DRIVER BEHAVIOUR ANALYSIS TO RECONSTRUCT VEHICLE TRAJECTORY IN URBAN AREA

PHÂN TÍCH HÀNH VI TÀI XẾ ĐỂ TÁI TẠO ĐƯỜNG ĐI CỦA XE TRONG KHU VỰC ĐÔ THỊ

Nguyen Manh Hung[1], Ching-Chun Huang[2]
[1]Ho Chi Minh City University of Technology and Education
[2]National Chung Cheng University- Chia-Yi County, Taiwan

Received 17/5/2016, Peer reviewed 14/6/2016, Accepted for publication 2/8/2016

ABSTRACT

Recently, many applications in video surveillance, intelligent traffic system and social security management require information of vehicle trajectories in an urban area. Conventional methods tried to track moving vehicles by using either appearance matching or spatial and temporal information to estimate vehicle trajectory. However, the authors have recognized a phenomenon that vehicles have tendencies to follow some main roads. Therefore, the trajectory could be reconstructed based on a small observation-set. By using a training process, we analyze driver behavior to find out main roads of an urban area. Besides, the authors propose a new idea to predict the vehicle trajectory based on these learned main roads. Based on an observed location, these main roads that the vehicle could go through have been estimated. The full trajectory is would be fusion process of these roads. Experiments prove our hypothesis about the existence of key roads that citizens prefer to use and proposed method could improve reconstruction performance.

Keywords: *User behavior analysis; Tendency learning; Non-overlapping camera network; Recommendation system; Trajectory prediction.*

TÓM TẮT

Ngày nay, rất nhiều ứng dụng đòi hỏi thông tin về đường đi của các phương tiện giao thông trong khu vưc đô thị như giám sát an ninh, hệ thống giao thông thông minh, quản lý trật tự xã hội. Các nghiên cứu trước đây về giám sát đường đi của phương tiện giao thông chủ yếu dựa vào việc tìm kiếm sự tương đồng giữa một mô hình mẫu với thông tin từ máy quay giám sát; hoặc dựa vào các thông tin về không gian, thời gian để ước lượng đường đi của phương tiện. Mặc dù vậy, tác giả nhận ra một hiện tượng là các phương tiện giao thông có khuynh hướng di chuyển theo một vài trục đường chính. Nhờ thế, đường đi của phương tiện đó có thể đượctái tạo nếu ta biết được một vài vị trí trên đường di chuyển. Bằng cách sử dụng một quá trình huấn luyện, tác giả đã phân tích dữ liệu người dùng để xác định các cung đường chính trong đô thị. Bên cạnh đó, tác giả đề xuất một phương pháp mới để tái tạo lại đường đi của một phương tiện giao thông dựa vào các cung đường chính đã được học. Dựa vào một sốrất ít vị trí trên quãng đường di chuyển, tác giả đã ước lượng được các cung đường chính mà tài xế đã đi qua. Quỹ đạo đường đi của phương tiện sẽ được suy diễn như là sự kết hợp giữa các cung đường chính mà tài xế đã đi qua. Kết quả thí nghiệm khẳng định giả thiết của tác giả về việc tồn tại những trục đường chính mà người dân có khuynh hướng sử *dụng và phương pháp đề nghị có khả năng nâng cao khả năng tái tạo lại đường đi của phương tiện.*

Từ khóa: *Phân tích hành vi người dùng; học tập khuynh hướng; hệ thống camera không trùng lắp; hệ thống đề nghị; dự báo đường đi.*

1. INTRODUCTION

A lot of intelligent applications such as video surveillance, traffic monitoring, social security management, public transportation and military rely on non-overlap camera network to estimate vehicle trajectory. In general, trajectory prediction could help to well estimate traffic flow or to know where the vehicle visited. That information is important in traffic management and criminal investigation. Many previous methods [1-3] were proposed to track the moving vehicles. Those methods mainly based on the measurement of space-time relationships and appearance similarity to recover the vehicle trajectory. However, the object appearance similarity may be corrupted due to dynamical illumination changes or inter-object occlusion. On the other hand, the transition time across cameras is irregular because of traffic condition and traffic light. Those factors may increase matching uncertainty and reduce system performance.

Recently, to overcome above challenges of a vision based system, the prediction has been modeled as a recommendation system [4]. Here, vehicle trajectories are modeled by a vector of cameras in an urban area. When a vehicle appears at a camera, the corresponding value is set to one and otherwise. If the information is not available, the value is empty. A recommendation system could help to predict empty value so that a full trajectory could be reconstructed. While avoid challenge of vision based system, the accuracy of the approach relies on the applied recommendation algorithm. Moreover, conventional recommendation algorithms do not explore raw data; therefore they work as black-boxes to reconstruct a trajectory. Hence, in this paper, we aim to propose a novel method that explore the raw data as well as introduce a method to infer vehicle trajectory in an efficient manner. Contributions of the paper are listed below:

- A proposed method to learn latent roads in an urban area. By analyzing user behavior, we find out there are main roads that users prefer to use. The learned process could help officer pay more attentions to avoid traffic jam on these roads.
- Moreover, an inference method to estimate vehicle trajectory base on a very few observations are also proposed. By using the learned road segment, the authors can infer full trajectory by fusing all appearing segment on the map. The properties provide a clear viewpoint about how the algorithm can work and improve performance significantly compare with based-line methods.

2. PROBLEM MODELING

In this paper, we assume that a vehicle is partially observed by some cameras in the public multi-camera surveillance networks. The proposed system aims to predict the full vehicle trajectory based on the partial camera observations. One example of input data is present in Fig.1. Here, the numbers present camera-IDs on a network and the green circles present the cameras where target is observed. In particular application, license plates could be used to identify a target vehicle. Due to occlusion effect and illumination effect, the license plate could not be detected at all camera nodes. Therefore, it is assumed that the plate is detected only at some camera nodes. Our method will predict a full trajectory as in Fig.2. The red line connects all cameras on the trajectory.

Fig.1 *Particle observation of input data.*

Fig.2 *Out put: full trajectory.*

3. THE PROPOSED METHOD

In this section, let us discuss how labeled data for training process were collected. Based on the data, the authors introduced a method to learn main roads as well as inferred a full trajectory based on the learnt road.

3.1 Data collection

In order to collect vehicle trajectories for training, we create a web interface, as shown in Fig. 3, which helps one to select the preferred trajectory from one place to the other place. First, a start point and a stop point are randomly selected in the map region of interest. Our system then automatically generates an initial trajectory connecting the start point and the stop point and shows the trajectory in Google Maps. Finally, users are required to modify the default trajectory to meet their personal preference. In our data collection process, we select a region in Kaohsiung city as our testing environment. Many users who are familiar with the environment are invited to collect the data.

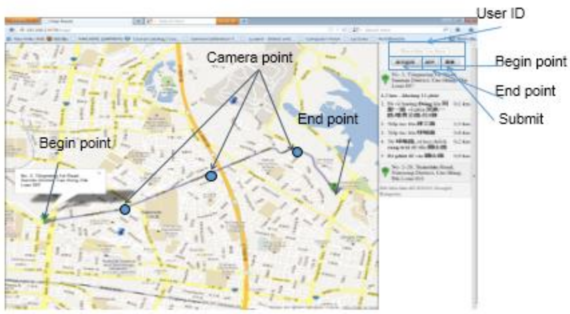

Fig.3 *Our Web interface used for data collection.*

Next, we define the available cameras inside our experimental region in Kaohsiung city. In our system, the location of each camera is recorded by its longitude and latitude. This information could be extracted from the Google MAP API. Now, a trajectory is presented as a group of identify numbers. The example of labeled data D is shown as in Table.1. Here the element Dij is set to one if the ith camera appears on the jth trajectory. Otherwise, if a camera is not appeared on a trajectory, the corresponding element is set to zero. By using the modeling method, each trajectory is presented by a n-by-1 vector in each column of the matrix D. Totally, we have collected 239 trajectories which include 627 cameras in our experimental region.

3.2 Learning process

To learn the main trajectories; we based on two assumptions. The first assumption is that a trajectory is a combination of several sort-main trajectories. It is reasonable because there are some key traffic nodes that help to move faster from a region to another region. The key traffic node will define main road segment. The second assumption is that if a segment is a main road, it should be used by many users. Therefore, these cameras of the segment had a high appearance frequency. Supported by two assumptions, we can learn the main road by factorize the labeled data in section 3.1 into two matrixes as in Fig.4. The first matrix presents the main road in each column whereas the second matrix present the contribution of each main roads in the weight. An intuition is showed in Fig.4; here a full trajectory is presented by some cameras in a column in D matrix. The full trajectory is a combination of three main trajectories. Our method aims to learn a M matrix that present main road in each column. Hence to present the full trajectory we only need to use three main road in M matrix. The corresponding in W matrix introduce which main road that used to create the full trajectory. Therefore, we can treat each element on D matrix is the property that a camera appeared on a trajectory, and an element on W matrix is the property that a camera appeared on each sort-main trajectory.

Table 1. *Labeled trajectory on the road*

	T1	T2	T3	T4			Tm
C1	1	0	1	0			0
C2	1	1	0	0			1
C3	0	1	1	1			0
C4	0	1	0	1			1
Cn	1	0	0	1			1

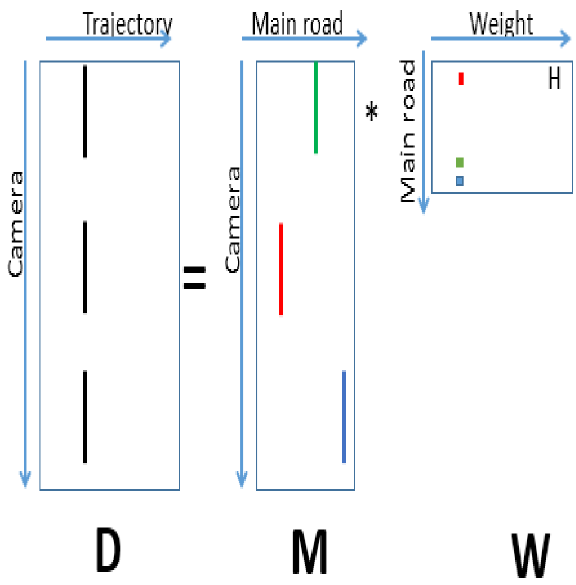

Fig.4 *Matrix factorization of labeled data*

To learn M matrix and W matrix, we need to optimize the energy function (1):

$$(\hat{M},\hat{W}) = \underset{(M,W)}{\operatorname{argmin}} \|D - M * W\|^2$$

st M,W>0 (1)

Here, the term $\|D - M * W\|^2$ is data error term; it is hoped that the learnt trajectory could fit well with labeled data. Moreover, the authors introduce Non-Negative Constraint to force each column of M is a part of a full trajectory. The concept is supported by result in [5] where non negative constraint can help to learn parts of face from a face dataset. In addition, we expect the weight of a main road is zeros if the segment does not have any connection to a trajectory. Therefore, the non-negative constraint in W matrix is introduced. The method in [6] is used to solved the problem in (1). The optimal solution is an integration process that presented in (2-3).

$$W_{ij} = W_{ij}\frac{(M^T D)_{ij}}{(M^T M W)_{ij}} \quad (2)$$

$$M_{ij} = M_{ij}\frac{(D H^T)_{ij}}{(M W W^T)_{ij}} \quad (3)$$

Under update rule in (2-3), the Euclidean distance $||D - M * W||$ is non-increase. Therefore, the solution could converge after an integration process.

3.3 Inference method

Given some partial observations at some camera node. If the vehicle appears at a camera, we name the camera is an appearing camera and denote as '1'. Otherwise, if the vehicle does not appear at a camera, we name the camera is non-appearing camera and denote as '0'.Other cameras on map are not observed, hence we call they are unknown cameras denoted by 'U'.

To infer status of these unknown camera, we proposed a hierarchical framework as presented in Fig.5. Here, we introduce a bottom-to-up and a top-to-down processes in a unified framework. For the bottom-to-up process, based on the main trajectory and given observations, we estimate a confident vector which present the ability that a camera is an appearing camera given by some observations. Consequently, each unknown camera is presented by a value. If the value is high, we predict the camera is an appearing camera. If the value is small, we predict the camera is a non-appearing camera. Others should be predicted as unsure cameras; this means we do not confident about their statuses. Then, for top-to-down process, we compare the confident vector with several hypothesis solutions. Which hypothesis could provide a small reconstruction error could be select as a best suitable solution. The method is based on an idea that, if a hypothesis is a real trajectory, the hypothesis is well reconstructed from training data. Otherwise, if a hypothesis is not a trajectory, the hypothesis will have a reconstruction error. Follow the method, we recommend top N candidates which have high abilities to be a ground true.

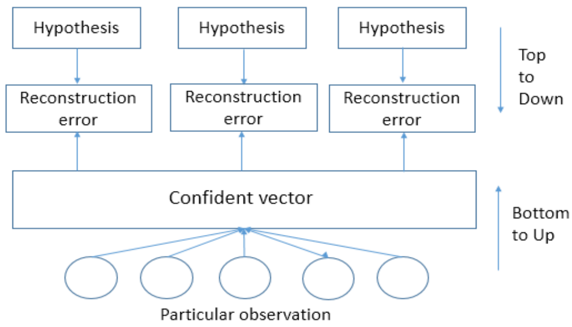

Fig.5 *System flow of the proposed method*

Section 3.3.1 introduces confident vector and section 3.3.2 discusses about hypothesis method.

3.3.1 Confident vector

By factorizing matrix D(m*n) into two matrixes, M(m*k) and W(k*n), we have k main trajectories which are columns of M matrix. Each M_{ij} element presents ability that i^{th} camera appears on the main trajectory j. We denote:

- X: a confident vector, X evaluates confidences of cameras could appear on a trajectory.
- X_i: a confident vector given by an observed i^{th} camera, Xi evaluates confidences of cameras could appear on a ground true if the i^{th} camera appears on the testing sample.
- X_{ij}: evaluate ability that cameras on main trajectory j appear on a ground true if the i^{th} camera appears on a testing sample.
- M(:,j): the j^{th} column of M, presents ability that cameras appear on main trajectory j.
- Sj: the set of appearing camera on the j^{th} main- trajectory

First, given one particle observation at the i^{th} camera and the j^{th} main trajectory, we compute the probability that cameras on the j^{th} trajectory could be an appearing camera. The value could be score in a vector X^{ij} as:

$$\begin{bmatrix} X_i^{ij} = 1 \\ X_{k\in S_j}^{ij} = M_{kj} \\ X_{k\ni S_j}^{ij} = 0 \end{bmatrix} \quad (4)$$

Follow the equation (4), where a camera is an appearing camera, we set its probability to be an appearing camera is one. Each camera on the trajectory could receive the probability in M matrix; others camera is set to zero mean that this camera does not have a change to appear on the trajectory.

The probability that cameras be an appearing camera given by all learnt trajectory is defined as equation (5); and the confident vector X is defined as (6).

$$X_i = \sum_{j\in \text{main trjectory set}} X^{ij} \quad (5)$$

$$X = \sum_{i\in \text{appearing camera set}} X_i \quad (6)$$

After estimate confident vector X, we identify all cameras on the map as three classes include appearing, not appearing, and unsure cameras. Figure (6) presents one example these classes. Here, 3rd, 9th, 11st, 12nd, 16th, 21st, 22nd, 25th cameras are not observed cameras. The 3rdcamera has a high confident value; hence we predict it is an appearing camera. The 9th, 11th and 12th cameras have moderate confident values, hence we predict they are unsure cameras. The 16th, 21st, 22nd, and 25th cameras have small confident values, hence we predict they are non-appearing cameras.

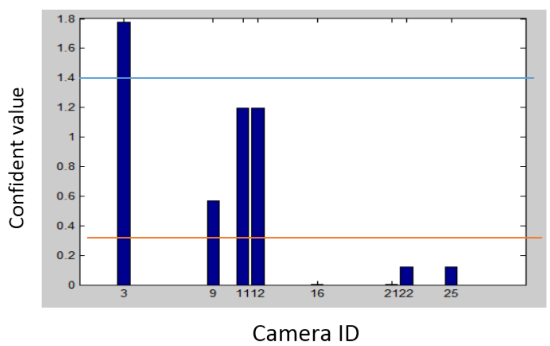

Fig.6 *One example of sub confident vector*

This vector shows confidences of not observed cameras.

The number of appearing cameras in each testing sample is different. Therefore, we should not set up a threshold to classify unknown cameras into the three groups. Hence, we suggest use k-means method [7] to automatically predict statuses of unknown cameras.

3.3.2 Hypothesis method

For each unsure camera, we have two available options. Therefore, for each n_1 unsure cameras, we can generate 2^{n_1} options. Each option is called a hypothesis. We reconstruct all hypotheses by using training data follow a process in Fig.7. An accuracy of the reconstruction is applied to predict what hypothesis could be aground truth trajectory. If the reconstruction error is small enough, it means the corresponding hypothesis has a high chance to be a truth trajectory.

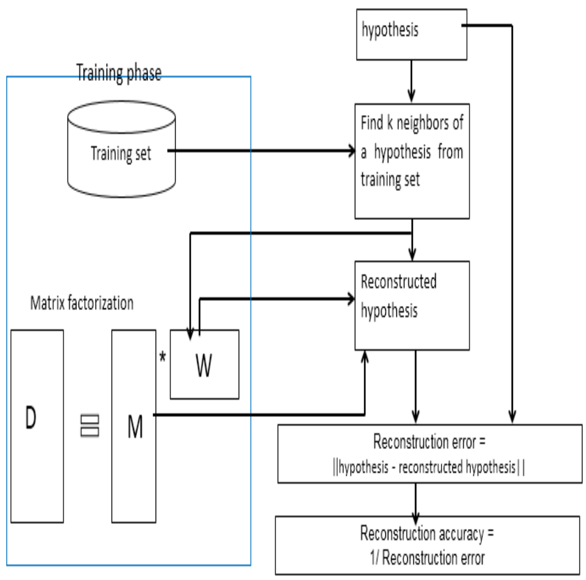

Fig.7 *Reconstruction process and reconstruction accuracy.*

First, we select k neighbors of a hypothesis by compute similarities between the hypothesis and all trajectories in training set as equations (7)

$$\mathrm{sim_i} = \frac{\mathrm{sum}(\mathrm{u_a}=\mathrm{D}(:,\mathrm{i}))}{\mathrm{m}} \qquad (7)$$

Here, u_a is a hypothesis, D(:,i) is the i^{th} column of D matrix which represents a training sample, m is the number of cameras on a map, sim_i is the similarity between the hypothesis and the training i^{th} trajectory. Then we compute the reconstructed hypothesis given by the k^{th} neighbor as in (8)

$$R_k = M * W(:,k) \qquad (8)$$

Here, $W(:,k)$ is the column k^{th} of the W matrix. The reconstructed hypothesis for a given u_a is a linear combination of each R_k as in (9).

$$R_a = \frac{\sum_{k\in neighbot\ set} sim_k * R_k}{\sum_{k\in neighbot\ set} sim_k} \qquad (9)$$

Reconstruction error of u_a is calculated as equation (10) and the corresponding accuracy is calculated as equation (11).

$$\mathrm{error_a} = \|\mathrm{u_a} - \mathrm{R}\| \qquad (10)$$

$$\mathrm{acc_a} = \frac{1}{\mathrm{error_a}} \qquad (11)$$

The selected hypothesis is based on the rule in (12).

$$\hat{R} = \underset{a}{\mathrm{argmin}}\, \mathrm{error_a} \qquad (12)$$

4. MAIN-ROAD LEARNING EVALUATION

To prove our idea about the existence of main road on a map, we analysis values in each column. If a camera has a high value on a column, the camera is identified as a camera on a main trajectory. We show the high confident cameras in each column on a map in Fig.8a and Fig8.b; the result shows that these cameras create a short trajectory on a road. Moreover, we show all main roads on the map in Fig.8c. Each color presents a main road.

a

b

c

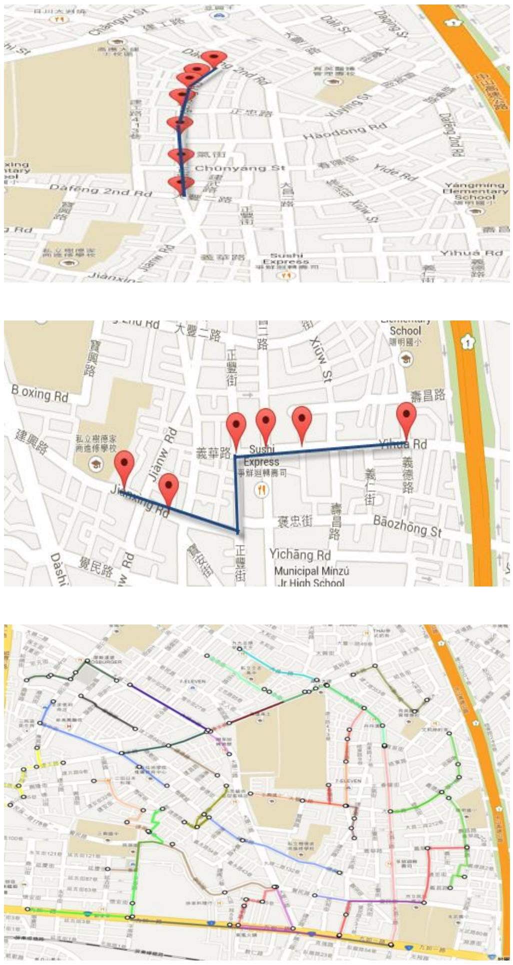

Fig.8 *Some main roads on a map*

5. RECONSTRUCTION EVALUATION

To evaluate our system, we compare our proposed method with the item based methods, user based method and latent based method as in [4]. For each comparison method, we try several parameters to select the best result, then we compare the result to our proposed method. This process prove our method is outstanding than the previous works in almost setting option.

To create the testing set, we randomly select 20 trajectories as the testing set and the other 219 trajectories as the training set. This experiment process is repeated ten times to make our evaluation trustable. We introduce evaluation metric in section 5.1 and detail of experimental result in section 5.2.

5.1 Performance Metric for Evaluation

Since the trajectory is represented as a camera status vector, which is a Boolean vector, we decided to use a confusion matrix described in Fig.9 to evaluate the performance. If a camera is predicted to appear in a reconstruction trajectory, and the camera also truly appears in the ground true, it is named as a true positive (TP). However, if the camera does not appear in the ground true then it is said to be a false positive (FP). Conversely, a true negative (TN) has occurred when a camera does not appear in both the reconstruction trajectory and the ground true. Moreover, a false negative (FN) indicates when a camera does not appear on the reconstruction trajectory while appears in the ground true.

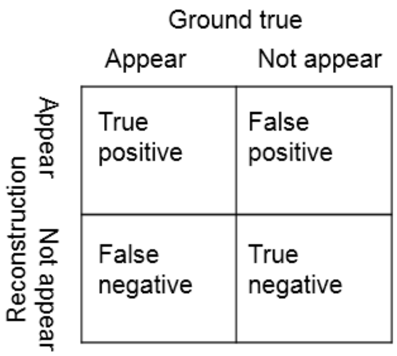

Fig.9.*Confusion matrix*.

The overlap between an output and ground true is described by a recall rate calculated by equation (13)

$$\text{recall} = \frac{\text{TP}}{\text{TP+FN}} \qquad (13)$$

Even if recall rate can measure the overlap between an output and ground true trajectory, it can't prove that the output is a good result. For instance, in the situation that many cameras are predicted as appearing cameras, the recall rate is very high. However, the prediction output may still far away from the ground true. Hence, the precision rate is also considered to measure the expandable ability of an output. The precision rate is calculated by equation (14)

$$\text{precision} = \frac{\text{TP}}{\text{TP+FP}} \qquad (14)$$

To find an optimal trade-off between precision and recall, a single-valued measure like the E-measure can be used.

$$\text{E}_{\text{measure}} = \frac{1}{\alpha\frac{1}{\text{precision}}+(1-\alpha)\frac{1}{\text{recall}}} \qquad (15)$$

In (21), parameter α controls the trade-off between precision rate and recall rate. In our paper, we assume that precision rate and recall rate are equivalent important. Hence the parameter α is set to 0.5, and E_measure become F_measure as equation (16).

$$\text{F\_measure} = \frac{2\ \text{precision.recall}}{\text{precision} + \text{recall}} \qquad (16)$$

5.2 Experimental Results

To evaluate the system performance, we tested the system in these situations that only 20%, 40%, or 60% of the camera statuses are observed. To be fair, the testing trajectory should be carefully generated. For instance, if we would like to test the 20% case, we generated a trajectory by removing 80% data with "1" camera statuses and removing 80% data with "0" camera statuses separately. Result for each experiment is presented in Fig.10, Fig.11, Fig.12 respectively.

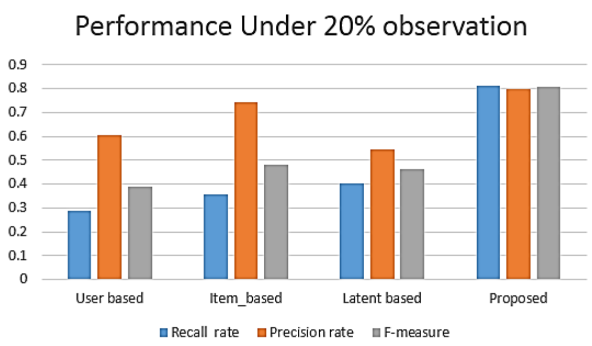

Fig.10 *Recall rate, precision rate, and F_measurement under 20% observation*

For the first experiment; since we rely on 20% camera on map, we expect the prediction should be a challenging task. Hence, our goal is not to build a system with high accuracy but to discover other helpful information that could be the complement of other previous methods. To discuss the experimental results, we firstly check the Recall rate. If the value is higher than the information we are given, it means the algorithm could provide positive help. Next, we check the Precision rate. Higher precision rate means the algorithm could give better recommendation. F-measure, it provides a measurement of overall performance. The experiments in Fig. 10, 11, 12 prove the outstanding of our method compare with previous work. The reason of the good performance is because our method can explore the structure of raw data; and used the structure efficiently.

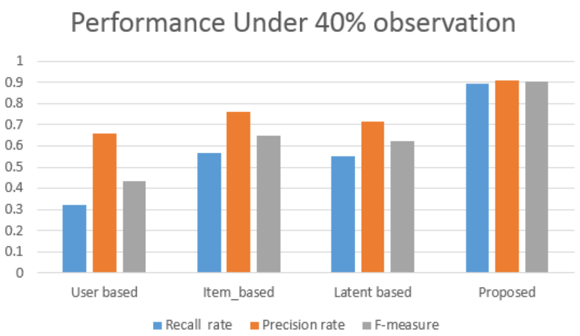

Fig.11*Recall rate, precision rate, and F_measurement under 40% observation*

The F measure of the proposed method is 0.8054, 0.9025, 0.9154 for 20%, 40%, 60% observed data respectively. Therefore, the proposed method can work well with a few (20% observed data). In addition, even 60% data is observed, the performance is only increase slightly compare with 40% data is observed. Therefore, we could rely on a smaller data to have a fair performance.

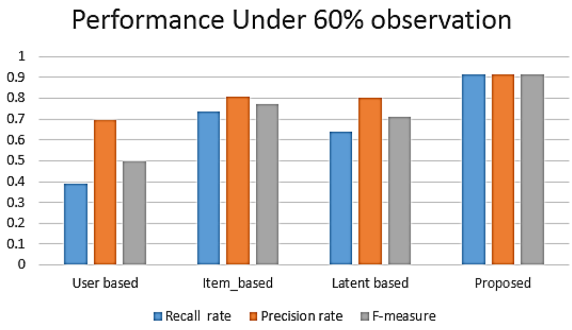

Fig.12 *Recall rate, precision rate, and F_mesurement under 60% observation*

Moreover, in all experiments, our proposed method provided approximate recall rate, precession rate, and F_measure score whereas the other method does not provide a recall rate as good as precision rate. It means that our proposed method does not provide too many positive samples or too many negative samples and work robustly.

6 CONCLUSION

In this paper, the authors proposed a method to investigate the main trajectory of an urban area and a prediction framework to estimate vehicular trajectory. By factorizing a matrix with non-negative constraint, main trajectories as columns of a component matrix have been learnt. By using the main trajectory, one can boost the information from a few observations to active a higher accuracy.

REFERENCES

[1] Omar Javed, Khurram Shafique, Zeeshan Rasheed, and Mubarak Shah, *Modeling Inter-Camera Space–Time and Appearance Relationshipsfor Tracking across Non-Overlapping Views*, Computer Vision and Image Understanding, pp. 146-162 , 2011.

[2] Guoyun Lian, Jianhuang Lai, and Wei-Shi Zheng, *Spatial–Temporal Consistent Labeling of Tracked Pedestrians across Non-Overlapping Camera Views*, Pattern Recognization, pp. 1121-1136 , 2011.

[3] Youlu Wang, Senem Velipasalar, and Mustafa Cenk Gursoy, *Wide-Area Multi-Object Tracking with Non-Overlapping Camera Views*, in International Conference on Multimedia and Expo, pp. 1-6, 2011.

[4] Ching-Chun Huang, Hung-Nguyen Manh, and Tai-Hwei Hwang, *Vehicle Trajectory Prediction across Non-overlapping Camera Networks*, International Conference onConnected Vehicles and Expo (ICCVE), pp:375 - 380, 2013.

[5] Moody Chu, Robert Plemmons, *Nonnegative Matrix Factorization and Applications*, International Linear Algebra Society, pp. 2-7, 2005.

[6] Hyunsoo Kim, and Haesun Park, *Non Negative Matrix Mactorization based on Alternating Nonnegativity Constrained Least Squares and Active Set Method*, Society Industrial and Applied Mathematics Journal on Matrix Analysis and Applications, pp. 713-730, 2008

[7] Christopher M. Bishop, *Pattern Recognition and Machine Learning*, Springer, ISBN:0387310738, 2006..

Corresponding author:

Nguyen Manh Hung

Ho Chi Minh City University of Technology and Education

Email: hungnm@hcmute.edu.vn