, no7, pp.1409-1422.
[7] Babenko B, Yang M H, Belongie S (2011). Robust object tracking with online multiple instance learning [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.33, no.8, pp.1619-1632.
[8] Zhang K H, Song H H (2013). Real-time visual tracking via online weighted multiple instance learning [J]. Pattern Recognition, vol.46, no.1, pp.397-411.