2011 International Conference on Computer Science and Information Technology (ICCSIT 2011)
IPCSIT vol. 51 (2012) © (2012) IACSIT Press, Singapore
DOI: 10.7763/IPCSIT.2012.V51.50

Prediction of Motion Trajectories Based on Markov Chains

Ling Gan[+] and Li'na Su

College of Computer Science and Technology, Chongqing University of Posts & Telecommunications, Chongqing, China

Abstract. As the research of object behaviors become more and more important of computer vision in recent years, Trajectories analysis became a hot topic as it is an basic problem of object behaviors learning and description. In this paper we present a predict object trajectories model based on MARKOV CHAINS for the learning of trajectory distribution patterns of event recognition. Due to MARKOV CHAINS predict trajectory model's repetition feature, it keeps correcting the prediction by calculated the data from abnormal behaviors, which called automatic learning. With the two different sets of data used to do the experimental that approved predict object trajectories model of MARKOV CHAINS has highly accuracy, efficiency and less dimensions compared with other learning of trajectory distribution patterns.

Keywords: MARKOV CHAINS, trajectory analysis and learning, anomaly detection, behavior prediction.

1. Introduction

As the tremendous potential value of visual surveillance, people pay more attention on it. Target detect, object classification, object tracking and event analysis which are the basic problems of visual surveillance obtained widely consideration especially event recognition. In this paper we focus on trajectory analysis which also is the basic problem of event recognition, without discussing other fields such as object tracking.

Most visual surveillance system and event recognition based on scenes that is already known, in which objects moving with a established way. In this case, each scene needs to define a set of object behaviors and keep updating since object behaviors changed. We can't predict object behaviors even in this way the environment is fixed, so it is useful and necessary to find a general method of event recognition for predicting object behaviors based on automatic generate model.

Johnson et al. present a statistics object trajectory model which generated from image sequences, in this model object behaviors are described as a set of sequence flow volumes, each volume contains 4 elements to express the object's position and velocity of image plane. The statistical model of object trajectories is formed with two two-layer competitive learning networks that are connected with leaky neurons. Paper 2 described a non-adaptive predict model, which predicts moving car direction in time k+1 based on a sequence statuses of the front k times. As without adaptive learning, predict accuracy can be low.

2. Markov chains

2.1. Markov process and Markov chain

Definition 1: a usually discrete stochastic process (as a random walk) in which the probabilities of occurrence of various future states depend only on the present state of the system or on the immediately preceding state and not on the path by which the present state was achieved —called also Markov chain.

Consider a discrete state space E and a random sequence $\{X(n), n = 0,1,2,...\}$, if any non-negative integers $n_1, n_2, \cdots n_m (0 \le n_1 < n_2 < \cdots < n_m)$ and natural number k, also with any sequence $i_1, i_2, \cdots i_m, j \in E$ which from discrete state space E can make below formula, then $\{X(n), n = 0,1,2,...\}$ is the Markov chain.

[+] *E-mail address*: Kanling_@163.com

$$P\{X(n_m+k)=j \mid X(n_1)=i_1, X(n_2)=i_2,\ldots,X(n_m)=i_m\} \\ = P\{X(n_m+k)=j \mid X(n_m)=i_m\} \quad (1)$$

In the formula (1), if n_m means the time of now, $n_1, n_2, \cdots n_{m-1}$ means the time of past, at time of n_m+k, status j only depends on time n_m's status, not depends on status of $n_1, n_2, \cdots n_{m-1}$,this feature called no after-effects of Markov chain.

Homogeneous Markov chain and k steps transition probability

$P\{X(n_m+k)=j \mid X(n_m)=i_m\}, k \geq 1$ Called k steps transition probability of Markov chain, marked as $p_{ij}(n, n+k)$. Transition probability means when time "n"'s status is "i", the probability of "j" which is k unit time after "n", if $p_{ij}(n, n+k)$ does not depend on Markov chain, which defined as homogeneous Markov chain. This condition only depends on transition start off status--i, transition steps--k and transition reached status j, nothing to do with n. Meanwhile k steps transition probability is marked as $p_{ij}(k)$, like below

$$p_{ij}(k) = p_{ij}(n, n+k) = P\{X(n+k)=j \mid X(n)=i\}, k>0 \quad (2)$$

In the formula $0 \leq p_{ij}(k) \leq 1, \sum_{j \in E} p_{ij}(k) = 1$.

2.2. The determination of multiple steps

Suppose k=1, $p_{ij}(1)$ called one step transition probability, also marked as p_{ij} for short. Matrix which contains all one step transition probability p_{ij} marked as P(1) means one step transition matrix at time m $(i, j \in E)$,as usual, we called it p for short. So all n steps transition probability is $p_{ij}(n)$.Matrix p(n) called n steps transition probability Markov chain, use C-K equation we can get the recurrence relations

$$P(n) = PP(n-1) = P(n-1)P \quad (3)$$

Then

$$P(n) = P^n \quad (4)$$

3. K Steps Transition Probability matrix And The Prediction

3.1.Build the k steps transition probability matrix

We successfully changed the model from complicated path prediction to fork-to-fork connection, each fork corresponds to the status of Markov chain. We suppose there are n forks which means the transition matrix is $n \times n$, p_{ij} means the probability of fork i connects to j in one step transition probability matrix $(1 \leq i, j \leq n)$ .obviously p_{ij} can be generated by statistics.

Take N_{ij} to express the times of fork i connects fork j from the vast real statistical data

$$p_{ij} = \frac{N_{ij}}{\sum_{j=1}^{n} N_{ij}} \quad (1 \leq i, j \leq n) \quad (5)$$

We can generate the one step transition probability matrix, according to the formula (4) we also can generate any k steps transition probability matrix.

$$P(k) = P^k \quad (6)$$

3.2.The basic theory and method of prediction

The prediction for choosing fork in the road based on the statistical of history information and trajectory. So the nearest previous fork which is chosen takes more affectivity on the prediction, meanwhile the early history is less import as we can ignore them. Then we can generate the Markov chain and the predict probability of each fork by weighting.

$$X(t) = a_1 S(t-1)P + a_2 S(t-2)P^2 + \ldots + a_k S(t-k)P^k \quad (7)$$

In the formula, t means the time of next fork, t-1 means the time of previous fork, the similar as others. X(t) means the predict probability from above formula calculates by weight,it is a $1 \times n$ matrix with the value of each element in it stands for the predict probability for the right fork to be the next. As S(i) , $t-k \le i \le t-1$ means the statuses of prev-i forks, it also is a matrix of $1\times n$,which the value of line one, row i is 1, the others are 0. It is a relative probability, totally added the value of all elements may over 1. $a_1, a_2, \cdots a_k$ is separately marked as how the impaction takes from the front 1,2,....,k forks to the next fork. These values can be gained from experience, firstly, we set $a_1 \ge a_2 \ge \cdots \ge a_k$. We also need to mark these forks as 0 which did not connect with the fork that is chosen just now, so we can the maximal element from X(t) and take the corresponding fork as the next fork.

3.3. Description of main arithmetic and analysis

Arithmetic 1: How to gain the matrix of 1 to k steps probability
Input:
$N = \{N_{ij}\}_{n\times n}$ // transition matrix of times
Output:
$M = \{N_{ijk}\}_{K\times n\times n}$ //1 to k steps transition matrix
//according to statistics the one step transition matrix P is generated
FOREACH $N_i \in N$
// cumulate is accumulate process
Sum=cumulate $N_{ij} \in N$
FOREACH $N_{ij} \in N_i$
$P_{ij} = N_{ij} / sum;$
$M_1 = P;$
// the matrix of 1 to k steps transition probability can be generated from formula (3)
FOR i=2 to k DO
//matrixMul saves the temp result of multiply matrixs
$M_i = matrixMul(M_{i-1}, P);$
ENDFOR
RETURN M;

The whole process need $2\times n\times n$ repeat times, after statistics we can generate matrix P from matrix N, complexity is $O(n\times n)$. There is no need to use formula (6) to gain 2 to k steps transition probability matrix from one steps transition probability matrix, just as formula (3) can be recursive and also gained 1 to k steps transition probability matrix.

Complexity of each multiply matrixs process is $O(n\times n\times n)$, so the complexity of one to k steps transition probability matrix is $O(K\times n\times n\times n)$.

Arithmetic 2 How to calculate prediction
Input:
$G = \{G_{ij}\}_{n\times n}$ //traffic net
$M = \{M_{ijk}\}_{K\times n\times n}$
$S = \{S_i\}_{1\times k}$
Now
$X = \{X_i\}_{1\times n}$
$A = \{A_i\}_{1\times K}$
Output:
Result //the most likely next fork
//clear X to 0 and calculate the prediction
setZero（ X ）;
FOR $i=1$ to k DO
// matrixMul saves the temp result of multiply matrixs
$Tepmatrix = matrixMul(S_i, M_i)$;
X=matrixAdd(X,Tepmatrix) ;

ENDFOR
// filter out the forks which could be reached through the information of G
FOR $i = 1$ to n DO
IF now , $i \notin G$ THEN
$X_i = 0$;
ENDIF
ENDFOR
//the fork with maximum value in X_i
result=selectMax($X_1, X_2, ..., X_n$) ;
RETURN result

There is need k-1 times of addition and multiplication to generated each prediction, as S_i is one-dimensional, so the time complexity of the whole process is $O(k \times n \times n + k \times n)$, it also needs n times to filter which complexity is $O(n)$, at last, the complexity of filter final result is $O(n)$, total complexity is $O(k \times n \times n)$.

4. Abnormal Trajectory Detection

Above trajectory prediction system based on history data and was generated by training MARKOV CHAINS, as we found when emergency happens locally, like traffic was been broken off by natural disasters, or getting busy by traffic accident, then we expect the system responds as the alert quickly. That is a predict system which can detect abnormal trajectory, it tracking cars and use exist MARKOV CHAINS model to predict and take it is normal if successfully predict which fork the car chosen, when predict failed the system will keep the record into failure list, at the same time marked the fork number that car chosen. Meanwhile the system calculates the times of failure timely, once the rate reaches to a threshold it will trigger a rectification use the records which were kept as abnormal trajectory happened.

4.1. Tracking car

Each fork of tracking car system which based on camera need to setup one to record the fork that car chosen and the characteristic of the car so that we can recognize it at next fork. From the exist methods like neural network, SVM, template matching we found at the same way many cars share the same characteristic, at this condition reorganization of characteristics may not accurate as vehicle detection. There are many stable and accurate methods of plate number identify, at each fork we set a camera to capture the plate number, then search with it at the background database for cars tracking.

4.2. Abnormal trajectory detection

For cars tracking, it is not only records the fork which the car taken and do the prediction, it also check the result from which the system can use to detect the abnormal trajectory.

Suppose that car V_i just passed by fork r_k, at r_k, it is need to identify the plate number and also checked out the history trajectory $r_1, r_2, ..., r_{k-1}$,then begin to train MARKOV CHAINS model by input these data, finally get the prediction result of fork r'_{k+1} that the car V_i probably will take.

When the car goes to the next fork r_{k+1}, the system do the exactly the same procedure as above, more it also checks the last prediction r'_{k+1}, If the result from current record is the same with the predication, shows predict success, then go on the next one r_{k+2}, otherwise means fail, at this condition it is need to record the real choice, and count the rate of fail, but it is no need to do the prediction. Once the rate of fail reached threshold, MARKOV CHAINS model needs a revise.

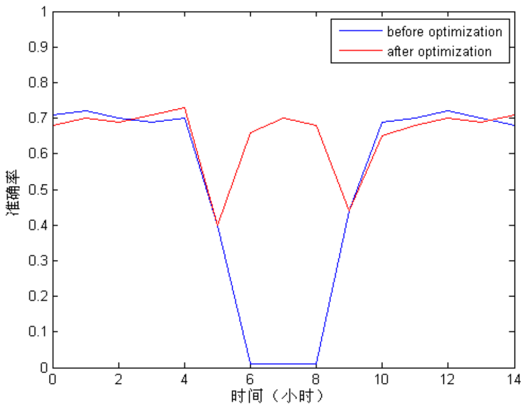
5. Experiment Result And Analysis

To verify above method's validity, we setup an experiment of path predict. In this experiment we statistics 10000 records of history trajectory with 5 forks to choose. Then calculated and saved one to k steps transition matrixes, after these preconditioning we used the MARKOV CHAINS model to do the prediction for the kth fork based on the 10000 history data. Form 1 shows the major parameters of the whole experiment.

Form 1: Parameters

Parameters	value	Meanings
N	5	The number of forks
K	3	Prediction of k steps history trajectory
$a_{1,\ldots,k}$	[4,1,0.25]	Weight array

This experiment takes monitor of cars at 5 forks, and identifies plate number also predicts moving trajectory, below shows the result:



Picture 1: Result of detect abnormal trajectory

Picture 1 shows that comparison of car tracking before and after the traffic accident, obviously after the time 4 when the accident happened, the predict accuracy of trajectory keeps falling down in those model who can not revise from abnormal trajectory. Since traffic police directed other cars to go by a roundabout route after the accident, between the time 6 and 8, models without the ability to detect abnormal trajectory failed to predict until at time 8, traffic became normal again. As our model which was been improved kept failing to predict at the begging just like the other models, but once the rate reached the threshold (it was been set as 0.4 in the experiment), it triggered a rectification of MARKOV CHAINS, at time 6 after the re-training, the predict system become effective. At time 8 the procedure taken as the same way until time 10 it recovered.

6. Summary

This paper present a predict trajectory method based on Markov chain, this model can predict the cars' trajectory effectively and can revise once there is abnormal behaviors, with experiment that proved this model has a higher accuracy and significance.

7. References

[1] N. Johnson and D. Hogg, "Learning the distribution of object trajectories for event recognition," Image and Vision Computing, vol. 14, no. 8, pp.609–615, 1996.

[2] PENG Qu,DING Zhi-ming,GUO Li-min.Prediction of Trajectory Based on Markov Chains. Computing Technology.vol,37.no.8.

[3] Weiming Hu, Dan Xie. A Hierarchical Self-Organizing Approach for Learning the Patterns of Motion Trajectories. IEEE TRANSACTIONS ON NEURAL NETWORKS, VOL. 15, NO. 1, JANUARY 2004.

[4] T. Collins, A. J. Lipton, and T. Kanade, "Introduction to the special sectionon video surveillance," IEEE Trans. Pattern Anal. Machine Intell.,vol. 22, pp. 745–746, 2000.

[5] R. J. Howarth and H. Buxton, "Conceptual descriptions from monitoring and watching image sequences," Image and Vision Computing, vol. 18, no. 9, pp. 105–135, 2000.

[6] R. J. Howarth and B. Hilary, "An analogical representation of space and time," Image and Vision Computing, vol. 10, no. 7, pp. 467–478, 1992.

[7] E. Andre, G. Herzog, and T. Rist, "On the simultaneous interpretation of real world image sequences and their natural language description: The System Soccer," in Proc. ECAI-88, Munich, 1988, pp. 449–454.

[8] K. Schaefer, M. Haag, W. Theilmann, and H. Nagel, "Integration of image sequence evaluation and fuzzy metric temporal logic programming," in KI-97: Advances in Artificial Intelligence, Lecture Notes in Computer Science, 1303, C. Habel, G. Brewka, and B. Nebel, Eds. New York: Springer, 1997, pp. 301–312.

[9] M. Brand and V. Kettnaker, "Discovery and segmentation of activities in video," IEEE Trans. Pattern Anal. Machine Intell., vol. 22, pp. 844–851,2000.