Motion models for multiple object tracking

Oleg Garanin[a]

[a] *The Branch of NRU "MPEI" in Smolensk, Smolensk, Russian Federation*

Abstract

In this paper, we consider the models for solving the problem of Multiple Object Tracking (MOT). We have compared the accuracy of the following models: autoregressive, moving average, Kalman filter. We compared the methods with using the *MAPE* metric and the 2dmot15 dataset. Prediction using an autoregressive model has good accuracy and can be used to build a MOT model.

Keywords 1

Multiple object tracking, motion model

1. Introduction

To develop a Multiple Object Tracking (MOT) method, the following models are usually used [4]: appearance, motion, interaction, exclusion, occlusion models.

Appearance models use a certain set of features to describe an object. Motion models investigate the dynamic behavior of an object and, based on the history, allow predicting in what position of space a given object will be at the next moments in time. Interaction models assess the influence of other objects on a given object (for example, if several people are moving in a crowd, a given person is in the crowd, then most likely he will move with the crowd). Exclusion models are based on the assumption that both objects cannot occupy the same position in space at the same time. Occlusion models take into account the overlap of part or all of an object by other objects.

Basically, to develop a method for tracking a set of objects, several models are used, most often an appearance model of an object and a motion model.

When constructing a motion model, it is quite important that this model allows predicting the position of an object with the required accuracy and at the same time has a low computational complexity.

The purpose of this work is to compare different motion models to construct a tracking method and to identify models that have the required accuracy and low computational complexity.

2. Review of existing motion models

The motion model evaluates the dynamic behavior of an object and, based on the history, allows predicting what position of space the given object will be in at the next moments in time. The predicted position can then be compared to the value obtained using the detector and corrected based on the data obtained.

Currently, linear and nonlinear motion models are used to solve the problem of tracking multiple objects. Mainly, linear models of motion with a constant position (the object does not move), with a constant speed or constant acceleration are used [4], since the motion of an object most often obeys a linear law.

[1] Russian Advances in Fuzzy Systems and Soft Computing: selected contributions to the 8-th International Conference Fuzzy Systems, Soft Computing and Intelligent Technologies (FSSCIT-2020), June 29 – July 1, 2020, Smolensk, Russia

EMAIL: hedgehog@mail.ru

For example, in ELP [5], the task of tracking a set of objects is solved using a linear regression-based motion model. This assessment is based on the assumption that movement over short periods of time can be represented as a linear movement pattern.

SORT [1] and Deep Sort [7] use the Kalman filter to predict the new state of objects. In models based on the Kalman filter, it is assumed that a moving object has a certain internal state, which is measured at each frame. Moreover, in [1], a state is understood as a set of characteristics: coordinates of the center of the bounding box, its area and aspect ratio. This model takes into account the rate of change in the position of the center of the rectangle and the area, while the aspect ratio is considered unchanged. In [7], a state is understood as a slightly different set of characteristics: coordinates of the center of the bounding box, aspect ratio and its width, while taking into account the rate of change of all values.

Non-linear models are used less frequently in MOT because they are more complex. Such models can be applied, for example, in cases when the linear model fails to describe the occlusion of objects on a large number of frames in a row. For example, in [6], the LSTM neural network is used to predict the speed of an object [2].

3. Description of the motion model for predicting the position of the object in space

The position of the object in space at a current point in time (the frame number is used as the point in time) is set using the following set: (w, h, x_0, y_0), where w is the width, h is the height, (x_0, y_0) is the center of the bounding box. Designations for the position of an object in space are shown in Figure 1.

Figure 1: Designation of the object position in space

The transformation of the set of object coordinates (x_1, y_1, x_2, y_2), where x_1, y_1 are the coordinates of the upper left corner of the bounding box, x_2, $y_2$are the coordinates of the upper lower corner of the bounding box, obtained using the detector, to this set of coordinates (w, h, x_0, y_0) is carried out as follows in (1)

$$\begin{pmatrix} w \\ h \\ x_0 \\ y_0 \end{pmatrix} = \begin{pmatrix} x_2 - x_1 \\ y_2 - y_1 \\ x_1 + \frac{x_2 - x_1}{2} \\ y_1 + \frac{y_2 - y_1}{2} \end{pmatrix}, \quad (1)$$

The calculation of the position of the object at the next moment in time is performed based on the previous values of the coordinates of the object for each component of the vector separately based on the motion model.

4. Comparison of the accuracy of different motion models to predict the position of an object in the next frame

As mentioned earlier, the motion model examines the dynamic behavior of an object and, based on the history, allows predicting in what position of space the given object will be at the next moments in time. Thus, such prediction can be considered as a time series forecasting problem.

To select a motion model (a method for predicting a time series), an experiment was carried out: 200 frames of object motion with a static camera were selected from the 2D MOT15 dataset [3] from the right side of the image to the left side, as shown in Figure 2.

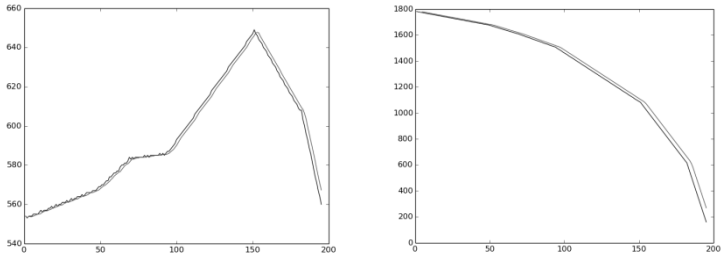
The 2D MOT15 dataset is designed to evaluate the effectiveness of the application of object tracking methods and contains more than 5 thousand training frames (11 sequences) with 500 tracks and 39905 objects and more than 5 thousand test frames (11 sequences) with 721 tracks and 61,440 objects. When forming the dataset, both static and moving cameras were used. It contains scenes with different lighting, object sizes and camera speed.

Then, training data from the dataset for this track were selected and graphs of the movement of the center of this object along the x and y coordinates were plotted, and the time series was forecast using the moving average method, using the Kalman filter, using autoregressive.

Figure 2: Object moving illustration

When calculating a moving average, prediction is performed as follows: the average value of the coordinate over the two previous frames is calculated, and the value on the next frame is considered equal to the value of this average. Figure 3 shows graphs of the object's movement along the *x* and *y*

coordinates and their forecast using a moving average. The *x*-axis is the frame number, the *y*-axis is the coordinate of the object in the image.

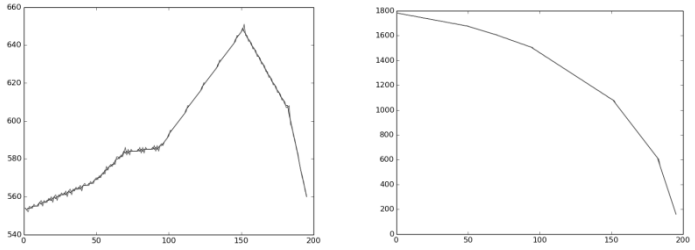


a) b)

Figure 3: The graph of the movement of an object along the *x* coordinate and its prediction using a moving average (a); on the *y* coordinate and its prediction using a moving average (b)

It follows from the graphs that a forecast using a moving average cannot be performed with high accuracy. In addition, calculating the moving average using more points further increases the error.

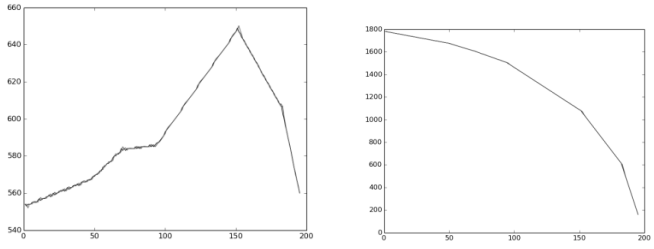
When calculating autoregression, the forecast is performed on the two previous frames with the construction of a linear relationship: the value in the next frame is calculated based on a linear function. The graphs are shown in Figure 4.



a) b)

Figure 4: The graph of the object's movement along the *x* coordinate and its prediction using autoregressive (a); along the *y* coordinate and its prediction using autoregressive (b)

Figure 5 shows the forecast using the Kalman filter with the parameters used in [1].



a) b)

Figure 5: The graph of the object's movement along the *x* coordinate and its prediction using the Kalman filter (a); in the *y* coordinate and its prediction using the Kalman filter (b)

Comparison of prediction accuracy was carried out using the mean absolute percentage prediction error (*MAPE*) in (2)

$$\varepsilon = \frac{1}{N} \cdot \sum_{t=1}^{n} \left| \frac{y_t - \tilde{y}_t}{y_t} \right| \cdot 100, \quad (2)$$

where y_t is the actual value of the predicted time series at time *t*; $\tilde{y}_t$ – forecast of the time series at time *t*; *N* is the number of time series samples.

Table 1

Method	Moving average	Autoregression	Kalman filter
MAPE (*x*), %	1,91	0,07	0,09
MAPE (*y*), %	0,24	0,13	0,1

It follows from Table 1 that forecasting using the autoregressive model has a fairly good accuracy and can be used for prediction.

However, it should be noted that in order to level the errors of the detector readings, one should use the history of motion based on a larger number of points, for example, 5–10, and perform an approximation, for example, based on the least squares method (OLS).

5. Conclusion

This article analyzes the existing models for solving the problem of Multiple Object Tracking: appearance, motion, interaction, exclusion, occlusion. We have compared the accuracy of the following models: autoregressive, moving average, Kalman filter. We compared the methods with using the *MAPE* metric and the 2dmot15 dataset. Prediction using an autoregressive model has good accuracy and can be used to build a MOT model.

6. References

[1] A. Bewley, Z. Ge, L. Ott, F. Ramos, B. Upcroft, Simple online and realtime tracking. arXiv preprint arXiv: 1602.00763, 2016.

[2] S. Hochreiter, J. Urgen, Schmidhuber Long Short-Term Memory. Neural Computation, 9(8) (1997) 1735-1780.

[3] L. Leal-Taixe, A. Milan, I. Reid, S. Roth, K. Schindler, MOTChallenge 2015: Towards a Benchmark for Multi-Target Tracking. arXiv preprint arXiv: 1504.01942, 2015.

[4] W. Luo, J. Xing, Multiple Object Tracking: A Literature Review. arXiv preprint arXiv: 1409.7618, 2017.

[5] N. McLaughlin, J. M. D. Rincon, P. Miller, Enhancing Linear Programming with Motion Modeling for Multi-target Tracking, In Proceedings of the IEEE Winter Conference on Applications of Computer Vision, 2015, 271–350.

[6] A. Sadeghian, A. Alahi, S. Savarese, Tracking The Untrackable: Learning To Track Multiple Cues with Long-Term Dependencies. arXiv preprint arXiv: 1701.01909, 2017.

[7] N. Wojke, A. Bewley, D. Paulus, Simple online and realtime tracking with a deep association metric. arXiv preprint arXiv: 1703.07402, 2017.