

Experimental Validation for CRFNFP Algorithm

Wang Mingjun, Yi Xinhua, Wang Xuefeng
School of mechanical engineering
Ningbo University of technology
Ningbo, China

Tu Jun
School of mechanical engineering
Shanghai Jiaotong University
Shanghai, China

Abstract—In 2010, we proposed CRFNFP[1] algorithm to enhance long-range terrain perception for outdoor robots through the integration of both appearance features and spatial contexts. And our preliminary simulation results indicated the superiority of CRFNFP over other existing approaches in terms of accuracy, robustness and adaptability to dynamic unstructured outdoor environments. In this paper, we further study on the comparison experiments for navigation behaviors of robotic systems with different scene perception algorithms in real outdoor scenes. We implemented 3 robotic systems and repeated the running jobs under various conditions. We also defined 3 criterion to facilitate comparison for all systems: Obstacle Response Distance (ORD), Time to Finish Job (TFJ), and Distance of the Whole Run (DWR). The comparative experiments indicate that, the CRFNFP-based navigating system outperforms traditional local-map-based navigating systems in terms of all criteria. And the results also show that the CRFNFP algorithm does enhance the long-range perception for mobile robots and helps planning more efficient paths for the navigation.

Keywords- autonomous navigation; stereo vision; machine learning; conditional random fields; scene analysis.

I. INTRODUCTION

Navigation in an unknown and unstructured outdoor environment is a fundamental and challenging problem for autonomous mobile robots. The navigation task requires identifying safe, traversable paths that allow the robot to progress toward a goal while avoiding obstacles.

Standard approaches to complete the task use ranging sensors such as stereo vision or radar to recover the 3-D shape of the terrain. Various features of the terrain such as slopes or discontinuities are then analyzed to determine traversable regions [2-5]. However, ranging sensors such as stereo visions only supply short-range perception and gives reliable obstacle detection to a range of approximately 5m[6]. Navigating solely on short-range perception can lead to incorrect classification of safe and unsafe terrain in the far field, inefficient path following or even the failure of an experiment due to nearsightedness [7, 8].

To address nearsighted navigational errors, near-to-far-learning-based, long-range perception approaches are developed, which collect both appearances and stereo information from the near field as inputs for training appearance-based models and then applies these models in the far field in order to predict safe terrain and obstacles farther out from the robot where stereo readings are unavailable [9-11]. We restrict our discussion to the online self-supervised learning

since the diversity of the terrain and the lighting conditions of outdoor environments make it infeasible to employ a database of obstacle templates or features, or other forms of predefined description collections. The winner of DARPA Grand Challenge[10] combines sensor information from a laser range finder and a pose estimation system to first identify a nearby patch (a set of neighboring pixels) of drivable surface. And then the vision system takes this patch and uses it to construct appearance models to find the drivable surface outward into the far range. Happold and Ollis[9] propose a method for classifying the traversability of terrain by combining unsupervised learning of color models that predict scene geometry with supervised learning of the relationship between geometric features and the traversability. A neural network is trained offline on hand-labeled geometric features computed from the stereo data. An online process learns the association between color and geometry, enabling the robot to assess the traversability of regions for which there is little range information by estimating the geometry from the color of the scene and passing this to the neural network. The system of Bajracharya[11] consists of two learning algorithms: a short-range, geometry-based local terrain classifier that learns from very few proprioceptive examples; and a long-range, image-based classifier that learns from geometry-based classification and continuously generalizes geometry to the appearance.

Appearance-based near-to-far learning methods mentioned above do support the long-range perception which provides the “look-ahead” capability for complementing the traditional short-range stereo- or LIDAR-based sensing. However, appearance-based methods assume that the near-field mapping from the appearance to traversability is the same as the far-field mapping. Such an assumption does not necessarily hold due to the complex terrain geometry and varying lighting conditions in unstructured outdoor environment. Therefore, how to use other strategies or information to compensate for the mapping deviation begins to draw more attention.

Lookingbill and Lieb[12] use a reverse optical flow technique to trace back the current road appearance to how it appeared in previous image frames in order to extract road templates at various distances. The templates can be then matched with distant possible road regions in the imagery. However, trackable features, on which the reverse flow technique is based, are subject to the image saturation and scene elements occurrence patterns. Furthermore, changing illuminant conditions can result in unacceptable rates of misclassification. Noting that the visual size of features scales inversely with the distance from camera, Hadsell and

Sermanet[13] normalize the image by constructing a horizon-leveled input pyramid in which similar obstacles have similar heights, regardless of their distances from the camera. However, the distance estimation for different regions of images introduces extra uncertainties. In addition, this approach does not consider the influence of changing lighting conditions on appearances. Procopio[8] proposes the use of classifier ensembles to learn and store terrain models over time for the application to future terrain. These ensembles are validated and constructed dynamically from a model library that is maintained as the robot navigates terrain toward some goal. The outputs of the models in the resulting ensemble are combined dynamically and in real time. The main contribution of the ensembles approach is to leverage robots' past experience for classification of the current scene. However, since the validation of models is based on the stereo readings from the current scene, this approach is still subject to the mapping deviation.

In 2010, we proposed the model of CRFNFP[1] to incorporate both the spatial contexts and appearance information to enhance the perception robustness and self-adaptability to changing illuminant conditions. And simulation results indicated the superiority of CRFNFP over other existing approaches. In this paper, we further implement the CRFNFP model in a robotic system to study on the navigating behaviors in real scenes.

An outline of this paper is as follows: We first briefly describe the CRFNFP framework in section II. The system implementation of the robot will be detailed in section III and section IV provides the experiment results. We conclude our paper in section V with our further research in this area. And section VI indicates our future research.

II. CRFNFP FRAMEWORK

A. Model Summary

CRFNFP framework is a near-to-far learning strategy to recognize the far-field of the current scene. We first over segment the current scene into superpixels (a superpixel is a set of neighboring pixels) and update the classification database using training samples from stereo readings of near-field of the current scene. Then we incorporate both local appearance of and spatial relationships (contexts) between regions in the CRFNFP framework to estimate the traversability of regions of the current scene.

The problem to be solved by CRFNFP is how to design a specific CRF framework with respect to the self-supervised, near-to-far learning in unstructured outdoor environments. To the best of our knowledge, ours is the first work that introduces and adapts the CRF-based framework to model the spatial contexts and to improve the long-range perception for mobile robots.

B. Model Definition

Let the observed data (local appearance) from an input image be given by $\mathbf{X} = \{\mathbf{x}_i\}_{i \in S}$, where S is the set of sites (one site corresponds to one superpixel in our application) and \mathbf{x}_i is the data from the i th site. The corresponding labels at the

image sites, which indicate the category of the traversability of a region, are given by $\mathbf{L} = \{l_i\}_{i \in S}$. In this work, we will be only concerned with binary classification, i.e., $l_i \in \{-1, 1\}$, -1 for ground and 1 for obstacle.

Our CRFNFP model is based on the Conditional Random Fields (CRF) model, so we first explain the CRF model.

CRF Definition: Let $G = (S, E)$ be a graph such that \mathbf{L} is indexed by the vertices of G . Then (\mathbf{L}, \mathbf{X}) is said to be a conditional random field if, when conditioned on \mathbf{X} , the random variables l_i obey the Markov property with respect to the graph: $P(l_i | \mathbf{X}, \mathbf{L}_{S-\{i\}}) = P(l_i | \mathbf{X}, \mathbf{L}_{N_i})$, where $S-\{i\}$ is the set of all nodes in the graph except the node i , N_i is the set of neighbors of the node i in G .

Given the observation \mathbf{X} , the CRFNFP defines the joint distribution over the labels \mathbf{L} as

$$P(\mathbf{L} | \mathbf{X}) = \frac{1}{Z} \exp \left\{ \sum_{i \in S} A_i(l_i, \mathbf{X}) + \sum_{i \in S} \sum_{j \in N_i} I_{ij}(l_i, l_j, \mathbf{X}) \right\} \quad (1)$$

Where Z is a normalizing constant known as partition function, and A_i and I_{ij} are the association and interaction potentials respectively.

The association potential is constructed by a Bayes Classifier, which directly maps appearance to traversability. And the interaction potential aims to incorporate spatial relationships and serves as the data-dependent smoothing function.

As a result, the CRFNFP framework not only includes appearance features as its prediction basis, but also incorporates spatial relationships between terrain regions in a principled way. Please refer to the reference[1] for the details of CRFNFP framework.

III. IMPLEMENTATION OF ROBOTIC SYSTEM

A. Summary of Hardware Components

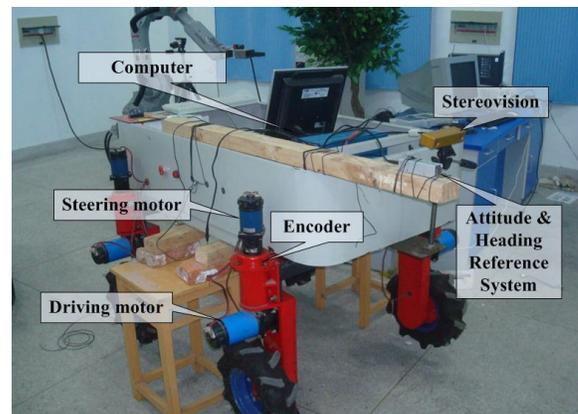


Figure 1. UGV used for navigation experiment.

The UGV is a four-wheeled, 8 DOF mobile robot with each wheel individually driven and steered to obtain the desired maneuverability. And the hardware of the UGV (as shown in Figure 1) mainly consist of vehicle body, an industrial personal computer (IPC), stereovision, AHRS (attitude and heading reference system), GPS (global positioning system). The hardware block diagram in Figure 2 shows the connection relationships among all components.

The GPS offers the global position while the AHRS combined with all the encoder provides the local position of the UGV. The stereo vision continuously takes picture of the current scenes, which is transmitted to the IPC. And the IPC will process all the information and provide the optimal control decision to further drive the UGV.

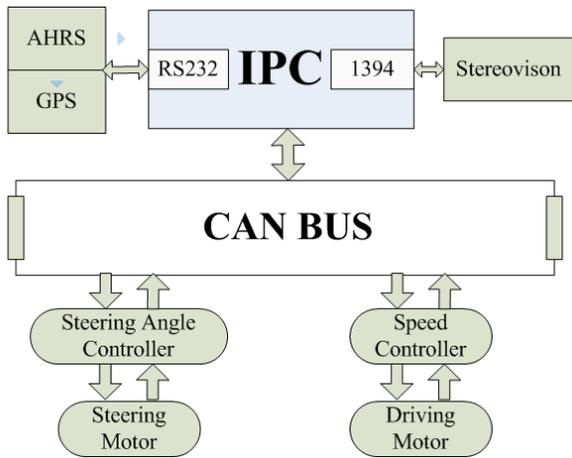


Figure 2. Hardware block diagram of UGV.

B. Flowchart Of Navigation Algorithm

The navigation job can be summarized as follows: given the target point, the robot goes from the start point to the endpoint while intelligently avoiding all the obstacles by taking corresponding actions.

The flowchart of Figure 3 shows that, the robot takes actions based on 3 sources of information:

- 1) *Near-field local mapping*: This mapping can model the local environment around the robot and provide the guidance for the robot to avoid close-range obstacles such as obstacles within 5 meters.
- 2) *Far-field path planning*: The inference results of far-field scenes can be used to generate the cost image, which represents the distant-range obstacle distribution (even obstacles up to 100 meters away). So the far-field path planning can lead the robot to avoid the distant obstacles ahead of time. And the corresponding trajectory can be shorter and smoother while the robot reaches the same target point.
- 3) *Directional deviation computation*: The directional deviation is defined as the angle between the target point direction and the forward direction of the UGV. The robot needs to approach the target point while minimizing the directional deviation as much as possible.

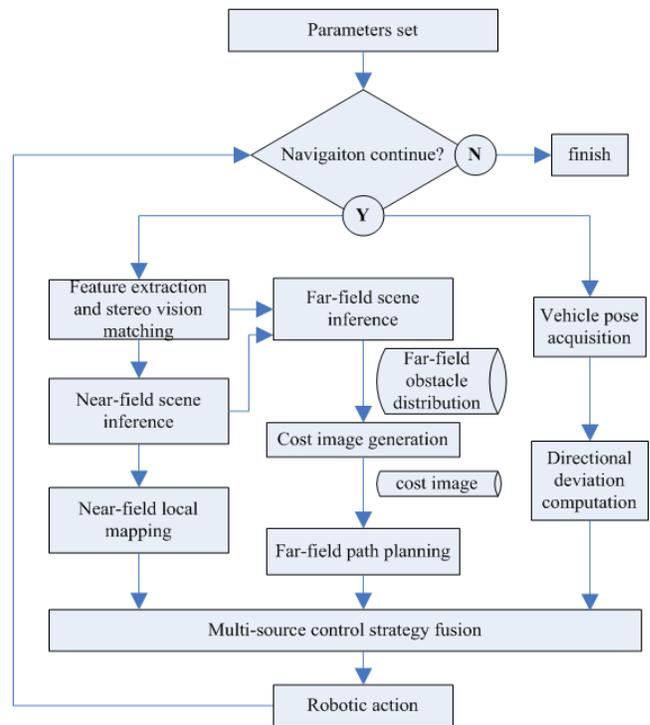


Figure 3. Flowchart of main algorithm of the navigation software.

IV. EXPERIMENTAL ANALYSIS

A. Experimental Design

The experiments were carried out in the playground of Nanjing Agricultural University in december of 2009. The playground was muddy and full of weeds. The corresponding obstacles were manually arranged rectangular banners with the height of 1.2 meters as shown in Figure 4.

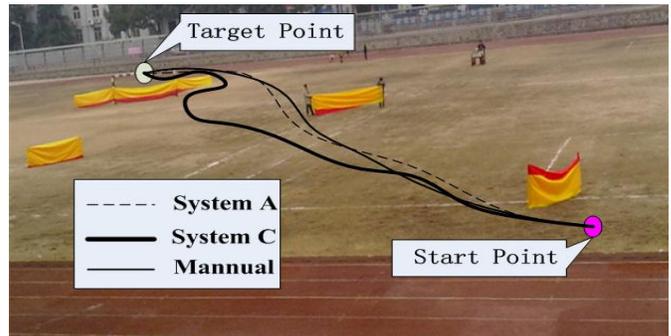


Figure 4. Playground for navigation experiments.

The experiments aim to validate that the CRFNFP model does enhance the long-range perception for mobile robots and helps to plan more efficient paths for the navigation job.

To achieve this goal, we arranged several sets of experiments under conditions of different obstacle color, system configuration and weather. There were 3 colors for obstacles: red(R), yellow(Y), and mixed(M, red banners combined with yellow ones). The weather condition contains sunny(S) and cloudy(C). The 3 system were configured as follows:

1) *System A*: CRFNFP-based far-field scene inference and subsequent far-field path planning, near-field local mapping, directional deviation computation;

2) *System B*: Bayes-classifier-based far-field scene inference and subsequent far-field path planning, near-field local mapping, directional deviation computation;

3) *System C*: Near-field local mapping, directional deviation computation.

The above mentioned Bayes-classifier can be regarded as a simplified CRFNFP model, which doesn't incorporate the spatial contexts for the recognition of scenes. In our implementation, all the algorithms are programmed under Visual C++ 6.0. Our CPU processor in the robot is 2.26 GHz Intel Core Duo P8400. And the running frequencies of A, B, C systems are 2Hz, 7Hz and 15Hz respectively with image resolution 320×240.

We ran system A and system B 5 times respectively for every combination of weather and obstacle color. And we ran system C only 2 times under the sunny and mixed obstacle color condition because system C were run only based on near-field local mapping and directional deviation computation, which means system C is not sensitive to the weather and obstacle color conditions. Furthermore, we manually drove the robot one time to collect data for comparison. In all runs, the walking speed of the robot is 0.3m/s. We list typical experimental data in TABLE 1.

Figure 5 shows the typical running trajectories of different system configurations and the corresponding obstacles positions. In each subplot, the start point (S) and the circle G represent the position of vehicle and the target point respectively. In our experiments, if the distance between the vehicle and the target point is within 2 meters, we consider the job is already finished. In Figure 5(a), we collected one running trajectory of each system configuration represented by different line styles. And Figure 5(b), Figure 5(c) and Figure 5(d) show two running trajectories of each system configuration respectively. All the quantitative data of the six trajectories are listed in TABLE 1 by bold-type.

B. Evaluation Criterion

To the best of our knowledge, the research of long-range terrain perception for outdoor robots was started just from several years ago. And there is no generally accepted evaluation criterion. In order to better illustrate the significance of our experimental results, we define 3 creterion as follows:

1) *Obstacle Response Distance (ORD for short)*: It is defined as the distance between the robot and the obstacle when the robot begins to recognize the obstacle steadily. Take **Error! Reference source not found.(a)** as an example, the distance between point C and the longest banner is the ORD of system A while the distance between point A and the longest banner is the ORD of system C.

2) *Time to Finish Job (TFJ for short)*: It is defined as the total time for the robot to finish the job.

3) *Distance of the Whole Run (DWR for short)*: It is defined as the distance that the robot experienced during the whole run.

Furthermore, we simplified the expressions of experimental conditions in TABLE 1 to facilitate the listing. For example, MIX-A-SUNNY-1 represents the condition of sunny, mixed color banner with the system A and Round 1.

TABLE I. Typical Running Results of Experiments

Experimental Condition	DWR (m)	TFJ (s)	ORD (m)
MIX-A-SUNNY-1	86.436	279	47
MIX-A-SUNNY-2	84.519	273	49
MIX-B-SUNNY-1	84.321	273	48
MIX-B-SUNNY-2	90.556	291	28
MIX-C-SUNNY-1	94.177	304	5
MIX-C-SUNNY-2	113.460	366	4
RED-A-SUNNY-1	87.345	283	46
RED-A-SUNNY-2	86.748	282	47
RED-B-SUNNY-1	88.574	285	40
RED-B-SUNNY-2	84.891	274	45
YELLOW-A-SUNNY-1	88.644	286	39
YELLOW-A-SUNNY-2	85.784	277	47
YELLOW-B-SUNNY-1	83.525	269	48
YELLOW-B-SUNNY-2	106.880	345	4
YELLOW-B-SUNNY-3	93.335	302	16
YELLOW-B-SUNNY-4	86.745	282	45
YELLOW-B-SUNNY-5	84.311	273	48
YELLOW-A-CLOUDY-1	87.675	282	42
YELLOW-A-CLOUDY-2	86.987	280.5	47
YELLOW-B-CLOUDY-1	84.587	274	49
YELLOW-B-CLOUDY-2	88.985	285	40
YELLOW-B-CLOUDY-3	95.366	307.5	15
YELLOW-B-CLOUDY-4	87.961	284	41
YELLOW-B-CLOUDY-5	92.388	298	22
MANUAL	83.500	269	—

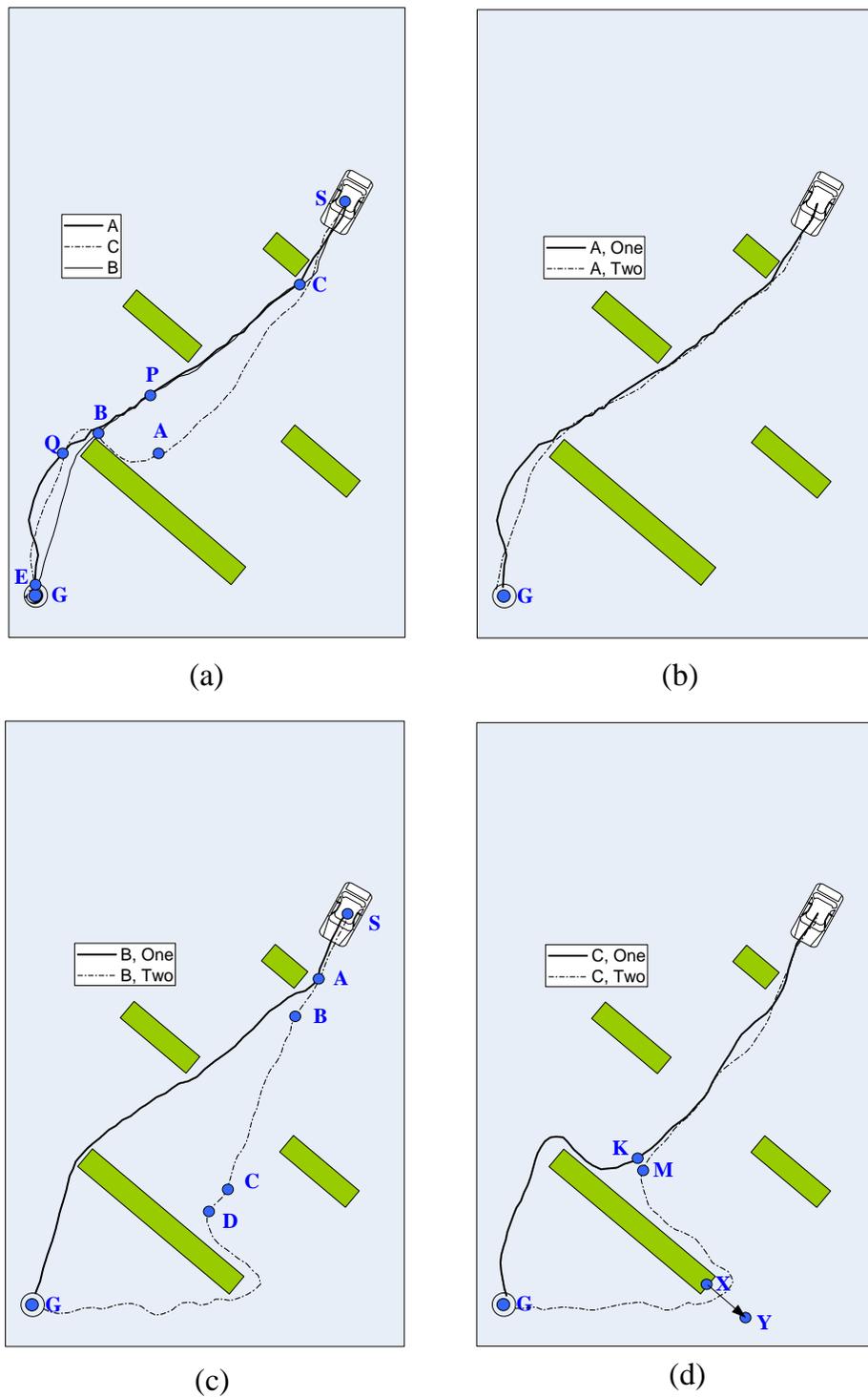


Figure 5. Trajectories of different runs of various system configuration.

C. Experimental Results

1) First, we investigate the navigation behavior of System C:

The robot based on system C doesn't infer any scenes, and the walking strategy can be summarized as:

a) If there are obstacles within the range of 8 meters around the robot, the robot turns left or right to avoid the obstacles;

b) If there are not obstacles within the range of 8 meters around the robot, the robot needs to calculate the directional

deviation and the corresponding the turning angle, which controls the robot to walk towards the target point.

As is shown by the dash-dotted line in Figure 5(a), the trajectory of the robot can be divided into 3 segments: SA, AB and BE.

During the segment of SA, there are no obstacles. The robot will generally head the direction of target point G. We found the actual direction of trajectory didn't coincide with that of target point but with small-amplitude oscillations. We considered it's caused by the oscillation of the control input, which, in turn, is resulted by GPS input drift and the large inertia of the UGV.

The robot found the close-range obstacle and decided to turn right sharply to avoid the obstacle during the segment of AB. The obstacle was almost 5 meters before the robot when the robot began to turn right. Therefore, we consider the obstacle response distance of system C in this run to be 5 meters. It's obvious that the total length of SA and AB is larger than that of other two trajectories. And the polyline walking of SA and AB can be thought as the first reason for the inefficiency of system C.

The controlling mode of system C in segment BE is similar to that of SA. The main difference of BE to other trajectories in Figure 5(a) lies in that when at the point B, the robot didn't head the target pointing G and the system C cost additional 30 control cycles to adjust the heading. So the extra time for heading adjustment can be thought as the second reason for the inefficiency of system C.

The third reason for the inefficiency of system C lies in the uncertainty of the turning direction decision when the robot confronted close-range obstacles. If the robot first recognize the banner as a left-anterior obstacle and it will turn right; and if the robot first recognize the banner as a right-anterior obstacle and it will turn left. Figure 5(d) shows the two trajectories of system C under the same experimental condition. The solid and the dash-dotted lines indicated the robot chose different turning directions in two rounds of running. The corresponding data of MIX-C-SUNNY-1 and MIX-C-SUNNY-2 in **Error! Reference source not found.** show that the decision of turning left made the robot cost another 62 seconds to reach the same target point (304 seconds and 366 seconds respectively). An extreme example is that if one end of the banner (point X) extends to the point Y, the robot at the point of M may still choose to turn left with a large probability, which will make the robot take more time to reach the same target point.

In summary, the polyline walking, heading adjustment and the uncertainty of the turning direction decision are the main reasons for the inefficiency of system C. And it's obvious that all these reasons, in turn, are caused by the incompetency of system C to incorporate the global obstacle distribution while making decisions.

2) *Second, we investigate the navigation behavior of System B:*

Different from system C, system B makes decision based on the distribution of both close-range and distant-range obstacles, which can help the robot to avoid the obstacle ahead of time. The solid trajectory in Figure 5(c) shows that the

obstacle response distance of system B in this round is 48 meters, which is much larger than that of system C.

However, we also found that, the dash-dotted trajectory (YELLOW-B-SUNNY-2 in TABLE 1) in in Figure 5(c) indicated that it's 23 meters longer than the other trajectory (YELLOW-B-SUNNY-1 in TABLE 1). To find the real reason for this inefficiency, we used the data accumulated during the experiment to offline simulate the whole run. The simulation showed that the uncertain recognition of distant-range obstacles during the segment BC caused the oscillation of the turning decision of system B. As a result, the robot randomly turns left or right during the segment BC. And on the other side, the solid trajectory turns right continuously to avoid the longest banner ahead of time, which made the whole trajectory smoother and shorter.

During the segment CD of the dash-dotted trajectory, the far-field scene inference and subsequent far-field path planning cannot provide meaningful guidance for the robot. The reason is that when the obstacle is too close to the robot, the system can't see the whole traversable region in the image plane because of the limitation of field-of-view of the stereo vision. Therefore, the long-range inference is not necessarily suitable for all kinds of scenes. And we plan to leave this problem to further research.

To summarize, the uncertain recognition of distant-range obstacles is the main reason for the inefficiency of system B. And the intrinsic reason for this is that the Bayes-classifier-based far-field scene inference used by system B only incorporates the appearance information while recognizing the scene, which is easily affected by the changing illumination of outdoor environments.

3) *Third, we investigate the navigation behavior of System A:*

System A performed better than system B and C under same conditions in terms of DWR, TFJ and ORD as shown in TABLE 1.

The ORD of system C is 4 or 5 meters while that of system A is at least 39 meters, which shows that the CRFNFP-based navigating system does enhance the long-range perception for mobile robots.

The average of DWR and TFJ of system A are about 86.8 meters and 280 seconds respectively, while the average of DWR and TFJ of system C are 103.8 meters and 335 seconds, which indicates that the CRFNFP model does help the robot to plan more efficient paths for the navigation.

V. CONCLUSIONS

We designed comparison experiments to further validate the CRFNFP algorithm, which is proposed by us in 2010, in real challenging scenes. The comparative experiments indicate that, the CRFNFP-based navigating system outperforms traditional local-map-based navigating systems in terms of all criterion defined by us in this paper.

And the results also show that the CRFNFP algorithm does enhance the long-range perception for mobile robots and helps planning more efficient paths for the navigation.

VI. FUTURE RESEARCH

Our further research may refer to the incorporation of other kinds of contexts, such as semantic contexts and temporal contexts. And we try to simulate the attention mechanism of human beings into perception algorithms.

VII. ACKNOWLEDGMENT

This work is supported by the national natural science foundation (Q12F030004) and Ningbo natural science foundation (2011A610198, 2011A610155).

REFERENCES

- [1] Wang, M.; Zhou, J.; Tu, J., et al. Learning Long-range Terrain Perception for Autonomous Mobile Robots[J]. International Journal of Advanced Robotic Systems, 2010, 7 (1), 55-66.
- [2] Matthies, L. Stereo vision for planetary rovers: Stochastic modeling to near real-time implementation[J]. International Journal of Computer Vision, 1992, 8 (1), 71-91.
- [3] Pagnot, R.; Grandjean, P. Fast cross-country navigation on fair terrains[J]. Proceedings - IEEE International Conference on Robotics and Automation, 1995, 3, 2593-2598.
- [4] Singh, S.; Simmons, R.; Smith, T., et al. Recent progress in local and global traversability for planetary rovers[J]. Proceedings - IEEE International Conference on Robotics and Automation, 2000, 2, 1194-1200.
- [5] Rieder, A.; Southall, B. Stereo perception on an off-road vehicle[J]. Proc. IEEE Intelligent Vehicle Symposium, 2002, 1, 221-226.
- [6] Ollis, M.; Huang, W. H.; Happold, M., et al. Image-based path planning for outdoor mobile robots[A]. IEEE International Conference on Robotics and Automation[C]. Pasadena, CA, 2008. pp 2723-2728.
- [7] Jackel, L. D.; Krotkov, E.; Perschbacher, M., et al. The DARPA LAGR program: Goals, challenges, methodology, and phase I results[J]. Journal of Field Robotics, 2006, 23 (11-12), 945-973.
- [8] Michael J. Procopio, J. M. G. G. Learning terrain segmentation with classifier ensembles for autonomous robot navigation in unstructured environments[J]. Journal of Field Robotics, 2009, 26 (2), 145-175.
- [9] Happold, M.; Ollis, M.; Johnson, N. Enhancing supervised terrain classification with predictive unsupervised learning[A]. Robotics: Science and Systems[C]. Philadelphia, PA. Cambridge, 2006.
- [10] Dahlkamp, H.; Kaehler, A.; Stavens, D., et al. Self-supervised monocular road detection in desert terrain[A]. Robotics: Science & Systems[C]. 2006.
- [11] Max Bajracharya, A. H. L. H. M. B. T. M. T. Autonomous off-road navigation with end-to-end learning for the LAGR program[J]. Journal of Field Robotics, 2009, 26 (1), 3-25.
- [12] Lookingbill, A.; Lieb, D.; Thrun, S. Optical Flow Approaches for Self-supervised Learning in Autonomous Mobile Robot Navigation (Autonomous Navigation in Dynamic Environments)[Z]. Springer Berlin/ Heidelberg: 2007; pp 29-44.
- [13] Hadsell, R.; Sermanet, P.; Ben, J., et al. Learning long-range vision for autonomous off-road driving[J]. Journal of Field Robotics, 2009, 26 (2), 120-144.
- [14] Lafferty, J. D.; McCallum, A.; Pereira, F. C. N. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data[A]. The Eighteenth International Conference on Machine Learning[C]. 2001.

AUTHORS PROFILE

Wang Mingjun, born in 1981, majored in machine vision and received PhD in 2010. His main research now focus on the machine learning and artificial intelligence.