

Automated Crack Detection on Concrete Bridges

Prateek Prasanna, Kristin J. Dana, Nenad Gucunski, Basily B. Basily, Hung M. La, *Member, IEEE*, Ronny Salim Lim, and Hooman Parvardeh

Abstract—Detection of cracks on bridge decks is a vital task for maintaining the structural health and reliability of concrete bridges. Robotic imaging can be used to obtain bridge surface image sets for automated on-site analysis. We present a novel automated crack detection algorithm, the *STRUM* (*spatially tuned robust multifeature*) classifier, and demonstrate results on real bridge data using a state-of-the-art robotic bridge scanning system. By using machine learning classification, we eliminate the need for manually tuning threshold parameters. The algorithm uses robust curve fitting to spatially localize potential crack regions even in the presence of noise. Multiple visual features that are spatially tuned to these regions are computed. Feature computation includes examining the scale-space of the local feature in order to represent the information and the unknown salient scale of the crack. The classification results are obtained with real bridge data from hundreds of crack regions over two bridges. This comprehensive analysis shows a peak STRUM classifier performance of 95% compared with 69% accuracy from a more typical image-based approach. In order to create a composite global view of a large bridge span, an image sequence from the robot is aligned computationally to create a continuous mosaic. A crack density map for the bridge mosaic provides a computational description as well as a global view of the spatial patterns of bridge deck cracking. The bridges surveyed for data collection and testing include Long-Term Bridge Performance program's (LTBP) pilot project bridges at Haymarket, VA, USA, and Sacramento, CA, USA.

Note to Practitioners—The automated crack detection algorithm can analyze an image sequence with full video coverage of the region of interest at high resolution (approximately 0.6 mm pixel size). The image sequence can be acquired with a robotic measurement device with attached cameras or with a mobile cart equipped with surface imaging cameras. The automated algorithm can provide a crack map from this video sequence that creates a seamless photographic panorama with annotated crack regions. Crack density (the number of cracks per region) is illustrated in the crack map because individual cracks are difficult to see at the magnification required to view large regions of the bridge deck.

Index Terms—AdaBoost, bridge deck inspection, bridge maintenance, computer vision, concrete, crack detection, crack pattern

Manuscript received April 02, 2014; accepted June 29, 2014. Date of publication October 07, 2014; date of current version April 05, 2016. This paper was recommended for publication by Associate Editor D. Song and Editor D. Tilbury upon evaluation of the reviewers' comments.

P. Prasanna and K. Dana are with the Department of Electrical and Computer Engineering, Rutgers University, Piscataway, NJ 08854 USA (e-mail: prateek.prasanna@gmail.com; kdana@ece.rutgers.edu).

N. Gucunski and B. Basily are with the Department of Civil and Environmental Engineering, Rutgers University, Piscataway, NJ 08854 USA.

R. Lim and H. Parvardeh are with the Center for Advanced Infrastructure and Transportation, Rutgers University, Piscataway, NJ 08854 USA (e-mail: gucunski@rci.rutgers.edu; hml42@rci.rutgers.edu; ronny.lim@rutgers.edu).

H. La is with Department of Computer Science and Engineering, University of Nevada, Reno, NV 89557 USA (e-mail: hla@unr.edu).

This paper has supplementary downloadable multimedia material available at <http://ieeexplore.ieee.org> provided by the authors. The Supplementary Material includes a presentation. This material is 12 MB in size.

Digital Object Identifier 10.1109/TASE.2014.2354314

recognition, homography, image mosaic, image stitching, Laplacian pyramid, machine learning, random forest, robotic imaging, robotic inspection, Seekur robot, structural health monitoring, structure from motion, STRUM classifier, support vector machine.

I. INTRODUCTION

CONDITION assessment of bridge decks plays a vital role in maintaining the structural health and reliability of concrete bridges. Early detection of small cracks on bridge decks is an important maintenance task. More than 100,000 bridges across the United States have exhibited early age bridge-deck cracking [3]. Many bridges exhibit defects in early stages immediately after construction. As cracks appear on the deck, paths are created for water and corrosive agents to reach the subsurface rods and steel reinforcements, requiring costly maintenance and repair.

The current method of site inspection is a time-consuming process for long-span bridges. Skilled inspectors go to the site and assess the deck condition, marking the corrosions and cracks on a chart, all under strict traffic control (Fig. 1). Automated and accurate condition assessment that requires minimal lane closure is highly desirable for fast large-area evaluation. Robotic bridge scanning is revolutionizing the process of bridge inspection [4]. However, a key challenge is automated interpretation of the large image dataset in order to infer bridge condition. We present a novel automated crack detection algorithm and demonstrate results using data from a state-of-the-art robotic bridge scanning system illustrated in Fig. 1. Many prior crack detection methods use simple edge detection or image thresholding; but these methods are non-robust to noise and require manual parameter setting and adjustment. When the cracks are high contrast regions against a near uniform background (see Fig. 2), the simple methods may perform well. However, real-world concrete images have cracks of variable appearance and competing visual clutter. Additionally, manual parameter adjustment is an impediment to automatically analyzing large datasets over multiple bridges. Our approach uses machine learning and optimization in order to successfully detect cracks while eliminating the need for tuning threshold parameters. We develop the spatially tuned robust multi-feature classifier (STRUMs) to obtain high-performance accuracy on images of real concrete. The STRUM algorithm starts with robust curve fitting to spatially localize potential crack regions even in the presence of noise and distractors. Visual features that are spatially tuned to these regions are computed. The computation of these visual features includes scale-space saliency to account for the unknown scale of the crack. A suite of possible combinations of visual features



Fig. 1. (Left) Bridge inspection at the Haymarket, VA, USA, testing site. The white dots seen in the image are the bridge deck grid-markings. (Right) RABIT - Robotic Assessment Bridge Inspection Tool: Long Term Bridge Performance (LTBP) program of the Federal Highway Administration (FHWA) [1] has developed a robotic scanning system for imaging bridge decks using nondestructive evaluation techniques (ground-penetrating radar, impact-echo measurements) and high-resolution images. Using this system, we collect image data from bridge decks for developing and testing an automated crack detection algorithm. Path-planning, sensor geometry, and collision detection for the robot scanning are discussed in [2].

is evaluated experimentally with three classifier methods: support vector machines [5], adaboost [6], and random forests [7]. The tests are done with real bridge data and hundreds of crack images from the Long-Term Bridge Performance program's (LTBP) [1] pilot project bridges at Haymarket, VA, USA, and Sacramento, CA, USA. In order to create a composite global view of a large bridge span, an image sequence from the robot is aligned computationally to create a continuous mosaic. A crack density map for the bridge mosaic provides a computational description as well as an at-a-glance view of the spatial patterns of bridge deck cracking.

A. Related Work

In prior work, many automated crack detection algorithms for bridge decks emphasize high-contrast visually distinct cracks. Standard image processing methods, including edge following, image thresholding, and morphology operations, are applicable for these cracks. Numerous successful approaches have been demonstrated with high-contrast, low-clutter crack regions as described in [8]–[15]. However, the crack images from the real concrete bridge decks are often immersed in significant visual clutter, as illustrated in Fig. 2, and are more difficult to detect in an automated manner. To illustrate the point, Fig. 3 shows the output of a recent crack detection algorithm [15] compared with our STRUM classifier on a sample image from our dataset.

Machine learning has been applied for visual recognition and classification and generally performs better than methods with hand-tuned parameters [16]–[21]. Example-based machine learning is referred to as *supervised learning* and enables statistical inference based on the relevant data without the need for manual parameter adjustment as in prior methods such as the percolation algorithm [9], [10] and binarization methods [8], [12], [14]. For the task of automated bridge crack detection, the trend toward using machine learning algorithms is relatively new. Machine learning is a large field and there is no best algorithm for all classification tasks. Constructing a suitable algorithm requires developing the right representation of the data. For example, neural nets are used to determine crack

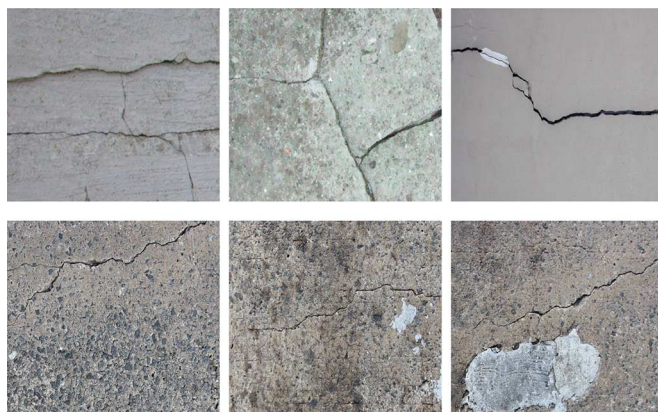


Fig. 2. (Top Row) Prior work uses images that are high contrast and low clutter similar to the images shown. Simple methods such as edge detection and thresholding for binarization can be applied for these images. (Bottom Row) Real bridge deck images from our work shows significantly more distractors and pose a more challenging crack detection problem.

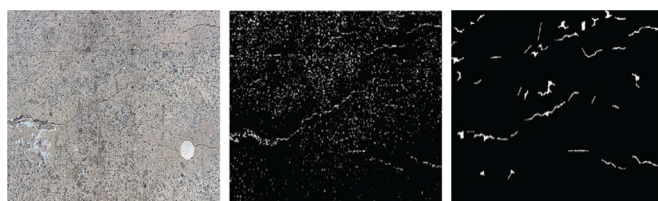


Fig. 3. (Left) Original image showing a surface with cracks. (Center) Image showing the output of a prior crack detection algorithm [15]. (Right) Image showing the output of our algorithm.

orientation [22], however, the representation relies on standard image binarization. Automated classifiers are used in [23] including support vector machines, nearest neighbor and neural nets with input features such as crack eccentricity, solidity and compactness. A drawback of this method is that input features depend on first segmenting the crack with standard manually tuned methods. If this segmentation fails, the features for the classifier are not directly meaningful. For our approach, a line-fitting is used to find crack segments; the method is robust because it is known to work well in the presence of noise and clutter. The features are computed relative to the local line fit and they are relevant for classification even if the line does not fall on a crack segment. Our approach of robust spatial tuning combined with a localized multifeature appearance vector is a key contribution of the STRUM algorithm.

II. METHODS

The STRUM classifier consists of three components: 1) a robust line segment detector; 2) spatially-tuned multiple feature computation; and 3) a machine learning classifier. The curve fit is done using RANdom SAMple Consensus (RANSAC) [24], where line segments are fit to pixels below a fixed percentage of the average intensity in pixel neighborhoods called blocks. The robustness of RANSAC is well known, and Fig. 7 provides a visual comparison to least-squares estimation for line fits. Small blocks are used so that curved cracks can be reasonably approximated with line segments within the small region. This approximation is equivalent to a Taylor series approximation where curved cracks are represented locally

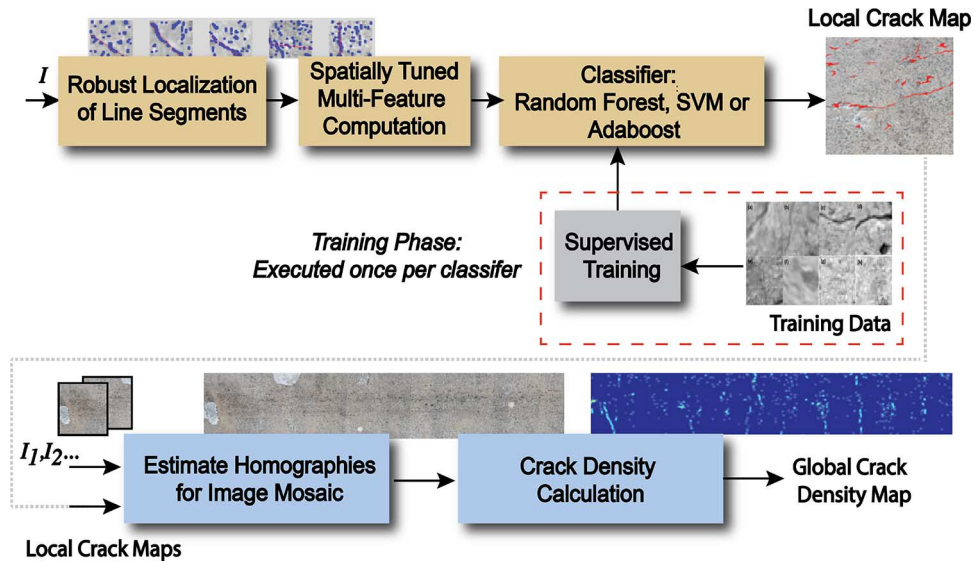


Fig. 4. Method overview. The STRUM (Spatially Tuned RobUst Multi-feature) classifier consists of three components: 1) a robust line segment detector; 2) spatially tuned multiple feature computation; and 3) a machine learning classifier. The resulting local crack maps over a bridge span are combined by forming an image mosaic where individual frames are aligned to a single coordinate frame and stitched for a composite image. A crack density map is computed providing a global view of the crack densities across the bridge.



Fig. 5. Seekur robot mounted with cameras used for image collection. The robot is remotely operated and data is sent to the base station located in a van at the end of the bridge-testing area.

with line segments. Line fits can also be replaced with a higher order curve fit, but lower order fits are typically more stable and reliable for small neighborhoods. The line segments are obtained for each block in the image and a machine learning classification is done to classify these segments into two classes: *crack* or *not-crack*. Training examples are provided that are manually labeled with the correct class, as shown in Fig. 6.

The key contribution of our method is the input to the standard classifier, i.e., the crack appearance vector computed as a spatially tuned multifeature vector. The appearance vector is constructed using components that each contribute a partial cue to the classification decision. We evaluate the performance of

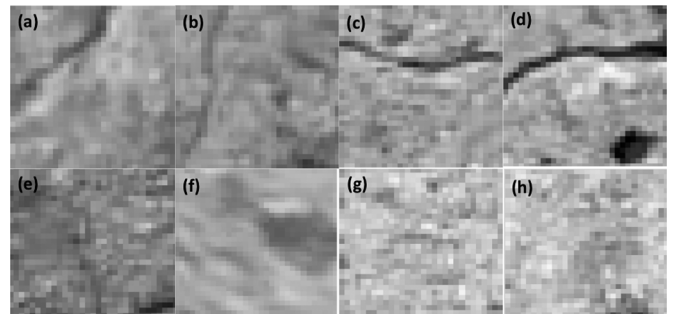


Fig. 6. Positive and negative training samples. (a)–(d) Shows 15×15 pixel image regions (blocks) with cracks. (e)–(h) Shows 15×15 pixel image regions without cracks. For our training and validation purposes, we construct a dataset of 2000 samples from two bridges, with equal number from each bridge, and equal number of positive and negative instances.

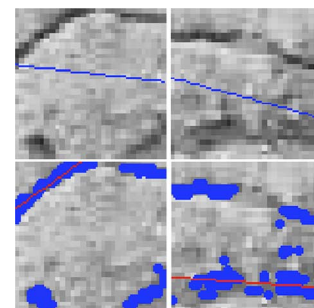


Fig. 7. Comparison between RANSAC and least square fit of curves. Red lines in (a) and (c) show the curves fit to the minimum intensity points using RANSAC. Blue lines in (b) and (d) show the curves fit to the minimum intensity points using least squares method that misses the cracks due to the presence outliers.

classification using combinations of the features and demonstrate that the multifeature appearance vector which integrates several weaker cues provides the best classifier. We investigate a suite of: 1) intensity based features; 2) gradient-based features;

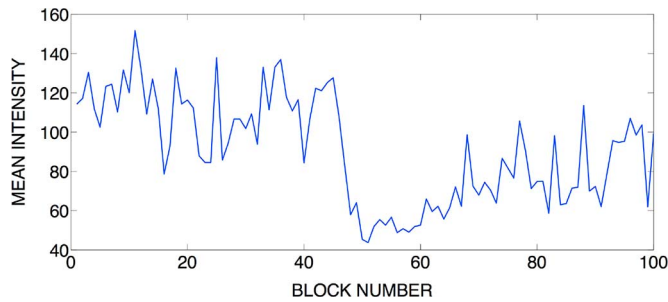


Fig. 8. An individual component of the crack appearance vector, μ_i , the mean pixel intensity along the line segments is less for crack regions (blocks 1–50) than non-crack regions (blocks 51–100). However, this feature alone would be insufficient to classify the crack.

and 3) scale-space features. Our experimental results show that combining these multiple features into one appearance vector provides optimal performance.

A. Intensity-Based and Gradient-Based Features

Spatially tuned features provide a quantitative description of appearance along the local line segment. As an example, the mean μ_i of the pixel intensity along the line segment is chosen as feature component because crack pixels are typically low intensity. However, that feature alone is not sufficient for high accuracy classification, as illustrated in Fig. 8 which shows the lower intensity trend for crack regions along with many violators of this trend. For very thin cracks, the darkness of the pixel intensities is not reliable. Also, a few dark pixels along the line segment in a non-crack region causes a low mean intensity. Multiple features that provide weak cues are combined in our method. Specifically, we use the following set of intensity-based features that are computed with pixels along the robustly detected line segment:

- Mean of intensity histogram (μ_i).
- Standard deviation of intensity (σ_i).
- Mean of gradient magnitudes (μ_g).
- Standard deviation of gradient magnitudes (σ_g)
- Ratio of the mean of intensity along the local line to the mean intensity in the local region (r_i).

The intensity standard deviation σ_i indicates that the crack segments have an approximate uniform intensity compared to non-crack segments. The component μ_g is used because the gradient magnitudes will be larger in the crack regions, and σ_g indicates that the gradient magnitude along a crack is expected to be more uniform than for line segments in non-crack regions. A discrete approximation of the derivative is used by a standard central difference filter. Finally, the feature component r_i provides a relative intensity measure of the line segment pixels compared to the background pixels and these photometric ratios have the advantage of being independent of global illumination. Each of these components provides a weak classification cue and the concatenation of all the components comprise the multifeature vector.

B. Scale-Space Features

The bridge deck images of interest have cracks of varying sizes, as thin as a millimeter to over a centimeter. Uniformity in the representation and processing of visual information over

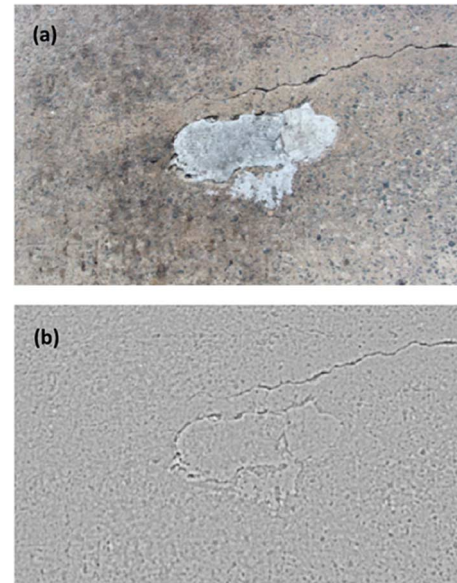


Fig. 9. (a) Original image. (b) Level-2 pyramid image upsampled to the same size as the original image. Laplace pyramid enhances edge features, which are more prominent at a characteristic scale.

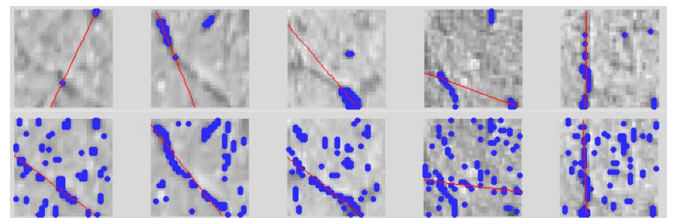


Fig. 10. Patches in the second row show the line fit to the points which have the lowest Laplacian pyramid values. Patches in the first row show the curves fit to the points of minimum intensity. These patches have very fine cracks. Curves fit to pixels having the lowest Laplacian pyramid values perform better than the curves fit to minimum intensity points.

multiple scales is an inherent property offered by visual systems. Laplacian pyramids are a classic coarse-to-fine strategy that assist in the search over scale space. Image structures, such as cracks, tend to have a particular salient scale. That is, they are most prominent at one level of the Laplace pyramid, as illustrated in Fig. 9. Laplacian pyramid images represent different spatial frequency bands so that level-0 contains image information in the highest spatial frequencies and the subsequent levels correspond to lower spatial frequency bands. The spatial tuning for these features is obtained by robust line fits to the minimum tenth percentile Laplacian pyramid values in the block, instead of pixel intensity values. The Laplacian pyramid value is the mean over pyramid levels 1, 2, and 3. As demonstrated in Fig. 10, thinner cracks can be detected when curves are fit to such points (scale-space extrema).

The scale-space features include the following spatially tuned Laplace pyramid features (computed along the local line segment):

- Maximum of Laplacian pyramid values across three levels (L_{\max}).
- Minimum of Laplacian pyramid values across three levels (L_{\min}).

- Mean of Laplacian pyramid level 1 values (μ_{L1}).
- Mean of Laplacian pyramid level 2 values (μ_{L2}).
- Mean of Laplacian pyramid level 3 values (μ_{L3}).

III. CLASSIFIER TRAINING AND TESTING

The crack appearance vector is used as input to machine learning classification. For statistical inference, classifiers are chosen based on empirical performance and we investigate the following classifiers: support vector machines (SVM) [5], adaboost [6], and random forest [7]. Features are used in different combinations to evaluate the performance of each of the three classifiers to yield the highest possible accuracy [25]. The performance on the training set is analyzed using a tenfold cross validation. Furthermore, a classification test that is *geographically mutually exclusive* is provided where the training data is obtained from one bridge and the test set is obtained from another bridge.

IV. GLOBAL VIEW: CRACK DENSITY MAPS

Crack location and characteristics are important to detect, evaluate and archive. Additionally, presentation of the crack detection results may be made to a human inspector. There is a challenge of communicating spatial patterns of cracks over the entire bridge deck. Subpixel image registration methods are used to create a large-scale mosaic from the robot scanner images. However, examining the full resolution mosaic annotated with cracks requires either wall-size monitors or lengthy scrolling across a large scale digital image. Downsampling the image to fit on a smaller window renders thin cracks invisible. We compute a simple crack density map where the pixel value is the crack density in the region. This computation can be slow since a spatial average must be computed for every pixel in the large mosaic. We employ integral images which is a computationally efficient technique useful when computing averages over windows for every pixel in the images [26]. The resultant crack density maps give a detailed overview about the surface degradation of the bridge deck.

For general camera motion, a 3×3 matrix H , called a homography, relates the pixel coordinates P_0 and P_1 in two images as

$$P_0 = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} P_1. \quad (1)$$

Homographies can be concatenated to relate points in reference frame to points in the current frame. The i th keypoint in the j th frame when transformed to the zeroth frame ${}^0P_{j,i}$ is related to the i th keypoint in the j th frame by

$${}^0P_{j,i} = {}^0H_j \times {}^jP_{j,i} \quad (2)$$

where j ($= 0, 1, 2, 3 \dots M$) is the frame number, i is the point pair index, and 0H_j is the concatenation of intermediate homographies between frame j and frame 0, given by

$${}^0H_j = {}^0H_1 \times {}^1H_2 \dots \times {}^{j-1}H_j. \quad (3)$$



Fig. 11. Stitched images before and after distortion suppression. (a) Stitched images obtained by minimizing the misalignment of matched points. (b) Stitched images with homography distortion removed by adding a distortion term to the optimization problem. There are eight images stitched together in this figure.

The problem with concatenation is an accumulation of errors over local frames. The best way to avoid this is a procedure called bundle adjustment [27]–[29] which computes all of the interframe homographies in a global optimization over all relevant images. However, this approach is very computationally intensive. Frame-to-frame alignment works well if the homography estimation is constrained so that the rotational components is small. We follow the approach of [30] to ensure undistorted homographies by encouraging the original rectangular shape of the image. This alignment is an optimization problem with an objective function that has two terms: reprojection error E_r to match points and distortion error E_d . The reprojection error is

$$E_r = \sum_{j=0}^M \sum_{i=1}^{N_j} ({}^0P_{j-1,i} - {}^0P_{j,i})^2 \quad (4)$$

where N_j is the number of keypoints in the j th frame and M is the number of frames. The distortion term E_d can be formulated as

$$E_d = \|H[1, 0, 0]^T - [1, 0, 0]^T\|^2 + \|H[0, 1, 0]^T - [0, 1, 0]^T\|^2. \quad (5)$$

Total error E_t is the sum of both terms given by

$$E_t = E_r + \alpha E_d. \quad (6)$$

Levenberg–Marquardt algorithm [31] is used for this nonlinear optimization step. Fig. 11 shows the image stitching results with and without the distortion term.

V. RESULTS

The experimental results use an image database collected by robotic scans of two bridge decks consisting of 100 images per bridge of 3.5 ft. \times 2.6 ft. concrete surface segments

TABLE I
COMPARISON OF SVM CLASSIFIER PERFORMANCE WITH DIFFERENT INTENSITY-BASED FEATURE VECTORS

Feature Vectors	Accuracy (%)	Sensitivity	Precision
μ_i, σ_i	87.2	0.78	0.844
μ_g	69	0.64	0.658
σ_g	68.4	0.628	0.646
μ_g, σ_g	75.6	0.624	0.69
r_i	84.7	0.788	0.796
$\mu_i, \mu_g, \sigma_g, r_i$ ($F1$)	89.5	0.796	0.869

TABLE II
CLASSIFIER PERFORMANCE WITH THE $F1$ FEATURE VECTOR

Classifier	Accuracy (%)	Sensitivity	Precision
SVM	89	0.832	0.849
Adaboost	91.2	0.74	0.974
Random Forest	92	0.784	0.916

TABLE III
COMPARISON OF SVM PERFORMANCE WITH DIFFERENT LAPLACE PYRAMID FEATURES AND THE COMBINED $F1, F2$ FEATURE VECTOR

Feature Vectors	Accuracy (%)	Sensitivity	Precision
L_{max}	53.2	0.38	0.55
L_{min}	58.2	0.404	0.63
L_{max}, L_{min}	60	0.488	0.629
$\mu_{L1}, \mu_{L2}, \mu_{L3}$	61	0.448	0.663
$L_{max}, L_{min}, \mu_1, \mu_2, \mu_3$ ($F2$)	64.6	0.44	0.748
$F1, F2$ (9×1 feature vector)	91.6	0.9	0.93

corresponding to 1920×1280 pixel image regions. The high-resolution images were captured using two Canon Rebel T3i DSLR cameras (with Canon EF-S 18–55 mm f/3.5–5.6 lenses) mounted on a Seekur robot, as shown in Fig. 1. Mechatronics and navigation of the robot system in described in [4].

For the experimental results, combinations of candidate features (intensity-based, gradient-based, and scale-space) are used. The validation and training sets are chosen from 1000 samples per bridge in the labeled database with equal number *crack* and *non-crack* samples. To quantify classifier performance, a standard tenfold cross validation is performed which varies the training and validation set over the entire dataset.

A. Classification With Intensity and Gradient-Based Features

Using the spatially tuned intensity-based features along the local line segments, as described in Section II, the blocks are classified and the performance metrics of the SVM classifier are shown in Table I. The ranking of features in order of their individual performance is r_i, μ_i, σ_g , and μ_g (in decreasing order of accuracy). $F1$, the combined feature vector using r_i, μ_i, σ_g , and μ_g performs best. We use the 4×1 feature vector $F1$ and evaluate the classifier performance on the validation set data, as shown in Table II. The combined intensity-based feature vector led to an increase in accuracy and precision of the classifiers. Random forests classifier has the highest accuracy and the adaboost classifier has the highest precision.

B. Classification Using Scale-Space Features

The SVM classifier performance for different combinations of Laplacian pyramid feature vectors described in Section II are shown in Table III. The performance of the three classifiers are compared in Table IV. Random forests has the highest accuracy,

TABLE IV
CLASSIFIER PERFORMANCE USING THE MULTIFEATURE VECTOR ($F1, F2$) A 9×1 FEATURE VECTOR AS DESCRIBED IN TABLES I AND III)

Classifier	Accuracy (%)	Sensitivity	Precision
SVM	91.6	0.9	0.93
Adaboost	93.7	0.904	0.89
Random Forest	94.7	0.9	0.911

TABLE V
CLASSIFIER PERFORMANCE WITH CALIFORNIA BRIDGE DATASET AS THE TRAINING SET AND THE VIRGINIA BRIDGE DATASET AS THE TEST SET WITH A 9×1 FEATURE VECTOR CONSISTING OF FEATURES $F1$ AND $F2$

Classifier	Accuracy (%)	Sensitivity	Precision
SVM	79.6	0.96	0.723
Adaboost	90.8	0.952	0.723
Random Forest	76.3	0.676	0.685

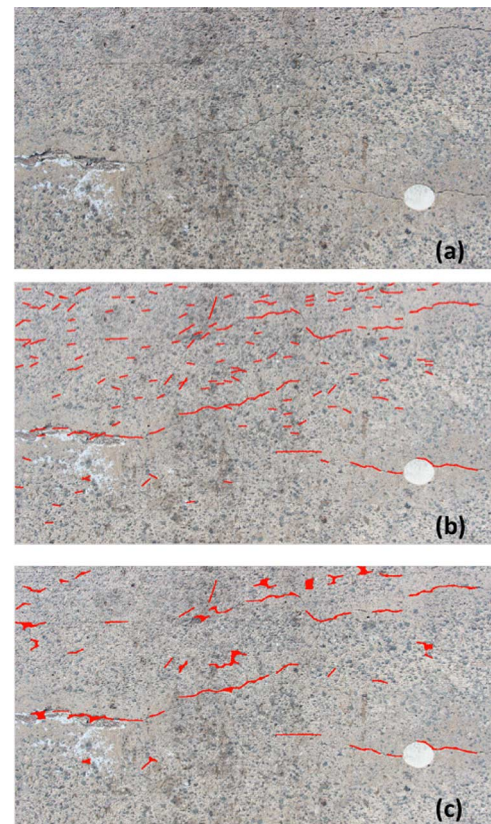


Fig. 12. Crack detection results. (a) Raw image from the Virginia bridge deck corresponding to a 2.6 ft \times 3.5 ft section on the bridge. (b) Image showing the detected cracks. (c) Morphological operations (closing and hole filling) remove small cracks.

while SVM and adaboost also perform well. A combination of the all features in a 9×1 vector results in an increase in accuracy for all three classification algorithms. The STRUM classifier therefore is defined with this multifeature vector as input.

C. Geographically Distinct Test Set

The classifier performance is evaluated using geographically distinct test data, i.e., from two different bridge datasets. For

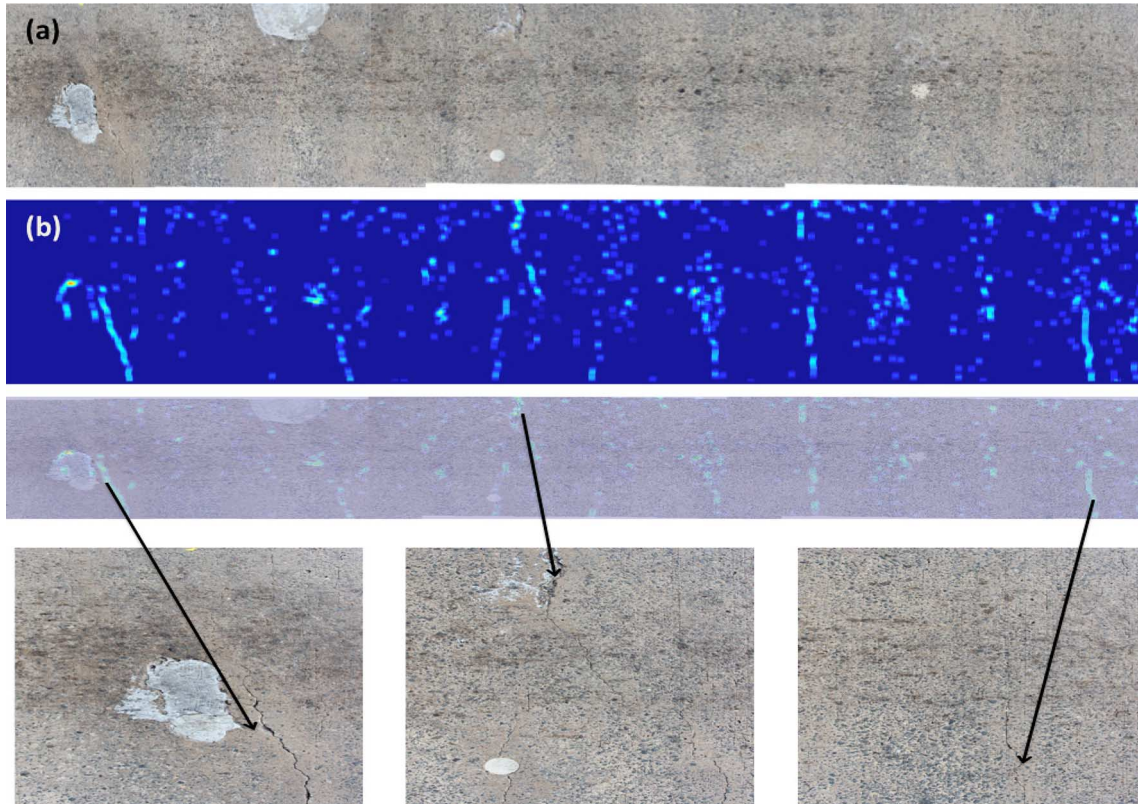


Fig. 13. Bridge deck mosaic and crack density map. (a) Mosaic of a 35 ft. \times 3.5 ft. section of the Virginia bridge deck. (b) Crack density map computed by averaging the number of cracks in a region. Notice how the cracks are not apparent in (a) because their width is subpixel in the scaled mosaic. But the crack density map gives a clear visual assessment of the pattern of fine cracks over the imaged span of the bridge. (Row 3) The crack density map in (b) is superimposed on the mosaic in (a). Three of the high crack-density regions have been shown zoomed for a clear depiction of the underlying cracks.

this purpose, we constructed training and test sets of images collected from bridges in both Haymarket, VA, USA, and Sacramento, CA, USA. The data from the Virginia bridge was collected under natural light conditions, while the data from the California bridge was collected at night under artificial lighting. For each of the two bridges, 1000 samples were labeled with equal positive and negative instances. The classifiers were evaluated with the multifeature vector $F1, F2$, combining the intensity-based and the Laplacian pyramid features. The classifier performance when trained on the California bridge-deck images and tested on the Virginia bridge-deck images is shown in Table V. The best performance was with the adaboost classifier. The STRUM classifier approach, as shown in Fig. 4, does not fix the choice of the machine learning classifier. The result shown suggests that the adaboost classifier provides a useful method for generalizing the trained model to novel bridge surfaces.

D. Crack Density Maps

Fig. 13 shows a bridge mosaic of a 35 ft. \times 3.5 ft. section of the Virginia bridge deck comprised of 12 individual images from the robot scan. The corresponding density map is obtained by running the STRUM classifier on the bridge mosaic. This color-map shows various levels of degradations indicated by the different colors where dark blue corresponds to region of low crack density and light blue corresponds to high crack density. The superposition of the crack map and the image mosaic is also

shown in Fig. 13. The global crack map can be used for quantitative analysis but also for visually assessing global crack patterns. For example, notice that an approximate periodic global pattern of cracking is detectable in Fig. 13.

E. Timing Discussion

On-site analysis enables knowledge inference from the large dataset collected by the robotic bridge scanning. The computation and measurement speed of our methods supports fast analysis in its current form. With a robot speed of 3 ft/second, the 35 ft. section shown in Fig. 13 is imaged in approximately 12 s. The individual image size is 2.6 ft. in the direction of motion and 3.5 ft. in the perpendicular direction. The robot takes images with approximately 40% overlap, i.e., the sample distance along the bridge span is approximately 1.5 ft. With a processor speed of 2.3 GHz, the total time for image collection, STRUM classification, and annotating the cracks on the 35 ft. section comprised of 12 images takes approximately 33 min, in a completely automated manner. The current implementation uses C++ and Matlab and can easily be optimized for significant speed gains. Time efficiency can be further improved with GPU parallelization and embedded vision hardware.

VI. CONCLUSION

The STRUM classifier for crack detection on bridge decks provides a method of inspection for use in on-site robotic scanning. Since automated scanning generates large image datasets,

automated analysis has clear utility in rapidly assessing bridge condition. Moreover, the results can be quantified, archived and compared over time. The methods of this work shows the first application of automated crack detection to robotic bridge scanning. The new algorithm uses a feature set that shows 90% accuracy on thousands of tests cracks. Additionally, geographically separate datasets have been provided for testing and training. A thorough evaluation of multiple features and multiple classifiers on real world data validates the methodology. A coherent spatial mosaic, along with the crack density map, serves as a new tool for inspectors to analyze bridge decks.

ACKNOWLEDGMENT

The authors would like to thank the Federal Highway Administration, U.S. Department of Transportation, for supporting this research as part of the Long-Term Bridge Performance Program. They also express their gratitude to K. Zhao, graduate student at Rutgers University, for his valuable contribution towards the deck-mosaicing application.

REFERENCES

- [1] S. Kruschwitz, C. Rascoe, R. Feldmann, and N. Gucunski, *Meeting LTBP Program Objectives Through Periodical Bridge Condition Monitoring by Nondestructive Evaluation*, ASCE, Structures Congress, 2009.
- [2] H. M. La, R. S. Lim, B. Basily, N. Gucunski, J. Yi, A. Maher, F. A. Romero, and H. Parvardeh, "Autonomous robotic system for high-efficiency non-destructive bridge deck inspection and evaluation," in *Proc. IEEE Int. Conf. Autom. Sci. Eng. (CASE)*, Aug. 2013, pp. 1053–1058.
- [3] B. Wan, T. McDaniel, and C. Foley, "What's causing cracking in new bridge decks?," Wisconsin Highway Research Program, 2010.
- [4] H. M. La, R. S. Lim, B. B. Basily, N. Gucunski, J. Yi, A. Maher, F. A. Romero, and H. Parvardeh, "Mechatronic systems design for an autonomous robotic system for high-efficiency bridge deck inspection and evaluation," *IEEE/ASME Trans. Mechatronics*, vol. 18, no. 6, pp. 1655–1664, Dec. 2013.
- [5] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.
- [6] Y. Freund and R. E. Schapire, "Experiments with a new boosting algorithm," in *Proc. Int. Workshop. Mach. Learn.*, 1996, pp. 148–156.
- [7] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.
- [8] T. Nishikawa, J. Yoshida, T. Sugiyama, and Y. Fujino, "Concrete crack detection by multiple sequential image filtering," *Comput.-Aided Civil Infrastructure Eng.*, vol. 27, no. 1, pp. 29–47, 2012.
- [9] Z. Zhu, S. German, and I. Brilakis, "Visual retrieval of concrete crack properties for automated post-earthquake structural safety evaluation," *Autom. Construction*, vol. 20, no. 7, pp. 874–883, 2011.
- [10] T. Yamaguchi and S. Hashimoto, "Fast crack detection method for large-size concrete surface images using percolation-based image processing," *Mach. Vision Appl.*, vol. 21, no. 5, pp. 797–809, 2010.
- [11] T. Yamaguchi, S. Nakamura, and S. Hashimoto, "An efficient crack detection method using percolation-based image processing," in *Proc. 3rd IEEE Conf. Ind. Electron. Appl.*, Jun. 2008, pp. 1875–1880.
- [12] X. Tong, J. Guo, Y. Ling, and Z. Yin, "A new image-based method for concrete bridge bottom crack detection," in *Proc. Int. Conf. Image Anal. Signal Process. (IASP)*, Oct. 2011, pp. 568–571.
- [13] H.-N. Nguyen, T.-Y. Kam, and P.-Y. Cheng, "A novel automatic concrete surface crack identification using isotropic undecimated wavelet transform," in *Proc. Int. Symp. Intell. Signal Process. Commun. Syst. (ISPACS)*, Nov. 2012, pp. 766–771.
- [14] R. Adhikari, O. Moselhi, and A. Bagchi, "Image-based retrieval of concrete crack properties for bridge inspection," *Autom. Construction*, vol. 39, pp. 180–194, 2014.

- [15] R. S. Lim, H. M. La, Z. Shan, and W. Sheng, "Developing a crack inspection robot for bridge maintenance," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2011, pp. 6288–6293.
- [16] P. Domingos, "A few useful things to know about machine learning," *Commun. ACM*, vol. 55, no. 10, pp. 78–87, Oct. 2012.
- [17] T. M. Mitchell, "The discipline of machine learning," Carnegie Mellon Univ., School of Comput. Sci., Mach. Learn. Dept., Tech. Rep. CMU-ML-06-108, Jul. 2006.
- [18] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York, NY, USA: Springer, 2006, vol. 1.
- [19] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. New York, NY, USA: Wiley, 2001.
- [20] T. M. Mitchell, *Machine Learning*. New York, NY, USA: McGraw-Hill, 1997.
- [21] T. M. Mitchell, "Does machine learning really work?," *Artif. Intell. Mag., Assoc. Adv. Arti. Intell.*, vol. 18, no. 3, pp. 11–20, 1997.
- [22] B. Lee, Y. Kim, S. Yi, and J. Kim, "Automated image processing technique for detecting and analyzing concrete surface cracks," *Structure Infrastructure Eng.*, vol. 9, no. 6, pp. 567–577, 2013.
- [23] M. Jahanshahi, S. Masri, C. Padgett, and G. Sukhatme, "An innovative methodology for detection and quantification of cracks through incorporation of depth perception," *Mach. Vision Appl.*, vol. 24, no. 2, pp. 227–241, 2013.
- [24] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, June 1981.
- [25] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The weka data mining software: An update," in *Proc. SIGKDD*, 2009, pp. 10–18.
- [26] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE Comput. Soc. Conf. Comput. Vision Pattern Recog. (CVPR)*, 2001, vol. 1, pp. I-511–I-518.
- [27] B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon, "Bundle adjustment—A modern synthesis," in *Vision Algorithms: Theory and Practice*, Berlin, Germany: Springer-Verlag, 2000, Lecture Notes in Computer Science, pp. 298–375.
- [28] C. Wu, S. Agarwal, B. Curless, and S. M. Seitz, "Multicore bundle adjustment," in *Proc. IEEE Conf. Comput. Vision Pattern Recog. (CVPR)*, 2011, pp. 3057–3064.
- [29] S. Agarwal, N. Snavely, S. M. Seitz, and R. Szeliski, "Bundle adjustment in the large," in *Computer Vision—ECCV*, 2010, pp. 29–42.
- [30] R. Garg and S. M. Seitz, "Dynamic mosaics," in *Proc. IEEE 2nd Int. Conf. 3D Imaging, Modeling, Process., Visualization, Transmission (3DIMPVT)*, 2012, pp. 65–72.
- [31] D. W. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters," *J. Soc. Indu. Appl. Math.*, vol. 11, no. 2, pp. 431–441, 1963.



Prateek Prasanna received the M.S. degree in electrical and computer engineering from Rutgers University, Piscataway, NJ, USA, in 2013, and the B.Tech. degree in electrical and electronics engineering from National Institute of Technology Calicut, India, in 2010. He is currently working towards the Ph.D. degree in biomedical engineering from Case Western Reserve University, Cleveland, OH, USA. He is a Graduate Research Assistant with the Center for Computational Imaging and Personal Diagnostics (CCIPD).

He was a Research Assistant with the Vision Laboratory, Department of Electrical and Computer Engineering, and the Center for Advanced Infrastructure and Transportation (CAIT), Rutgers University, Piscataway, NJ, USA. He was involved in a Long-Term Bridge Performance Program (LTBP) research project with the Federal Highway Administration (FHWA). He was a member of the CAIT team that developed the Robotics Assisted Bridge Inspection Tool (RABIT) for FHWA. His research interests include computer vision, machine learning, biomedical image analysis, and computer-aided diagnosis.



Kristin J. Dana received the B.S. degree from the Cooper Union University, New York, NY, USA, in 1990, the M.S. degree from the Massachusetts Institute of Technology, Cambridge, MA, USA, in 1992, and the Ph.D. degree from Columbia University, New York, NY, USA, in 1999.

She is an Associate Professor with the Department of Electrical and Computer Engineering, Rutgers, The State University of New Jersey. Her research interests in computer vision include computational photography, machine learning, illumination modeling, texture and reflectance, bioimaging, motion estimation, optical devices, optimization in vision, and applications of robotics. She is the inventor of the 'Texcam texture camera' for convenient measurement of reflectance and texture. She is also a member of the Rutgers Center for Cognitive Science and a member of the Graduate Faculty of the Computer Science Department. From 1992 to 1995, she was on the Research Staff at Sarnoff Corporation developing real-time motion estimation algorithms for applications in defense, biomedicine, and entertainment industries.

Dr. Dana is a recipient of the National Science Foundation Career Award (2001) for a program investigating surface science for vision and graphics and a team member recipient of the Charles Pankow Innovation Award in 2014 from the ASCE.



Nenad Gucunski received the B.S. degree in civil engineering from the University of Zagreb, Zagreb, Croatia, and the M.S. degree and the Ph.D. degree in civil engineering from the University of Michigan, Ann Arbor, MI, USA.

He is currently a Professor and Chairman of the Department of Civil and Environmental Engineering, Rutgers University, Piscataway, NJ, USA. He is also the Director of the Infrastructure Condition Monitoring Program at the Rutgers' Center for Advanced Infrastructure and Transportation (CAIT).

His research interests include NDT/NDE of transportation infrastructure. He is leading a number of important infrastructure-related research projects.

Dr. Gucunski is the PI of the National Institute of Standards and Technology-Technology Innovation Program (NIST-TIP) ANDERS project, the Strategic Highway Research Project 2 (SHRP 2), project on NDE for bridge decks, the Lead of the NDE Team for FHWA's Long-Term Bridge Performance (LTBP) Program, and the PI on several other projects. He is an active member of a number of societies and is serving as the Chair of the Geophysical Engineering Committee of the American Society of Civil Engineers (ASCE).



Basily B. Basily received the B.S. and M.S. degrees from Cairo University, Cairo, Egypt, in 1969 and 1972, respectively, and the Ph.D. degree from the University of Aston, Birmingham, U.K., in 1977, all in mechanical engineering.

He is currently a Research Professor with the Department of Civil and Environmental Engineering, Rutgers University, Piscataway, NJ, USA. He is also a Professor in the Department of Mechanical Design and Production, Cairo University, Giza, Egypt. He is the author of nine international patents

and 32 publications in his areas of interest. His research has been funded by the U.S. Navy, the National Science Foundation, the Department of Defense, the Department of Transportation, Honda Corporation, and other industrial companies. His research interests include experimental stress analysis, metal forming, and machine design.

Dr. Basily is a Member of the American Society of Engineering Education. He received the 2011 Thomas Alva Edison Patent Award from the New Jersey Commission on Science and Technology for his design of a novel folding process and associated machine that manufactures three-dimensional folded core sheet structures.



Hung M. La (M'09) received the B.S. and M.S. degrees in electrical engineering from Thai Nguyen University of Technology, Thai Nguyen, Vietnam, in 2001 and 2003, respectively, and the Ph.D. degree in electrical and computer engineering from Oklahoma State University, Stillwater, OK, USA, in 2011.

He is an Assistant Professor with the Department of Computer Science and Engineering, University of Nevada, Reno, NV, USA. During 2001–2007, he was a Lecturer at the Department of Electrical Engineering, Thai Nguyen University of Technology.

From 2011 to 2014, he was a Postdoc and then a Research Associate with the Center for Advanced Infrastructure and Transportation (CAIT), Rutgers University, Piscataway, NJ, USA. He was a key team member of the CAIT team that developed the Robotics Assisted Bridge Inspection Tool (RABIT) for the Federal Highway Administration (FHWA). He has been actively involved in research projects with the FHWA, the National Institute of Standards and Technology (NIST), the U.S. Department of Transportation (DoT), the Department of Defense (DoD), and the National Science Foundation (NSF). He has authored over 30 papers published in major journals, book chapters, and international conference proceedings. His current research interests include mobile robotic systems, mobile sensor networks, cooperative control, learning and sensing, and intelligent transportation systems.

Dr. La and his team received the 2014 American Society of Civil Engineers Charles Pankow Award for Innovation for the Robotics Assisted Bridge Inspection Tool (RABIT). He is the recipient of two Best Paper Awards and a Best Presentation Award at international conferences.



Ronny Salim Lim received the B.S. degree from Pelita Harapan University, Tangerang, Indonesia, in 2008, and the M.S. degree in electrical and computer engineering from Oklahoma State University, Stillwater, OK, USA, in 2011.

He is currently an Application Developer with the Center for Advanced Infrastructure and Transportation, Rutgers University, Piscataway, NJ, USA. His research interests include robotics and automation for civil infrastructure, intelligent transportation systems, and multiagent robotic control.



Hooman Parvardeh received the B.S. degree in computer science from Amirkabir University of Technology, Tehran, Iran, in 2008, and the M.S. degree in industrial engineering from Rutgers University, New Brunswick, NJ, USA, in 2011.

He is currently a Research Engineer with the Center of Advanced Infrastructure and Transportation, Rutgers University, Piscataway, NJ, USA. His research interests are nondestructive testing and nondestructive evaluation (NDE) of bridge decks along with bridge data management and bridge

management systems.