Rapid Color Grading for Fruit Quality Evaluation Using Direct Color Mapping

Dah-Jye Lee, *Senior Member, IEEE*, James K. Archibald, *Senior Member, IEEE*, and Guangming Xiong

***Abstract*—Color grading is a crucial step in the processing of fruits and vegetables that directly affects profitability, because the quality of agricultural products is often associated with their color. Most existing automatic color grading systems determine color quality either by directly comparing product color against a predefined and fixed set of reference colors or by using a set of color separating parameters, often in three-dimensional color spaces. Using these methods, it is not convenient for the user to adjust color preferences or grading parameters. In this paper, we present an effective and user-friendly color mapping concept for automated color grading that is well suited for commercial production. User friendliness is often viewed by the industry as a very important factor to the acceptance and success of automation equipment. This color mapping method uses preselected colors of interest specific to a given application to calculate a unique set of coefficients for color space conversion. The three-dimensional RGB color space is converted into a small set of color indices unique to the application. In contrast with more complex color grading techniques, the proposed method makes it easy for a human operator to specify and adjust color-preference settings Tomato and date maturity evaluation and date surface defect detection are used to demonstrate the performance of this novel color mapping concept.**

***Note to Practitioners*—The quality and maturity stage of agricultural products is often associated with their color. For example, in fresh produce markets such as red delicious apples and peaches, consumers prefer dark red fruit. Color is also used to determine fruit maturity and to detect surface blemishes and defects. In general, fruits such as tomatoes and dates are harvested before they fully ripen. After harvesting, they continue to ripen and their color changes. For example, color is used to determine the length of time the tomatoes can be transported and the type of drying process to ripen the dates. Most existing color grading systems are not user-friendly and do not allow the operator to adjust grading parameters according to his/her color perception and color preference. The color mapping concept presented in this paper is easy to implement and is well suited for commercial production.**

***Index Terms*—Machine vision, automatic color grading, visual inspection, color space conversion, fruit quality, food processing.**

Manuscript received February 20, 2010; revised July 30, 2010; accepted October 06, 2010. Date of publication November 15, 2010; date of current version April 06, 2011. This paper was recommended for publication by Associate Editor P. Sastry and Editor S. Sarma upon evaluation of the reviewers' comments. This work was supported in part by Datepac, LLC of Yuma, Arizona, USA.

D.-J. Lee and J. K. Archibald are with the Department of Electrical and Computer Engineering, Brigham Young University, Provo, UT 84602 USA (e-mail: djlee@byu.edu).

G. Xiong is with the School of Mechanical Engineering, Beijing Institute of Technology, Beijing 100081, China.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TASE.2010.2087325

I. Introduction

In recent years, machine vision has been used in many applications requiring visual inspection. As examples, some machine vision systems are designed specifically for factory automation tasks such as intelligent packing [1], versatile online visual inspections [2], [3], lumber production [4], microscopic imaging for biology [5], poultry production [6], camera image contrast enhancement for surveillance and inspection tasks [7], patterned texture material inspection [8], and closed-loop online process control [9]. These systems are examples that show the importance of visual inspection technology in automation.

Some machine vision systems are designed using color image processing techniques specifically for automation tasks. For example, an automatic color shade grading system compares color histograms for industrial inspection of randomly textured multicolored ceramic tiles [10]. Another similar system combines wavelet texture analysis with color information to address deficiencies of the color histogram method [11]. The color histogram method is improved by using fuzzy logic for color grading of wood boards [12]. Statistical analysis of chromaticity and luminance differences is developed to evaluate common solid-state and fluorescent lamps with respect to reference illuminants [13]. An imaging device using a charge-coupled device (CCD) and a tunable bandpass filter performs color fading analysis of an art print that has undergone a light forced exposure [14].

Specific to the agricultural industry, color image processing has been employed in many applications involving color grading. The development of color CCD cameras and CMOS image sensors—widely used in cellular phones, digital cameras, and other consumer-electronics devices—has provided inexpensive technology for vision-based automation for the agricultural industry. Color machine vision systems have been developed for real-time color grading in industrial food product production for palm oil fresh fruit bunches [15], beef color evaluation [16], detection of melanin spots in Atlantic salmon fillets [17], measurement of red grapefruit juice color [18], color feature detection specific to food processing [19], [20], etc. The aforementioned vision systems for automation [1]–[9], and for non-agricultural [10]–[14] or agricultural [15]–[20] applications show the importance of machine vision technology to automation.

Color grading, the focus of this work, is the primary method for determining product value in the food processing and agriculture industries. In its most general sense, the term "color grading" is sometimes used to refer to processes that include

the enhancement of images to be viewed by a human. In the context of this paper, the processing of fruits and vegetables, color grading refers to the procedure of assigning each product a quality or maturity value based on its color.

For many agricultural products, certain colors are preferred and command higher selling prices. This is true for apples [21]–[25], broccoli [26], and cranberries [27]. In fresh produce markets such as red delicious apples and peaches, dark red represents higher quality than light red. Without consistent and reliable color grading, the evaluation of product quality is erratic and product value cannot be maximized.

Color can be used to evaluate fruit maturity for tomatoes [28]–[31], watermelons [32], bananas [33], and dates [34]. Consider, for example, the problem of determining time to market for tomatoes. In general, tomatoes must be harvested while still green. After harvesting, the fruit continue to ripen and their color turns lighter green, then pink, and eventually red. Due to transportation delays, tomatoes that are already ripe (red) when picked must be sold to local markets. Green tomatoes can be shipped to customers over much greater distances.

Another example of using color for maturity evaluation is to determine the proper drying process to help dates to ripen [34]. Date growers harvest all fruit in a very short window of 3 to 4 weeks, yielding dates with varying stages of maturity. Unripe dates are sent to gain moisture in a hydrating building (most mature), to ripen in the sun (medium), or to dry in a heated building (least mature). Insufficient drying can cause dates to rot and turn sour, while excessive drying can cause the fruit's skin to peel, reducing the value and quality of the fruit.

Color can also be used to detect surface blemishes or defects on apples [35], pears [36], and oranges [37]. Another unique example is skin separation detection on dates [38]. After the drying process, some dates end up in lower quality grades because of skin separation (peel) that occurs as they dry. It is therefore critical that the maturity of each harvested date is evaluated accurately before drying, and that skin separation defects are detected reliably before packing.

A wide variety of methods are used in the aforementioned color grading systems. Color grading generally involves two steps: color feature representation and color categorization. Color must be represented in a way that allows it to be categorized. Color representations include color histograms [10]–[12], [22], [37], RGB [14], [16], [17], [28], [29], [36], hue saturation, intensity (HSI) [16], [22], [25], [36], CIE Lab [16], [18], [26], [27], [30], [33], and surface texture and feature [11], [19], [20], [24]. Color categorization is performed to grade colors using simple thresholding [18], [22], [31], [38], color distance to reference colors [10], [16], [21], [36], statistical analysis [13], [17], [19], [20], [27], fuzzy logic [12], [15], [32], neural networks [23], [26], [35], [37], a genetic algorithm [24], a Bayes classifier [25], and 3-D lookup tables [28], [29].

Much of the research in image processing techniques for agricultural applications focuses on practical issues arising when theory is matched with practice in a real-world setting. Our own investigations have led us to conclude that machine learning techniques such as fuzzy logic, neural networks, genetic algorithms, and 3-D lookup tables are particularly difficult to deploy in real-world commercial applications and require extensive training of the user. In a high-volume production setting, it is problematic to perform frequent routines to adapt to variations in the colors of the produce being graded. A second critical concern is that grading results obtained by automated training will not necessarily agree with the preferences of an experienced operator. Moreover, when the results are not in agreement, there is no straightforward or intuitive way of making slight adjustments to classification thresholds that more accurately reflect the preferences of a human expert.

In this paper, we present a simple but effective color mapping concept for color grading. One of our overriding goals at the outset was to make it easy for a human operator to adjust the system to match color preferences for specific color grading applications. Our method does not incorporate machine learning or artificial intelligence techniques because of the challenges mentioned above. Instead, it converts a specified range of colors of interest in the 3-D RGB color space into a smooth and continuous 1-D color space. We will show that this technique results in color grading that reliably meets industry standards for accuracy. Moreover, setting the color preferences and making subsequent adjustments are both straightforward and easily accomplished by an operator without specialized training in image processing. It involves only sliding a threshold bar to achieve the effect of "slightly darker" or "slightly greener," etc. We use tomato and date maturity evaluation and date surface defect detection to demonstrate the effectiveness of this algorithm.

The details of our new method are presented in Section II, including the motivation, color representation, color mapping, and calibration. We demonstrate the robustness of this novel algorithm with experimental results of maturity evaluation of tomatoes in Section III and dates in Section IV. Section V discusses the use of this algorithm for date surface defect detection. We summarize our work and conclude this paper in Section VI.

II. Direct Color Mapping

A. Challenges in Color Grading

Most color image processing and computer vision algorithms such as image segmentation and object recognition work in distinct color spaces. A wide variety of color spaces exist (e.g., RGB, CMY, HSI, YIQ, YUV, CIE Lab, CIE Luv), each interpreting and modeling color in a particular way. No single color space is most appropriate for all applications. The choice of an appropriate color space is critical in the design of any automated machine vision system, as it will directly affect the overhead and efficiency of required image processing tasks [39].

At the outset of our project, based on our experience working with color grading vision system operators, we identified two major requirements for a user-friendly color grading technique.

1) Easy determination of color preference settings.
2) Easy adjustment of the color preference settings to match the desired color separations.

Papers in the literature show that RGB, HSI, and CIE Lab are the most commonly used color spaces for color grading. However, for purposes of color grading, neither RGB, HSI, nor CIE Lab color spaces are particularly attractive alternatives. Of greatest concern is that selected target hues or colors seldom result in a convenient or intuitive partitioning of the multidimensional

Fig. 1. A color bar of colors of interest for tomato color grading.

Fig. 2. Converted color index values of Fig. 1.

color spaces. In particular, it is difficult to define and adjust color grade boundaries in the 3-D RGB color space. Furthermore, color discontinuity in hue makes it impossible to meet the second requirement [34].

Consider a simple example to illustrate the underlying idea of a friendly color grading system. Fig. 1 shows the color range of tomatoes at all possible maturity stages, with six distinct grades identified according to the United States Department of Agriculture (USDA) classification standard: Green, Breakers, Turning, Pink, Light Red, and Red [40]. Ideally, the user should be able to modify the color preferences simply by adjusting the line that separates two adjacent grades. For example, if the user prefers to include slightly darker green tomatoes in Grade 2, then the cutoff point between Grades 1 and 2 needs only to be moved slightly to the left. Similarly, to include fruit that is darker red in Grade 5, the separation point between Grades 5 and 6 would be moved slightly to the right. The new color mapping method for color grading described in the next section provides this functionality for any desired application-specific range of colors.

B. Color Representation

Most color cameras used for machine vision and automation output RGB signals. Each of these RGB signals is usually digitized into an 8-bit value between 0 and 255. Thus, it takes 24 bits (specifying a total of more than 16 million unique colors) to represent the color information for a single pixel in the acquired image. In many cases, although the full 24-bit color information is stored, the *absolute* color values are not as critical as the *relative* differences between colors in distinct regions of interest.

Furthermore, most products have a limited number of colors of interest—the range of observed colors in grading a particular product will be a very small subset of the full 16M colors. Colors that are known to never occur should be ignored. For example, while grading Golden Delicious apples, the system can safely reduce the number of bits for storing or processing the blue channel and focus primarily on the red and green channels. If the range of colors of interest is well defined for each specific product, then color grading can be significantly simplified. For tomato maturity evaluation, the application for which we present experimental results, colors of interest range from dark green to dark red, with green, orange, pink, and light red in between, as shown in Fig. 1.

C. Mapping Colors to 1-D Color Space

Since a limited number of colors are of interest for any given product, it is possible to map the range of colors of interest to a 1-D color space that uses a single value or index to represent each color in the specified range such that increasing values reflect increasing product preference. Using tomato maturity evaluation as an example, if the six major color groups (green, breakers, turning, pink, light red, and red) can be mapped into color values such as 25, 60, 90, 130, 170, and 230, respectively, then the color grade boundaries can be set by selecting a

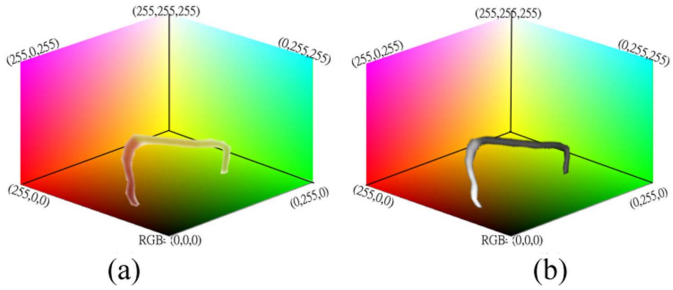

Fig. 3. (a) Colors of interest. (b) Mapping colors to 1-D color space.

cutoff point for each grade. Fig. 2 shows the converted 1-D color index values of the color bar in Fig. 1 interpreted on a 256-level grayscale. This also allows the user to fine tune color preferences corresponding to "slightly redder" or "slightly greener" by simply sliding the appropriate threshold.

Fig. 3 illustrates the essential idea of our new technique. Fig. 3(a) shows the colors of interest (shown as a narrow bending tube) for tomato color grading in RGB color space. Each color of interest in the tube has its own unique RGB value. As the figure shows, only a very small subset of 16M colors in the tube is of interest in the grading of tomatoes; purple or blue tomatoes are not likely to turn up during grading, and colors not present in tomatoes can be excluded. Fig. 3(b) shows an example of the resulting color values or indices ranging from 0 to 255. (Displayed images interpret the 8-bit index value on a 256-level grayscale.) Note that lower values represent green fruit, while higher values represent redder fruit that is more mature.

It is important to note that, as shown in the next subsection, the proposed color mapping is a least squares solution. This means that, although not all colors can be considered and used for calibration, colors that vary slightly from the defined colors will still be mapped to their closest indices. The diameter of the bending tubes can be larger as shown in Fig. 3 to accommodate slight color variations. Moreover, this approach can also be used for color segmentation (to separate fruit from background) by simply assigning, say, color index 255 to pixels with colors very close to the background that should be excluded from color grading. Colors outside the selected color range are very likely to be mapped to either negative values (which are truncated to 0) or very high values (which are capped at 255 in our implementation). Finally, this color mapping method can be easily used to detect surface defects, discoloration, or blemishes by simply assigning, say, color index 0 for unwanted colors. These three functionalities will be demonstrated in Sections III and IV.

D. Direct Color Mapping

The proposed color mapping method maps the RGB values of colors of interest into color indices using a polynomial equation. After careful experiments, we chose to employ a third-order term $(\mathrm{R}\times\mathrm{G}\times\mathrm{B})$ and the full rank of second-order polynomials

($\mathrm{R}^2, \mathrm{G}^2, \mathrm{B}^2, \mathrm{R}\times\mathrm{G}, \mathrm{R}\times\mathrm{B}, \mathrm{G}\times\mathrm{B}, \mathrm{R}, \mathrm{G}, \mathrm{B}$, and a constant value) to convert 3-D colors to 1-D color indices. Using higher than third-order terms increases computation complexity but does not improve accuracy for our applications. However, based on our 3-D to 1-D color mapping concept, we believe that higher order or even nonlinear functions could perform as well as our formula or possibly needed for applications that color preferences vary in different range of colors of interest. However, the improvement from using a more complicated formula for our example applications is probably not worth the extra computations. We use the calibration results presented in Section III to prove that higher order terms are not needed.

The formula of this conversion is shown in (1) as

$$\begin{aligned}\text{Color Index} = {} & c_1 \cdot R \cdot G \cdot B \\ & + \sum_{i=2}^{7} c_i \cdot R^l \cdot G^m \cdot B^n \\ & + \sum_{i=8}^{10} c_i \cdot R^o \cdot G^p \cdot B^q + c_{11} \end{aligned} \tag{1}$$

where l, m, n, o, p, and q are all greater than or equal to 0 and $l+m+n=2$ to form all second-order terms and $o+p+q=1$ to form all first-order terms. Note that a total of 11 coefficients, $c_1, c_2, c_3, \ldots, c_{11}$, must be specified to use this formula. These coefficients are obtained through calibration using a selected set of colors of interest in RGB values and a set of desired linear color indices. Note that the colors of interest are specific to each application, as is the corresponding set of coefficients that result.

Once the coefficients are obtained, each pixel having a color in the predefined range will be represented with a real-number value (but normalized to an integer between 0 and 255) in the resulting linear space. Colors differing slightly from the predefined colors will be mapped to the closest value. Only a small set of the full range of 16M colors will be condensed and represented by a set of color indices. This reduction in resolution is possible only because a large number of colors do not occur in any given application.

E. Calibration

As noted above, one set of preselected colors of interest must be used to calculate the coefficients. At least 11 sample colors selected from the application's color range are needed to solve for the 11 coefficients. Each sample color provides a distinct set of RGB values for calibration. Equation (1) with multiple sample colors can be written in matrix form as shown in (2) at the bottom of the page.

In the equation, R_i, G_i, and B_i for $i = 0, 1, 2, \ldots, n$ are the RGB values of n preselected color samples and $v_1, v_2, v_3, \ldots, v_n$ are the corresponding desired linear color indices. The total number of rows (n) depends on how many sample colors are used. By setting the corresponding desired 1-D color indices according to the requirements of the specific application, the coefficients can be obtained by solving (2).

The coefficients can be calculated using a least-squared error method. Direct computation of the pseudo-inverse for the least-squared error method is numerically inferior to computation using a matrix factorization such as LU factorization, Cholesky factorization, QR factorization, or Singular Value Decomposition (SVD). LU is applied to general square matrices in a Gaussian elimination sense. Cholesky factorization is for symmetric, positive definite matrices, while QR factorization is for rectangular matrices. SVD provides a robust solution for both overdetermined and under-determined least-squared problems [41].

SVD was selected because a single solution of (2) may not exist due to imperfect selection of sample colors and desired color value. Moreover, it allows the use of more than 11 sample colors to solve for the best solution. A given matrix A $(m \times n)$ can be factored as $\mathrm{A} = \mathrm{U}\Sigma\mathrm{V}^{\mathrm{H}}$, where $\mathrm{U}(m\times m)$ and $\mathrm{V}(n\times n)$ are both unitary matrices, and $\Sigma(m \times n)$ has the form $\Sigma = \mathrm{diag}(\sigma^1, \sigma^2, \ldots, \sigma^P)$, where $p = \min(m, n)$. The diagonal elements of Σ are called the singular values of A and usually ordered so that $\sigma^1 \geq \sigma^2 \geq \cdots \geq \sigma^P \geq 0$. Matrix Σ is written as a diagonal matrix, even though it is not a square matrix if $m \neq n$. The elements $\sigma^1, \sigma^2, \ldots, \sigma^P$ are written along the main diagonal, and rows or columns of zeros are appended as necessary to obtain the proper dimension for Σ if it is not a square matrix $(m \neq n)$.

Suppose that $m \geq n > r = \mathrm{rank}(\mathrm{A})$. Then, the economy SVD of A is $\mathrm{A} = \mathrm{U}_1\Sigma_1\mathrm{V}_1^{\mathrm{H}}$, where U_1 is the first r columns of U, and Σ_1 is an $r \times r$ diagonal matrix with $\sigma^1 \geq \sigma^2 \geq \cdots \geq \sigma^r \geq 0$. $\mathrm{V}_1^{\mathrm{H}}$ is the first r rows of V^H. If A is nearly rank-deficient, some of its singular values may be close to zero. When taking the inverse of Σ_1, the last few elements of ${\Sigma_1}^{-1}$ could be very large. To avoid numerical problems, these large elements are set to zero in forming the pseudo-inverse. This is equivalent to replacing A with a lower-rank estimate [41].

III. Tomato Maturity Evaluation

This section presents results obtained when our novel color-mapping scheme was applied to automated color grading for tomato maturity evaluation. The results demonstrate the robustness of the proposed approach.

A. Calibration Results

Thirteen sample colors were selected to calibrate the system for tomato maturity evaluation. Sample colors were arranged

$$\begin{bmatrix} R_1G_1B_1 & R_1^2 & G_1^2 & B_1^2 & R_1G_1 & R_1B_1 & G_1B_1 & R_1 & G_1 & B_1 & 1 \\ R_2G_2B_2 & R_2^2 & G_2^2 & B_2^2 & R_2G_2 & R_2B_2 & G_2B_2 & R_2 & G_2 & B_2 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ R_nG_nB_n & R_n^2 & G_n^2 & B_n^2 & R_nG_n & R_nB_n & G_nB_n & R_n & G_n & B_n & 1 \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_{11} \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix} \tag{2}$$

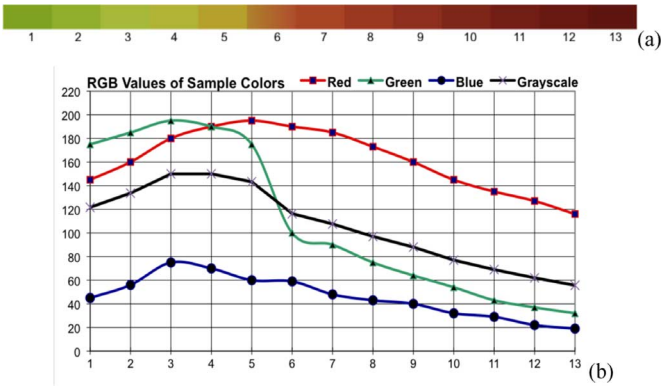

Fig. 4. (a) Thirteen sample colors ranging from green to dark red and (b) their RGB and grayscale values.

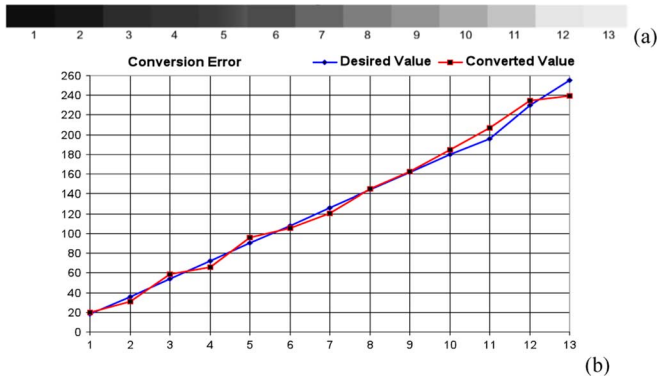

Fig. 5. Comparison between the desired values and the converted values.

from green to red, as shown in Fig. 4(a). Corresponding RGB and grayscale values are shown in Fig. 4(b). An experienced human grader manually selected tomato samples at different maturity stages and ranked them according to the six-stage USDA standard described in Section II-A. These 13 samples were images and their RGB values were used to create a 13×11 matrix as shown in (2), with one row per sample. The colors (rows) were arranged in the order of color preference, from least mature (dark green) to most mature (dark red) fruit.

The human expert specified a desired linear color index in the range from 0 to 255 for each color sample. The range of 0 to 255 (an 8-bit integer) was selected for ease of display and subsequent image processing steps. Wider ranges such as 16 bits could be used if desired. The resulting values were arranged into a 1×13 matrix with values $[18\ 36\ 54\ 72\ 90\ 108\ 126\ 144\ 162\ 180\ 196\ 230\ 255]^{\mathrm{T}}$—corresponding to vector v in (2). These values are assigned by the user according to the selected color samples. The coefficients were then obtained by solving these 13 equations using SVD, resulting in the values: $[-0.000250, -0.014245, 0.017014, 0.258901, 0.052357, -0.086940, -0.052598, 5.958549, -8.903804, -2.089897, 0.118389]$.

We then used these coefficients to convert the sample colors to 1-D color indices to evaluate the conversion accuracy. Fig. 5(a) shows the image that results if each resulting index is interpreted as a 256-level grayscale value. As can be seen, the samples transition smoothly from black to white. Fig. 5(b) shows the desired linear color indices (v_1 to v_{13} in (2)) that are selected for calibration (in blue) and the resulting linear color indices (in red) that are converted using (1). There exist small discrepancies between the desired and the converted values because the conversion coefficients are obtained as a least-square solution. Although including high order terms in (1) and (2) could potentially eliminate these small discrepancies, the small improvement in accuracy is not worth the added complexity in computations.

For the 13 data points, the average error was 4.5 units on the color index scale. As we will see, this error is small enough to allow very accurate color grading. We conclude that our approach using the third-order polynomial works very well for this application. In short, it converted a range of 24-bit colors of interest into linear indices that are a close reflection of the color preferences of an experienced user.

Note that none of the four curves (R, G, B, and grayscale) shown in Fig. 4(b) exhibits a one-to-one relationship between their values and sample colors, which means that using only grayscale or any individual color channel for color grading will not work. For example, both sample colors 3 and 4 have the same grayscale value of 150, while sample 2 has the same grayscale value (134) as a color between colors 5 and 6. Similarity, colors 4 and 6 have the same red value (190) and sample color 2 has a red value very close to that of sample color 9. Green and blue channels exhibit the same problem. This multiple-to-one relationship causes confusion in determining color grades using a single channel. This problem would be more severe for other fruits or applications that have the four curves vary their values more or intersect with each other more than those shown in Fig. 4(b). On the other hand, the resulting color index curve using our direct color mapping method shown in Fig. 5(b) exhibits a one-to-one mapping, ensuring that every color in the defined color range has a unique color index value.

B. Color Grading Results

This direct color mapping method was implemented using OpenCV (an open source image function library) in a machine vision system for the color-based grading of fruit intended for fresh produce markets. Two major applications that have shown impressive success are the grading of apples and tomatoes. In both cases, color grade was evaluated based on a combination of color value and the consistency of the surface color. The following discussion focuses on tomato maturity evaluation.

Grading is performed by first converting the input color image into an 8-bit image (pixel values ranging from 0 to 255) in the new linear color space. Tomatoes are graded based on their color value (average of color indices) and their color consistency (the inverse of the standard deviation of the histogram normalized to 100%). Color value and color consistency cutoff points for each grade are set by the user and can be adjusted easily by sliding bars in graphical user interface (GUI). We set up the cutoff points to separate tomatoes into 8 grades (2 more than the USDA standard) to show the accuracy of the system.

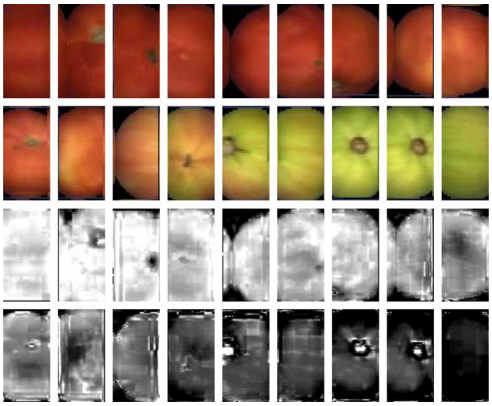

Fig. 6. Tomato images and their corresponding color values in the 1-D linear color space.

TABLE I
AVERAGE COLOR VALUES, COLOR CONSISTENCY, AND GRADE

color index	229	219	215	202	190	185	178	162	140
consistency	88	66	72	75	73	72	73	58	62
grade	A	A	A	A	B	B	B	C	D
color index	136	130	101	80	71	57	46	40	11
consistency	68	55	60	74	61	74	60	63	90
grade	D	D	D	F	G	H	H	H	H

Fig. 6 shows 18 color images that were captured to demonstrate the accuracy of the system. Some images include a small portion of another piece of fruit. These rare cases happen when two pieces touching each other. Since it is only a very small portion, it has minimal effect on system performance. These 18 images were arranged from dark red to dark green in the order of their maturity stage. The tomatoes were to be graded into eight maturity levels, more accurate than the USDA standard. As previously explained, finding thresholds or cutoff points for these RGB images in 3-D RGB color space would be very difficult. The top two rows show the original images in the level of maturity. Images at corresponding locations in the bottom two rows show the converted images in the linear color space, interpreted as 256-level grayscale images.

Tomato maturity was calculated according to the average color index and color consistency, as described above. Table I shows these measurements and the resulting grades using the color preference settings selected by the operator. Measurements and the grade in Table I correspond to the locations of test images shown in Fig. 6. Using the functionality described in Section II-C, colors of visible stem areas were intentionally mapped to 0 so that they were excluded from color consistency. Colors of those stem pixels that were close to defined colors and not mapped to 0 resulted in lower color consistency than those fruit for which the stem is not visible. However, the overall grading results—based on both color index and consistency averages—are consistent with human grading.

This new color mapping technique has been implemented and proven successful for commercial production of tomato grading for maturity. Using a 640×480 color camera to cover a $8'' \times 6''$ area (80 pixels per inch), the system is able to meet the USDA standard to separate tomatoes into at least six groups with at least 95% accuracy. The system also meets the industry requirement of grading at least 48 pieces of fruit per second using a single commercial personal computer (Pentium-IV 3.0 GHz). Higher throughput can be achieved with a more powerful multicore system or high-performance computing platforms such as Field Programmable Gate Arrays (FPGAs) or Graphics Processing Units (GPUs).

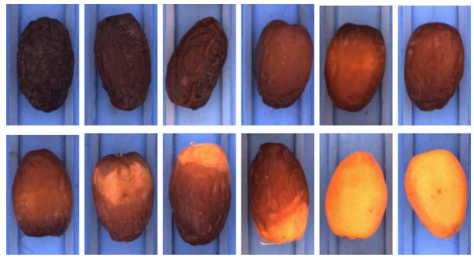

Fig. 7. Different date maturity stages.

Fig. 8. Colors of interest and converted color index values.

IV. DATE MATURITY EVALUATION

Our direct color mapping method was also used for date maturity evaluation to demonstrate its robustness in measuring color and segmenting fruit area from the background. A color grading vision system based on this method was developed and installed at a date processing facility for commercial use. The system using a commercial PC is capable of processing 60 dates per second. Fig. 7 shows dates from dark red (the most mature stage), to orange (medium stage), to yellow (least mature).

A. Calibration Results

For date maturity evaluation, as shown in Fig. 8, colors range from yellow to dark red, with orange and light red in between. Many colors such as green and blue will not occur and are excluded. Using the new direct color mapping method, the resulting color indices are shown in Fig. 8. The preferred color (dark red) is assigned the highest color index value. There are striations in the color indices in Fig. 8 (as well as Fig. 2). These are caused by the small discrepancies explained in Section III-A and displayed in Fig. 5 and the round off error for display.

Thirteen date samples with colors ranging from yellow to red as shown in Fig. 8 were selected by an experienced grader to calibrate the system. This set of 13 RGB values was selected to cover the full range of colors at all maturity stages. The values were used to create a 13×11 matrix, with one row per sample color. The colors (rows) were arranged in the order of color preference, from least mature to most mature fruit. For each color sample, a human expert specified a desired color index in the range from 0 to 235. Index values for fruit colors were limited to 235 to distinguish the fruit pixels from shadow pixels on the blue carrier that are darker than fruit. Color index values close to 255 are reserved for these shadow pixels. The full range (0 to 255) was used for tomatoes because the tomatoes do not have dark (close to black) colors. The desired index values were arranged into a 1×13 matrix

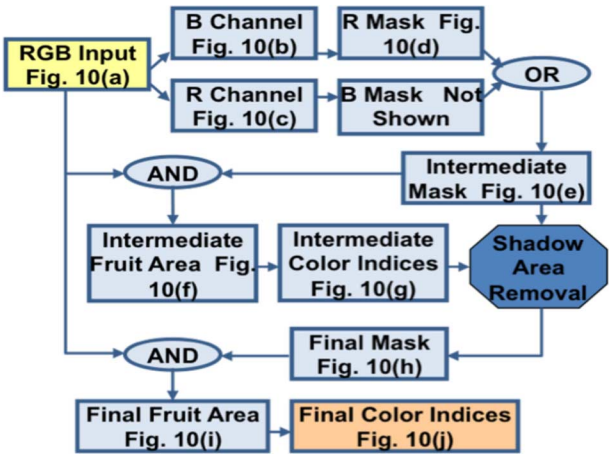

Fig. 9. Flowchart of fruit detection process.

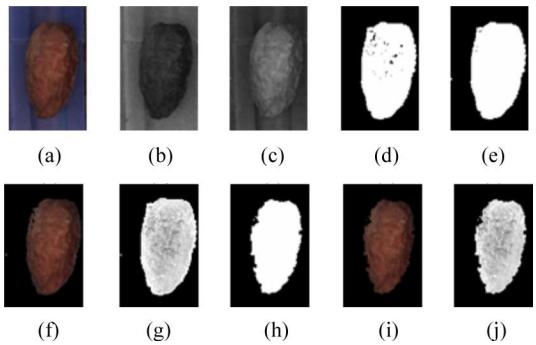

Fig. 10. (a) Original image. (b) Blue channel. (c) Red channel. (d) Binary image of blue channel. (e) Binary image of blue ORed with binary image of red. (f) Segmented fruit image using (e) as a mask. (g) Color indices of (f). (h) Binary image after filtering out dark blue background. (i) Segmented fruit image using (h) as a mask. (j) Color indices of (i).

with values [18 36 54 72 90 108 126 144 162 180 196 220 235]. The 13 equations were then solved using SVD (as described in Section III), resulting in these 11 optimal coefficient values: $[0.000955, 0.000541, 0.018773, 0.25405, -0.025364, -0.055644, -0.391265, 0.510995, 10.321283, 2.867669, 0.080986]$. These coefficients were used to convert RGB color values to color indices for further processing.

B. Fruit Segmentation

Fruit area must be separated or segmented from the background before color analysis is carried out. As shown in Fig. 7, a blue plastic belt was chosen to build the fruit carrier for imaging for two reasons. First, blue does not occur naturally in dates. Second, blue is one of the three channels in the RGB color space, making it easier to filter than colors that require multiple channels to represent. Fruit is segmented from the blue background by creating masks corresponding to the fruit area using the blue and red channels. The new color mapping method is then used to remove areas of shadow and increase the accuracy of the mask. Fig. 9 summarizes the segmentation steps, and Fig. 10 shows results on sample images.

Fig. 10(a) shows the original RGB image. The blue and red channels [Figs. 10(b) and (c), respectively] are first binarized to generate blue and red masks. For each pixel in the image, the corresponding value in the blue mask [Fig. 10(d)] is set to 0 if the value of the blue channel is high (blue background), otherwise it is set to 255 (fruit). Values in the red mask (not shown) are set to 255 if the red channel is high (fruit), otherwise, they are set to 0. The two masks are then combined using a logical binary OR operation to create an intermediate mask [Fig. 10(e)] without the holes and missing areas commonly seen in individual blue and red masks [see Fig. 10(d)]. By using the logical binary AND operator on corresponding pixel and mask values, the intermediate mask is used to select the fruit area from the original RGB image [Fig. 10(f)].

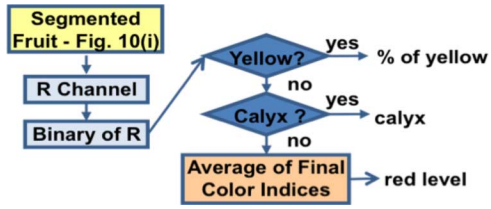

Fig. 11. Flowchart of maturity evaluation algorithm.

At this point, pixels in the fruit area can be converted to color index values using the proposed color space conversion method [Fig. 10(g)], but it can be seen (e.g., upper left portion of date) that the fruit area includes pixels from shadows cast by the fruit on the carrier that are close in color to dark red fruit. It is desirable to remove these shadow pixels before color analysis as they would adversely affect grading accuracy. Using the technique described in Section II-C, the system is calibrated in such a way that these shadow colors are converted to a color index of 255. A final mask [Fig. 10(h)] is generated by removing areas of shadow from the intermediate mask. This is accomplished by setting mask pixels to 0 if the corresponding pixels in Fig. 10(g) have a color index of 255 (shadow). Occasionally, some dark red pixels inside the fruit area are removed mistakenly because they are indistinguishable from shadow pixels, but these voids are filled in by a later step that includes connected component analysis. Using the final mask with shadow areas removed, the fruit area can be accurately segmented [Fig. 10(i)] and the proposed color space conversion method can be used to obtain final color indices, as shown in Fig. 10(j).

C. Measuring Maturity

The date maturity evaluation process called presort or field sort is done right after harvest. The goal of this evaluation is to determine the subsequent drying process as discussed in Section I. Fig. 11 shows the flowchart of the algorithm for maturity evaluation. The algorithm starts with the segmented fruit image as shown in Fig. 10(i). Since yellow fruit has a strong red color component, a binarization based on the R channel can reliably detect the yellow regions of the fruit. Fruit that is categorized as yellow is least mature and should be sent to a heated building to dry. Here the percentage of yellow surface is used to determine how long they should be dried. Fruit with very different yellow percentages must be separated because they require different lengths of time to dry and mature.

The calyx area of fruit (the end opposite the stem) often has a bright red color component. This area is generally very small

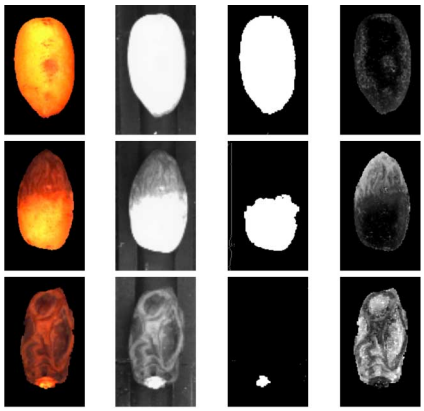

Fig. 12. Yellow fruit and fruit with calyx.

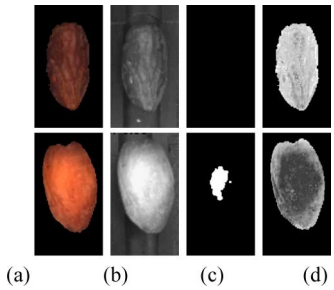

Fig. 13. Red and orange fruit.

(3 to 10 pixels). It is equivalent to 0.0375″ to 0.125″ with an 80 pixel-per-inch resolution. It usually appears at the very top or bottom of the fruit in the image. Based on these characteristics, the calyx can be detected based on the size and location of the binarized red component. After yellow and the calyx are detected, the average of fruit color indices shown in Fig. 10(j) is calculated to obtain the red level (dark or light) which is a good indicator of whether they are ready or should be dried in the sun and for how long.

D. Maturity Evaluation Results

Fig. 12 shows the processing result of yellow fruit and fruit with calyx. Column (a) in Fig. 12 shows the segmented fruit image obtained using the fruit segmentation algorithm described in Section IV-B. Column (b) shows the R channel of each image. Notice the high intensity in the yellow and calyx regions. A user-selected threshold (240 in our experiments) on red intensity values is able to separate the yellow regions from non-yellow regions (shown in Column (c)). Column (d) shows the color indices obtained using our color quantization method. Fruit with yellow area more than a user-selected threshold (13% in our experiments) are considered yellow fruit, and their maturity level was evaluated based on the percentage of yellow area of the whole fruit. Fruit pieces with a very small yellow area (less than 2%) near the top or bottom end of the fruit were categorized as "calyx." The calyx area is considered a defect and pieces so classified must be separated from the rest of the fruit. Fruit that are not considered "yellow" or "calyx" will undergo further processing.

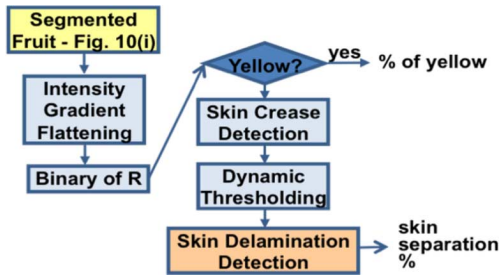

Fig. 14. Flowchart of skin quality evaluation algorithm.

Fruit with the yellow area smaller than 13% and not on the end of the fruit are considered either red, orange, or having a skin separation surface defect. Fig. 13 shows the processing result of red and orange fruit samples. As shown in Fig. 13, red (ready for packing) fruit has 0% yellow, while orange fruit has a small yellow area generally near the center of the fruit (clearly not a calyx). However, fruit with surface defects could also have a yellow area smaller than 13% (including 0%). Column (d) of Fig. 13 shows the color indices of the two samples. It can be seen that color indices of the red fruit are very high (bright) and color indices of the orange fruit are much lower (darker). Fruit that are obviously red or orange can be separated from each other by examining the average of the fruit color indices. A high threshold of 200 was selected in our experiments to detect red fruit. A low threshold of 140 was selected to separate orange fruit.

A total of 664 yellow or partial yellow, 320 orange, 322 red, and 812 defective Medjool dates at differing maturity stages were randomly selected for testing. All yellow fruit pieces were correctly categorized and the percentage of yellow was calculated with fairly high accuracy (relative to a conventional visual inspection). Moreover, 94.7% of the red fruit, 86.8% of orange fruit, and 89.6% of the fruit with skin separation were accurately classified. In both accuracy and consistency, the performance of this color grading system exceeded that of manual grading during production. This human error is mainly due to fatigue and the level of supervision, which is why automation is so important for maintaining high product quality.

V. Date Surface Defect Detection

Fruit pieces that are classified as either red or light red (ready for packing) are then processed further to evaluate skin quality. In a similar fashion to maturity evaluation, skin quality is evaluated based on the percentage of the fruit surface with separated/detached skin. This skin quality evaluation process is also called grading or packing.

A. Detecting Skin Separation

The same calibration process discussed in Section IV-A was performed before the skin separation detection process. Fig. 14 shows the flowchart of the skin quality evaluation algorithm. It includes three steps: intensity gradient flattening, yellow spot detection, and skin separation detection. Most fruit surfaces are round. Fruit surface curvature causes the light to reflect differently on the fruit surface. Image brightness on the edge of the fruit is darker than the brightness in the middle of the fruit.

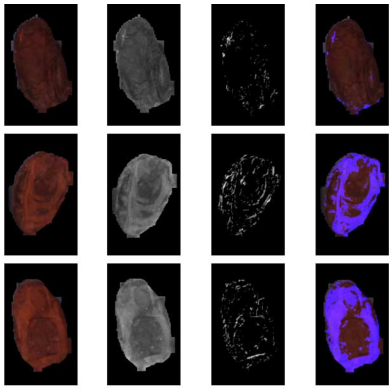

Fig. 15. Fruit with and without skin separation surface defect.

This effect can be observed easily in the yellow fruit (lower right in Fig. 7). In order to detect skin separation accurately, this brightness gradient must be corrected or flattened. Our gradient compensation algorithm [42] was adopted here to compensate for this effect as the first step of the skin quality evaluation algorithm.

Some ripe fruit contains a small spot of yellow and is considered low quality fruit. Since yellow color has a strong red component, a binarization based on the R channel can reliably detect the yellow spot of the fruit. If the yellow spot (percentage of the entire fruit surface) exceeds a predefined percentage (programmable by the operator), then the fruit is considered yellow and low quality. Fruit that is not considered yellow will then be processed for skin separation detection.

Ripe fruit with high percentages of separated skin may appear similar in color to unripe (orange) fruit with no skin separation. The two are distinguished from each other by an additional processing step called skin crease detection. A skin crease or fold can be detected based on the contrast of the fruit color indices of nearby pixels, as shown in Fig. 10(j). The Canny edge operator is applied to the R channel image. The resulting number of edge pixels is calculated and its ratio to the entire fruit surface can be obtained. An adaptive threshold is determined based on the percentage of skin crease. A high skin crease percentage results in a low skin detection threshold, which detects more skin separation than a high threshold. Skin separation percentage (of the entire fruit surface) can be calculated to determine fruit quality (lower is better).

B. Defect Detection Results

Fruit that are not categorized as yellow are distinguished from each other by quantifying the amount of skin separation. Fig. 15 shows one sample of orange fruit (top row) and two samples of fruit with skin separation defects. Column (b) of Fig. 15 shows the color index images of the fruit samples. As can be seen, skin separation appears as a "crease" that is apparent in the color index image. The Canny edge operator was used to detect the crease and the result is shown in column (c). The orange fruit in Fig. 15 (top row) has far fewer creases in the color index image. The other fruit samples in Fig. 15 have more creases that reflect the amount of skin separation. However, each crease represents just the boundary edge of a skin region. To determine the actual amount of skin separation, the color index images (column (c) of Fig. 15) are thresholded using an adaptive threshold determined by the amount of crease. Column (d) shows detected skin separation areas superimposed on the original images. The severity of the skin separation defect is evaluated as the percentage of skin that is separated.

A color grading vision system based on this method was developed and installed at a date processing facility for testing. The system using a commercial PC is capable of processing 60 dates per second.

VI. Conclusion

We have presented a new color mapping concept of converting 3-D color spaces to 1-D color indices for automated color grading. Our approach is based on the use of a third-order polynomial to convert 3-D RGB values into a simple 1-D color space for our example applications. However, based on our 3-D to 1-D color mapping concept, higher order polynomials or nonlinear functions could perform as well as our formula or possibly needed for applications that color preferences vary in different range of colors of interest.

Unlike other color grading techniques, this approach makes the selection and adjustment of color preferences much easier. The user can change color grade thresholds in a manner consistent with human color perception, simply sliding a cutoff point to include fruit that is "slightly darker," "a bit lighter green," or "lighter red." Furthermore, these changes in preferred color ranges (defined entirely by the user) can be completed without reference to precise reference colors.

The implementation of this new color mapping technique and the results presented in Sections III–V demonstrate the simplicity and accuracy of this method. It is simple but effective. All that the system requires is that a set of colors of interest be selected by an experienced grader, and that each color be assigned a desired color index value on a linear scale. Provided that the selected color samples cover the complete range of colors that can occur, accurate color grading can be performed.

A fully automated date maturity evaluation system based on the proposed method has been installed and operated for commercial production since August 2009. Another system for date surface defect detection is being tested. This new direct color mapping concept can be applied to a variety of color grading applications that require the easy of setting and adjustment of color preferences.

There are many color grading applications can utilize the proposed color mapping method. Future improvements include exploring the possibility of using other statistical analysis methods such as color index histogram analysis, using median (instead of average) value for color quality. Based on the converted color indices, different analysis methods can be used for specific applications.

References

[1] A. Bouganis and M. Shanahan, "A vision-based intelligent system for packing 2-D irregular shapes," *IEEE Trans. Autom. Sci. Eng.*, vol. 4, no. 3, pp. 382–394, Jul. 2007.

[2] H. C. Garcia and J. R. Villalobos, "Automated refinement of automated visual inspection algorithms," *IEEE Trans. Autom. Sci. Eng.*, vol. 6, no. 3, pp. 514–524, Jul. 2009.

[3] H. C. Garcia, J. R. Villalobos, and G. C. Runger, "An automated feature selection method for visual inspection systems," *IEEE Trans. Autom. Sci. Eng.*, vol. 3, no. 4, pp. 394–406, Oct. 2006.
[4] S. M. Bhandarkar, X. Luo, R. F. Daniels, and E. W. Tollner, "Automated planning and optimization of lumber production using machine vision and computed tomography," *IEEE Trans. Autom. Sci. Eng.*, vol. 5, no. 4, pp. 677–695, Oct. 2008.
[5] Y. H. Anis, M. R. Holl, and D. R. Meldrum, "Automated selection and placement of single cells using vision-based feedback control," *IEEE Trans. Auto. Sci. Eng.*, vol. 7, no. 3, pp. 598–606, Jan. 2010.
[6] Z. Du, M. K. Jeong, and S. G. Kong, "Band selection of hyperspectral images for automatic detection of poultry skin tumors," *IEEE Trans. Autom. Sci. Eng.*, vol. 4, no. 3, pp. 332–339, Jul. 2007.
[7] N. M. Kwok, Q. P. Ha, D. Liu, and G. Fang, "Contrast enhancement and intensity preservation for gray-level images using multiobjective particle swarm optimization," *IEEE Trans. Autom. Sci. Eng.*, vol. 6, no. 1, pp. 145–155, Jan. 2009.
[8] H. Y. T. Ngan and G. K. H. Pang, "Regularity analysis for patterned texture inspection," *IEEE Trans. Autom. Sci. Eng.*, vol. 6, no. 1, pp. 131–144, Jan. 2009.
[9] Y. Cheng and M. A. Jafari, "Vision-based online process control in manufacturing applications," *IEEE Trans. Autom. Sci. Eng.*, vol. 5, no. 1, pp. 140–153, Jan. 2008.
[10] C. Boukouvalas, J. Kittler, R. Marik, and M. Petrou, "Color grading of randomly textured ceramic tiles using color histograms," *IEEE Trans. Ind. Electron.*, vol. 46, no. 1, pp. 219–226, Feb. 1999.
[11] J. Ai, D. Liu, and X. Zhu, "Combination of wavelet analysis and color applied to automatic color grading of ceramic tiles," in *Proc. IEEE Int. Conf. Pattern Recognit.*, 2004, vol. 3, pp. 235–238.
[12] J. Faria, T. Martins, M. Ferreira, and C. Santos, "A computer vision system for color grading wood boards using fuzzy logic," in *Proc. IEEE Int. Symp. Ind. Electron.*, June 2008, pp. 1082–1087.
[13] A. Zukauskas, R. Vaicekauskas, F. Ivanauskas, H. Vaitkevicius, P. Vitta, and M. S. Shur, "Statistical approach to color quality of solid-state lamps," *IEEE J. Sel. Topics Quantum Electron.*, vol. 15, no. 6, pp. 1753–1762, Nov. 2009.
[14] P. Fiorentin, E. Pedrotti, and A. Scroccaro, "A multispectral imaging device for monitoring of color in art works," in *Proc. IEEE Instrument. Meas. Technol. Conf.*, May 2009, pp. 356–360.
[15] N. Jamil, A. Mohamed, and S. Abdullah, "Automated grading of palm oil fresh fruit bunches (FFB) using neuro-fuzzy technique," in *Proc. Int. Conf. Soft Comput. Pattern Recognit.*, Dec. 2009, pp. 245–249.
[16] X. Sun, H. J. Gong, F. Zhang, and K. J. Chen, "A digital image method for measuring and analyzing color characteristics of various color scores of beef," in *Proc. Int. Conf. Image and Signal Process.*, Oct. 2009, pp. 1–6.
[17] J. R. Mathiassen, E. Misimi, and A. Skavhaug, "A simple computer vision method for automatic detection of melanin spots in Atlantic salmon fillets," in *Proc. Int. Conf. Mach. Vision and Image Process.*, Sep. 2007, pp. 192–200.
[18] H. S. Lee, "Objective measurement of red grapefruit juice color," *J. Agric. Food Chem.*, vol. 48, no. 5, pp. 1507–1511, Apr. 2000.
[19] K. M. Lee, Q. Li, and W. Daley, "Effects of classification methods on color-based feature detection with food processing applications," *IEEE Trans. Autom. Sci. Eng.*, vol. 4, no. 1, pp. 40–51, Jan. 2007.
[20] K. M. Lee, W. Daley, and Q. Li, "Artificial color contrast for machine vision and its effects on feature detection," in *Proc. AIM*, Monterey, CA, Jul. 2005, pp. 219–224.
[21] N. Kondo, "Fruit grading robot," in *Proc. Int. Conf. Advanced Intell. Mechatronics (AIM)*, Jul. 2003, vol. 2, pp. 1366–1371.
[22] H. Ji and J. Yuan, "The application study of apple color grading by particle swarm optimization neural networks," in *Proc. 6th World Congr. Intell. Control Autom.*, 2006, vol. 1, pp. 2651–2654.
[23] K. Nakano, "Application of neural networks to the color grading of apples," *Comput. Electron. Agriculture*, vol. 18, no. 2–3, pp. 105–116, Aug. 1997.
[24] X. Zou, J. Zhao, and Y. Li, "Apple color grading based on organization feature parameters," *Pattern Recognit. Lett.*, vol. 28, no. 15, pp. 2046–2053, Jun. 2007.
[25] F. Guo and Q. Cao, "Study on color image processing based intelligent fruit sorting system," in *Proc. World Congr. Intell. Control Autom.*, Jun. 2004, pp. 4802–4805.
[26] K. Tu, K. Ren, L. Pan, and H. Li, "A study of broccoli grading system based on machine vision and neural networks," in *Proc. Int. Conf. Mechatronics Autom.*, Sep. 2007, pp. 2332–2336.
[27] Y. Chherawala, R. Lepage, and G. Doyon, "Food grading/sorting based on color appearance trough machine vision: The case of fresh cranberries," in *Proc. Inform. Commun. Technol.*, Oct. 2006, vol. 1, pp. 1540–1545.
[28] C. A. Gunawardena, L. J. Clark, and T. J. Dennis, "A spot-type defect detection and colour identification system for agricultural produce," in *Proc. Int. Conf. Ind. Electron., Control, Instrumentation*, 1991, vol. 3, pp. 2531–2534.
[29] C. A. Gunawardena, T. J. Dennis, and L. J. Clark, "Color identification and quality inspection system for agricultural produce," in *Proc. 33rd Midwest Symp. Circuits and Systems*, Aug. 1990, pp. 657–660.
[30] W. M. Syahrir, A. Suryanti, and C. Connsynn, "Color grading in tomato maturity estimator using image processing technique," in *Proc. Int. Conf. Comput. Sci. Inf. Technol.*, Aug. 2009, pp. 276–280.
[31] D. J. Lee, "Color space conversion for linear color grading," in *Proc. SPIE Intell. Robots Comput. Vision XIX*, 2000, vol. 4197, pp. 358–366.
[32] F. Y. A. Rahman, S. R. M. S. Baki, A. I. M. Yassin, N. M. Tahir, and W. I. W. Ishak, "Monitoring of watermelon ripeness based on fuzzy logic," in *Proc. World Congr. Comput. Sci. Inf. Eng.*, July 2009, vol. 6, pp. 67–70.
[33] N. Mustafa, N. Fuad, S. Ahmed, A. Abidin, Z. Ali, W. Yit, and Z. Sharrif, "Image processing of an agriculture produce: Determination of size and ripeness of a banana," in *Proc. Int. Symp. Inf. Technol.*, Aug. 2008, pp. 1–7.
[34] D. J. Lee, J. K. Archibald, Y. C. Chang, and C. R. Greco, "Robust color space conversion and color distribution analysis techniques for date maturity evaluation," *J. Food Eng.*, vol. 88, no. 3, pp. 364–372, Oct. 2008.
[35] G. Rennick, Y. Attikiouzel, and A. Zaknic, "Machine grading and blemish detection in apples," in *Proc. IEEE Int. Conf. Signal Process. Its Appl.*, 1999, vol. 2, pp. 567–570.
[36] Y. Zhao, D. Wang, and Qian, "Machine vision based image analysis for the estimation of pear external quality," in *Proc. Int. Conf. Int. Comput. Technol. Autom.*, Oct. 2009, pp. 629–632.
[37] M. Recce, J. Taylor, A. Piebe, and G. Tropiano, "High speed vision-based quality grading of oranges," in *Proc. Int. Workshop on Neural Networks for Identification, Control, Robotics, and Signal/Image Processing*, Aug. 2002, pp. 136–144.
[38] D. J. Lee, Y. C. Chang, J. K. Archibald, and C. R. Greco, "Color quantization and image analysis for automated fruit quality evaluation," in *Proc. IEEE Conf. Autom. Sci. Eng. (CASE)*, 2008, pp. 194–199.
[39] H. Stokman and T. Gevers, "Selection and fusion of color models for image feature detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, pp. 371–381, 2007.
[40] K. H. Choi, G. H. Lee, Y. J. Han, and J. M. Bunn, "Tomato maturity evaluation using color image analysis," *Trans. ASAE*, vol. 38, no. 1, pp. 171–176, 1995.
[41] T. K. Moon and W. C. Stirling, *Mathematical Methods and Algorithms for Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 2000.
[42] D. J. Lee, J. K. Archibald, X. Q. Xu, and P. C. Zhan, "Using distance transform to solve real-time machine vision inspection problems," *Mach. Vision Appl. J.*, vol. 18, no. 2, pp. 85–93, Apr. 2007.

Dah-Jye Lee (SM'03) received the B.S. degree from the National Taiwan University of Science and Technology, Taipei, in 1984, and the M.S. and Ph.D. degrees in electrical engineering from Texas Tech University, Lubbock, in 1987 and 1990, respectively. He also received the M.B.A. degree from Shenandoah University, Winchester, VA, in 1999.

He is currently a Professor with the Department of Electrical and Computer Engineering, Brigham Young University (BYU), Provo, UT. He worked in the machine vision industry for 11 years prior to joining BYU in 2001. His research work focuses on medical informatics and imaging, hardware implementation of real-time vision algorithms and machine vision applications.

James K. Archibald (SM'05) received the B.S. degree in mathematics from Brigham Young University, Provo, UT, in 1981 and the M.S. and Ph.D. degrees in computer science from the University of Washington, Seattle, in 1983 and 1987, respectively.

He has been with the Department of Electrical and Computer Engineering, Brigham Young University, since 1987. His research interests include robotics, multiagent systems, and machine vision.

Dr. Archibald is a member of the ACM and Phi Kappa Phi.

Guangming Xiong received the B.E. degree in mechanical and electrical engineering (Honors) from Nanchang University, Nanchang, Jiangxi, China, in 1997 and the Ph.D. degree in mechanical and vehicular engineering from Beijing Institute of Technology, Beijing, China, in 2005.

His research interests include robotic vision, intelligent vehicle, mobile robots. He is a lecturer at the School of Mechanical Engineering, Beijing Institute of Technology, and is currently an Adjunct Research Professor at Brigham Young University, Provo, UT.