A Robotic System Towards Concrete Structure Spalling And Crack Database

Liang YANG[1,2], Bing LI[1], Wei LI[1], Zhaoming LIU[2], Guoyong YANG[2], Jizhong XIAO [1,2*]

Abstract— Concrete spalling and crack inspection is a labor intensive and routine task. However, it plays an important role in structure health monitoring (SHM) of civil infrastructures. Autonomous inspection with robots has been regarded as one of the best ways to reduce both error and cost. This paper presents an automated approach using Unmanned Aerial Vehicle(UAV) and towards a Concrete Structure Spalling and Crack database (CSSC), which is by far the first released database for deep learning inspection. We aim locate the spalling and crack regions to assist 3D registration and visualization. For deep inspection, we provide a complete procedure of data searching, labeling, training, and post processing. We further present a visual Simultaneously Localization and Mapping(SLAM) approach for localization and reconstruction. Comparative experiments and field tests are illustrated, results show that we can achieve an accuracy over 70% for field tests, and more than 93% accuracy with CSSC database.

I. INTRODUCTION

Manual inspection and evaluation is a common procedure for structural health monitoring (SHM). The human inspectors need to periodically detect the visible surface defects (such as cracks) and the interior subsurface defects (such as delamination or voids) using non-destructive evaluation (NDE) devices such as high resolution cameras, impact echo (IE) [1] and ground penetration radar (GPR) [2] to assess the serviceability conditions of civil infrastructures. However, such manual inspections are time consuming, labor intensive, and often require the setup of scaffolding equipment to access hard to reach places, which is costly and leads to safety concerns.

Machine vision and automated inspection and evaluation procedures have become very popular for detecting surface flaws in civil engineering communities in the past decade [3]. Conventional image processing methods are applied for detection by emphasizing the high contrast distinct visual features. Edge detection algorithms, such as fast Haar transform (FHT), fast Fourier transform (FFT), Sobel operator and Canny edge detector, have been deployed for crack identification [4]. These algorithms are also used with other algorithms such as image segmentation, image thresholding (like OSTU) and morphology operations [5], and work well on uniform background. But these are still relying on segmentation and handcrafted feature detectors (like edge, line) for training, and lead to failures for various defect types.

The concept of using deep convolutional neural networks (CNN) for tunnel visual inspection was presented in [6], where related techniques and descriptions were elaborated. Authors in [6] mainly used the CNN for feature extraction and Multi-Layer Perceptron (MLP) for classification. The paper didn't make it clear that what is the database, and what kind of defection can be detected by the system. Crack detection is discussed in [7], and it is said to have a better accuracy over SVM and boosting without notifying the corresponding database and field tests. Authors in [8] did a good job for providing a database for crack inspection and labeling, and evaluation of the performance.

To automate the concrete structure crack inspection, various robotic systems have been explored. Authors in [9] developed an early vision-based inspection vehicle robot which used batch processes of crack detecting algorithms and data storage in bridge management system (BMS) database. Vision-based UAV robots were deployed for bridge inspection and condition assessment in [10], and conventional image processing approaches were applied for crack identification. Therefore, based on our previous research on non-destructive evaluation (NDE) robotics for concrete deck inspection [11], it is very promising to deploy UAV inspection robot for low-cost and omnibearing detection. By automatically collecting increasing amount of inspection data along with corresponding spatial information, data-driven based classification and recognition approaches like CNN shows the potential to provide more robust inspection detection result for SHM assessment than conventional approaches.

This paper present a low-cost automated inspection approach using UAV equipped with stereo-vision system. Our contributions are two folds. First, we build a database by collecting and labelling images of spalling and cracks on concrete structures for deep training purpose, and we evaluate the deep learning approach performance for inspection based a proposed cascading shotting approach. Second, we leveraging our previous work on UAV navigation and SLAM [12], [13], [14], [15]. We further developed a novel method to project the labelled 2D flaws to point cloud model for 3D visualization. To the best of our knowledge, this paper firstly utilizes the robotics automated UAV for deep learning based visual inspection on concrete structure.

II. SYSTEM ARCHITECTURE

As shown in Fig.1, the inspection system consists of three subsystems, including: 1) Control and Mission system (CMS): a quadrotor UAV with mission planner and navigation control system [13], [12]. It fuses visual odometry with
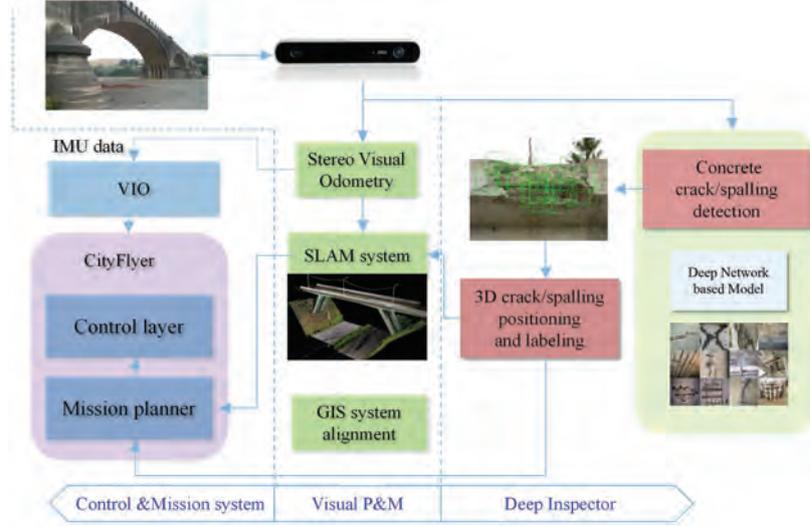
Fig. 1. The inspection system consists of CityFlyer UAV with mission planner and navigation control system, Deep-Inspector software for spalling and crack detection and labelling, and VO-based positioning and mapping module. The final 3D map can be merged into GIS system, which can help build seamless data net for monitoring purpose.

inertial measurement unit (IMU) data, called visual inertial odometry (VIO) with an output frequency of 100 HZ, for robust positioning control. 2) Deep-Inspector: Towards a CSSC database system with very deep CNN for spalling and crack detection and labelling. It is performed on the ground station via wireless data transition at 2 HZ. 3) visual positioning and mapping (P&M): loop closing is introduced to reduce drift of visual odometry (VO) to guarantee accurate positioning and 3D point cloud mapping with defect area registration for visualization. Hardware of CMS is consist of CityFlyer[13], [12] and a laptop based ground system. Since the paper mainly discuss the deep database system and 3D positioning with defect registration, we just provide simple knowledge of VIO.

A. Visual Inertial Odometry

We introduce Mulit-state-EKF(MS-EKF)[16] to fuse IMU measurement and VO in a direct approach, that is, the IMU measurement acts as the propagation and the VO acts as correction.

For IMU, its evolving state vector is,

$$X_{IMU} = [^W\mathfrak{R}_I^T \quad ^WV_I^T \quad ^Wa_I^T \quad ^I_Wq \quad b_a \quad b_g] \tag{1}$$

Where $^W\mathfrak{R}_I^T$ denotes the position of IMU in the world frame W. ^I_Wq is unit quaternion that represents the rotation from the world frame W to the IMU frame I. $^WV_I^T$ and $^Wa_I^T$ are the IMU velocity and acceleration with respect to the world. b_a and b_g denote the the biases affecting the gyroscope and accelerometer measurements.The system derivative form can be partially presented as following using East-North-Up(ENU) coordinate system (partly referred in [16], [17],

$$
\begin{aligned}
^W\dot{\mathfrak{R}}_I &= {}^W V_I + {}^W a_I \cdot \Delta T \\
^W\dot{V}_I &= {}^W a_I \cdot \Delta T = {}^W_I C(a_m - b_a) + g \\
\dot{a}_I &= j + \chi \times V + w_m \times a_m \\
^I_W\dot{q} &= \frac{1}{2}\Omega(w_m)^I_W\dot{q}
\end{aligned}
\tag{2}
$$

Where $^W_I C$ is the translation from the IMU frame to the world frame, a_m is the acceleration measurement, w_m is the angular velocity measurement, ΔT denotes the time interval, g denotes the gravity. The a_I acceleration subjects to a rotation and translation in inertial frame, w_I denotes the angular velocity. Ω is the matrix product referred in [16].

Once, VO system outputs the pose $P_{vo} = [r,t]$ (r denotes the rotation, and t denotes the translation), we can update the system based on the measurement model,

$$Z = HX_{IMU} + V \tag{3}$$

Where Z denotes the measurement, V denotes the measurement noise. H denotes the output measurement matrix. Then, the prediction from IMU propagation can be corrected by updating through EKF approach.

III. DEEP BASED DETECTION

The core of the system is the deep inspection system with database with data-collection ability. To our knowledge, there does not exist any database containing this kind of information, especially for both spalling and crack. With web search and real data collection in Manhattan, New York), we built the CSSC database. We also keep the field test data for public. In this section, we discuss the complete procedure of data preparation, labeling, training and post processing to vote defective region for 3D registration.
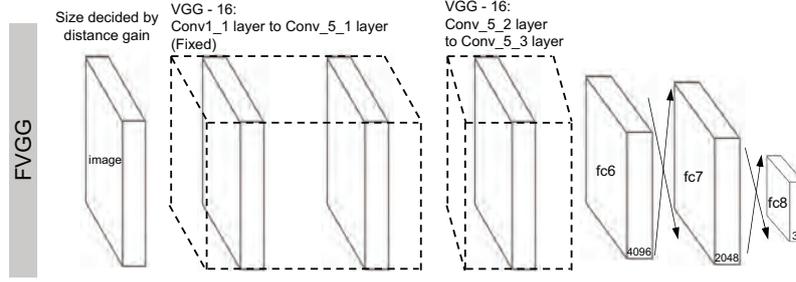
Fig. 2. Architecture of the proposed Fine-Tuned VGG-16 (FVGG) model. FVGG is robust to input resolution to enable precise detection.

A. Data Preparation And Labeling

It is should be noted that web keyword search provides random results, and we have tried the following key words:

- **For concrete spalling**: Concrete spalling, Concrete delamination, concrete bridge spalling, concrete column spalling, concrete spalling from fire, concrete spalling repair, concrete wall, etc. We used search engines Google, Yahoo, Bing, and Flickr, and found total of 22,268 images. From all these images, only 278 images were deemed useful for training.
- **For concrete crack**: Concrete crack, crack repair, concrete scaling, concrete crazing, concrete crazing texture. We found a total of 16,215 images, and 954 images were used for crack detection.

The 278 spalling images and 954 crack make up the initial CSSC database. Since the randomly posted images on the Web are not well tagged with the flaw area, and the shape of the spalling and crack region tends to be stochastic in distribution (illustrated in Fig.3(a) Fig.3(b)), we label the area manually as illustrated in lower part of Fig.3(a) and Fig.3(b).) by referring the experts in civil engineering. It should be noted that: 1) spalling defects are normally quite apparent, the color information information should be included for further detection. 2) the civil engineers care more on unobvious cracks as discussed in [18], thus should be more careful when labeling.

The defect region distribution of spalling and crack tends to be unorganized and cannot be marked with a fixed rectangle region, also the method proposed in [19] does not work in our robot system as the distance varies with flying. For training purpose, this paper cuts the images with region of interests(ROI) into different size of sub-parts, such as $100*100$ and $130*130$ illustrated in Fig.4).

In order to decide whether a sub-image should be selected or not, we define a rule to determine whether a random cut part can be used as ROI training positive input. Since we have the manually labeled results as illustrated in Fig.3, this paper propose to use pixel threshold to pick the positive and negative data. Let define the manually labeled region contains N pixel level labels. For each random generated small parts I_s, given the following rule,

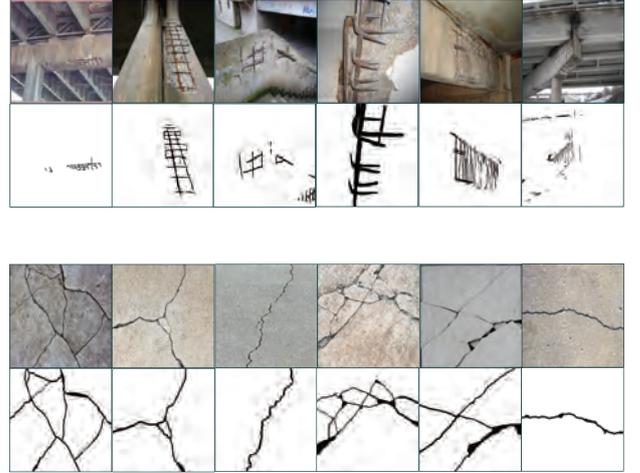$$flag = \begin{cases} 0, if \ n(I_s) <= 0 \\ 1, if \ n(I_s) >= N*k \end{cases} \quad (4)$$

Fig. 3. Representation of spalling and crack images with manually labeling. The above two rows show the crack images with its manual labeling. In this paper, we label the images to enable distribution estimation of the images, which then helps to evaluate the health condition of the concrete structure. The below two rows represent the spalling images with its manual labeling, which then enables the sub-cutting for training.

Fig. 4. Examples of generated positive training images based on proposed selection criteria. The spalling images are presented with size of $100*100$ with $k = 0.04$, and $130*130$ with $k = 0.06$ for training. The crack images are also presented with $100*100$ with $k = 0.04$, and $130*130$ with $k = 0.06$ for training.

Where k denotes the percentage.

B. The CNN Model and Training

This paper uses a fine-tuned VGG-16 (FVGG) for classification [20] which has shown its ability to deeply understand image high level features and provide effective representation feature. The network architecture is presented in Fig.2, where
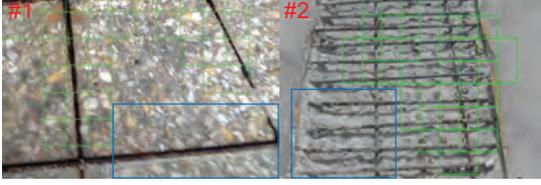
Fig. 6. Missed detected examples in CSSC database by using our model.

Fig. 9. The over detection and missed detection cases in field test 1. The orange rectangles denote over detection, and the missed detection are represented with blue rectangles.

the image input is decided with a distance gain. FVGG was fine-tuned with a large number of settings, and we finalized the network as: 1) we fixed the first 11 layers parameters, 2) we release the layers from 12 to 16, and changes the last fully-connected layer to 3 channels.

The training performs 400 iterations with a batch size 40 over 15,950 images. We set the learning rate as $1.0E - 3$ for the first 200 iterations, and then adjust to $5.0E - 4$ for the second 200 iterations. We tried to change the learning to a smaller scale, but got worse detection rate. The overall objective loss deploys the following form:

$$L = -\sum_{x^I} p(x^I)log(q(x^I)) \qquad (5)$$

where $p(x^I)$ denotes the true probability of input x^I, $q(x^I)$ is the predicted probability.

C. Post Processing For Defects Labeling

After the model is well trained, we utilize the model to do detection. In Section III.B, we discussed that the input images are varied with the distance. The distance factor can be easily achieved with stereo camera. For a constant sub-image size I_s, we use a sliding window to do detection through the whole image. Then each image will be labeled with crack true, or spalling true, or no defects. Since the accuracy of the model cannot be guaranteed to be 100%, we store all the detection results of the same place and the final decision is made through the averaging the probability.

IV. VISUAL POSITIONING AND INSPECTION REGISTRATION

3D representation of the construction is commonly used for reverse engineering applications and structure analysis in civil engineering, besides 3D model can help by providing depth information as well as the distribution for better post assessment [21]. In this work, the 3D mapping system is built based on the visual-SLAM, then the deep based detection result helps to register the spalling and crack region in the 3D map.

A. Loop closing Aided VO

Visual Odomery cannot eliminate the long term drift without adopting the loop closing to correct the error. For SLAM, we just record the key frames $^FK = \{I_i, P_i | i \in \{1, 2, ..., m\}\}$ (i.e vertex) based on detection of threshold movement, where I_i is the image, P_i is the pose. We know that the step transformation between consecutive two frames, and the transformation $T_{i,j}$ between any two frames i, j can

be derived the same way. The best aspect of SLAM is that it can help find the correlation between frames, that is, *edge*. Then, we can optimize the key frame poses with the form [22],

$$(^KP_i^*, ^KP_j^*) = arg\min_{^KP_i}(^K\hat{P}_j - T_{i,j}.^K\hat{P}_i)^T \Omega_{i,j}(^K\hat{P}_j - T_{i,j}.^K\hat{P}_i) \qquad (6)$$

Where $\Omega_{i,j}$ denotes the information matrix, $*$ denotes the optimized pose, $^\wedge$ denotes the estimation. After each pose optimizing process, we only consider the errors of the last key frame to be corrected, and we have the error with, $^KT_{error} = ^K P_{last}^* .^K P_{last}^{-1}$. Then, we can correct the current VO output with $P_{current}^* = ^K T_{error} \cdot P_{current}$.

B. Spalling and Crack Alignment

For spalling the area and depth information is interesting to engineers. The deep detection tells the regions, it enables marking of the region with special color in the 3D model, which in turn provides much easier way for evaluation. For crack, engineers are more interested in the tiny cracks with width information. The regions can give more information to do post detection for crack and spalling distribution.

V. EXPERIMENTS

Since the robustness of the CityFlyer control system has been demonstrated in [12], [13], we skip this part. The simulation is carried out to detect the region of spalling and crack in CSSC. We provide a detailed analysis for success and failure. The two field tests were performed in Manhattan with CityFlyer, both under bridge area.

A. Deep Network Based Detection with CSSC

The initial CSSC database is divided into three parts, that is, 70% for training, 10% for validation, and 20% for testing. Some results are illustrated in Fig.5. For test, we achieved a 93.36% mean accuracy(see in Table. I). The failure detection is mainly caused by image blur or low contrast light.

In Fig.5, images #1,#2,#3 are spalling detection results, and images #4,#5 are crack detection results. For an expert

Fig. 5. Results generated by our deep based detection model with CSSC database, where the green rectangle denotes the region of spalling and crack. In images #6 and #7, the red features denote the post-labeling of the crack for evaluation, where the width of the crack can be achieved via stereo-vision.
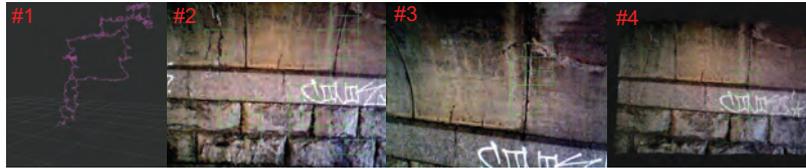

Fig. 7. The detection results achieved by CityFlyer in field test 1. Image #1 denotes the trajectory of the CityFlyer, #2 and #3 are detected results, and #4 is the 3D registered model.

Fig. 8. The detection results achieved by CityFlyer in field test 2. Image #1 denotes the trajectory of the CityFlyer, #2 and #3 are detected results, and #4 is the 3D registered model.

TABLE I
QUANTIFIED RESULT OF DETECTION WITH CSSC DATABASE

Database	Average Precision (%)	Partial Incomplete Detection (%)	Total Image
CCNY-CSSC	93.36	6.64	1232

TABLE II
FIELD TEST RESULT AT MANHATTAN 155 ST

Test No.	Average Precision (%)	Blurred Image (frames)	Average Precision Without Blur(%)	Over Estimated (%)	Total Image
No.1	72.45	149	76.73	97.18	4998
No.2	67.65	55	71.19	24.3	2650
Further Tuned With Field Data					
No. 1	83.69	149	87.97	93.34	4998
No. 2	81.38	55	84.92	33.57	2650

engineer, the most difficult part is not to find the spalling, but to find the crack(especially minor crack). Thus, our method adopts a two steps approach: 1) deep network based detection to the region containing the crack;2) feature detection to find the crack distribution. Then, we can see in Fig.5, images #6,#7 with red labeling indicates the distribution of the crack. Mis-detection is presented in Fig.6, where image #1 dneotes the case of image blur, and image #2 denotes the case of low contrast.

B. Field Tests in Manhattan

The field tests were carried out at a bridge located at upper west side of 155th Broadway in Manhattan. The ground station for monitoring and deep CNN processing was a Dell XPS 15 laptop, with a Nvidia 960M graphic card. Two sets of field tests were implemented with different scenario.

1) Field Test 1: The first experiment was performed at the middle part of the bridge where a darker light affected

the inspection. The flying duration was $210s$ with a total of 4,998 images captured. It is presented in Fig.7. The purple line in image #1 denotes the trajectory of CityFlyer with the goal to capture the spalling and crack at the right angle.

The expected spalling and crack region number is two, which is shown in Fig.7 and in image #2 and a closer look of right part is shown in image #3. For all the detection results we found so far, shows 76.73% accuracy without image blur (see in Table.II). Our 3D registration has an advantage to do 3D visualization, and it enables professional health evaluation. It is shown in image #4, our method can label the region in 3D space for inspection.

Simulations with CSSC show that image blur can affect the detection, which is also found in field test as illustrated in Fig.9. The blue rectangles denote the missed detection. By analyzing all the images, we found that the image brightness, image contrast, and blurred degree affect the detection accuracy, which then should be considered to be pre-processed to improve the accuracy.

2) Field Test 2: The second experiment was carried out in the entrance of the under bridge area, where a brighter light exists. Fig.8 shows the CityFlyer's trajectory and several representative results, and the 3D registered model is presented in image #4. CityFlyer was set at the left side of the spalling and crack region, then it tried to move toward the front angle of spalling and crack for inspection (see in image #1). This site contains two spalling and crack regions as illustrated in images #2 and #3. For all the images collected, we have an average accuracy of 71.19% without blurred image.

Compared to Test 1, the brighter image gave better inspection results. It is shown in Table.II, the *over detection* is represented by the orange rectangles in Fig.9, image #1,#2. Over detection is casue by darker light, which lowers the feature comparison between palling and crack region with other regions. Thus, Test 1 has a 97.18% over estimation compared to 24.3% of brighter images.

As discussed in simulation and experiment 1, image blur and brightness affect the detection a lot. Besides, we further fine-tuned the model with field test image data with the same process as discussed in Section.III. The comparative results are illustrated in Table.II, we can see that we got 11.14% and 13.73% improvement after fine-tuned with filed images.

VI. CONCLUSIONS

The paper proposed a automated UAV-based system for concrete structure inspection with the ability to locate spalling and crack area and then to provide depth and width information for evaluation. Inspection system consists of CityFlyer for image capturing with a visual-inertial odometry, deep based spalling and crack detection, and 3D registration for visualization. The system can significantly lower the labor costs. Moreover, it has an advantage of accurate automated detection. The system shows a 93.36% detection accuracy with CSSC database, and over 70% accuracy with field tests. The system supports real-time localization with detection processing and can be used as a reliable system for concrete structure health monitoring.

REFERENCES

[1] Sansalone, M.J., Streett, W.B.: Impact-echo. nondestructive evaluation of concrete and masonry. (1997)
[2] Daniels, J.J., et al.: Ground penetrating radar fundamentals. Prepared as an appendix to a Report to the US EPA, Region V (2000) 1–21
[3] Zaurin, R., Catbas, F.: Integration of computer imaging and sensor data for structural health monitoring of bridges. Smart Materials and Structures **19**(1) (2009) 015019
[4] Abdel-Qader, I., Abudayyeh, O., Kelly, M.E.: Analysis of edge-detection techniques for crack identification in bridges. Journal of Computing in Civil Engineering **17**(4) (2003) 255–263
[5] Sinha, S.K., Fieguth, P.W.: Automated detection of cracks in buried concrete pipe images. Automation in Construction **15**(1) (2006) 58–72
[6] Makantasis, K., Protopapadakis, E., Doulamis, A., Doulamis, N., Loupos, C.: Deep convolutional neural networks for efficient vision based tunnel inspection. In: In Proc. Of IEEE International Conference on Intelligent Computer Communication and Processing, IEEE (2015) 335–342
[7] Zhang, L., Yang, F., Zhang, Y.D., Zhu, Y.J.: Road crack detection using deep convolutional neural network. In: Image Processing (ICIP), 2016 IEEE International Conference on, IEEE (2016) 3708–3712
[8] Eisenbach, M., Stricker, R., Debes, K., Gross, H.M.: Crack detection with an interactive and adaptive video inspection system
[9] Oh, J.K., Jang, G., Oh, S., Lee, J.H., Yi, B.J., Moon, Y.S., Lee, J.S., Choi, Y.: Bridge inspection robot system with machine vision. Automation in Construction **18**(7) (2009) 929–941
[10] Khan, F., Ellenberg, A., Mazzotti, M., Kontsos, A., Moon, F., Pradhan, A., Bartoli, I.: Investigation on bridge assessment using unmanned aerial systems. In: Structures Congress 2015. (2015) 404–413
[11] Xiao, J., Agrawal, A.: Robotic inspection of bridges using impact-echo technology. UTRC-RITA Project Final Report (2015)
[12] Dryanovski, I., Valenti, R.G., Xiao, J.: An open-source navigation system for micro aerial vehicles. Autonomous Robots **34**(3) (2013) 177–188
[13] Valenti, R.G., Jian, Y.D., Ni, K., Xiao, J.: An autonomous flyer photographer. In: In Proc. Of IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems, IEEE (2016) 273–278
[14] Yang, L., Qi, J., Jiang, Z., Song, D., Han, J., Xiao, J.: Guiding attraction based random tree path planning under uncertainty: Dedicate for uav. In: In Proc. Of IEEE International Conference on Mechatronics and Automation (ICMA), IEEE (2014) 1182–1187
[15] Yang, L., Qi, J., Cao, Y., He, Y., Han, J., Xiao, J.: Uav path planning framework under kinodynamic constraints in cluttered environments. In: International Conference on Intelligent Robotics and Applications, Springer (2015) 248–259
[16] Mourikis, A.I., Roumeliotis, S.I.: A multi-state constraint kalman filter for vision-aided inertial navigation. In: In Proc. Of IEEE international conference on Robotics and automation, IEEE (2007) 3565–3572
[17] Armesto, L., Tornero, J., Vincze, M.: Fast ego-motion estimation with multi-rate fusion of inertial and vision. The International Journal of Robotics Research **26**(6) (2007) 577–589
[18] Prasanna, P., Dana, K.J., Gucunski, N., Basily, B.B., La, H.M., Lim, R.S., Parvardeh, H.: Automated crack detection on concrete bridges. IEEE Transactions on Automation Science and Engineering **13**(2) (2016) 591–599
[19] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C.: Ssd: Single shot multibox detector. In: European conference on computer vision, Springer (2016) 21–37
[20] Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
[21] Liu, W., Chen, S.e.: Reliability analysis of bridge evaluations based on 3d light detection and ranging data. Structural Control and Health Monitoring **20**(12) (2013) 1397–1409
[22] Kümmerle, R., Grisetti, G., Strasdat, H., Konolige, K., Burgard, W.: g 2 o: A general framework for graph optimization. In: In Proc. Of IEEE International Conference on Robotics and Automation, IEEE (2011) 3607–3613