(19) Europäisches Patentamt / European Patent Office / Office européen des brevets

(11) **EP 2 638 508 B1**

(12) # EUROPEAN PATENT SPECIFICATION

(45) Date of publication and mention of the grant of the patent:
**07.07.2021 Bulletin 2021/27**

(21) Application number: **11840428.4**

(22) Date of filing: **27.09.2011**

(51) Int Cl.:
***G06K 9/00*** *(2006.01)*

(86) International application number:
**PCT/US2011/053438**

(87) International publication number:
**WO 2012/064416 (18.05.2012 Gazette 2012/20)**

(54) **SYSTEM AND METHOD FOR IDENTIFYING COMPLEX TOKENS IN AN IMAGE**

SYSTEM UND VERFAHREN ZUR IDENTIFIZIERUNG KOMPLEXER TOKENS IN EINEM BILD

SYSTÈME ET PROCÉDÉ PERMETTANT D'IDENTIFIER DES JETONS COMPLEXES DANS UNE IMAGE

(84) Designated Contracting States:
**AL AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HR HU IE IS IT LI LT LU LV MC MK MT NL NO PL PT RO RS SE SI SK SM TR**

(30) Priority: **10.11.2010 US 927244**

(43) Date of publication of application:
**18.09.2013 Bulletin 2013/38**

(73) Proprietor: **Tandent Computer Vision, LLC**
**San Francisco, California 94103 (US)**

(72) Inventor: **SMITH, Casey Arthur**
**Grand Junction, CO 81505-7041 (US)**

(74) Representative: **Reichert & Lindner**
**Partnerschaft Patentanwälte**
**Stromerstr. 2A**
**93049 Regensburg (DE)**

(56) References cited:

| | |
|---|---|
| **US-A1- 2007 177 797** | **US-A1- 2007 242 848** |
| **US-A1- 2007 242 878** | **US-A1- 2009 245 651** |
| **US-A1- 2010 278 448** | **US-A1- 2010 278 448** |
| **US-B1- 6 594 388** | **US-B1- 6 731 792** |
| **US-B1- 7 433 540** | **US-B1- 7 751 639** |

- **DREW ET AL.: 'Recovery of chromaticity image free from shadows via illumination invariance' IEEEWORKSHOP ON COLOR AND PHOTOMETRIC METHODS IN COMPUTER VISION, [Online] 2003, pages 32 - 39., XP055015434 Retrieved from the Internet: <URL:https://docs.goog!e.com/viewer?a=v&q=cache:6GmufJBAyfYJ:citeseerx.ist.psu.edu/v)ewdoc/d ownload?doi%3D10.1.1.124.2145%26rep%3Drep1% 26type%3Dpdf+log+chromaticity&hl=en&gl=us8pid=bl8srcid=ADGEESiSQh-pgdg -JX-Xm9gZe1 qH3pbrplwH4krqAmtb-IZwt410vTFlXcsZ3H66Q6k 3p HGFIqB941 kgZLRN7- PAYz6jMdPRD99qSgy6Yl> [retrieved on 2012-01-31]**
- **STONE ET AL.: 'Color Gamut Mapping and the Printing of Digital Color Images' ACM TRANSACTIONS ON GRAPHICS, [Online] vol. 7, no. 4, October 1988, pages 249 - 292, XP000600596 Retrieved from the Internet: <URL:http://dl.acm.org/citation.cfm?id=4804 5> [retrieved on 2012-01-31]**
- **TAPPEN ET AL.: 'Recovering Intrinsic Images from a Single Image', [Online] September 2002, CAMBRIDGE, MA, XP001512627 Retrieved from the Internet: <URL:http:l/18.7.29.232/bitstream/handle/1721.1/6703/AIM-2002-015.pdf?sequence=2> [retrieved on 2012-01-31]**
- **None**

EP 2 638 508 B1

## Description

Background of the Invention

**[0001]** Many significant and commercially important uses of modern computer technology relate to images. These include image processing, image analysis and computer vision applications. In computer vision applications, such as, for example, object recognition and optical character recognition, it has been found that a separation of illumination and material aspects of an image can significantly improve the accuracy of computer performance. Document US 2010/278448 provides in that respect a method for said separation according to techniques improved in the present specification.

Summary of the Invention

**[0002]** The present invention provides a method and system comprising image techniques that accurately and correctly identify regions of an image that each correspond to a single material reflectance in a scene depicted in the image. The invention is defined by the appended claims.

**[0003]** The present invention contemplates a computer readable media as any product that embodies information usable in a computer to execute the methods of the present invention, including instructions implemented as a hardware circuit, for example, as in an integrated circuit chip. The automated, computerized methods can be performed by a digital computer, analog computer, optical sensor, state machine, sequencer, integrated chip or any device or apparatus that can be designed or programed to carry out the steps of the methods of the present invention.

Brief Description of the Drawings

**[0004]**

Figure 1 is a block diagram of a computer system arranged and configured to perform operations related to images.

Figure 2 shows an n X m pixel array image file for an image stored in the computer system of figure 1.

Figure 3a is a flow chart for identifying Type C token regions in the image file of figure 2, according to a feature of the present invention.

Figure 3b is an original image used as an example in the identification of Type C tokens.

Figure 3c shows Type C token regions in the image of figure 3b.

Figure 3d shows Type B tokens, generated from the Type C tokens of figure 3c, according to a feature of the present invention.

Figure 4 is a flow chart for a routine to test Type C tokens identified by the routine of the flow chart of figure 3a, according to a feature of the present invention.

Figure 5 is a graphic representation of a log color space chromaticity plane according to a feature of the present invention.

Figure 6 is a flow chart for determining a list of colors depicted in an input image.

Figure 7 is a flow chart for determining an orientation for a log chromaticity space, according to a feature of the present invention.

Figure 8 is a flow chart for determining log chromaticity coordinates for the colors of an input image, as determined through execution of the routine of figure 6, according to a feature of the present invention.

Figure 9 is a flow chart for augmenting the log chromaticity coordinates, as determined through execution of the routine of figure 8, according to a feature of the present invention.

Figure 10 is a flow chart for clustering the log chromaticity coordinates, according to a feature of the present invention.

Figure 11 is a flow chart for assigning the log chromaticity coordinates to clusters determined through execution of the routine of figure 10, according to a feature of the present invention.

Figure 12 is a flow chart for detecting regions of uniform reflectance based on the log chromaticity clustering according to a feature of the present invention.

Detailed Description of the Preferred Embodiments

**[0005]** Referring now to the drawings, and initially to figure 1, there is shown a block diagram of a computer system 10 arranged and configured to perform operations related to images. A CPU 12 is coupled to a device such as, for example, a digital camera 14 via, for example, a USB port. The digital camera 14 operates to download images stored locally on the camera 14, to the CPU 12. The CPU 12 stores the downloaded images in a memory 16 as image files 18. The image files 18 can be accessed by the CPU 12 for display on a monitor 20, or for print out on a printer 22.

**[0006]** Alternatively, the CPU 12 can be implemented as a microprocessor embedded in a device such as, for example, the digital camera 14 or a robot. The CPU 12 can also be equipped with a real time operating system for real time operations related to images, in connection with, for example, a robotic operation or an interactive operation with a user.

**[0007]** As shown in figure 2, each image file 18 comprises an n X m pixel array. Each pixel, p, is a picture element corresponding to a discrete portion of the overall image. All of the pixels together define the image represented by the image file 18. Each pixel comprises a digital value corresponding to a set of color bands, for example, red, green and blue color components (RGB) of the picture element. The present invention is applicable to any multi-band image, where each band corresponds to a piece of the electro-magnetic spectrum. The pixel array includes n rows of m columns each,
starting with the pixel p(1,1) and ending with the pixel p(n, m). When displaying or printing an image, the CPU 12 retrieves the corresponding image file 18 from the memory 16, and operates the monitor 20 or printer 22, as the case may be, as a function of the digital values of the pixels in the image file 18, as is generally known.

**[0008]** In an image operation, the CPU 12 operates to analyze the RGB values of the pixels of a stored image file 18 to achieve various objectives, such as, for example, to identify regions of an image that correspond to a single material depicted in a scene recorded in the image file 18. A fundamental observation underlying a basic discovery of the present invention, is that an image comprises two components, material and illumination. All changes in an image are caused by one or the other of these components. A method for detecting of one of these components, for example, material, provides a mechanism for distinguishing material or object geometry, such as object edges, from illumination and shadow boundaries.

**[0009]** Such a mechanism enables techniques that can be used to generate intrinsic images. The intrinsic images correspond to an original image, for example, an image depicted in an input image file 18. The intrinsic images include, for example, an illumination image, to capture the intensity and color of light incident upon each point on the surfaces depicted in the image, and a material reflectance image, to capture reflectance properties of surfaces depicted in the image (the percentage of each wavelength of light a surface reflects). The separation of illumination from material in the intrinsic images provides the CPU 12 with images optimized for more effective and accurate further processing.

**[0010]** Pursuant to a feature of the present invention, a token is a connected region of an image wherein the pixels of the region are related to one another in a manner relevant to identification of image features and characteristics such as an identification of materials and illumination. The pixels of a token can be related in terms of either homogeneous factors, such as, for example, close correlation of color among the pixels, or inhomogeneous factors, such as, for example, differing color values related geometrically in a color space such as RGB space, commonly referred to as a texture. The present invention utilizes spatio-spectral information relevant to contiguous pixels of an image depicted in an image file 18 to identify token regions. The spatio-spectral information includes spectral relationships among contiguous pixels, in terms of color bands, for example the RGB values of the pixels, and the spatial extent of the pixel spectral characteristics relevant to a single material.

**[0011]** According to one exemplary embodiment of the present invention, tokens are each classified as either a Type A token, a Type B token or a Type C token. A Type A token is a connected image region comprising contiguous pixels that represent the largest possible region of the image encompassing a single material in the scene (uniform reflectance). A Type B token is a connected image region comprising contiguous pixels that represent a region of the image encompassing a single material in the scene, though not necessarily the maximal region of uniform reflectance corresponding to that material. A Type B token can also be defined as a collection of one or more image regions or pixels, all of which have the same reflectance (material color) though not necessarily all pixels which correspond to that material color. A Type C token comprises a connected image region of similar image properties among the contiguous pixels of the token, where similarity is defined with respect to a noise model for the imaging system used to record the image.

**[0012]** Referring now to figure 3a, there is shown a flow chart for identifying Type C token regions in the scene depicted in the image file 18 of figure 2, according to a feature of the present invention. Type C tokens can be readily identified in an image, utilizing the steps of figure 3a, and then analyzed and processed to construct Type B tokens, according to a feature of the present invention.

**[0013]** A $1^{st}$ order uniform, homogeneous Type C token comprises a single robust color measurement among contiguous pixels of the image. At the start of the identification routine, the CPU 12 sets up a region map in memory. In step 100, the CPU 12 clears the region map and assigns a region ID, which is initially set at 1. An iteration for the routine, corresponding to a pixel number, is set at i = 0, and a number for an N x N pixel array, for use as a seed to determine the token, is set an initial value, $N = N_{start}$ . $N_{start}$ can be any integer > 0, for example it can be set at 11 or 15 pixels.

**[0014]** At step 102, a seed test is begun. The CPU 12 selects a first pixel, i = 1, pixel (1, 1) for example (see figure 2), the pixel at the upper left corner of a first N x N sample of the image file 18. The pixel is then tested in decision block 104 to determine if the selected pixel is part of a good seed. The test can comprise a comparison of the color value of the selected pixel to the color values of a preselected number of its neighboring pixels as the seed, for example, the N x N array. The color values comparison can be with respect to multiple color band values (RGB in our example) of the pixel. If the comparison does not result in approximately equal values (within the noise levels of the recording device) for the pixels in the seed, the CPU 12 increments the value of i (step 106), for example, i = 2, pixel (1, 2), for a next N x N seed sample, and then tests to determine if i = $i_{max}$ (decision block 108).

**[0015]** If the pixel value is at $i_{max}$, a value selected as a threshold for deciding to reduce the seed size for improved results, the seed size, N, is reduced (step 110), for example, from N = 15 to N = 12. In an exemplary embodiment of the present invention, $i_{max}$ can be set at a number of pixels in an image ending at pixel (n, m), as shown in figure 2. In this manner, the routine of figure 3a parses the entire image at a first value of N before repeating the routine for a reduced value of N.

**[0016]** After reduction of the seed size, the routine returns to step 102, and continues to test for token seeds. An $N_{stop}$ value (for example, N = 2) is also checked in step 110 to determine if the analysis is complete. If the value of N is at $N_{stop}$, the CPU 12 has completed a survey of the image pixel arrays and exits the routine.

**[0017]** If the value of i is less than $i_{max}$, and N is greater than $N_{stop}$, the routine returns to step 102, and continues to test for token seeds.

**[0018]** When a good seed (an N x N array with approximately equal pixel values) is found (block 104), the token is grown from the seed. In step 112, the CPU 12 pushes the pixels from the seed onto a queue. All of the pixels in the queue are marked with the current region ID in the region map. The CPU 12 then inquires as to whether the queue is empty (decision block 114). If the queue is not empty, the routine proceeds to step 116.

**[0019]** In step 116, the CPU 12 pops the front pixel off the queue and proceeds to step 118. In step 118, the CPU 12 marks "good' neighbors around the subject pixel, that is neighbors approximately equal in color value to the subject pixel, with the current region ID. All of the marked good neighbors are placed in the region map and also pushed onto the queue. The CPU 12 then returns to the decision block 114. The routine of steps 114, 116, 118 is repeated until the queue is empty. At that time, all of the pixels forming a token in the current region will have been identified and marked in the region map as a Type C token.

**[0020]** When the queue is empty, the CPU 12 proceeds to step 120. At step 120, the CPU 12 increments the region ID for use with identification of a next token. The CPU 12 then returns to step 106 to repeat the routine in respect of the new current token region.

**[0021]** Upon arrival at N = $N_{stop}$, step 110 of the flow chart of figure 3a, or completion of a region map that coincides with the image, the routine will have completed the token building task. Figure 3b is an original image used as an example in the identification of tokens. The image shows areas of the color blue and the blue in shadow, and of the color teal and the teal in shadow. Figure 3c shows token regions corresponding to the region map, for example, as identified through execution of the routine of figure 3a (Type C tokens), in respect to the image of figure 3b. The token regions are color coded to illustrate the token makeup of the image of figure 3b, including penumbra regions between the full color blue and teal areas of the image and the shadow of the colored areas.

**[0022]** While each Type C token comprises a region of the image having a single robust color measurement among contiguous pixels of the image, the token may grow across material boundaries. Typically, different materials connect together in one Type C token via a neck region often located on shadow boundaries or in areas with varying illumination crossing different materials with similar hue but different intensities. A neck pixel can be identified by examining characteristics of adjacent pixels. When a pixel has two contiguous pixels on opposite sides that are not within the corresponding token, and two contiguous pixels on opposite sides that are within the corresponding token, the pixel is defined as a neck pixel.

**[0023]** Figure 4 shows a flow chart for a neck test for Type C tokens. In step 122, the CPU 12 examines each pixel of an identified token to determine whether any of the pixels under examination forms a neck. The routine of figure 4 can be executed as a subroutine directly after a particular token is identified during execution of the routine of figure 3a. All pixels identified as a neck are marked as "ungrowable." In decision block 124, the CPU 12 determines if any of the pixels were marked.

**[0024]** If no, the CPU 12 exits the routine of figure 4 and returns to the routine of figure 3a (step 126).

**[0025]** If yes, the CPU 12 proceeds to step 128 and operates to regrow the token from a seed location selected from among the unmarked pixels of the current token, as per the routine of figure 3a, without changing the counts for seed size and region ID. During the regrowth process, the CPU 12 does not include any pixel previously marked as ungrowable. After the token is regrown, the previously marked pixels are unmarked so that other tokens may grow into them.

**[0026]** Subsequent to the regrowth of the token without the previously marked pixels, the CPU 12 returns to step 122 to test the newly regrown token. Neck testing identifies Type C tokens that cross material boundaries, and regrows the identified tokens to provide single material Type C tokens suitable for use in creating Type B tokens.

**[0027]** Figure 3d shows Type B tokens generated from the Type C tokens of figure 3c, according to a feature of the present invention. The present invention provides a novel exemplary technique using log chromaticity clustering, for constructing Type B tokens for an image File 18. Log chromaticity is a technique for developing an illumination invariant chromaticity space.

**[0028]** A method and system for separating illumination and reflectance using a log chromaticity representation is disclosed in U. S. Patent No. 7,596,266. The techniques taught in U. S. Patent No. 7,596,266 can be used to provide illumination invariant log chromaticity representation values for each color of an image, for example, as represented by Type C tokens.

**[0029]** Logarithmic values of the color band values of the image pixels are plotted an a log-color space graph.

The logarithmic values are then projected to a log-chromaticity projection plane oriented as a function of a bi-illuminant dichromatic reflection model (BIDR model), to provide a log chromaticity value for each pixel, as taught in U. S. Patent No. 7,596,266. The BIDR Model predicts that differing color measurement values fall within a cylinder in RGB space, from a dark end (in shadow) to a bright end (lit end), along a positive slope, when the color change is due to an illumination change forming a shadow over a single material of a scene depicted in the image.

**[0030]** Figure 5 is a graphic representation of a log color space, bi-illuminant chromaticity plane according to a feature of the invention disclosed in U. S. Patent No. 7,596,266. The alignment of the chromaticity plane is determined by a vector N, normal to the chromaticity plane, and defined as $N = \log(Bright_{vector}) - \log(Dark_{vector}) = \log(1 + 1/S_{vector})$. The co-ordinates of the plane, u, v can be defined by a projection of the green axis onto the chromaticity plane as the u axis, and the cross product of u and N being defined as the v axis. In our example, each log value for the materials A, B, C is projected onto the chromaticity plane, and will therefore have a corresponding u, v coordinate value in the plane that is a chromaticity value, as shown in figure 5.

**[0031]** Thus, according to the technique disclosed in U. S. Patent No. 7,596,266, the RGB values of each pixel in an image file 18 can be mapped by the CPU 12 from the image file value p(n, m, R, G, B) to a log value, then, through a projection to the chromaticity plane, to the corresponding u, v value, as shown in figure 5. Each pixel p(n, m, R, G, B) in the image file 18 is then replaced by the CPU 12 by a two dimensional chromaticity value: p(n, m, u, v), to provide a chromaticity representation of the original RGB image. In general, for an N band image, the N color values are replaced by N - 1 chromaticity values. The chromaticity representation is a truly accurate illumination invariant representation because the BIDR model upon which the representation is based, accurately and correctly represents the illumination flux that caused the original image.

**[0032]** According to a feature of the present invention, log chromaticity values are calculated for each color depicted in an image file 18 input to the CPU 12 for identification of regions of the uniform reflectance (Type B tokens). For example, each pixel of a Type C token will be of approximately the same color value, for example, in terms of RGB values, as all the other constituent pixels of the same Type C token, within the noise level of the equipment used to record the image. Thus, an average of the color values for the constituent pixels of each particular Type C token can be used to represent the color value for the respective Type C token in the log chromaticity analysis.

**[0033]** Figure 6 is a flow chart for determining a list of colors depicted in an input image, for example, an image file 18. In step 200, an input image file 18 is input to the CPU 12 for processing. In steps 202 and 204, the CPU 12 determines the colors depicted in the input image file 18. In step 202, the CPU 12 calculates an average color for each Type C token determined by the CPU 12 through execution of the routine of figure 3a, as described above, for a list of colors. The CPU 12 can be operated to optionally require a minimum token size, in terms of the number of constituent pixels of the token, or a minimum seed size (the N x N array) used to determine Type C tokens according to the routine of figure 3a, for the analysis. The minimum size requirements are implemented to assure that color measurements in the list of colors for the image are an accurate depiction of color in a scene depicted in the input image, and not an artifact of blend pixels.

**[0034]** Blend pixels are pixels between two differently colored regions of an image. If the colors between the two regions are plotted in RGB space, there is a linear transition between the colors, with each blend pixel, moving from one region to the next, being a weighted average of the colors of the two regions. Thus, each blend pixel does not represent a true color of the image. If blend pixels are present, relatively small Type C tokens, consisting of blend pixels, can be identified for areas of an image between two differently colored regions. By requiring a size minimum, the CPU 12 can eliminate tokens consisting of blend pixel from the analysis.

**[0035]** In step 204, the CPU 12 can alternatively collect colors at the pixel level, that is, the RGB values of the pixels of the input image file 18, as shown in figure 2. The CPU 12 can be operated to optionally require each pixel of the image file 18 used in the analysis to have a minimum stability or local standard deviation via a filter output, for a more accurate list of colors. For example, second derivative energy can be used to indicate the stability of pixels of an image.

**[0036]** In this approach, the CPU 12 calculates a second derivative at each pixel, or a subset of pixels disbursed across the image to cover all illumination conditions of the image depicted in an input image file 18, using a Difference of Gaussians, Laplacian of Gaussian, or similar filter. The second derivative energy for each pixel examined can then be calculated by the CPU 12 as the average of the absolute value of the second derivative in each color band (or the absolute value of the single value in a grayscale image), the sum of squares of the values of the second derivatives in each color band (or the square of the single value in a grayscale image), the maximum squared second derivative value across the color bands (or the square of the single value in a grayscale image), or any similar method. Upon the calculation of the second derivative energy for each of the pixels, the CPU 12 analyzes the energy values of the pixels. There is an inverse relationship between second derivative energy and pixel stability, the higher the energy, the less stable the corresponding pixel.

**[0037]** In step 206, the CPU 12 outputs a list or lists of color (after executing one or both of steps 202 and/or 204). All of the further processing can be executed using the list from either step 202 or 204, or vary the list used (one or the other of the lists from steps 202 or 204) at each subsequent step; the method claimed uses lists from both step 202 and step 204. Figure 7 is a flow chart for determining an orientation for a log chromaticity representation, according to a feature of the present invention. For example, the CPU 12 determines an orientation for the normal N, for a log chromaticity plane, as shown in figure 5. In step 210, the CPU 12 receives a list of colors for an input file 18, such as a list output in step 206 of the routine of figure 6. In step 212, the CPU 12 determines an orientation for a log chromaticity space. The claimed method here uses a list based on tokens according to step 202 of figure 6.

**[0038]** As taught in U. S. Patent No. 7,596,266, and as noted above, alignment of the chromaticity plane is represented by N, N being a vector normal to the chromaticity representation, for example, the chromaticity plane of figure 5. The orientation is estimated by the CPU 12 thorough execution of any one of several techniques. For example, the CPU 12 can determine estimates based upon entropy minimization, manual selection by a user or the use of a characteristic spectral ratio for an image of an input image file 18, as fully disclosed in U. S. Patent No. 7,596,266.

**[0039]** For a higher dimensional set of colors, for example, an RYGB space (red, yellow, green, blue), the log chromaticity normal, N, defines a sub-space with one less dimension than the input space. Thus, in the four dimensional RYGB space, the normal N defines a three dimensional log chromaticity space. When the four dimensional RYGB values are projected into the three dimensional log chromaticity space, the projected values within the log chromaticity space are unaffected by illumination variation.

**[0040]** In step 214, the CPU 12 outputs an orientation for the normal N. As illustrated in the example of figure 5, the normal N defines an orientation for a u, v plane in a three dimensional RGB space.

**[0041]** Figure 8 is a flow chart for determining log chromaticity coordinates for the colors of an input image, as identified in steps 202 or 204 of the routine of figure 6, according to a feature of the present invention. In step 220, a list of colors is input to the CPU 12. The list of colors can comprise either the list generated through execution of step 202 of the routine of figure 6, or the list generated through execution of step 204. In step 222, the log chromaticity orientation for the normal, N, determined through execution of the routine of figure 7, is also input to the CPU 12.

**[0042]** In step 224, the CPU 12 operates to calculate a log value for each color in the list of colors and plots the log values in a three dimensional log space at respective (log R, log G, log B) coordinates, as illustrated in figure 5. Materials A, B and C denote log values for specific colors from the list of colors input to the CPU 12 in step 220. A log chromaticity plane is also calculated by the CPU 12, in the three dimensional log space, with u, v coordinates and an orientation set by N, input to the CPU 12 in step 222. Each u, v coordinate in the log chromaticity plane can also be designated by a corresponding (log R, log G, log B) coordinate in the three dimensional log space.

**[0043]** According to a feature of the present invention, the CPU 12 then projects the log values for the colors A, B and C onto the log chromaticity plane to determine a u, v log chromaticity coordinate for each color. Each u, v log chromaticity coordinate can be expressed by the corresponding (log R, log G, log B) coordinate in the three dimensional log space. The CPU 12 outputs a list of the log chromaticity coordinates in step 226. The list cross-references each color to a u, v log chromaticity coordinate and to the pixels (or a Type C tokens) having the respective color (depending upon the list of colors used in the analysis (either step 202(tokens) or 204 (pixels))).

**[0044]** Figure 9 is a flow chart for optionally augmenting the log chromaticity coordinates for pixels or Type C tokens with extra dimensions, according to a feature of the present invention. In step 230, the list of log chromaticity coordinates, determined for the colors of the input image through execution of the routine of figure 8, is input to the CPU 12. In step 232, the CPU 12 accesses the input image file 18, for use in the augmentation.

**[0045]** In step 234, the CPU 12 optionally operates to augment each log chromaticity coordinate with a tone mapping intensity for each corresponding pixel (or Type C token). The tone mapping intensity is determined using any known tone mapping technique. An augmentation with tone mapping intensity information provides a basis for clustering pixels or tokens that are grouped according to both similar log chromaticity coordinates and similar tone mapping intensities. This improves the accuracy of a clustering step.

**[0046]** In step 236, the CPU 12 optionally operates to augment each log chromaticity coordinate with x, y coordinates for the corresponding pixel (or an average of the x, y coordinates for the constituent pixels of a Type C token) (see figure 2 showing a P (1,1) to P (N, M) pixel arrangement). Thus, a clustering step with x, y coordinate information will provide groups in a spatially limited arrangement, when that characteristic is desired.

**[0047]** In each of steps 234 and 236, the augmented information can, in each case, be weighted by a factor $w_1$ and $w_2$, $w_3$ respectively, to specify the relative importance and scale of the different dimensions in the augmented coordinates. The weight factors $w_1$ and $w_2$, $w_3$ are user-specified. Accordingly, the (log R, log G, log B) coordinates for a pixel or Type C token is augmented to (log R, log G, log B, $T^*w_1$, $x^*w_2$, $y^*w_3$) where T, x and y are the tone mapped intensity, the x coordinate and the y coordinate, respectively.

**[0048]** In step 238, the CPU 12 outputs a list of the augmented coordinates. The augmented log chromaticity coordinates provide accurate illumination invariant representations of the pixels, or for a specified regional arrangement of an input image, such as, for example, Type C tokens. According to a feature of the present invention, the illumination invariant characteristic of the log chromaticity coordinates is relied upon as a basis to identify regions of an image of a single material or reflectance, such as, for example, Type B tokens.

**[0049]** Figure 10 is a flow chart for clustering the log chromaticity coordinates, according to a feature of the present invention. In step 240, the list of augmented log chromaticity coordinates is input the CPU 12. In step 242, the CPU 12 operates to cluster the log chromaticity coordinates. The clustering step can be implemented via, for example, a known k-means clustering. Any known clustering technique can be used to cluster the log chromaticity coordinates to determine groups of similar log chromaticity coordinate values. The CPU 12 correlates each log chromaticity coordinate to the group to which the respective coordinate belongs. The CPU 12 also operates to calculate a center for each group identified in the clustering step. For example, the CPU 12 can determine a center for each group relative to a (log R, log G, log B, log T) space.

**[0050]** In step 244, the CPU 12 outputs a list of the cluster group memberships for the log chromaticity coordinates (cross referenced to either the corresponding pixels or Type C tokens) and/or a list of cluster group centers.

**[0051]** As noted above, in the execution of the clustering method, the CPU 12 can use the list of colors from either the list generated through execution of step 202 of the routine of figure 6, or the list generated through execution of step 204. In applying the identified cluster groups to an input image, the CPU 12 can be operated to use the same set of colors as used in the clustering method (one of the list of colors corresponding to step 202 or to the list of colors corresponding to step 204), or apply a different set of colors (the other of the list of colors corresponding to step 202 or the list of colors corresponding to step 204). If a different set of colors is used, as is the case for the method claimed, the CPU 12 proceeds to execute the routine of figure 11.

**[0052]** Figure 11 is a flow chart for assigning the log chromaticity coordinates to clusters determined through execution of the routine of figure 10, when a different list of colors is used after the identification of the cluster groups, according to a feature of the present invention. In step 250, the CPU 12 once again executes the routine of figure 8, this time in respect to the new list of colors. For example, if the list of colors generated in step 202 (colors based upon Type C tokens) was used to identify the cluster groups, as is the case for the claimed method, and the CPU 12 then operates to classify log chromaticity coordinates relative to cluster groups based upon the list of colors generated in step 204 (colors based upon pixels), step 250 of the routine of figure 11 is executed to determine the log chromaticity coordinates for the colors of the pixels in the input image file 18.

**[0053]** In step 252, the list of cluster centers is input to the CPU 12. In step 254, the CPU 12 operates to classify each of the log chromaticity coordinates identified in step 250, according to the nearest cluster group center. In step 256, the CPU 12 outputs a list of the cluster group memberships for the log chromaticity coordinates based upon the new list of colors, with a cross reference to either corresponding pixels or Type C tokens, depending upon the list of colors used in step 250 (the list of colors generated in step 202 or the list of colors generated in step 204).

**[0054]** Figure 12 is a flow chart for detecting regions of uniform reflectance based on the log chromaticity clustering according to a feature of the present invention. In step 260, the input image file 18 is once again provided to the CPU 12. In step 262, one of the pixels or Type C tokens, depending upon the list of colors used in step 250, is input to the CPU 12. In step 264, the cluster membership information, from either steps 244 or 256, is input to the CPU 12.

**[0055]** In step 266, the CPU 12 operates to merge each of the pixels, or specified regions of an input image, such as, for example, Type C tokens, having a same cluster group membership into a single region of the image to represent a region of uniform reflectance (Type B token). The CPU 12 performs such a merge operation for all of the pixels or tokens, as the case may be, for the input image file 18. In step 268, the CPU 12 outputs a list of all regions of uniform reflectance (and also of similar tone mapping intensities and x, y coordinates, if the log chromaticity coordinates were augmented in steps 234 and/or 236). It should be noted that each region of uniform reflectance (Type B token) determined according to the features of the present invention, potentially has significant illumination variation across the region.

**[0056]** U. S. Patent Publication No. US 2010/0142825 teaches a constraint/solver model for segregating illumination and material in an image, including an optimized solution based upon a same material constraint. A same material constraint, as taught in U. S. Patent Publication No. US 2010/0142825, utilizes Type C tokens and Type B tokens, as can be determined according to the teachings of the present invention. The constraining relationship is that all Type C tokens that are part of the same Type B token are constrained to be of the same material. This constraint enforces the definition of a Type B token, that is, a connected image region comprising contiguous pixels that represent a region of the image encompassing a single material in the scene, though not necessarily the maximal region corresponding to that material. Thus, all Type C tokens that lie within the same Type B token are by the definition imposed upon Type B tokens, of the same material, though not necessarily of the same illumination. The Type C tokens are therefore constrained to correspond to observed differences in appearance that are caused by varying illumination.

**[0057]** Implementation of the constraint/solver model according to the techniques and teachings of U. S. Patent Publication No. US 2010/0142825, utilizing the Type C tokens and Type B tokens obtained via a log chromaticity clustering technique according to the present invention, provides a highly effective and efficient method for generating intrinsic images corresponding to an original input image. The intrinsic images can be used to enhance the accuracy and efficiency of image processing, image analysis and computer vision applications.

**Claims**

**1.** An automated, computerized method for processing an image, comprising the steps of:

providing an image file depicting an image, in a computer memory;
collecting colors from tokens in the image, to obtain a list of colors, a token being a connected image region comprising pixels of equal color within the noise levels of a recording device used to record the image;
determining a normal for a plane in a space of logarithmic color band values from the list of colors, the direction of the normal corresponding to a color change which is due to an illumination change;
projecting logarithmic color band values from the color list obtained from the tokens onto the plane of the previous step to obtain log chromaticity coordinates, wherein the log chromaticity coordinates are the coordinates in the plane;
clustering the log chromaticity coordinates to provide clusters of log chromaticity coordinates;
calculating a center of each cluster to provide a list of cluster centers;
generating a list of colors from the pixels of the image;
projecting the colors from the list of colors from the pixels in log chromaticity space onto said plane along the normal of the plane determined from the color list obtained from the tokens to generate log chromaticity coordinates for the colors of the pixels;
assigning the log chromaticity coordinates for the colors of the pixels to clusters according to the nearest cluster center from the list of cluster centers;
and
identifying regions of uniform reflectance in the image as a function of the clusters of log chromaticity coordinates as determined in the previous step by merging all the pixels corresponding to the log chromaticity coordinates belonging to a single cluster into a single region, wherein a region of uniform reflectance is defined as a region encompassing a single material.

**2.** A computer system which comprises:

a CPU; and
a memory storing an image file containing an image;
the CPU arranged and configured to execute a routine to perform the method according to claim 1.

**3.** A computer program product, disposed on a computer readable media, the product including computer executable process steps operable to control a computer to perform the method according to claim 1.

**Patentansprüche**

**1.** Automatisiertes, computergestütztes Verfahren zum Verarbeiten eines Bilds, wobei das Verfahren die folgenden Schritte umfasst:

Bereitstellen einer Bilddatei, welche ein Bild darstellt, in einem Computerspeicher;
Sammeln von Farben aus Token in dem Bild, um eine Liste von Farben zu erlangen, wobei ein Token ein verbundener Bildbereich ist, welcher Pixel von gleicher Farbe innerhalb der Rauschpegel einer Aufzeichnungsvorrichtung umfasst, welche dazu verwendet wird, das Bild aufzuzeichnen;
Bestimmen einer Normale für eine Ebene in einem Raum von logarithmischen Farbbandwerten aus der Liste von Farben, wobei die Richtung der Normale einer Farbänderung aufgrund einer Illuminationsänderung entspricht;
Projizieren von logarithmischen Farbbandwerten aus der Farbliste, welche aus den Token erlangt wurden, auf die Ebene des vorherigen Schritts, um Log-Chromatizitätskoordinaten zu erhalten, wobei die Log-Chromatizitätskoordinaten die Koordinaten in der Ebene sind;
Clustern der Log-Chromatizitätskoordinaten, um Cluster von Log-Chromatizitätskoordinaten bereitzustellen;
Berechnen einer Mitte jedes Clusters, um eine Liste von Clustermitten bereitzustellen;
Erzeugen einer Liste von Farben aus den Pixeln des Bildes;
Projizieren der Farben aus der Liste von Farben von den Pixeln in Log-Chromatizitätsräumen auf die Ebene entlang der Normale der Ebene, welche aus der Farbliste bestimmt wird, welche aus den Token erlangt wird, um Log-Chromatizitätskoordinaten für die Farben der Pixel zu erzeugen;
Zuweisen der Log-Chromatizitätskoordinaten der Farben der Pixel an Cluster gemäß der nächstgelegenen Clustermitte aus der Liste von Clustermitten; und

Identifizieren von Bereichen einheitlicher Reflexion in dem Bild als eine Funktion des Clusters von Log-Chromatizitätskoordinaten, wie in dem vorhergehenden Schritt bestimmt, durch ein Verschmelzen all der Pixel, die den Log-Chromatizitätskoordinaten entsprechen, welche zu einem einzelnen Cluster gehören, in einen einzelnen Bereich, wobei ein Bereich einheitlicher Reflexion definiert ist als ein Bereich, welcher ein einzelnes Material umgibt.

**2.** Computersystem, welches Folgendes umfasst:

eine CPU; und
einen Speicher, welcher eine Bilddatei speichert, welche ein Bild enthält;
wobei die CPU dazu angeordnet und konfiguriert ist, eine Routine auszuführen, um das Verfahren nach Anspruch 1 durchzuführen.

**3.** Computerprogrammprodukt, welches auf einem computerlesbaren Medium angeordnet ist, wobei das Produkt computerausführbare Prozessschritte beinhaltet, welche betriebsfähig sind, um einen Computer zu steuern, um das Verfahren nach Anspruch 1 durchzuführen.

## Revendications

**1.** Procédé informatisé automatisé pour traiter une image, comprenant les étapes consistant à :

fournir un fichier d'image représentant une image, dans une mémoire d'ordinateur ;
collecter des couleurs à partir d'occurrences dans l'image, pour obtenir une liste de couleurs, une occurrence étant une région d'image connectée comprenant des pixels de couleur égale au sein des niveaux de bruit d'un dispositif d'enregistrement utilisé pour enregistrer l'image ;
déterminer une normale pour un plan dans un espace de valeurs de bande de couleur logarithmique à partir de la liste de couleurs, la direction de la normale correspondant à un changement de couleur qui est dû à un changement d'illumination ;
projeter des valeurs de bande de couleur logarithmique depuis la liste de couleurs obtenue à partir des occurrences sur le plan de l'étape précédente afin d'obtenir des coordonnées de chromaticité log, dans lequel les coordonnées de chromaticité log sont les coordonnées dans le plan ;
rassembler en grappes les coordonnées de chromaticité log pour fournir des grappes de coordonnées de chromaticité log ;
calculer un centre de chaque grappe pour fournir une liste de centres de grappe ;
générer une liste de couleurs à partir des pixels de l'image ;
projeter les couleurs depuis la liste de couleurs à partir des pixels dans un espace de chromaticité log sur ledit plan suivant la normale du plan déterminée à partir de la liste de couleurs obtenue à partir des occurrences pour générer les coordonnées de chromaticité log pour les couleurs des pixels
attribuer les coordonnées de chromaticité log pour les couleurs des pixels à des grappes selon le centre de grappe le plus proche à partir de la liste de centres de grappe ; et
identifier des régions de réflectance uniforme dans l'image en fonction des grappes de coordonnées de chromaticité log telles que déterminées dans l'étape précédente par fusion de l'ensemble des pixels correspondant aux coordonnées de chromaticité log appartenant à une seule grappe dans une seule région, dans lequel une région de réflectance uniforme est définie comme une région englobant un seul matériau.

**2.** Système d'ordinateur qui comprend :

une CPU ; et
une mémoire stockant un fichier d'image contenant une image ;
la CPU étant agencée et configurée pour exécuter une routine afin de réaliser le procédé selon la revendication 1.

**3.** Produit-programme d'ordinateur, disposé sur un support lisible par ordinateur, le produit comportant des étapes de processus exécutables par ordinateur servant à commander un ordinateur afin de réaliser le procédé selon la revendication 1.

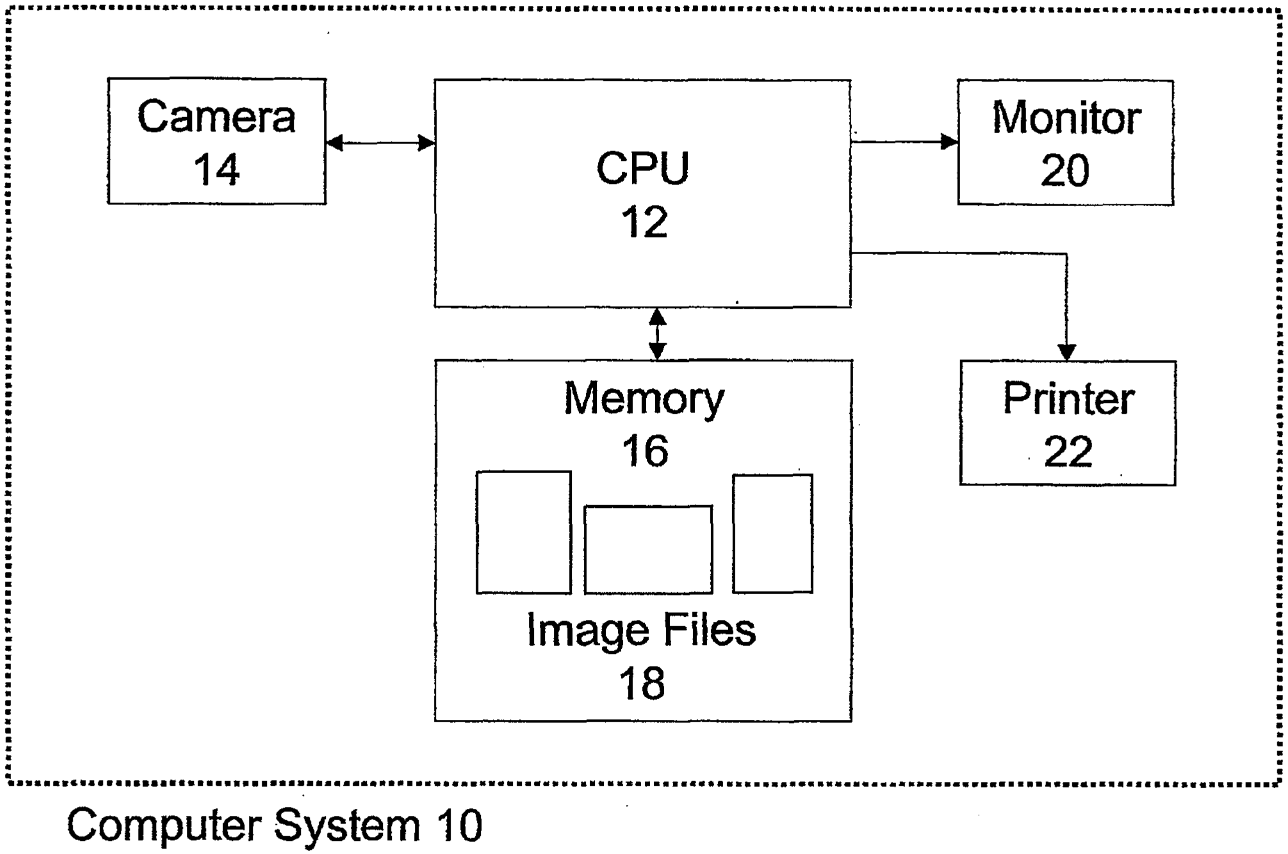



Figure 1: Computer System Configured to Operate on Images

Figure 2: Pixel Array for Storing Image Data

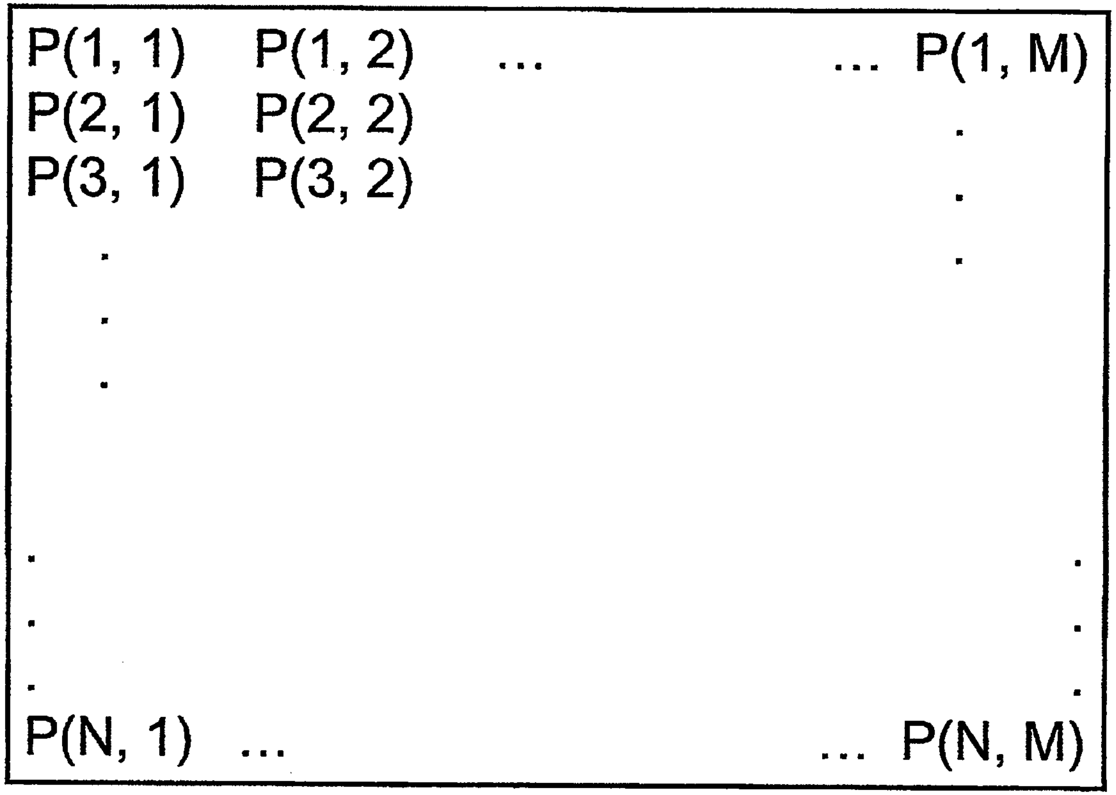



Image File 18

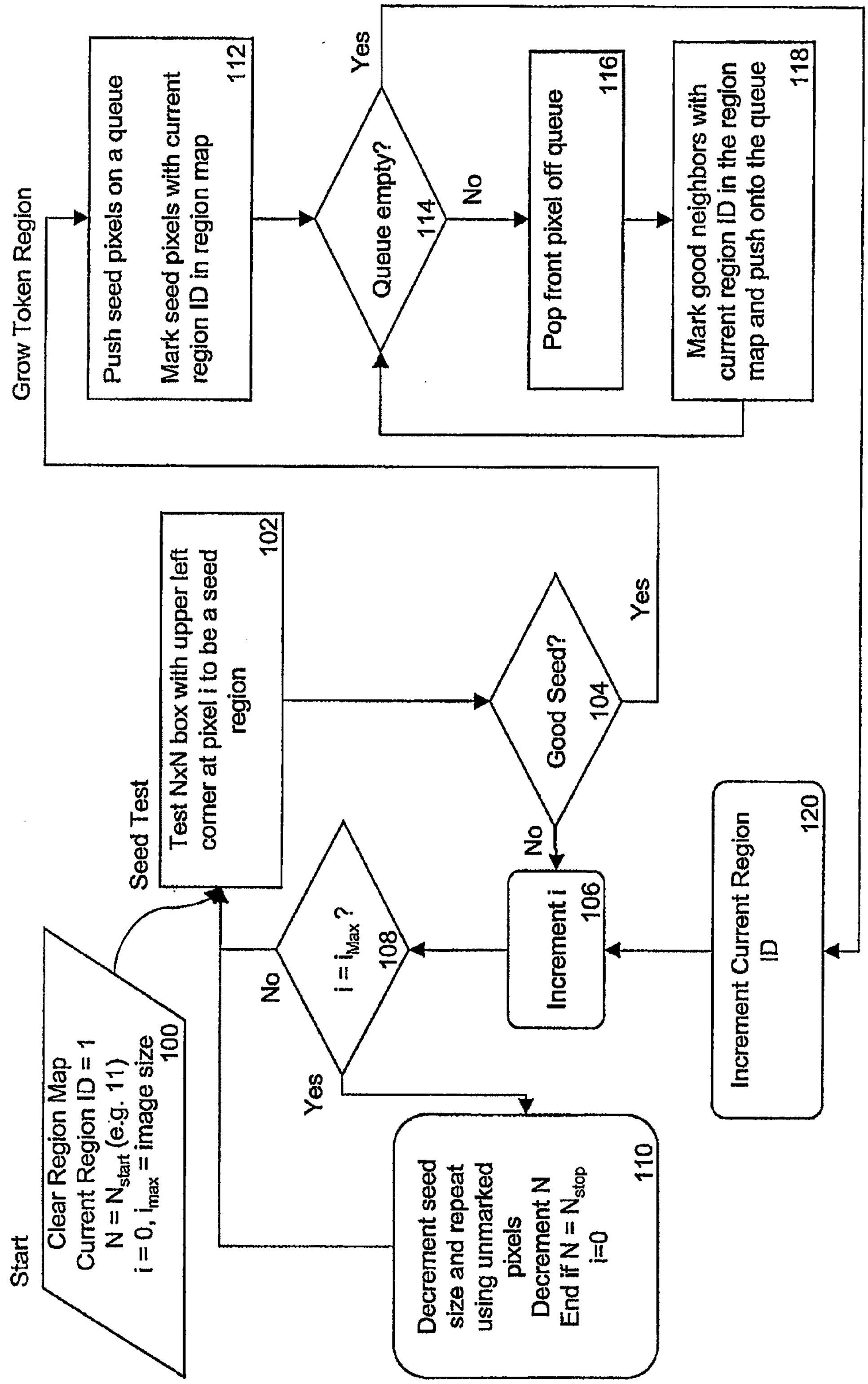



Figure 3A: Identifying Token Regions in an Image

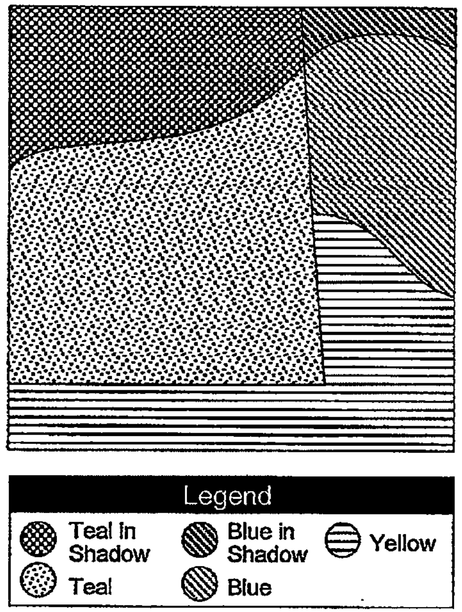



Figure 3B: Original Image

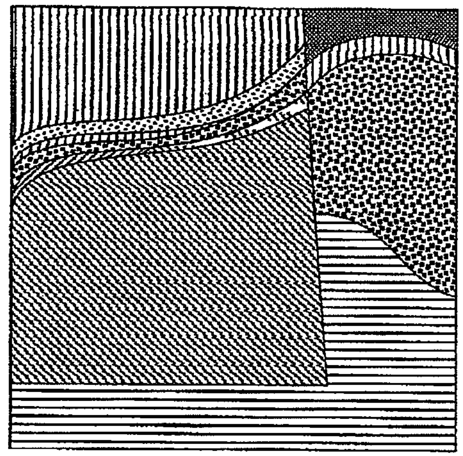


Figure 3C: Token Regions

Figure 3B, 3C: Examples of Identifying Token Regions in an Image

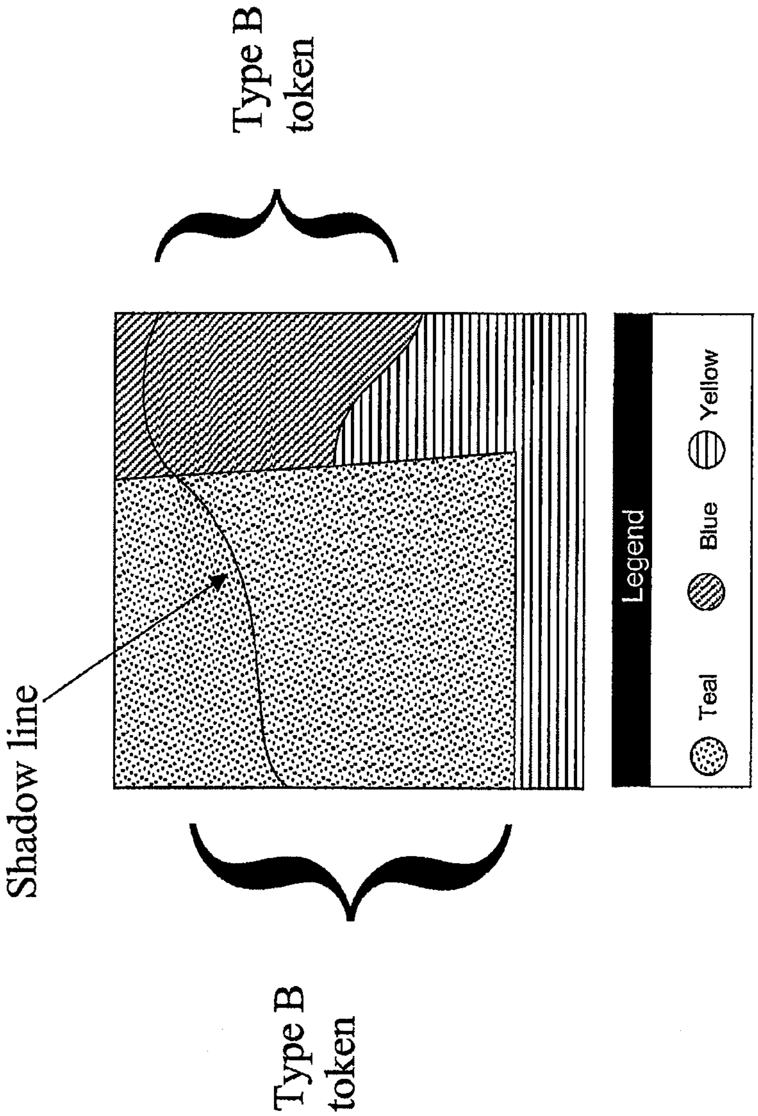



Figure 3D: Type B Tokens

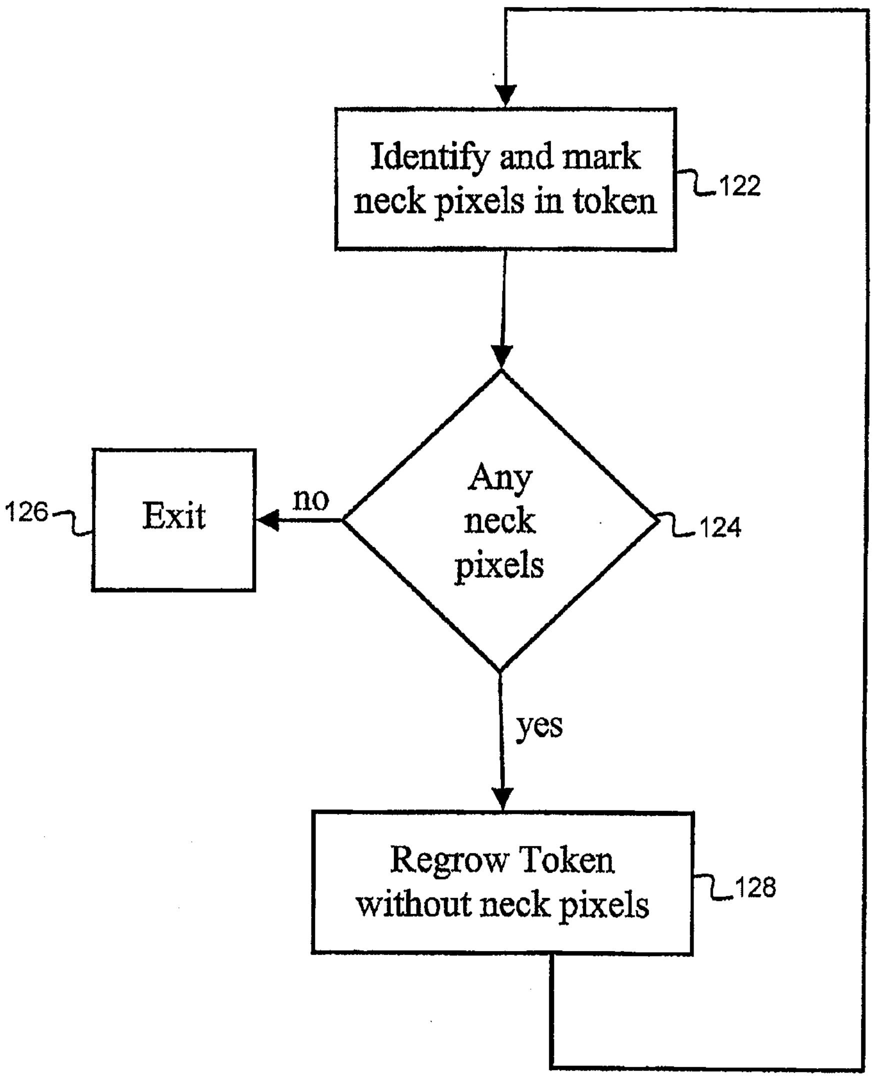
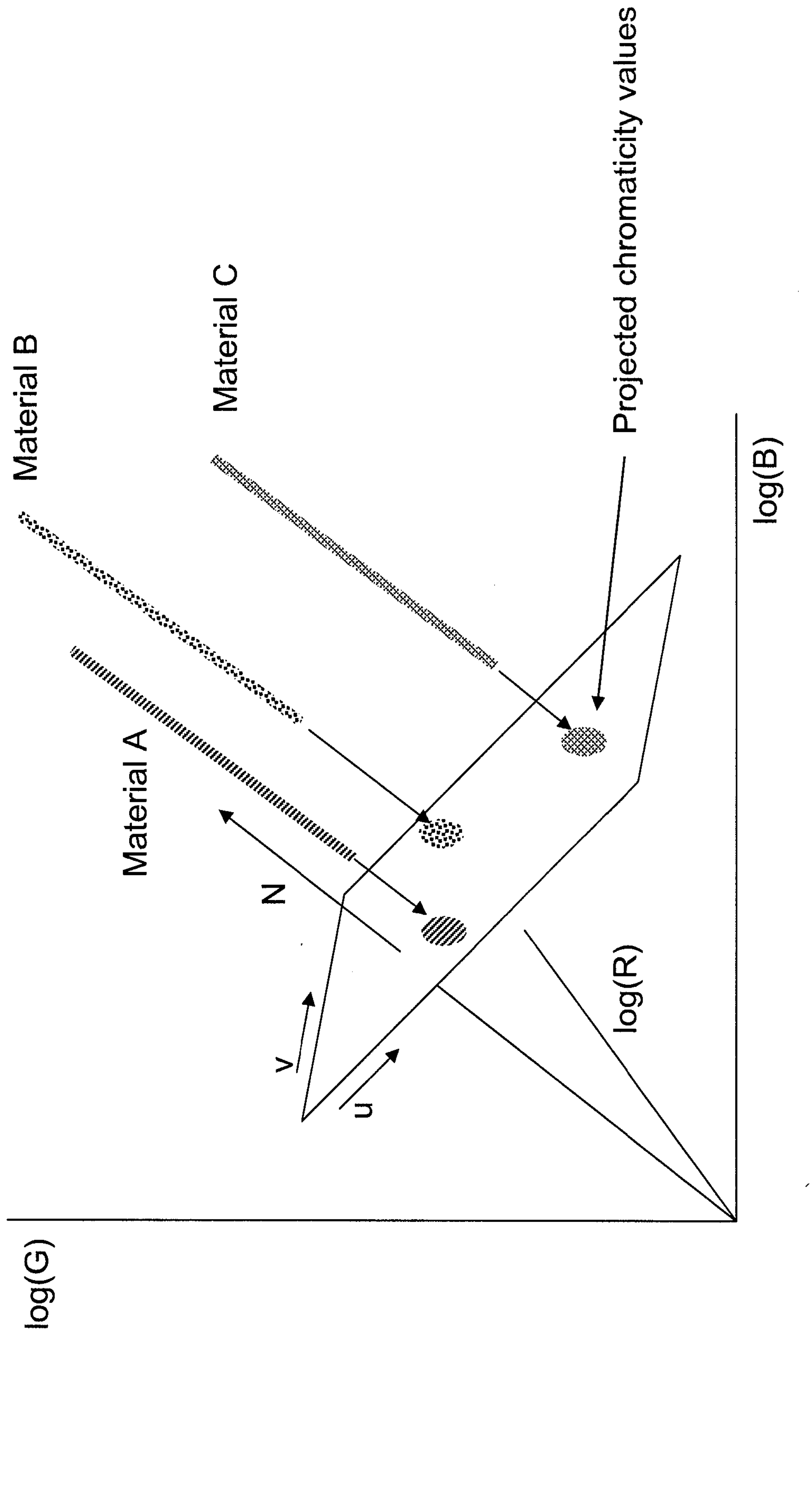

Identify and mark neck pixels in token
122
Any neck pixels
124
no
Exit
126
yes
Regrow Token without neck pixels
128

Figure 4




N = Log Space Chromaticity Plane Normal

Figure 5: Log Color Space Chromaticity Plane

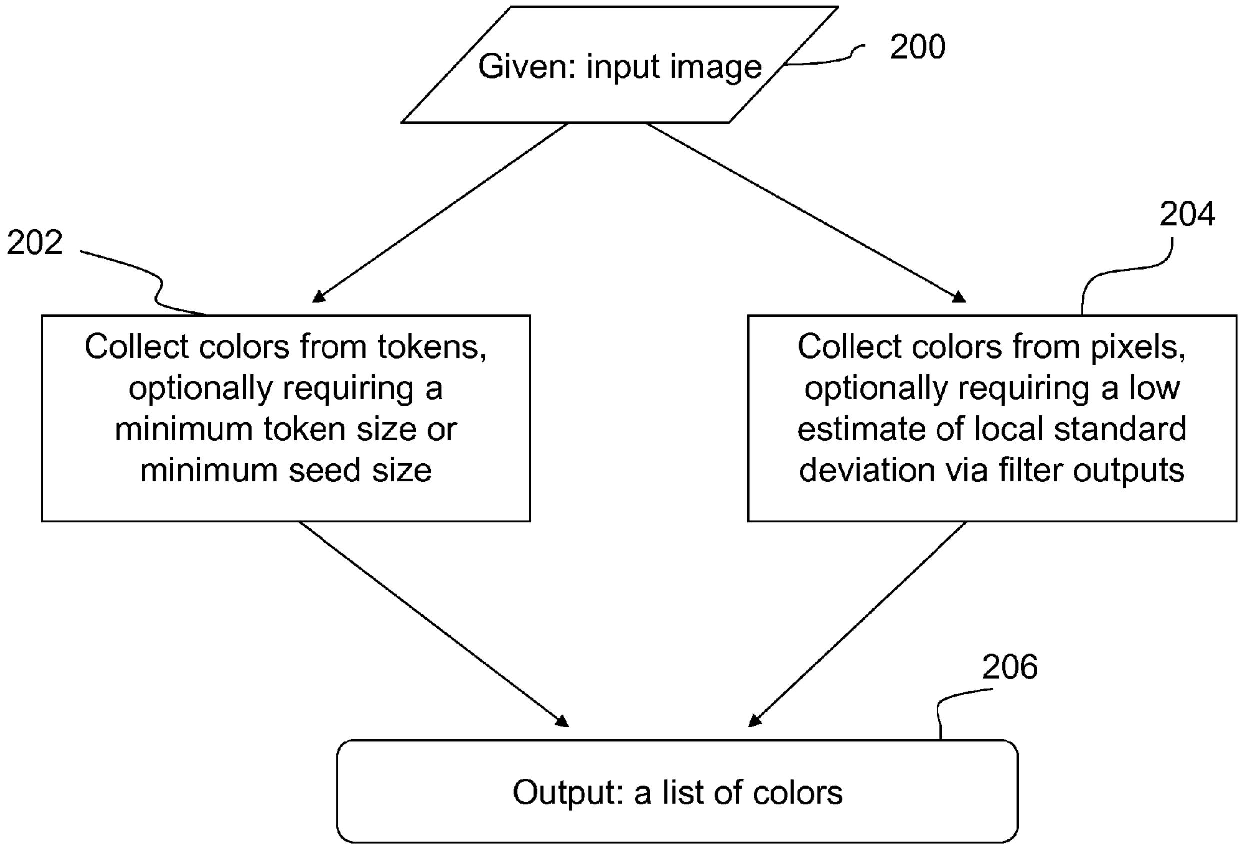



Figure 6: Selecting colors from an image

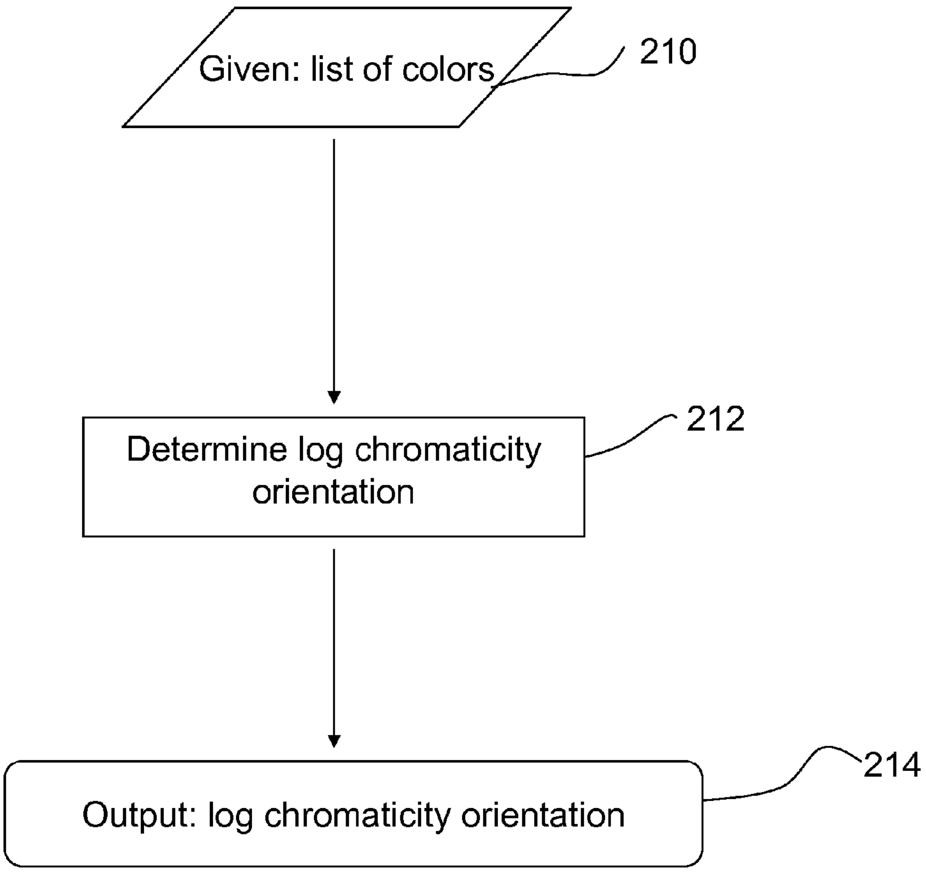



Figure 7: Determining the log chromaticity orientation

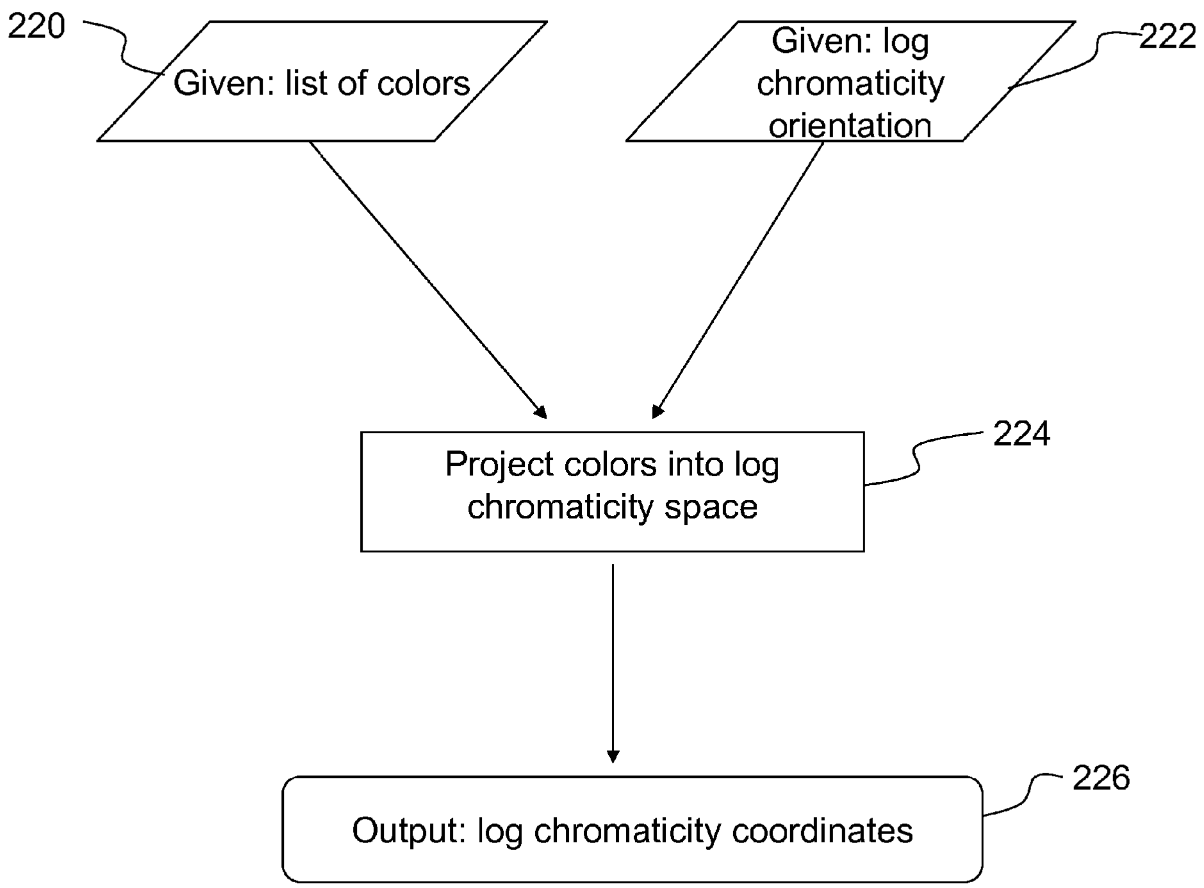



Figure 8: Determining log chromaticity coordinates

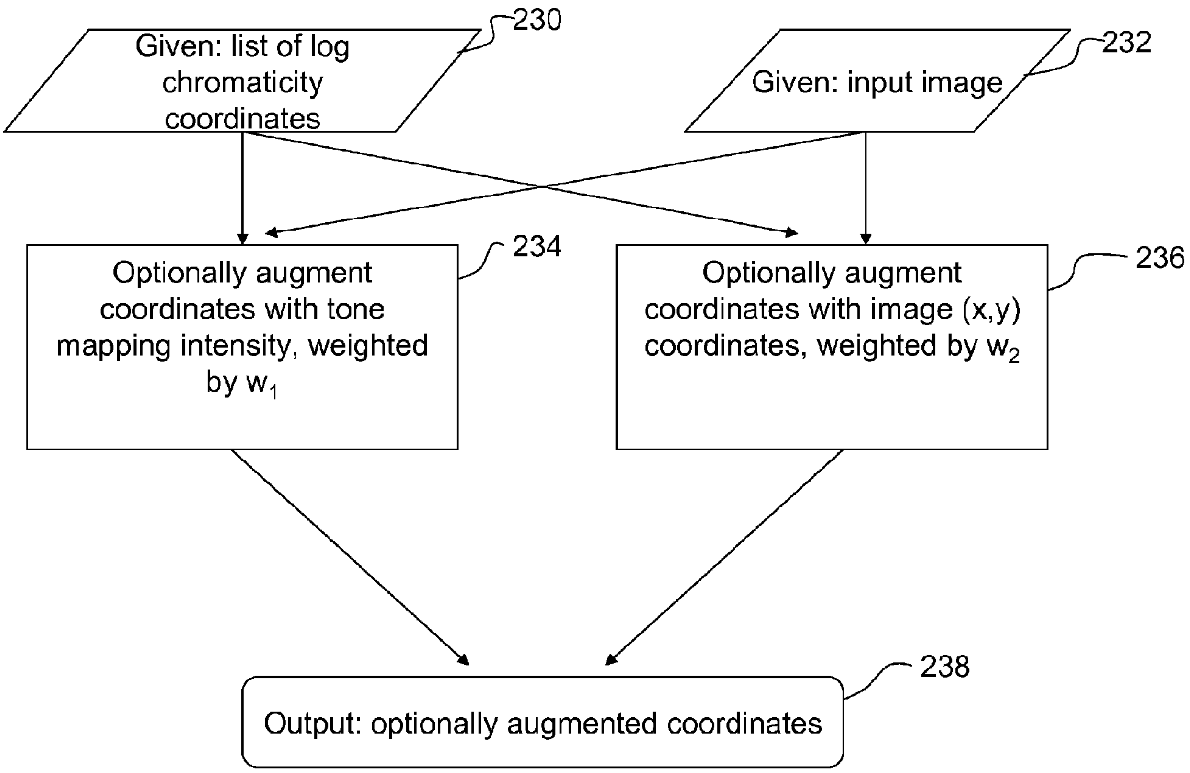



Figure 9: Optionally augmenting log chromaticity coordinates

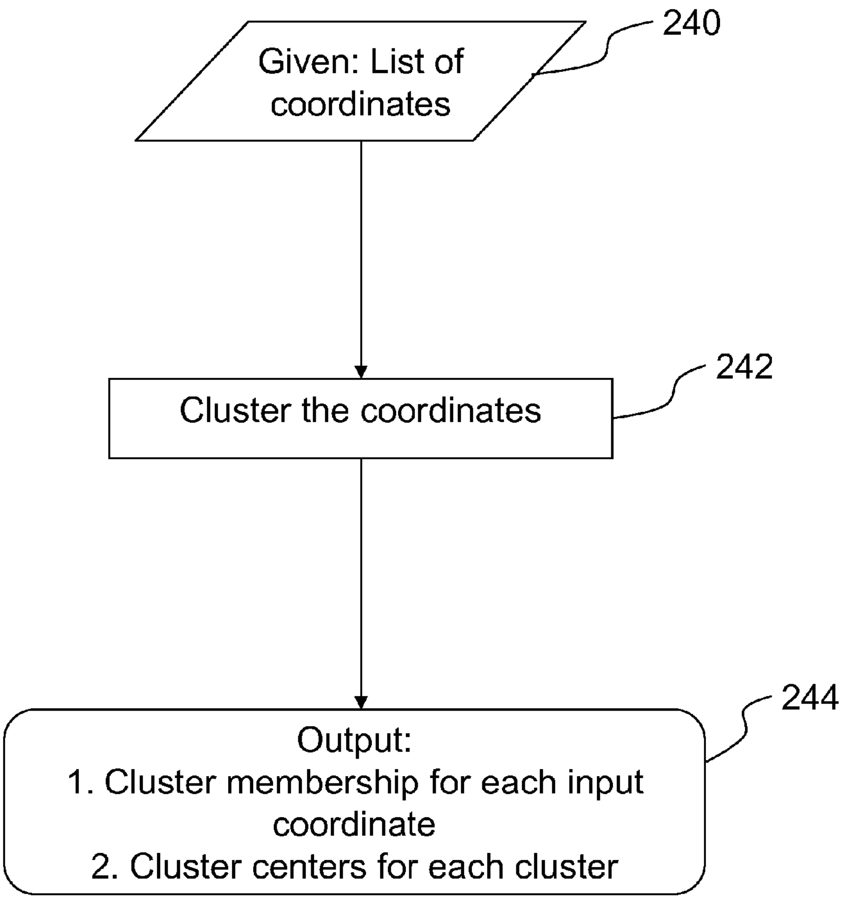



Figure 10: Clustering log chromaticity coordinates

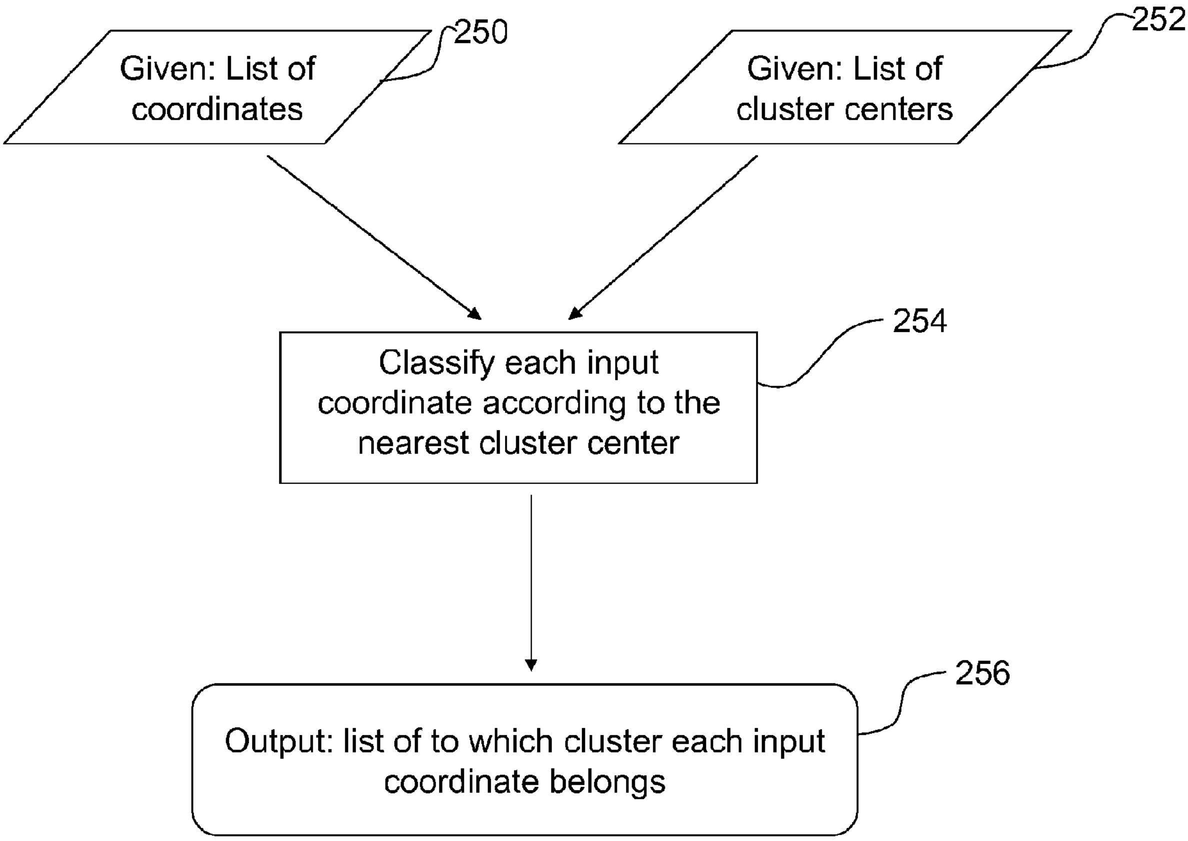



Figure 11: Assigning coordinates to clusters

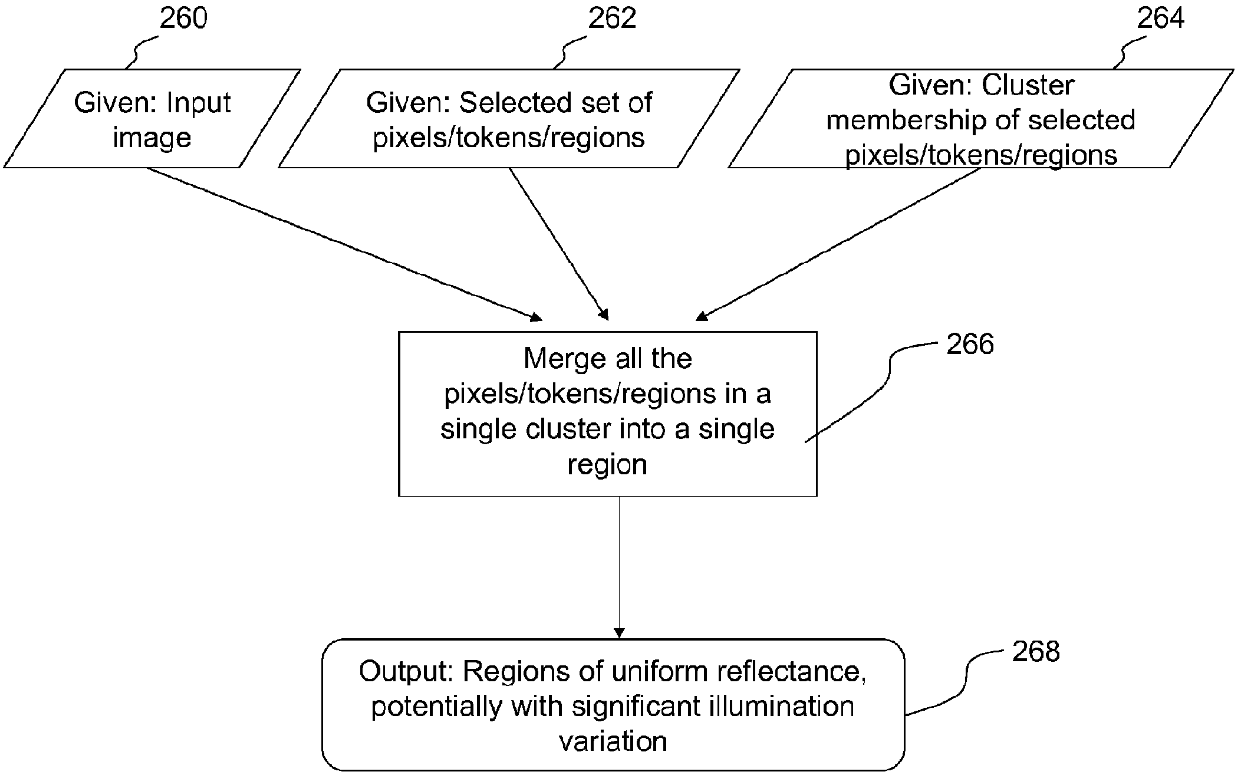



Figure 12: Detecting regions of uniform reflectance based on log chromaticity clustering

## REFERENCES CITED IN THE DESCRIPTION

*This list of references cited by the applicant is for the reader's convenience only. It does not form part of the European patent document. Even though great care has been taken in compiling the references, errors or omissions cannot be excluded and the EPO disclaims all liability in this regard.*

### Patent documents cited in the description


- US 2010278448 A **[0001]**
- US 7596266 B **[0028] [0029] [0030] [0031] [0038]**
- US 20100142825 A **[0056] [0057]**