Computer-Aided Civil and Infrastructure Engineering **26** (2011) 368–380

Automated Object Identification Using Optical Video Cameras on Construction Sites

Seokho Chi

School of Urban Development, Queensland University of Technology, QLD 4001, Australia

&

Carlos H. Caldas*

Department of Civil, Architectural, and Environmental Engineering, University of Texas at Austin, TX 78712, USA

Abstract: *Visual recording devices such as video cameras, CCTVs, or webcams have been broadly used to facilitate work progress or safety monitoring on construction sites. Without human intervention, however, both real-time reasoning about captured scenes and interpretation of recorded images are challenging tasks. This article presents an exploratory method for automated object identification using standard video cameras on construction sites. The proposed method supports real-time detection and classification of mobile heavy equipment and workers. The background subtraction algorithm extracts motion pixels from an image sequence, the pixels are then grouped into regions to represent moving objects, and finally the regions are identified as a certain object using classifiers. For evaluating the method, the formulated computer-aided process was implemented on actual construction sites, and promising results were obtained. This article is expected to contribute to future applications of automated monitoring systems of work zone safety or productivity.*

1 INTRODUCTION

The demand for real-time site monitoring using video cameras has rapidly grown for applications on work progress control, productivity data collection, accident investigation, and collaborative communications in the construction industry (Leung et al., 2008; Abeid et al., 2003; Shih et al., 2006). The site monitoring systems integrated with wireless technologies and web platforms have contributed to improvements in communications among project team members by providing them with captured images or videos of a construction site and letting them discuss and exchange ideas about on-going activities. Through the systems, the team members can remotely observe work progress without visiting the sites, and they thus can share their ideas in real time without the delay that later site visits would entail. They are even able to playback recorded video at any time and in any place. However, human intervention for analyzing the results of such monitoring is time consuming because the time required to view the video is almost equal to that spent on the original on-site labor (Everett et al., 1998). Although compressed time-lapse video is able to shorten lengthy videos, different time intervals (frame rates) need to be applied to different operations for minimizing observation errors (Everett et al., 1998; Kang and Choi, 2005). Thus, automated interpretation of captured data would result in benefits to monitoring systems because it would save time and money.

This article primarily investigates a method for automated object identification using standard video cameras on heavy-equipment-intensive sites, those in which several pieces of heavy equipment are operating simultaneously. This exploratory method enables detecting and classifying both on-site heavy equipment machinery and workers. Although this article does not claim

*To whom correspondence should be addressed. E-mails: *caldas@mail.utexas.edu; toalexdad@gmail.com*.

DOI: 10.1111/j.1467-8667.2010.00690.x

that the proposed method resolves all automated site-monitoring issues, the method has distinct advantages in relieving project team members from manual examination of captured data, and furthermore, it would contribute to future applications of automated monitoring systems of work zone productivity or safety.

This article is organized into five sections. After this introduction, Section 2 offers a literature review of visual-based construction site monitoring. Section 3 describes a research process for automated object identification. Section 4 shows experimental results and analyzes the resulting performance of the monitoring. Section 5 concludes the article by suggesting avenues of further research.

2 BACKGROUND REVIEW

Studies on site monitoring using video cameras have been performed for many years in the construction industry. In an early study, Eldin and Egger (1990) proposed some benefits of the use of a video camera for improving productivity on construction projects. They presented a case study during which a camcorder system was used as a management tool to improve labor productivity. They collected data for selected work items, and the results showed measurable advantages in improving communications between management and labor, enhancing decision making about productivity problems, and providing records of construction activities for training, safety, performance evaluation, and possible legal disputes. Everett and Slocum (1993) examined problems with productivity and safety of crane operations because of misinterpretation of visual cues; crane operators highly rely on hand signals relayed among craftsmen. They introduced the CRANIUM, a video monitoring system designed to improve communications by providing images from a camera mounted on the crane boom to a television monitor in the crane cab. This mechanism allowed crane operators an improved view of what was happening on the ground. Similarly, Ruff (2003) mounted a video camera on the rear of sanding trucks to assist truck operators in monitoring blind areas around the equipment to prevent collisions with workers on foot or other objects. He emphasized the efficiency of video monitoring, which provided actual views of the areas behind the trucks. Abeid et al. (2003) developed a computer-aided monitoring system, PHOTO-NET II, for construction project management. Their system (1) recorded time-lapse digital videos of construction activities from on-site surveillance cameras, (2) communicated over the Internet with the cameras, and (3) linked the recorded footage to a database that contained CPM (Critical Path Method) schedule information. The authors promoted a contractual environment where schedule-related claims were discussed and resolved more easily based on factual and objective information available to all stakeholders. Shih et al. (2006) proposed a panoramic image database management system (PIDMS) to integrate construction-related records. They monitored a remote renovation site using a set of panoramic cameras and recorded panoramic images and videos for managing work progress and site resources. The system was able to increase the efficiency and effectiveness of the site supervision of work schedules, manpower, and inspections of material, machinery, and clean-up. Leung et al. (2008) presented a real-time construction site monitoring system integrating a wireless network, video cameras, and web-based collaborative software. Multiple users were able to access the system simultaneously, so project team members could view the scenes of the construction site in real time and discuss their ideas with each other as in a video conference. Experimental results from actual construction sites showed that the proposed system contributed to the creation of a collaborative environment that facilitated effective communication among team members.

The studies reviewed above proved the feasibility of video monitoring for active and effective project management and contributed to the improvement of productivity and safety on construction sites. However, such studies required extensive human intervention for analyzing the monitoring results, and this human intervention was time consuming because the time required to view the video was almost equal to the time spent on the on-site labor (Everett et al., 1998). Though lengthy videos can be compressed by only capturing images at predetermined intervals, determining how often to capture such images is difficult because the less frequently the images are captured, the more information is lost (Everett et al., 1998; Kang and Choi, 2005). Thus, the automated interpretation of captured data would result in benefits to monitoring systems by relieving project team members from having to examine all the video themselves.

3 AUTOMATED OBJECT IDENTIFICATION

Among many devices employed for object detection and identification studies such as three-dimensional imaging sensors (Chi et al., 2009), radio-frequency identification (RFID) technology (Chae, 2009), global positioning system (GPS) (Abderrahim et al., 2005), terrestrial laser scanners (Park et al., 2007), or light detection and ranging (LIDAR) technology (Cai and Rasdorf, 2008), this article primarily presents an exploratory method for automated object identification using standard video cameras. Much research has been conducted in the field of computer vision study to develop robust object identification algorithms. The algorithms on

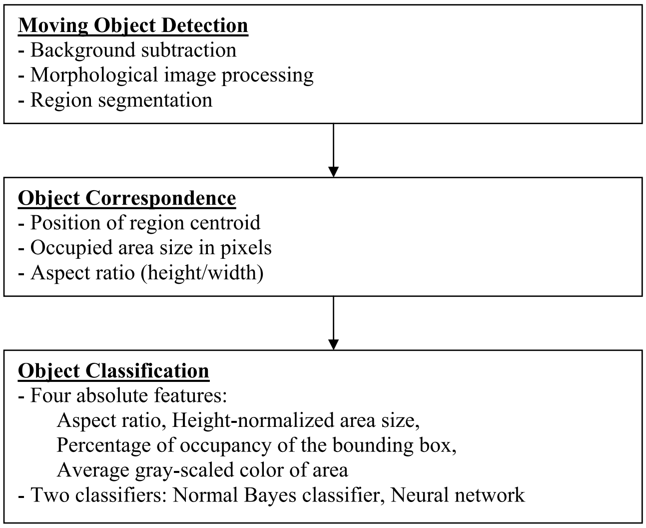


Fig. 1. Flowchart for automated object identification.

existing studies (Collins et al., 2000; Collins et al., 2001; Stauffer and Grimson, 2000; Javed and Shah, 2002; Bose and Grimson, 2003; Bose and Grimson, 2004; Bose, 2005; Hu et al., 2004; Lalonde et al., 2007; Shah et al., 2007) mainly follow three steps: (1) moving object detection, (2) object correspondence within an image sequence, and (3) object classification, all of which provide the functional requirements of the proposed identification method (Figure 1). From the video stream (an image sequence), the stationary background regions are first subtracted and the dynamic foreground regions of moving objects are extracted based on the foreground detection and segmentation algorithm. Incomplete foreground regions with holes and disconnections are then reconstructed by applying morphological image processing and the foreground pixels are grouped into one region using the connected component algorithm so that the individual target region can be extracted. The connected regions now represent moving objects, and their correspondences are found within an image sequence. The object information including object shape and appearance is then put into classifiers and finally objects in the image are identified using the classifiers. The remaining sections will cover the detailed explanation on each process.

3.1 Moving object detection

The first step, "moving object detection," aims at segmenting the regions of motion corresponding to moving objects from the rest of an image (Hu et al., 2004); that is, this process extracts information of moving objects from the construction environment (Stauffer and Grimson, 2000). For detecting moving regions, the background subtraction algorithm first needs to be implemented to eliminate the static regions (Stauffer and Grimson, 2000; Lalonde et al., 2007; Fisher et al., 2005; Radke et al., 2005). The goal of background subtraction is to determine which pixels belong in the background (the area of the scene behind an object or objects of interest—the static environment), and which pixels belong in the foreground (the area of the scene in which the object of interest lies—the moving environment). The value of the each pixel is continuously updated over time, and these data are then used to update the background and foreground identification.

Among the many innovative background subtraction algorithms, the foreground object detection and segmentation algorithm introduced by Li et al. (2003) was selected for this research because this algorithm was theoretically designed for detecting both gradual and sudden environmental changes. The authors in this article believed this characteristic offers advantages for implementation on actual construction sites in which a large number of uncertainties, including weather conditions and worker behavior, exist. The foreground object detection and segmentation algorithm distinguishes foreground from background by using color information of image pixels. When a pixel is associated with a stationary background object, its colors remain stable at the same place within an image sequence. Using this information about the distinction between stationary and moving objects, the algorithm first filters static pixels out.

Once the static pixels are removed, the algorithm then determines whether each moving pixel belongs to the background or the foreground by considering the color co-occurrences of sequential images. For example, pixels may be associated with moving background objects such as shaking trees. Even though the colors of the

(a) (b)

Fig. 2. Performance analysis of the background subtraction algorithms: (a) original images, (b) results of the foreground detection and segmentation algorithm.

pixels are different because the shaking trees are moving, the color co-occurrences are similar because similar changes statistically happen at the same place within an image sequence. The algorithm labels such changes as sudden changes. Other moving pixels, pixels not from moving background objects, build the foreground, and these pixels are labeled as gradual changes. In this way, the algorithm effectively identifies both gradual and sudden changes in the background, and finally, it detects the foreground from the image as a whole, which contains both stationary and moving background objects.

For analyzing algorithm performance, videos were recorded from actual construction sites, and the foreground detection and segmentation algorithm was implemented. For implementation, Intel Open Source Computer Vision Library (OpenCV) was employed and the function "cvCreateFGDStatModel" was utilized. Figure 2 shows an example of performance result of the subtraction algorithm. Figure 2a shows the original image for an operating backhoe. The background subtraction result using the algorithm is shown in Figure 2b.

The number of foreground objects (objects with white pixels in Figure 2b) of each image was recorded to determine algorithm accuracy. To do so, 120 sequential image frames were examined. Because only the backhoe was operating within the camera's field of view, the number of moving objects should be one theoretically. Table 1 shows results of accuracy checking. On the average, more than nine objects were detected from each image because the foreground region contained several holes and disconnected parts as shown in Figure 2b. This implementation result did not satisfy the authors' expectation. Thus, the authors needed to investigate post-image-processing techniques for improving modeling accuracy.

Table 1
Performance of morphological image processing

Total 120 image frame	*Before morphological image processing*	*After morphological image processing*
Average number of foreground objects	9.85	1.61

One such technique is morphological image processing. This technique offers advantages for reconstructing these kinds of incomplete regions by filling in holes or joining disconnected parts (Collins et al., 2000; Fisher et al., 2005). Morphological image processing is a technique that analyzes the spatial pattern of an image, and expands or reduces the image to improve image quality. There are two fundamental operations in morphological processing: erosion and dilation. Erosion processing is the operation of reducing a binary object with respect to the background. In general, 3×3 mask with the value "1," which represents an occupied white pixel, is used for binary images. If pixel values of an image are equal to the mask values, the erosion operation assigns the value "1" in its result image. However, if any one pixel value is different from the mask value, the operation assigns the value "0," which means a nonoccupied black pixel, in the result image. In this way, the erosion processing expands the background and reduces remaining parts of the image. This operation is good at filtering out any isolated object regions and separating incorrectly connected regions that are only connected with a thin section. Conversely, dilation processing is the operation of expanding an object with respect to the background. 3×3 mask with the value "0" is used. If pixel values of an image are equal to the mask values, the dilation operation assigns the value "0" in its result image. But, if any one pixel value is different from the mask value, the operation assigns the value "1" in the result image. As a result, the dilation processing reduces the background and expands remaining parts of the image. This operation has advantages because it can fill in any small holes in the object regions and join close but disconnected regions.

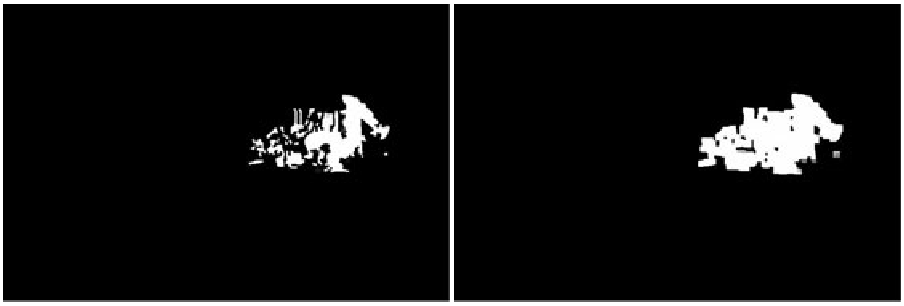


Fig. 3. Implementation results of morphological image processing: (a) results before morphological processing, (b) results after morphological processing.

Figure 3 shows an example of implementation result of the morphological image processing, and Table 1 summarizes how the processing improved the background subtraction performance. The images were dilated twice and eroded once.

Figure 3a shows the foreground image before morphological processing was applied. Figure 3b illustrates how morphological processing improved the subtraction process. In relief, the missing and disconnected regions of objects were effectively reconstructed after morphological processing. As shown in Table 1, less than two objects were detected from the foreground image on the average. Because of this improvement, the foreground detection and segmentation algorithm became more suitable for construction applications when combined with morphological image processing.

After background subtraction, the extracted foreground pixels need to be grouped into regions by the connected component algorithm for representing moving objects (Fisher et al., 2005; Samet and Tamminen, 1988). This algorithm joins neighboring pixels into regions by detecting unconnected regions in images. The algorithm assigns the first label for one foreground pixel and starts searching for occupied pixel positions from neighboring pixels in the foreground image (Shapiro and Stockman, 2001). If the occupied neighboring pixels exist, the algorithm assigns the current pixel label to them. The algorithm then keeps searching for another foreground pixel. If a newly found pixel does not have any label, the algorithm finds the neighbor's label and assigns it to the pixel. If there are no labeled neighbors, the algorithm assigns a new label to the pixel and its neighbors and continues searching. In this way, the foreground pixels are grouped and labeled, and these labels represent different objects in the image.

In summary, moving object detection includes two major processes: background subtraction and region segmentation. Background subtraction algorithms first generate a foreground image from moving pixels in the image, and then the incomplete foreground image is reconstructed using morphological image processing. Next, the pixels in the foreground image are grouped into regions based on the connected component algorithm, which represent moving objects in the image. The connected moving regions are now ready for object correspondence and classification.

3.2 Object correspondence

The second step, "object correspondence," aims at taking the segmented moving regions and matching them to find a corresponding region within an image sequence. For example, if there are three moving objects in the current image and only two objects in the previous one, the algorithm matches the two previously existing object regions within the two images and identifies the newly apparent object in the current image. Such object correspondence also offers advantages in terms of noise removal. For instance, if the size of the new object region is smaller than the predetermined empirical threshold size, the region can be regarded as noise. Similarly, if an object region suddenly appears or disappears without a smooth flow of motion, then it can be designated as noise.

To find the best corresponding region pair, regional characteristics such as shapes and appearances are generally compared. In this research, three object features were used for regional correspondence: position of region centroid, occupied area size in pixels, and aspect ratio (height/width). Position of region centroid was selected for checking temporal continuity with smooth flow of motion; if an object moves smoothly, two positions of object centroids in two sequential frames should be close each other. Conversely, if object positions are very far from each other, it is difficult to say that they are the same objects. Second, to find the size correspondence of an object, the occupied area size in pixels was chosen. If the sizes of two object regions are very different from each other within an image sequence, two objects might be the different objects. Finally, the

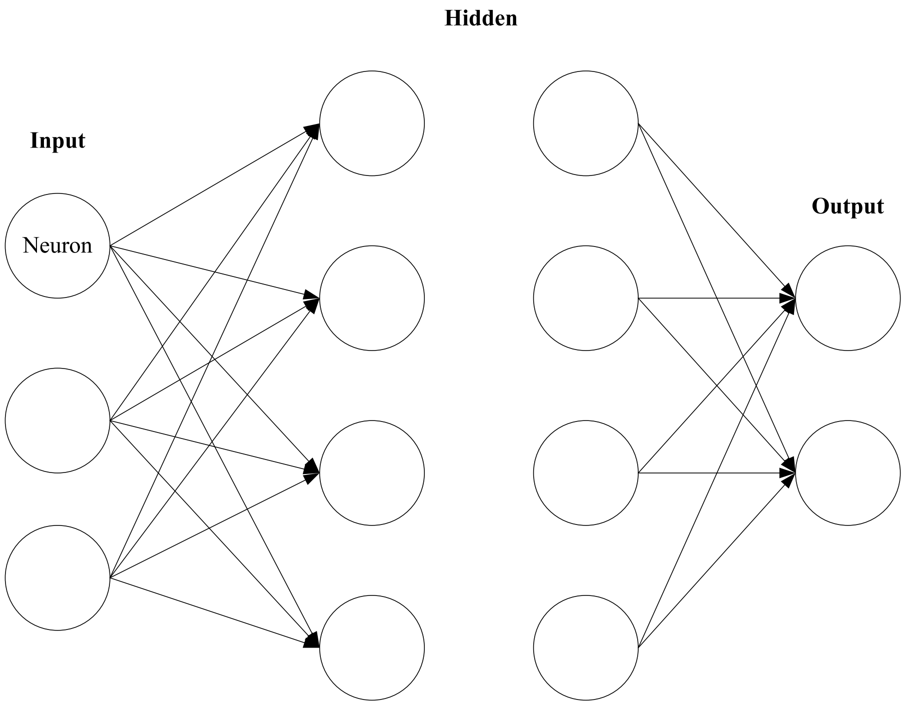


Fig. 4. Neural networks with multilayer perceptrons.

aspect ratio was picked for shape correspondence because the same object holds similar aspect ratio within an image sequence. The last two features were also compared with predetermined minimum and maximum size thresholds for noise removal which involved removing blobs below or over critical sizes.

For implementation, a simple approach based on a frame-to-frame matching cost function was used with the above three features: position of region centroid (p), occupied area size (s), and aspect ratio (r). The matching cost between a known object O_{known} in the previous image and a candidate moving region $O_{\text{candidate}}$ in the current image can be determined by the equation below.

$$C(O_{\text{known}}, O_{\text{candidate}}) = f(|p_{\text{known}} - p_{\text{candidate}}|, |s_{\text{known}} - s_{\text{candidate}}|, |r_{\text{known}} - r_{\text{candidate}}|) \quad (1)$$

The matching regions are close enough in cost space and a candidate moving region having the lowest matching cost is considered to be a potential match.

3.3 Object classification

The last step, "object classification," aims at classifying moving regions by using common shape, appearance, or motion (Stauffer and Grimson, 2000). The algorithm analyzes features of moving regions, inputs feature values into a classifier, and finally finds the best-matched object classes from training data. In this research, class categories include workers and several types of heavy equipment machinery. Four main features were selected for classification: aspect ratio (height/width), height-normalized area size (occupied area size in pixels/centroid of the area height), percentage of occupancy of the bounding box (number of occupied pixels within the bounding box/the bounding box size), and average gray-scaled color of the area. These features were selected because they were not affected by the apparent area size of the screen. An example of an image's size changing, for instance, would be if a worker approaches the camera position, causing the apparent area size to become larger within an image sequence. In this case, the area size at the nearer distance is larger than the size at the distance that is farther away. However, the aspect ratios of the worker (height/width) remain the same at both distances. Such absolute features are then inputted into a classifier to find the best-matched object class.

Two common classifiers (the normal Bayes and neural network classifiers) were employed in this research. The normal Bayes classifier (Fukunaga, 1990) is a simple classification method that assumes the whole data distribution displays a Gaussian mixture, one component per class. This method assumes that feature vectors from each class are normally distributed, but not necessarily independently distributed (cf. a naïve Bayes classifier is based on applying Bayes' theorem with strong

Fig. 5. Different three-dimensional postures of a backhoe captured from different viewpoints.

(a) (b)

Fig. 6. Background subtraction results: (a) original images, (b) foreground images.

independence assumptions (Tan et al., 2005). In simple terms, a naïve Bayes classifier assumes that a feature vector is unrelated to any other features.). From a sub set of the training data, the classification algorithm estimates mean values and covariance matrices between feature vectors for every class and then the estimated values are used to predict new unknown cases. In other words, the Bayes classifier computes the probability that one object region falls into a certain class, the classifier determines the class that has the largest probability with minimum errors among all the classes based on the principle of maximum likelihood estimation, and it classifies the object region as being of most likely class.

The other classifier, the neural network (Rojas, 1996), is a standard three-layer network which uses a back propagation algorithm for hierarchical learning. The multilayer perceptrons (MLP) (Rojas, 1996), the most commonly used type of neural networks which maps sets of input data onto a set of appropriate output, consists of an input layer, output layer, and one or more

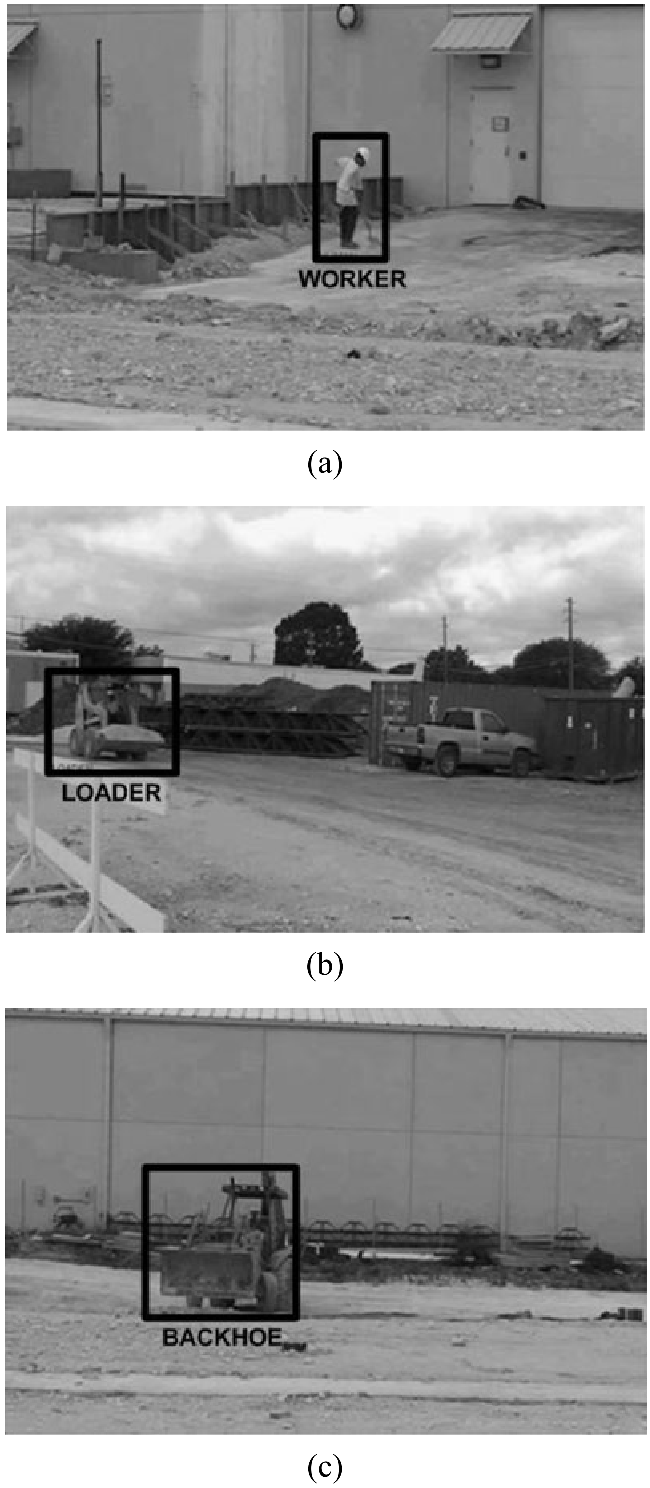


Fig. 7. Object classification: (a) worker, (b) loader, (c) backhoe.

hidden layers (Figure 4). Each layer of the MLP includes one or more neurons that are directionally linked with the neurons from the previous and the following layer.

Each neuron takes the output values from several neurons in the previous layer on input and passes the response to several neurons in the next layer. The values retrieved from the previous layer are added with certain weights, individually determined for each neuron, and the sum is transformed using the activation function that may also be different for different neurons. In more detail, the training data first builds network decision rules. The input layer is then constructed by the feature vectors obtained from a newly captured image. The input values are passed to the hidden layer and the outputs of the hidden layer are computed using the weights and the activation functions. The sizes of the input and output layers are determined from the training data (specifically, the number of input features and class types), and the number of hidden layers can be customized for optimizing classification results. The larger the number of hidden layers and the larger their sizes, the more potential network flexibility there is. In addition, the error on the training set could decrease slightly under these conditions.

4 EXPERIMENTAL RESULTS

Experiments on actual construction sites were conducted for testing and validating the proposed object identification method. The algorithm codes were written using the C++ programming language in Microsoft Visual C++ 6.0 (Microsoft Corporation, Redmond, WA). The author employed Intel Open Source Computer Vision Library (OpenCV) (Intel Corporation, Santa Clara, CA) for image processing and its visualization. A laptop computer (3.2 GHz Intel Pentium 4 CPU and 1.5 GB of RAM) was used for program implementation. The video data were collected using a Sony HDR-SR7 digital camcorder (Sony Electronics Inc., San Diego, CA).

For the first step of the experiments, a training data set was built regarding three objects: a skid steer loader, a backhoe, and a worker. The training data consisted of four variables (aspect ratio, height-normalized area size, percentage of occupancy of the bounding box, and average gray-scaled color of the area) obtained from multiple poses of the above-mentioned moving objects (Figure 5). A total of 750 images (250 for each individual object) were trained to build a final data set. The training data were then used to classify objects in testing images.

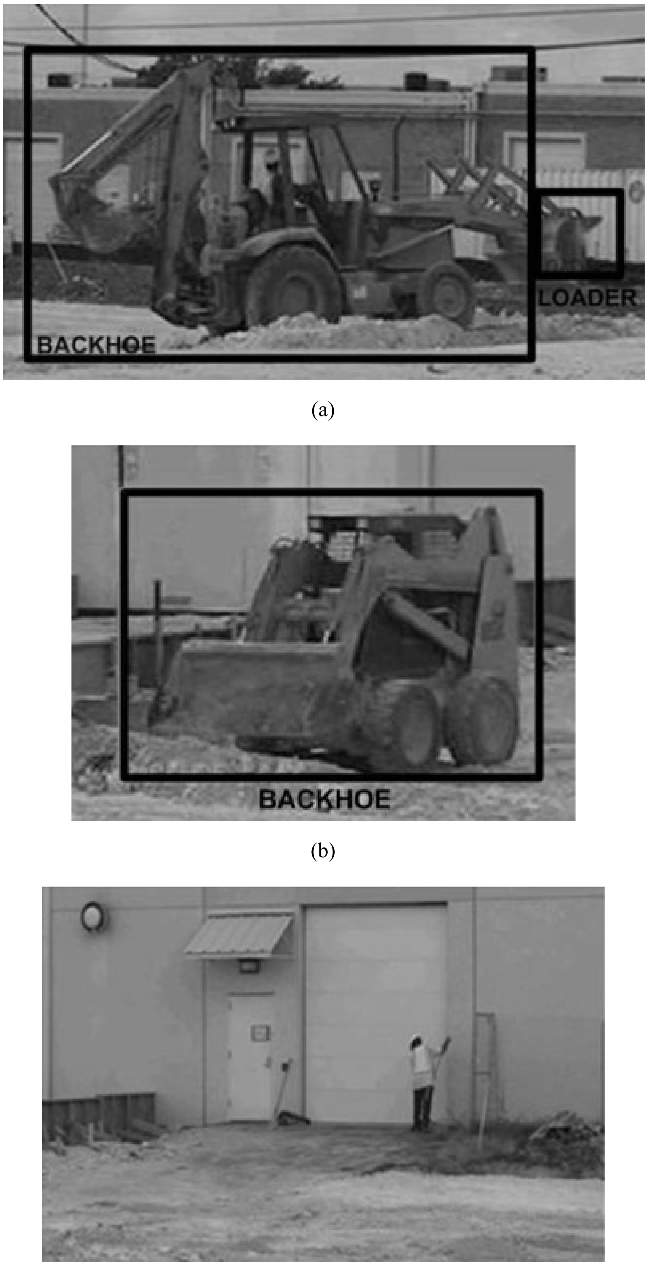
Once the training data set was built, the proposed object identification method was tested. First, the foreground regions were extracted based on the foreground detection and segmentation algorithm using morphological image processing. For implementation, OpenCV function "cvCreateFGDStatModel" and two morphological functions "cvErode" and "cvDilate" were utilized. Figure 6 shows examples of the implementation results using this background subtraction algorithm. The images were dilated twice and eroded once. Figure 6a shows the original images and Figure 6b shows the foreground regions obtained from the images. In the top images, regions for three objects, including a loader, a backhoe and a worker, were correctly extracted. In the bottom images, reliable regions for two workers were obtained.

After background subtraction, the foreground regions were grouped and corresponded with each object within an image sequence. To implement the connected component algorithm, OpenCV function "cvFloodFill," which fills a connected component with given color, was used. The calculated features of foreground regions were then inputted into one of two classifiers, which were constructed using the training data set, either into the normal Bayes classifier (Bayes) or the neural network. Using inputted variables, the classifier classified each object as a worker, a loader, or a backhoe. Figure 7 shows examples of the classification results for one moving object.

Table 2 summarizes the performance results of different classifiers. "RSE" stands for region segmentation errors due to incorrect foreground image from background subtraction (Figure 8a); "CE" means classification errors (Figure 8b); and "FDE" represents

Table 2
Performance analysis of object classification

Classifier	*Object*	*Total images*	*CE images*		*RSE images*	*FDE images*
Bayes	Worker	485	(L) 0 (0.00%)	(B) 0 (0.00%)	11 (2.27%)	2 (0.41%)
	Loader	347	(H) 1 (0.29%)	(B) 16 (4.61%)	6 (1.73%)	6 (1.73%)
	Backhoe	450	(H) 2 (0.44%)	(L) 24 (5.33%)	16 (3.56%)	1 (0.22%)
	Total	1,282	3.35%		2.57%	0.70%
Neural network	Worker	485	(L) 0 (0.00%)	(B) 0 (0.00%)	11 (2.27%)	2 (0.41%)
	Loader	347	(H) 0 (0.00%)	(B) 13 (3.75%)	6 (1.73%)	6 (1.73%)
	Backhoe	450	(H) 10 (2.22%)	(L) 27 (6.00%)	16 (3.56%)	1 (0.22%)
	Total	1,282	3.90%		2.57%	0.70%
Processing time			1,282 images per 427 sec = 3 frames/sec			



(a)

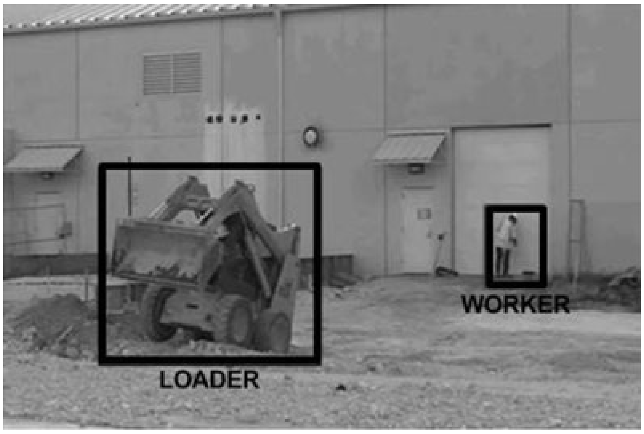
(b)

(c)

Fig. 8. Object classification error: (a) region segmentation error, (b) classification error, (c) foreground detection error.

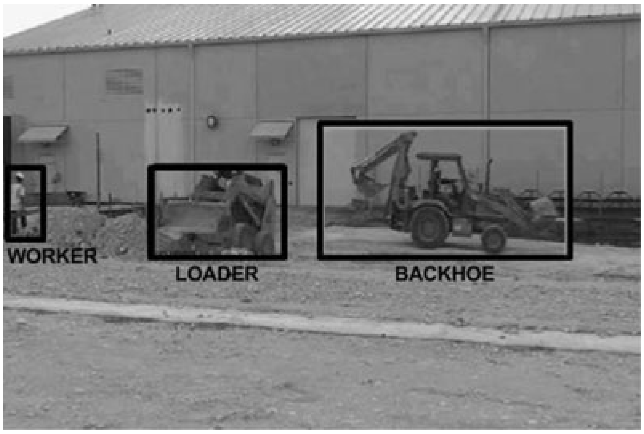
foreground detection errors resulting from sudden light changes or camera instability (Figure 8c). In Figure 8a, the body of the backhoe was broken into two pieces; one piece was classified as the backhoe, but the other small piece was classified as the loader. In Figure 8b, the loader was incorrectly classified as the backhoe. In Figure 8c, no object was detected because the whole image frame moved and was extracted as the foreground due to sudden light changes or camera instability due to wind.

A total of 1,282 images were analyzed and the classification method was processed three times per second, which satisfied near real-time applications. As shown in the table, the overall classification errors of two classifiers were under 4% (Bayes: 3.35% and NN: 3.90%), which showed an acceptable rate of accuracy for the



(a)

(b)



(c)

Fig. 9. Multiple-objects classification: (a) loader and worker, (b) worker and backhoe, (c) worker, loader, and backhoe.

algorithms. However, note that the classification accuracy for a worker was much more precise than that for a loader or a backhoe. Such results seemed to result from the shape similarities between a loader and a backhoe in certain object positions.

The proposed object identification method also showed successful experimental results with multiple-object classifications (Figure 9). Multiple sets of objects including a loader and a worker (Figure 9a), a worker and a backhoe (Figure 9b), and a worker, a loader, and a backhoe (Figure 9c) were correctly classified. However, the algorithms sometimes generated errors when two objects were occluded. Because of the two-dimensional-based line-of-sight issues with video cameras, the front object sometimes hid part of an object behind it; that is to say, the algorithm failed to separate two different objects when they aligned along the same line-of-sight as the camera position.

5 CONCLUSIONS AND RECOMMENDATIONS

This article highlights the demand for automated interpretation of video monitoring and proposes an exploratory method for automated object identification using video cameras in heavy-equipment-intensive construction sites. The experimental results showed that the proposed object identification method both successfully detected on-site objects and classified what they were in real time. This method would provide benefits on site monitoring with reduced human intervention, lower device cost than one of other sensors, and applicability to existing monitoring devices such as video cameras, CCTVs, and webcams.

Nevertheless, there are still limitations, improvement opportunities, and future research challenges to be addressed. The experimental results showed that in a few instances the proposed object detection algorithms split the whole body of an object into separate small pieces, and this separation produced classification errors. In addition, sudden light changes (illumination invariance) or camera instability due to wind caused less-precise detection outcomes. Moreover, region segmentation algorithms failed to separate occluded objects. Because the foreground detection process plays the most fundamental role through the entire process, further research should be conducted to improve its performance. In addition, more sophisticated techniques, such as occlusion reasoning for merging or splitting close objects (Malinovskiy et al., 2009), people counting algorithms to distinguish laborers working in a small group (Zhao and Nevatia, 2004), or material identification methods to track movement of construction materials (Brilakis and Soibelman, 2008), need to be investigated.

The experimental results showed that the proposed classification algorithms achieved an accuracy rate of approximately 96%. To further improve classification performance, more variables should be involved in the classification process. In addition, it would be effective to increase the number of training data because this would allow capturing more features of objects.

REFERENCES

Abderrahim, M., Garcia, E., Diez, R. & Balaguer, C. (2005), A mechatronics security system for the construction site, *Automation in Construction*, **14**(4), 460–66.

Abeid, J., Allouche, E., Arditi, D. & Hayman, M. (2003), PHOTO-NET II: a computer-based monitoring system applied to project management, *Automation in Construction*, **12**(5), 603–16.

Bose, B. (2005), Classifying tracked moving objects in outdoor urban scenes, Research Qualifying Examination Report, Electrical Engineering and Computer Science Department, Massachusetts Institute of Technology, Cambridge, MA.

Bose, B. & Grimson, E. (2003), Learning to use scene context for object classification in surveillance, in *Proceedings of the Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance (VS-PETS)*, Nice, France.

Bose, B. & Grimson, E. (2004), Improving object classification in far-field video, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Washington, DC.

Brilakis, I. K. & Soibelman, L. (2008), Shape-based retrieval of construction site photographs, *Journal of Computing in Civil Engineering*, **22**(1), 14–20.

Cai, H. & Rasdorf, W. (2008), Modeling road centerlines and predicting lengths in 3-D using LIDAR point cloud and planimetric road centerline data, *Computer-Aided Civil and Infrastructure Engineering*, **23**(3), 157–73.

Chae, S. (2009), Development of warning system for preventing collision accident on construction site, in *Proceedings of 2009 International Symposium on Automation and Robotics in Construction (ISARC)*, Austin, TX.

Chi, S., Caldas, C. H. & Kim, D. Y. (2009), A methodology for object identification and tracking in construction based on spatial modeling and image matching techniques, *Computer-Aided Civil and Infrastructure Engineering*, **24**(3), 199–211.

Collins, R. T., Lipton, A. J., Fujiyoshi, H. & Kanade, T. (2001), Algorithms for cooperative multi-sensor surveillance, *Proceedings of the IEEE*, **89**(10), 1456–77.

Collins, R. T., Lipton, A. J. & Kanade, T. (2000), A system for video surveillance and monitoring (VSAM), in *Proceedings of the American Nuclear Society (ANS) 8th International Topical Meeting on Robotics and Remote Systems*, Pittsburgh, PA.

Eldin, N. N. & Egger, S. (1990), Productivity improvement tool: camcorder, *Journal of Construction Engineering and Management*, **116**(1), 100–11.

Everett, J. G., Halkali, H. & Schlaff, T. G. (1998), Time-lapse video applications for construction project management, *Journal of Construction Engineering and Management*, **124**(3), 204–9.

Everett, J. G. & Slocum, A. H. (1993), CRANIUM: device for improving crane productivity and safety, *Journal of Construction Engineering and Management*, **119**(1), 23–39.

Fisher, R. B., Dawson-Howe, K., Fitzgibbon, A., Robertson, C. & Trucco, E. (2005), *Dictionary of Computer Vision and Image Processing*, John Wiley & Sons Ltd, Chichester, England.

Fukunaga, K. (1990), *Introduction to Statistical Pattern Recognition*, 2nd edn. Academic Press, NY.

Hu, W., Tan, T., Wang, L. & Maybank, S. (2004), A survey on visual surveillance of object motion and behaviors, *IEEE Transactions on Systems, Man, and Cybernetics – Part C: Applications and Reviews*, **34**(3), 334–52.

Javed, O. & Shah, M. (2002), Tracking and object classification for automated surveillance, in *Proceedings of the*

7th European Conference on Computer Vision (ECCV), Copenhagen, Denmark.

Kang, J. & Choi, J. W. (2005), Observation error of time-lapsed photos in construction operation monitoring, in *Proceedings of the 2005 ASCE International Conference on Computing in Civil Engineering*, Cancun, Mexico, 100–9.

Lalonde, M., Foucher, S., Gagnon, L., Pronovost, E., Derenne, M. & Janelle, A. (2007), A system to automatically track humans and vehicles with a PTZ camera, in *Proceedings of SPIE Defense and Security: Visual Information Processing XVI (SPIE #6575)*, Orlando, FL.

Leung, S., Mak, S. & Lee, B. L. P. (2008), Using a real-time integrated communication system to monitor the progress and quality of construction works, *Automation in Construction*, **17**(6), 749–57.

Li, L., Huang, W., Gu, I. Y. & Tian, Q. (2003), Foreground object detection from videos containing complex background, in *Proceedings of the 11th ACM International Conference on Multimedia*, Berkeley, CA, 2–10.

Malinovskiy, Y., Zheng, J. & Wang, Y. (2009), Model-free video detection and tracking of pedestrians and bicyclists, *Computer-Aided Civil and Infrastructure Engineering*, **24**(3), 157–68.

Park, H. S., Lee, H. M., Adeli, H. & Lee, I. (2007), A new approach for health monitoring of structures: terrestrial laser scanning, *Computer-Aided Civil and Infrastructure Engineering*, **22**(1), 19–30.

Radke, R. J., Andra, S., Al-Kofahi, O. & Roysam, B. (2005), Image change detection algorithms: a systematic survey, *IEEE Transactions on Image Processing*, **14**(3), 294–307.

Rojas, R. (1996), *Neural Networks—A Systematic Introduction*, Springer-Verlag, Berlin, NY.

Ruff, T. M. (2003), Evaluation of systems to monitor blind areas behind trucks used in road construction and maintenance: phase 1, Report of investigations: 9660, Pittsburgh Research Laboratory, National Institute for Occupational Safety and Health, U.S. Department of Health and Human Services, Pittsburgh, PA.

Samet, H. & Tamminen, M. (1988), Efficient component labeling of images of arbitrary dimension represented by linear bintrees, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **10**(4), 579–86.

Shah, M., Javed, O. & Shafique, K. (2007), Automated visual surveillance in realistic scenarios, *IEEE Multimedia*, **14**(1), 30–9.

Shapiro, L. & Stockman, G. (2001), *Computer Vision*, Prentice Hall, Englewood Cliffs, NJ, 69–73.

Shih, N., Lai, J. & Tsai, Y. (2006), The application of a panorama image database management systems (PIDMS) for information integration of construction sites, *Journal of Information Technology in Construction*, **11**, 641–54, http://www.itcon.org/2006/44/.

Stauffer, C. & Grimson, E. (2000), Learning patterns of activity using real-time tracking, *IEEE Transactions on Pattern Recognition and Machine Intelligence (TPAMI)*, **22**(8), 747–57.

Tan, P., Steinbach, M. & Kumar, V. (2005), *Introduction to Data Mining*, Pearson Education, Boston, MA.

Zhao, T. & Nevatia, R. (2004), Tracking multiple humans in complex situations, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **26**(9), 1208–21.