Designing Highly Reliable Fiducial Markers

Mark Fiala, *Member*, *IEEE*

Abstract—Fiducial markers are artificial landmarks added to a scene to facilitate locating point correspondences between images, or between images and a known model. Reliable fiducials solve the interest point detection and matching problems when adding markers is convenient. The proper design of fiducials and the associated computer vision algorithms to detect them can enable accurate pose detection for applications ranging from augmented reality, input devices for HCI, to robot navigation. Marker systems typically have two stages, hypothesis generation from unique image features and verification/identification. A set of criteria for high robustness and practical use are identified and then optimized to produce the *ARTag* fiducial marker system. An edge-based method robust to lighting and partial occlusion is used for the hypothesis stage, and a reliable digital coding system is used for the identification and verification stage. Using these design criteria large gains in performance are achieved by ARTag over conventional ad hoc designs.

Index Terms—Augmented reality, fiducial marker systems, computer vision.

1 Introduction

FIDUCIAL markers are useful in many situations where object recognition or pose determination is needed with a high reliability, where natural features are not present in sufficient quantity and uniqueness, and where it is not inconvenient to afix markers. Example applications include indoor augmented reality, handheld objects for user pose input, message tags to trigger a behavior, or generic pose determination in industrial settings. Despite future improvement in markerless computer vision, there will probably always be applications such as featureless indoor scenes where adding markers is superior.

Marker design is often performed ad hoc due to an absence in the literature of adequate analytical approaches. With proper design markers can be made to be very reliable and outlier detections or matches can be reduced to occurring with very low probabilities. The *ARTag* fiducial marker system [1], [2] was designed with the approaches detailed herein. Specifically, digital coding techniques for the *verification and identification* stage are described. Several augmented reality software development kits use ARTag for reliable pose determination in cluttered scenes despite uncontrolled lighting.

A fiducial marker system consists of some unique patterns along with the algorithms necessary to locate their projection in camera images. Such a system should reliably report one or more points per marker when seen in the image. The patterns should be distinct enough not to be confused with the environment. Ideally, the system should have a library of many unique markers that can be distinguished one from another. The image processing should be robust enough to find the markers in situations of uncontrolled lighting, image noise and blurring, unknown scale, and partial occlusion. Preferably, the markers should be passive (not requiring electrical power) planar patterns for convenient printing and mounting and should be detectable with a minimum of required image pixels to maximize the range of usage.

• *The author is with the Department of Computer Science, Ryerson University, 350 Victoria Street, Toronto, ON, Canada M5B 2K3. E-mail: mark.fiala@ryerson.ca.*

Manuscript received 25 Feb. 2008; revised 13 Oct. 2008; accepted 5 July 2009; published online 24 July 2009.
Recommended for acceptance by Y. Sato.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-2008-02-0118.
Digital Object Identifier no. 10.1109/TPAMI.2009.146.

0162-8828/10/$26.00 © 2010 IEEE Published by the IEEE Computer Society

The most common markers used are constellations of circular dots or colored markings. In recent years, many augmented reality (AR) projects have used *ARToolkit* [3] marker, which consist of a square outline with an internal pattern identified by correlation. The *ARTag* marker system is a more recent system gaining popularity in AR projects due to its improved performance.

A fiducial marker system is designed to solve the following problem: Given an input image (either a static image or a frame from a video stream), provide a list of markers found in the image. Fig. 1 shows ARTag patterns being recognized in an image.

This extracted information can be used in different ways, the simplest being to use only the marker ID to trigger some behavior dependent only on the presence of a marker such as logging into a kiosk. The image position information can also be used, which makes it possible to use a marker or array or markers as a handheld computer mouse pointer. Finding pose from one or more markers finds use in applications such as augmented reality where the camera pose is used to align a virtual camera with the real one to overlay virtual objects (Fig. 1b).

The paper is organized as follows: Criteria are identified to evaluate a marker system, a number of bar code, and planar marker systems are introduced. The ARTag marker system and its design according to the criteria is described, a general design methodology for digital marker codes is given, and, finally, some applications are presented that are only possible with a highly reliable system created using the approaches given herein.

2 Measuring Fiducial Marker System Performance

A fiducial marker system can be rated by how reliably it finds image points matching physical points on the markers. Failure modes typically include poor tolerance to lighting conditions and falsely reporting objects similar in appearance to markers in cluttered scenes. A nonrobust marker system will provide such false outlier measurements, which may complicate a system design. The shortcomings of a tracking system are readily apparent in Augmented Reality applications when augmentations do not register well with real objects, if they flicker or vibrate, or if the wrong virtual object appears.

The robustness and usefulness of a fiducial marker system can be characterized by some numerical metrics and some careful qualitative observations. The former, if properly stated, give quantities to optimize. Eleven practical evaluation criteria, which other fiducial marker systems do not fully address;

1. the *false positive* rate,
2. the *intermarker confusion* rate,
3. the *false negative* rate,
4. the *minimal marker size*,
5. the *vertex jitter* characteristics,
6. the *marker library size*,
7. *immunity to lighting conditions*,
8. *immunity to occlusion*,
9. *perspective support*,
10. *immunity to photometric calibration*, and
11. the *speed performance*.

All but number seven or eight can be determined theoretically or with experimentation. The failure to properly address any of these 11 criteria greatly reduces the usability of a marker system.

The *false positive* rate is the rate of falsely reporting the presence of a marker when none is present. The *intermarker confusion* rate is the rate of when the wrong id is reported, i.e., one marker was mistaken for another. The *false negative* rate is the probability that a marker is present in an image but not reported. Another metric is criterion 4, the *minimal marker size*, which is the size in pixels required for reliable detection. The smaller the marker needs to be

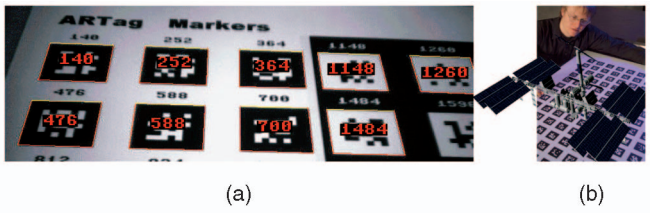

Fig. 1. ARTag markers detected (a) in an image and (b) augmented reality application. (a) Overlaid red border and ID number show automatic detection. (b) Camera pose is calculated using markers to correctly overlay space station model in augmented reality demo.

Fig. 2. A standard one-dimensional bar code and some two-dimensional bar code systems. The standard bar code and PDF417 are intended for scanning lasers. Data Matrix, Maxicode, and QR are suitable for conveying information when held in front of a camera but are not suitable for a fiducial.

in the image, the larger the usable range of distance from camera to markers. The *vertex jitter*, criterion 5, is the noise in the marker corner positions and affects the stability of an augmented object. The *library size* is the number of unique markers that the system can support; a large library is needed to enable applications such as the instrumented room application described in Section 5. *Immunity to lighting conditions* and *immunity to occlusion* are important because a robust fiducial marker system must recognize the marker patterns despite the ambient lighting and partial covering of the pattern. If perspective distortion is expected in the imagery, as with commonly available Webcams and video cameras with a wide field of view, *perspective support* is necessary, meaning that, for planar markers, at least four salient points must exist on the marker. The *immunity to photometric calibration* refers to whether the marker material and camera light gathering optical properties need measurement. Last is the practical issue of speed performance. In order to be useful for AR, a vision-based fiducial marker tracking system must function in real time with common low cost computing power.

ARTag was designed to address all of these issues. It consists of a library of patterns with a square border and unique interior digital signature, along with the detection algorithms to find them in imagery. The false positive rate is estimated to be <0.0039 percent of the quadrilaterals found in an image. The intermarker confusion rate depends on the image noise but, as will be shown later with Hamming distance analysis, is minimal. The markers can be seen with a minimal marker size of 13 pixels wide. Experiments with several cameras report a vertex jitter of a standard deviation of 0.03 pixels and a maximum of 0.09 pixels. The library size is 1,001 markers, or 2,002 depending on configuration. It functions despite large lighting changes, even across the same image. Markers can be partially occluded on several sides with occlusions reaching into the data area. ARTag markers are square, thus providing four corner points necessary for perspective support. The processing speed depends on the platform and the number of markers in the image but is typically on the order of 10-50 ms, fast enough for near real-time performance.

3 Planar Marker Systems

Some existing systems and their two main stages of processing are described. Discussed first are what are *not* fiducial marker systems: bar codes.

3.1 Bar Codes

Encoding information on a planar surface for detection with optical techniques has long been in ubiquitous use in the form of standard bar codes, usually read by laser scanners. Two-dimensional bar codes were developed to be read by a passive camera instead of an active laser. They are intended to carry information, not to localize, as is needed as a fiducial for applications such as augmented reality. Common bar codes are also shown in Fig. 2.

DataMatrix [4] , Maxicode, and QR all have a common thread of encoding data using binary values for reflectance. The pattern is typically bitonal reducing the decision made per pixel to a binary decision. This reduces the lighting and camera sensitivity requirement and removes the need for linearization of the signal (i.e., no attempts are made to identify shades of gray). In general DataMatrix, Maxicode, and QR are useful for encoding information but are not as useful for fiducial marker systems for two reasons. First, they are not intended for, and won't function well, in situations with a large field of view which introduces perspective distortion and, when detected, they do not provide enough image points for 3D pose calculation. DataMatrix can only adjust for affine warping by using the "L" shaped locator, i.e., with three points instead of the four required for perspective support. Second, they typically require a large area in the image, limiting the range at which the markers can be used. However, the lessons of bitonality and robust digital methods can be applied to fiducial design.

3.2 Fiducial Marker Systems

One approach is to design a marker with a lower information content than a 2D bar code. The extreme example is to use dots or discs and find only their center; a 3D version is a set of spheres or LEDs (Light Emitting Diodes) identified by a unique relative position relationship. In these systems, either the markers are active beacons (LEDs) or spheres covered with retroreflective material. Several manufacturers produce such systems for 3D tracking.[1] In this case, the configuration of the LEDs or spheres is a type of large 3D marker, the usage is limited to application domains such as indoor tracking for VR, and the equipment is specialized and limited to specially equipped labs.

A related idea is to use the planar equivalent: flat circular dots. Flat dots or checker-board patterns are typically used in camera calibration systems and applications with uncluttered scenes. Matching often needs to be confirmed manually to remove outliers. Dots can be expanded to have multiple IDs by encoding a ring of segments around a circular dot [5], [6]. Naimark and Foxlin describe a system encoding data inside the circular boundary [7]. Cho et al. [8] use concentric differently colored rings. The circular systems can provide only a single point with a high degree of accuracy; the plane a circular marker lies upon is more inaccurate to determine. This limits the pose accuracy possible from a single marker.

More unique information must be added to provide sufficient uniqueness and a larger marker library. Instead of attempting to add ID info to circular dots or open regions, one can add ID info to triangular or quadrilateral shapes. Quadrilateral shapes provide four salient points, which is useful both for computing pose from a single marker and for decoding the markers.

It is useful at this point to introduce the two stage model before discussing other marker systems. Fiducial marker systems typically function in two stages; hypothesis generation and verification/identification.

3.2.1 *First Stage: Hypothesis Generation from Unique Feature(s)*

To identify a planar pattern under perspective projection a homography between the pattern and the image plane is needed.

1. Some examples: Vicon, ART, Aicon, and 3rd Tech (UNC's HiBall).

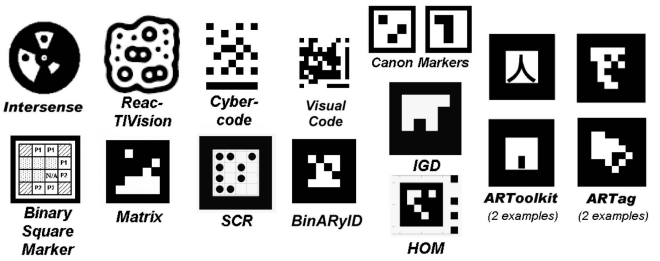

Fig. 3. Several fiducial marker systems.

If the perspective projection is approximated by an affine one, six parameters are still necessary to test the pattern. The space of possible homographies or affine transforms is too large to test an image for valid patterns by exhaustive search. Therefore, a first stage must find candidate regions by locating one or more relatively unique features of the marker. A geometric shape or shapes, such as a dot, bar, ellipse, triangle, square, etc., provides an anchor to form a marker detection hypothesis.

The result of using one or more features in the first stage of detection is a list of regions with homography or affine parameters presented to the second stage, which checks if the region is indeed a marker or just an object in the environment. With the aforementioned coded dot systems, an affine warp can be calculated on the ellipse's major and minor axes. Fig. 3 shows several markers systems, all of which assume planarity and use an affine or homography mapping for analysis by the second stage of verification/identification.

Several systems use *blob detection*, finding connected components of interest in a binary image formed by thresholding, to find unique features for hypothesis generation. The perimeter is examined for four straight line segments in systems using a quadrilateral as a unique feature such as ARToolkit. The second order moment statistics are used to filter blobs to identify long bar elements in *Cybercode* [9] and *Visual Codes* [10].

Matrix [11], *ARToolkit*, *ARToolkit Plus* [12], *BinARyID*, and *ARTag* all use square markers and their projection in the camera image under perspective projection is a quadrilateral. These five systems are known to first find the quadrilateral feature. Details are not available on the other square markers shown in Fig. 3 (the Canon markers and the IGD, HOM, and SCR [13] markers from the ARVIKA project [14]), but one can deduce that they must follow this same first step.

Klinker et al. [15], [16] use rectangular black rectangles on a white background as AR landmarks. These are detected by finding matching light/dark and dark/light transitions along image rows and columns, removing the need for an intensity threshold.

ARToolkit [3] is probably the best known system described in this paper, and it is widely used by AR and Human Computer Interaction (HCI) systems due to its available source code. Thresholding, connectivity, and perimeter analyses are used to find quadrilaterals whose corners are used to create a marker hypothesis with a homography mapping.

Almost all marker systems for which details are known use thresholding, connectivity, and blob analyses (perimeter or second moment analysis). The use of thresholding and connectivity introduces error modes shown in Section 4.1, specifically the choice of threshold amidst uncontrolled lighting and imperfections in the binary connected region's outline in the presence of occlusion.

How well the first stage is implemented affects the false negative detection rate (for example, if a threshold is incorrect markers will be missed), minimal marker size, vertex jitter characteristics (how the feature points are calculated), and especially immunity to lighting conditions and immunity to occlusion. If the hypothesis mapping is a homography, then the marker system has perspective support, a system using only affine mappings will have trouble with larger markers and cameras with a small focal length (such as most commodity Webcams or cameras built into portable devices).

3.2.2 Second Stage: Verification/Identification

When candidate image regions have been detected by the first stage, the second stage of *verification and identification* is performed to determine if the regions actually correspond to the fiducial markers or are just similar looking objects in the environment. Despite the practical benefits of the edge-based first stage, the novelty of ARTag lies more with this second stage. A binary yes or no decision is made on the region to determine if it is a marker (*verification*). If the marker library has more than one marker in it, then the *identification* step decides its identity within the library. How the markers and detection algorithms are designed will affect the false positive, intermarker confusion, and false negative rates, as well as the minimal marker size, and marker library size.

In ARToolkit, the homography mapping for each hypothesis is used to sample an N by N grid of points within the quadrilateral border. The normalized vector dot product between this and 12 stored prototypes for each marker is performed; a marker is reported if the highest is above a threshold. Prototype images have to be captured for each marker with the same camera and lighting, and this threshold has to be finely tuned to try balance the false positive and false negative rates. Overall usage is not scene independent. Owen et al. [17] proposes an ARToolkit library based on spatial frequency components to extend the library to on the order of 30 markers under controlled circumstances. Despite the apparent high dimensionality of the pattern space, it appears difficult to get a large and reliable marker library with correlation methods.

Following the trend in communication systems which have moved from analog to digital systems, fiducial markers can have symbolic codes in the pattern and use robust digital coding techniques to achieve higher performance. To achieve this, the pattern is divided into several *cells*, and each cell is decoded into a symbol. The symbols can then be processed to reach a decision on a potential marker's validity and identity.

A key design decision is how to encode information in the cells. Applying a single gray scale or color shade to a cell permits the symbol extraction to work down to a theoretical minimum of one cell per image pixel, thus improving (reducing) criterion 4, minimum marker size. Due to the lower spatial resolution of the color information and the small number of distinct colors that can be identified under uncontrolled printing, camera, and lighting circumstances, it can be argued [9] that the increase in symbol coding for a cell is offset by the reduced effective resolution using color.

Similarly, to obviate the need for photometric calibration by assuming a monotonic response between printed and observed brightness for elements in close proximity, cells can simply be bitonal (shaded either black or white). This lends itself to a binary symbol scheme of **0** and **1**. Of the systems listed, it is known that digital symbols (binary bits in a bitonal marker) are used in the Intersense, ARToolkit Plus, and BinARyID systems. The Intersense marker simply uses the 15 encoded bits as a 15-bit ID, while ARToolkit Plus repeats a 9-bit ID four times in its 36 bits. A user of BinARyID creates a library of patterns of incrementing 16-bit numbers that do not have rotational symmetry. Recently, Rice et al. [18] addressed the trade-off of data size versus detectability in their marker simulator where users can specify the number of cells.

Historically, Rekimoto's *Matrix* was published in 1998, *Artoolkit* was created in 1999, and the IGD, HOM, SCR markers were created during the ARVIKA project from 1999 to 2003 but are not available for public use. The Intersense system, created for a commercial positioning system, was published in 2002. ARToolkit became the most popular system in the academic community; its

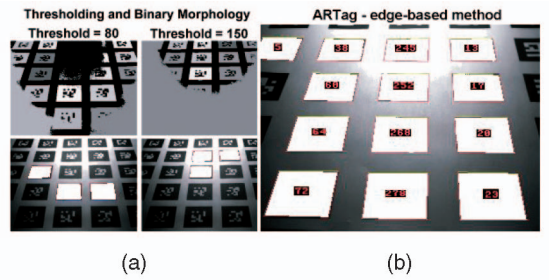

Fig. 4. (a) Images show only three detected, each using thresholding. (b) Image shows that with identical lighting, all the markers are found when using the edge-based method.

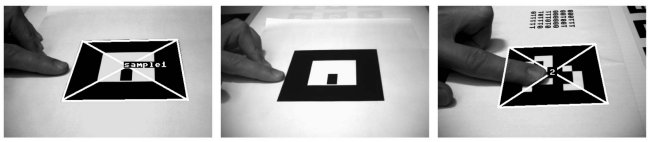

Fig. 5. Marker detection with occlusion. ARToolkit performance (a) and (b) shows how detection is lost with even small occlusion of the boundary. (c) Shows how ARTag markers can be recognized even despite large occlusions.

algorithms were efficiently optimized for the commonly available processing power of the day. ARTag, the system described in this paper, was created in 2004 to take advantage of improved processing power and to address ARToolkit's shortcomings. Its success spawned the creation of ARToolkit Plus and BinARyID; however, both are built upon ARToolkit's unique feature detection stage and share the same sensitivity to lighting and occlusion. ARToolkit, ARTag, and ARToolkit Plus are compared in [19]. Comparisons to ARTag, and the weaknesses of ad hoc digital coding schemes are shown in Section 4.4.

4 ARTag

The ARTag fiducial marker system [1] was created using the criteria in Section 2. For the unique feature, a square border was adopted to provide perspective support (criterion 9) and minimal pose jitter (criterion 5) [2] by using the outmost points, the corners. The outer periphery provides the greatest number of pixels relative to the marker area to maximize the accuracy of line equations formed from the border sides. These border line segments are intersected to find the marker corners to a subpixel resolution. Unlike other systems, this border is detected by an edge detection approach to remove the need for a threshold and to allow the markers to be detected despite unknown lighting, partial occlusion, and unknown photometric calibration (criteria 7, 8, and 10) [19].

For the verification and identification stage, a digital symbol approach was used instead of a pattern wide matching scheme (such as correlation in ARToolkit). No comparisons are made to preloaded expected patterns; the expected patterns are implicit in the decoding algorithm. The digital coding system used consists of a six by six grid of bitonal cells in the interior of the marker. The marker is entirely bitonal, only black and white shading, allowing either gray scale or color cameras to be used and to satisfy criterion 7 of not requiring photometric calibration. The border width was chosen to be two cell widths wide to maximize the edge detector response when the marker is small in the image. A binary codeword of 36 digital symbols is encoded in the marker containing a marker ID, and checksum and error correcting codes. The library is either 1,001 or 2,002 markers depending on if both marker polarities are used (black border on white background and vice versa). Several ARTag markers are shown in Fig. 1.

4.1 Unique Feature Detection: Finding Quads

Quadrilateral contours are points in the image which may belong to the outside border of a marker. This quadrilateral border must first be found; in ARTag this is done by combining line segments found in a gradient image. Using edges replaces the necessity of an absolute threshold value that binarization requires with a threshold on the gradient. The choice of an absolute threshold is problematic since this value will depend on lighting. Furthermore, partially occluded markers can still be detected using line segments based on heuristics of missing corners and sides. This is described in detail in [2]. Figs. 4 and 5 demonstrate the decreased sensitivity to lighting and occlusion, respectively. In principle, any method that finds the border using edge gradients as opposed to intensity value should be used.

4.2 Verification and Identification of ARTag Markers

A principled approach to provisioning the data cells in a digital symbol coding scheme can greatly affect several of the criteria of Section 2; most importantly, the false positive, false negative, and intermarker confusion rates.

The marker must be detectable at different orientations, and therefore, the marker must have some element that indicates orientation, or the patterns must be chosen so that no marker in the library be rotationally symmetric with any other (including itself). The circular Intersense markers and Binary Square Marker [20] allocate some specific part of the marker to indicate orientation. This could be considered a design weakness since the marker detection is more fragile to a misread in one part of the marker than another. The latter approach of making the library rotationally asymmetric was chosen for ARTag. This comes at the cost of requiring that a marker hypothesis be tested in all four orientations.

The patterns chosen for the BinARyID system are derived directly from this rotational asymmetry requirement. For a square marker, this is approximately less than a quarter of all possibilities. For example, in a 4×4 grid there are $<2^{14}$ possible markers. In contrast, all possibilities of the Intersense marker cells are considered as valid markers due to the orientation detection in the decoding step. It is desirable to reduce the number of possibilities that represent a valid marker in order to lower the false positive rate (criterion 1) and the intermarker confusion rate (criterion 2). Otherwise, the false positive rate will be 100 percent of the objects in the image that pass the first stage (unique feature test for systems like ARTag). Similarily, if all possibilities are used, it only takes one misread cell to cause the wrong ID to be reported.

One method to reduce the number of possibilities that map to a marker is to utilize a *checksum*. Several of the bits in ARTag were reserved for a checksum; if $\mathbf{c}$ bits are reserved for a checksum, then the false positive rate for an object passing the unique feature test is at most $\frac{1}{2(\mathbf{c}-2)}$ (the -2 is due to the rotational symmetry, $log_2 4 = 2$; each pattern appears four times in the set of all possibilities). The phrase "at most" is used because $\frac{1}{2(\mathbf{c}-2)}$ assumes equal likelihoods for all nonmarker objects in the scene, whereas in practice most nonmarker quadrilaterals typically have blank interiors with sparse frequency content.

A second possibility, when using a digital symbol approach, is to add *error correction* capability, $\mathbf{e}$ of the bits can be set aside to provide redundancy so that the up to $\mathbf{corr}$ falsely read bits can be repaired.

The ARTag system was designed with both a checksum and error correction mechanism, which is a novel contribution of ARTag. The marker ID was scrambled with the checksum, error correcting codes, and a fixed mask designed to increase the entropy (frequency content) in order to differentiate markers from the typical low frequency content of nonmarker quadrilaterals.

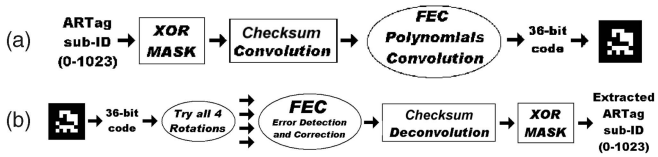

Fig. 6. (a) Digital encoding process: Creating ARTag markers. A sub-ID number (the lower 10 bits of the ARTag ID) generates a 36-bit pattern of black and white cells. (b) Digital decoding process: Identifying ARTag IDs in the binary pattern from the interior of an ARTag marker.

The system can be abstractly described as a communication system, where an ID is data which is encoded (Fig. 6a), transmitted through a medium of printing and image acquisition, and decoded (Fig. 6b). In ARTag, the marker ID is composed of both the border polarity (black square on white background or vice versa) and the sub-ID, which is encoded in the marker interior.

The process for detecting an ARTag marker is as follows: Once quadrilateral border contours for hypothetical markers have been located as per Section 4.1, the internal region is sampled with a $\mathbf{w}$ by $\mathbf{w}$ grid and assigned digital symbols "0" or "1" according to a threshold derived from the intensities found around the quad border. All subsequent processing to verify and identify the marker is performed digitally. Four $\mathbf{n}$-bit binary sequences are obtained from the 2D $\mathbf{w}$ by $\mathbf{w}$ digital symbol array, one for each of the four possible rotation positions ($\mathbf{n} = \mathbf{w}^2$). Only one of the four sequences will end up being validated in the decoding process. The $\mathbf{n}$-bit binary sequence encoded in the marker encapsulates a *sub-ID*. The extra bits provide redundancy to reduce the chances of false detection and identification (including uniqueness over the four possible rotations), and error correcting capability.

4.3 Provisioning Data Cell Usage to Optimize Criteria

With the choices made for a square marker with an internal bitonal grid of cells containing ID, checksum, and error correcting codes, the next design decisions are to plan how large to make the data grid and how to provision the cell usage.

With the marker pattern being a square grid of width $\mathbf{w}$ cells, there are $\mathbf{n} = \mathbf{w}^2$ cells. $\mathbf{i}$ bits are allocated for the *sub-ID* identity, $\mathbf{c}$ bits are for the checksum, and $\mathbf{e}$ are for the error correcting codes. In total $\mathbf{n} = \mathbf{i} + \mathbf{c} + \mathbf{e}$. The maximum number of sub-IDs is $2^{\mathbf{i}}$. A balance has to be made between having a low $\mathbf{n}$ (criterion 6, minimum marker size), having a large checksum $\mathbf{c}$ to minimize the false positive (criterion 1) and intermarker confusion rates (criterion 2), and having many redundant error correcting bits $\mathbf{e}$ to minimize the false negative rate (criterion 3).

If the probability of a cell being misread, due to image noise, specularities, position imperfections in the border detection, etc., can be estimated to be $\mathbf{p}$ and the number of cells is $\mathbf{n}$, then the probability of a false negative is $1 - (1 - \mathbf{p})^{\mathbf{n}}$. For $\mathbf{n} = 36$ and a cell misread rate of $\mathbf{p} = 10^{-4}$, the false negative rate will be 0.0036. If $\mathbf{corr}$ errors can be corrected, then the rate falls to $1 - \sum_{i=0}^{\mathbf{corr}} C_i^{corr}(1-\mathbf{p})^{\mathbf{i}}\mathbf{p}^{(n-\mathbf{i})}$ according to the binomial theorem. The rate falls to 6.3×10^{-6} for $\mathbf{corr} = 1$, to 7.1×10^{-9} for $\mathbf{corr} = 2$, and to 5.9×10^{-12} for $\mathbf{corr} = 3$. $\mathbf{corr}$ should not be raised too high as the false positive rate is degraded (raised) as the number of potential combinations that map to a verified marker detection increase. For $\mathbf{corr} = 1$, the false positive rate falls to $\frac{n}{2^{\mathbf{c}}}$, for $\mathbf{corr} = 2$, the false positive rate falls to $\frac{n(n-1)}{2^{\mathbf{c}}}$, etc. Also, since $\mathbf{e}$ of the bits are being consumed to provide $\mathbf{corr}$ of correctible errors, increasing $\mathbf{corr}$ also increases the marker size or consumes the bits left for the ID and checksum. In this case, $\mathbf{e}$ is approximately equal to $\mathbf{corr} \cdot log_2\mathbf{n}$ using the error correction scheme used [2].

The best possible intermarker confusion rate (criterion 2) can be estimated by the minimum *Hamming distance* between any two markers in the library considering all four rotation positions. The Hamming distance is the smallest number of bit changes to convert one marker pattern into another. The rate will be shown in (1) to be a summation of terms with Hamming distance as exponents, with the dominant term being due to the minimum distance. Assuming the marker patterns were all equally spaced by Hamming distance, a checksum of $\mathbf{c}$ bits would provide a minimum Hamming distance of $\mathbf{c} - log_2 4 = \mathbf{c} - 2$ bits. The error correcting codes will reduce this margin to $\mathbf{c} - 2 - \mathbf{corr}$ bits that can be corrupted for a marker to be falsely identified. The markers will probably not be ideally spaced by this amount; therefore, the quantity of average minimum Hamming distance $\mathbf{AHD} = \mathbf{c} - 2 - \mathbf{corr}$ is introduced. The higher $\mathbf{AHD}$ is, the lower the intermarker confusion rate. Table 1 shows several possible choices for $\mathbf{w}$, $\mathbf{n}$, $\mathbf{i}$, $\mathbf{c}$, and $\mathbf{e}$, and the resulting library size and example error rates assuming $\mathbf{p} = 10^{-4}$.

In order to keep the minimal marker size small, a grid of size $\mathbf{w} = 6$ was chosen since it provided the library size and error rates deemed to be acceptable. The configuration that was chosen is $\mathbf{w} = 6$ (therefore, $\mathbf{n} = 36$), $\mathbf{i} = 10$ for a library size of $\approx$1,024 markers, a checksum of size $\mathbf{c} = 16$ bits for an estimated false positive rate of 6.1×10^{-5} of hypothesis quadrilateral features, and an error correcting capability of $\mathbf{corr} = 2$ bits which consumes $\mathbf{e} = 10$ bits. With an example data cell misread rate of $\mathbf{p} = 10^{-4}$ this gives an estimated false negative rate of 7.1×10^{-9}. It should be

TABLE 1
Example Different Combinations of Design Parameters for a Square Bitonal Array of Width $\mathbf{w}$, $\mathbf{n} = \#$ of Cells, $\mathbf{i}$, $\mathbf{c}$, and $\mathbf{e} = \#$ of Bits for Marker ID, Checksum, and Error Correcting Codes

i	c	e	corr	lib	fp	fn	ahd
$\mathbf{w} = 4$, $\mathbf{n} = 16$							
16	0	0	0	$< 2^{14}$	1	10^{-3}	1
12	4	0	0	2048	0.25	10^{-3}	2
8	8	0	0	256	0.015	10^{-3}	6
4	12	0	0	16	10^{-3}	10^{-3}	10
8	4	4	1	256	0.25	10^{-6}	1
4	8	4	1	16	0.015	10^{-6}	5
$\mathbf{w} = 5$, $\mathbf{n} = 25$							
25	0	0	0	$< 10^7$	1	10^{-3}	1
21	4	0	0	10^6	0.25	10^{-3}	2
17	8	0	0	10^5	0.015	10^{-3}	6
13	12	0	0	8182	10^{-3}	10^{-3}	10
8	12	5	1	256	10^{-3}	10^{-6}	9
3	12	10	2	8	10^{-3}	10^{-9}	8
$\mathbf{w} = 6$, $\mathbf{n} = 36$							
36	0	0	0	$< 10^{10}$	1	10^{-3}	1
28	8	0	0	10^8	0.015	10^{-3}	6
20	16	0	0	10^6	10^{-4}	10^{-3}	14
12	24	0	0	2048	10^{-7}	10^{-3}	22
10	26	0	0	1024	10^{-7}	10^{-3}	8
8	28	0	0	256	10^{-8}	10^{-3}	6
10	21	5	1	1024	10^{-6}	10^{-5}	18
10	**16**	**10**	**2**	**1024**	$\mathbf{10^{-5}}$	$\mathbf{10^{-8}}$	**12**
10	11	15	3	1024	10^{-3}	10^{-11}	6
12	19	5	1	256	10^{-5}	10^{-5}	16
12	14	10	2	256	10^{-3}	10^{-8}	10
12	9	15	3	256	10^{-2}	10^{-11}	4
$\mathbf{w} = 7$, $\mathbf{n} = 49$							
49	0	0	0	$< 10^14$	1	10^{-3}	1
37	12	0	0	10^11	10^{-3}	10^{-3}	10
25	24	0	0	10^7	10^{-7}	10^{-3}	22
17	32	0	0	10^5	10^{-9}	10^{-3}	30
25	16	8	1	10^7	10^{-5}	10^{-5}	13
17	16	16	2	10^5	10^{-5}	10^{-8}	12
9	16	24	3	512	10^{-5}	10^{-11}	11
5	12	32	4	32	10^{-3}	10^{-14}	6

lib *is the library size,* $\mathbf{corr} = \#$ *of correctible bits,* $\mathbf{fpr}$, $\mathbf{fnr} = false$ *positive, false negative rates assuming a bit misread rate of* $\mathbf{p} = 10^{-4}$. $\mathbf{ahd}$ *is the average* Hamming distance. *The row in bold under* $\mathbf{w} = 6$ *are the parameters chosen for the ARTag system.*

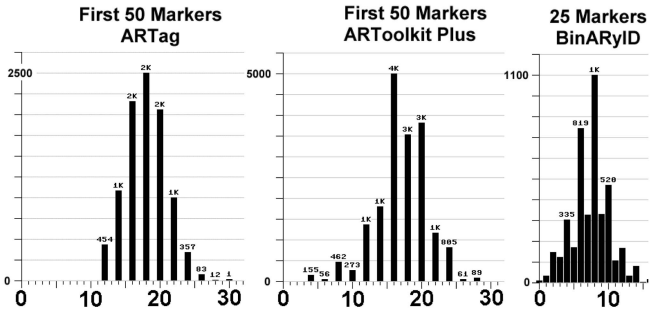

Fig. 7. Cumulative histogram of Hamming distances between markers considering rotation and reflection.

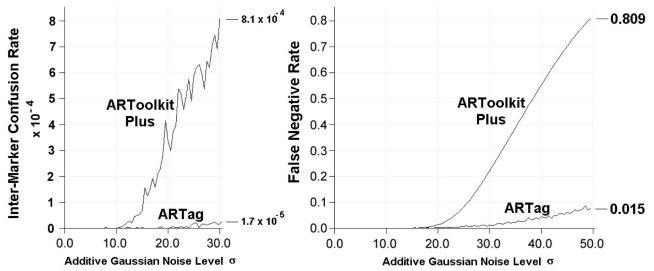

Fig. 8. Synthetic experiments: Gaussian noise added to test image to produce detection errors.

noted again that the false positive rate will be lower by a large but unquantifiable amount due to the low frequency content of most quadrilateral shapes in typical scenes. These values were chosen to give the best trade-off between all of the relevant parameters. Having about 1,000 markers was sufficient for the applications envisaged, such as the instrumented room. Having two bits of correctibility was felt to be important, not only to reduce the false negative rate, but also to allow partial occlusion. Typically, four cells can be occluded since, mostly, probably half will be thresholded to the same shade (light or dark) as the cell itself.

4.4 Reducing Intermarker Confusion

The above analysis yields the basic structure of our digital marker including a *Hamming distance* [21] approximation which is maximized for a low intermarker confusion rate (criterion 2). More design work remains to complete the marker library to more accurately optimize this rate. The user should have a high confidence that one marker won't be mistaken for another. With proper marker system design, the rate at which this occurs can be minimized, ideally to very low levels that don't appear in practice.

Each marker pattern can be turned into another if the right "1" and "0" symbols are changed. Ideally, the Hamming distance should be as high as possible between all possible markers, taking rotation and optionally mirroring into consideration. The probability that a digital code can be mistaken for another in a set of codes is given by a summation of the probabilities that the given marker can be falsely recognized as each of the other codes in the set. This probability can be expressed as a sum of the probabilities of all possibilities where one marker can be misread as another through a specific set of misreads. This is equal to the likelihood of a specific number of bits misread multiplied by the frequency of occurrence, which is measured in a histogram of Hamming distances (1). The histogram is created by grouping together all of the cases of equal Hamming distance, where $HD(d)$ is the number of cases where a Hamming Distance of d occurs relative to a single marker A. The probability of intermarker confusion is now divided into two parts; the system dependent probabilities of a cell misread $\mathbf{p}$ due to camera, lighting, image acquisition, sensor noise, corner position error, etc., and the component due to the distinctiveness of the markers themselves represented by the Hamming distance histogram. This Hamming distance histogram is a metric to optimize, i.e., to reduce intermarker confusion independent of the specific conditions where the system is used. Since the probability is most affected by the lower order terms of (1), we seek to reduce the frequency of those of low values of $\mathbf{d}$. The more that the histogram can be pushed out to the right (as plotted as in this paper) by having empty histogram bins for low values of $\mathbf{d}$, and the lower $HD(\mathbf{d})$ is for the first few nonzero bins, the more immune to intermarker confusion a marker set will be

$$P(\neq A) = \sum_{d=1}^{\mathbf{n}} HD(n) \cdot p^d (1-p)^{\mathbf{n}-d}. \quad (1)$$

The ARTag marker set was specifically designed based on Hamming analysis. Using the system shown in Fig. 6, we see the parameters are the 10-bit XOR mask, the checksum convolution code and the convolution codes used for the FEC. These are our final variables in designing the marker system. The XOR mask does not affect the Hamming distances since it merely flips the same bits for all patterns; it only affects which sub-ID should not be used to avoid an all white or all black marker interior which will be confused with commonly found empty quadrilaterals. This leaves the choice of convolution polynomials chosen for use in the checksum and error correction as variables in order to obtain an optimal histogram of Hamming distances measured for the full marker set. Several combinations were tested; the optimal combination produced the histogram with the minimum area less than $AHD = 12$. The set of all possible markers was then slightly reduced to remove the markers that contributed to the nonzero $HD(\mathbf{d})$ histogram bins with the lowest $\mathbf{d}$, 23 sub-IDs were removed reducing the set to 1,001 sub-IDs. The final library size is 1,001 with a single border polarity, or 2,002 with both possible border polarities.

To further reduce the intermarker confusion, the sub-IDs were ordered in a list according to maximal Hamming distance from each other. A user takes the top k markers in order from the top of this list to obtain a lower intermarker confusion rate than the entire set. In this way, an application can be made very immune to intermarker confusion. The histogram of the first 50 ARTag markers is shown as the left plot in Fig. 7 with corresponding histograms from the first 50 ARToolkit Plus markers and the 25 BinARyID markers provided in [22]. Rotations and reflections are considered, more plots can be found in [2]. The important information in the histograms is of how many entries there are for low Hamming distance numbers (left side of plots) since these values will dominate the probability calculation. With ARTag, the left side of the histogram is 0 for distances up till 12 ($HD(d) = 0 : d < 12$). Since ARToolkit Plus (the version available at the time of this writing) encodes IDs by simply repeating a 9-bit code four times and applying an XOR mask, we expect a small Hamming distance of $HD(d) = 4$ between markers with consecutive IDs.

BinARyID codes were designed by incrementing a 16-bit code and checking for rotational asymmetry; thus many markers only have a Hamming distance of $HD(d) = 1$ between consecutive IDs. Also, due to reflection symmetry, there are several that fill the $HD(d) = 0$ bin, which means intermarker confusion events are guaranteed if a marker is seen in a reflection.

Fig. 8 demonstrates the effect of optimizing the Hamming distances. Two test images were captured: one with an array of ARTag markers and one with an array of ARToolkit Plus markers. Both images had the same size and arrangement of markers. Several million test images were created from these images with additive gaussian noise, and all images were input to the corresponding marker detection algorithm. When markers were misidentified or missed an intermarker confusion event or false negative event was recorded. Despite having the same size and

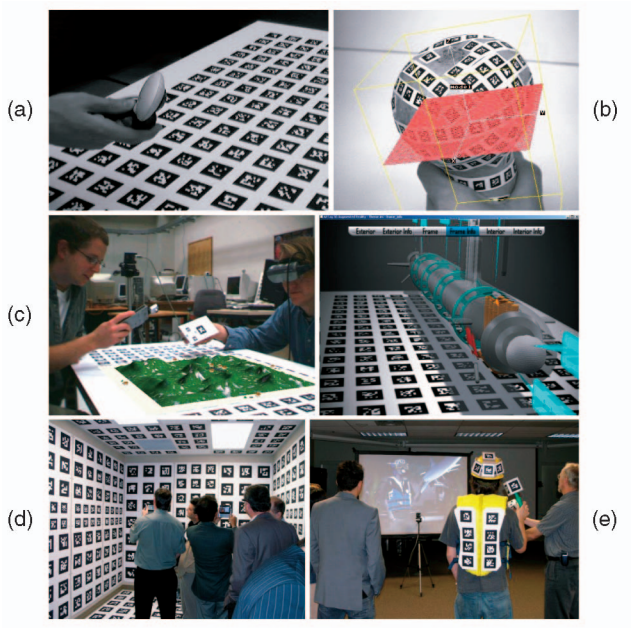

Fig. 9. Applications of the ARTag fiducial marker system. (a) Automatic camera calibration. (b) 6-DOF object tracking. (c) Table-top video see-through augmented reality with handheld device. (d) Instrumented room, markers on all surfaces for real-time positioning for augmented and virtual reality with handheld and wearable computers. (e) Magic Mirror—display shows mirrored camera image with virtual objects appearing attached to handheld and wearable objects.

number of data cells ($n = 36$), ARTag performed several orders of magnitude better, as predicted by this analysis. An executable version of BinARyID was not available to perform these tests but the intermarker confusion rate is expected to be very high.

The result of applying this analysis to develop ARTag is a system where an intermarker confusion event has not been seen in normal operation[2] despite being used over thousands of frames with hundreds of markers visible in each frame as in our camera calibration experiments. And a false positive event has yet to be recorded.

5 Fiducial Marker Applications

The ARTag fiducial marker system was designed to meet the 11 criteria from Section 2. This section describes some applications of the ARTag system.

The robust detection and very low intermarker confusion enables fully automatic camera calibration [23] without the manual stages normally necessary for point identification or cleanup of semiautomatic techniques (shown in Fig. 9a).

The high confidence in the ID of a detected ARTag marker allows it to be used for human computer interaction ranging from simple behaviors based on detection of single markers such as kiosk interaction with printed markers [24] and flash cards for a scuba diver to signal instructions to an underwater robot [25] to a full 6-DOF real-time input device of an object covered with markers [26] as shown in Fig. 9b.

ARTag was principally designed for augmented reality applications to allow table, floor, or entire room surfaces to be covered with markers. A handheld or wearable device calculates pose in real time given the matching points (four points per marker, one for each corner) between the extracted markers and an a priori model such that virtual models, animations, and video game content appears to belong to the real scene [27] (Figs. 9c and 9d).

2. Some intermarker confusion events have been seen when the patterns were bent or folded on themselves, but this is not considered normal operation.

The large library size of ARTag markers (1,001) allows an entire room (Fig. 9d) to have floor, wall, and ceiling surfaces covered with markers, providing an instrumented space for augmented and virtual reality systems using only passive computer vision software from the devices' on-board cameras to give reliable pose. This was not possible with previous marker systems due to the small marker sizes and intermarker confusion. Handheld and wearable items are augmented when looking in the "Magic Mirror" Fig. 9e.

6 Conclusion

A design approach was described for designing highly reliable fiducial marker systems considering a set of performance criteria. Fundamental considerations to satisfy these criteria were provided to allow a principled approach to what is typically an ad hoc design process. A general purpose marker system should provide reliable information with unknown printing methods and cameras under uncontrolled lighting. Of prime importance for many applications is that markers should not be falsely detected, or the wrong ID reported. Marker systems were shown to have two stages, hypothesis generation from unique image features and verification/identification. Methods to reduce sensitivity to lighting and occlusion were shown for the first stage, and Hamming distance-based digital code analysis demonstrated for the second stage. The *ARTag* fiducial marker system was designed using these principles, comparisons were made to several systems in the literature and industry, and several applications that are enabled by ARTag were shown.

References

[1] M. Fiala, "ARTag, a Fiducial Marker System Using Digital Techniques," *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition*, pp. 590-596, Aug. 2005.
[2] M. Fiala, "ARTag Revision 1, a Fiducial Marker System Using Digital Techniques," Nat'l Research Council Publication 47419/ERB-1117, Nov. 2004.
[3] H. Kato and M. Billinghurst, "Marker Tracking and HMD Calibration for a Video-Based Augmented Reality Conferencing System," *Proc. Second IEEE and ACM Int'l Workshop Augmented Reality*, vol. 85, pp. 20-21, Oct. 1999.
[4] Electronics Industry Association Website, http://www.eia.org/, 2009.
[5] V. Knyaz and A. Sibiryakov, "The Development of New Coded Targets for Automated Point Identification and Non-Contact 3D Surface Measurements," *Proc. Int'l Conf. Computer Graphics and Vision*, 1998.
[6] http://www.photomodeler.com, 2009.
[7] L. Naimark and E. Foxlin, "Circular Data Matrix Fiducial System and Robust Image Processing for a Wearable Vision Inertial Self-Tracker," *Proc. IEEE and ACM Int'l Symp. Mixed and Augmented Reality*, Sept. 2002.
[8] Y. Cho, J. Lee, and U. Neumann, "A Multi-Ring Color Fiducial System and an Intensity-Invariant Detection Method for Scalable Fiducial-Tracking Augmented Reality," *Proc. Int'l Workshop Augmented Reality*, pp. 147-165, 1998.
[9] J. Rekimoto and Y. Ayatsuka, "CyberCode: Designing Augmented Reality Environments with Visual Tags," *Proc. DARE 2000 on Designing Augmented Reality Environments*, pp. 1-10, Oct. 2000.
[10] M. Rohs and B. Gfeller, "Using Camera-Equipped Mobile Phones for Interacting with Real-World Objects," *Advances in Pervasive Computing*, pp. 265-271, Springer, 2004.
[11] J. Rekimoto, "Matrix: A Realtime Object Identification and Registration Method for Augmented Reality," *Proc. Third Asia Pacific Computer Human Interaction*, pp. 63-68, 1998.
[12] D. Wagner and D. Schmalstieg, "ARToolKitPlus for Pose Tracking on Mobile Devices," *Proc. 12th Computer Vision Winter Workshop*, Feb. 2007.
[13] X. Zhang, S. Fronz, and N. Navab, "Visual Marker Detection and Decoding in AR Systems: A Comparative Study," *Proc. IEEE and ACM Int'l Symp. Mixed and Augmented Reality*, pp. 97-106, Sept. 2002.
[14] "ARVIKA Webpage," http://www.arvika.de/www/index.htm, 2009.
[15] G. Klinker, D. Stricker, and D. Reiners, "Augmented Reality: A Balancing Act Between High Quality and Real-Time Constraints," *Proc. Int'l Symp. Mixed Reality*, pp. 97-106, 1999.
[16] D. Stricker, G. Klinker, and D. Reiners, "A Fast and Robust Line-Based Optical Tracker for Augmented Reality Applications," *Proc. Int'l Workshop Augmented Reality*, pp. 129-145, 1999.
[17] C. Owen, F. Xiao, and P. Middlin, "What Is the Best Fiducial?" *Proc. First IEEE Int'l Augmented Reality Toolkit Workshop*, Sept. 2002.

[18] A. Rice, A. Beresford, and R. Harle, "Cantag: An Open Source Software Toolkit for Designing and Deploying Marker-Based Vision Systems," *Proc. Fourth Ann. IEEE Int'l Conf. Pervasive Computing and Comm.*, pp. 10-21, Mar. 2006.
[19] M. Fiala, "Comparing ARTag and ARToolkit Plus Fiducial Marker Systems," *Proc. IEEE Int'l Workshop Haptic Audio Visual Environments and Their Applications*, pp. 147-152, Oct. 2005.
[20] P. Boulanger, "Application of Augmented Reality to Industrial Tele-Training," *Proc. Canadian Conf. Computer and Robot Vision*, pp. 320-328, May 2004.
[21] R. Hamming, "Error Detecting and Error Correcting Codes," *Bell Systems Technology J.*, Vol. 29, pp. 147-160, Apr. 1950.
[22] D. Flohr and J. Fischer, "A Lightweight ID-Based Extension for Marker Tracking Systems," *Proc. Eurographics Symp. Virtual Environments*, pp. 59-64, 2007.
[23] M. Fiala and C. Shu, "Self Identifying Patterns for Plane-Based Camera Calibration," *Machine Vision and Applications*, Springer, Aug. 2007.
[24] M. Crease and M. Fiala, "Presentation Assistant and Kiosk Interaction with Fiducial Markers," *Proc. IEEE Int'l Workshop Haptic, Audio and Visual Environments and Games*, pp. 43-43, Oct. 2007.
[25] G. Dudek, J. Sattar, and A. Xu, "A Visual Language for Robot Control and Programming: A Human Interface Study," *Proc. IEEE Int'l Conf. Robotics and Automation*, pp. 2507-2513, Apr. 2007.
[26] M. Fiala, "The SQUASH 1000 Tangible User Interface System," *Proc. IEEE and ACM Int'l Symp. Mixed and Augmented Reality*, pp. 180-181, 2005.
[27] M. Fiala and G. Roth, "Magic Lens Augmented Reality: Table-Top and Augmentorium," *Proc. ACM SIGGRAPH '07*, p. 152, 2007.

▷ **For more information on this or any other computing topic, please visit our Digital Library at** www.computer.org/publications/dlib.