

(12) **Patent Application Publication**
Doermann et al.

(43) **Pub. Date:** **Nov. 8, 2007**

Publication Classification

(52) **U.S. Cl.** 345/606

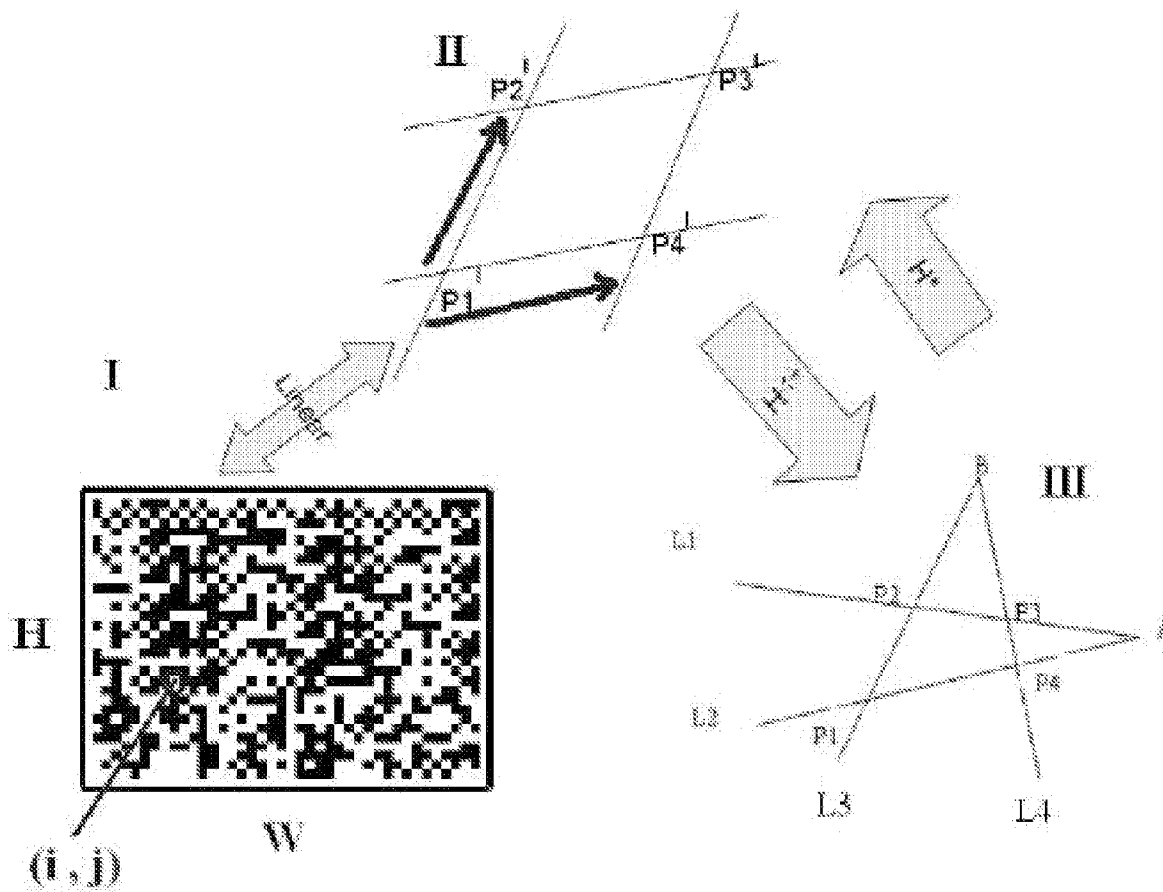
(57) **ABSTRACT**

A system and method for using camera enabled personal digital assistant (PDA) or cell phone hardware to provide enhanced imaging capabilities. The system and method enhances images taken on a mobile camera device to enable the mobile device, for example, a personal digital assistant (PDA) or cell phone, to provide enhanced imaging capabilities. A method comprising the steps of pre-calculating a pixel value at each point on a grid and storing said pre-calculated pixel values in a lookup table, using one bit to represent each pixel in said image, quantizing said image at a small step interval such that each pixel in the image corresponds to one point on said grid, and interpolating said image through a memory-indexing process. The method may further comprise the step of performing clustering based contrast enhancement on said image prior to said step of using one bit to represent each pixel in said image.

(22) Filed: **May 8, 2007**

Related U.S. Application Data

(60) Provisional application No. 60/806,081, filed on Jun. 28, 2006, provisional application No. 60/746,752, filed on May 8, 2006, provisional application No. 60/746,755, filed on May 8, 2006, provisional application No. 60/806,083, filed on Jun. 28, 2006.



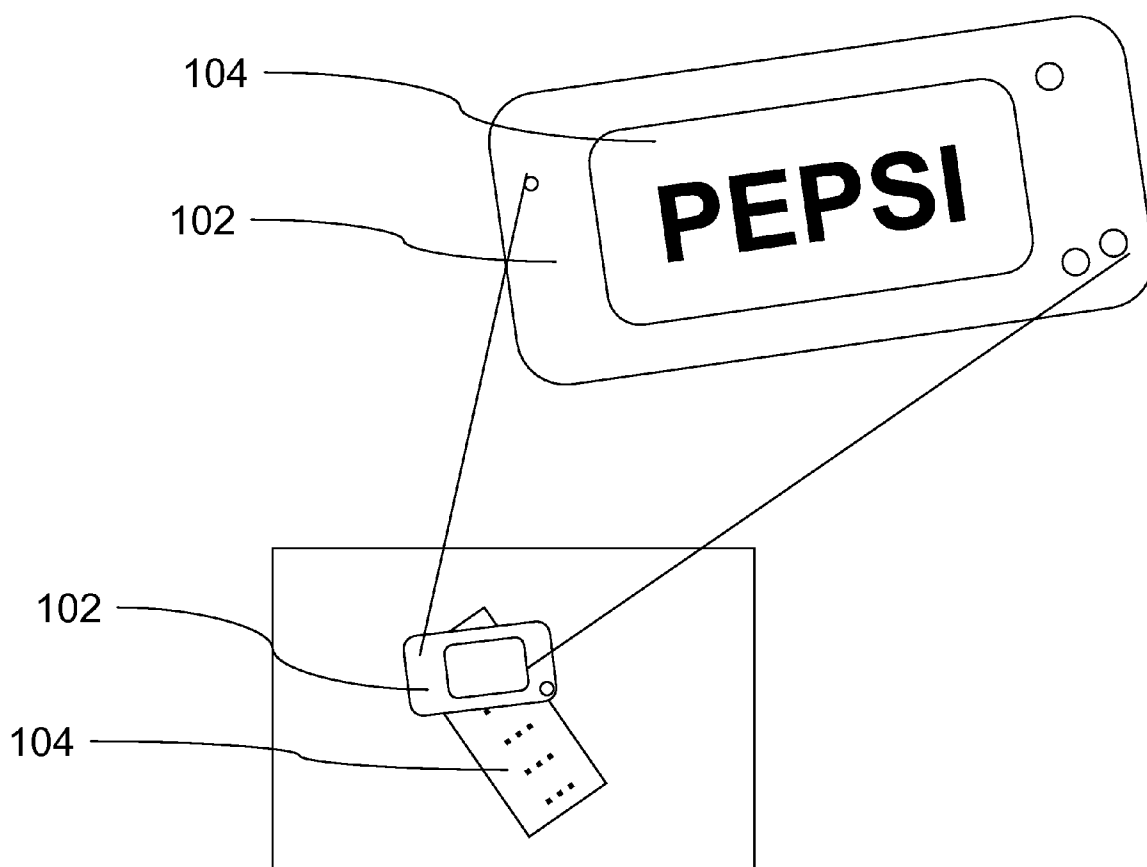


FIG. 1

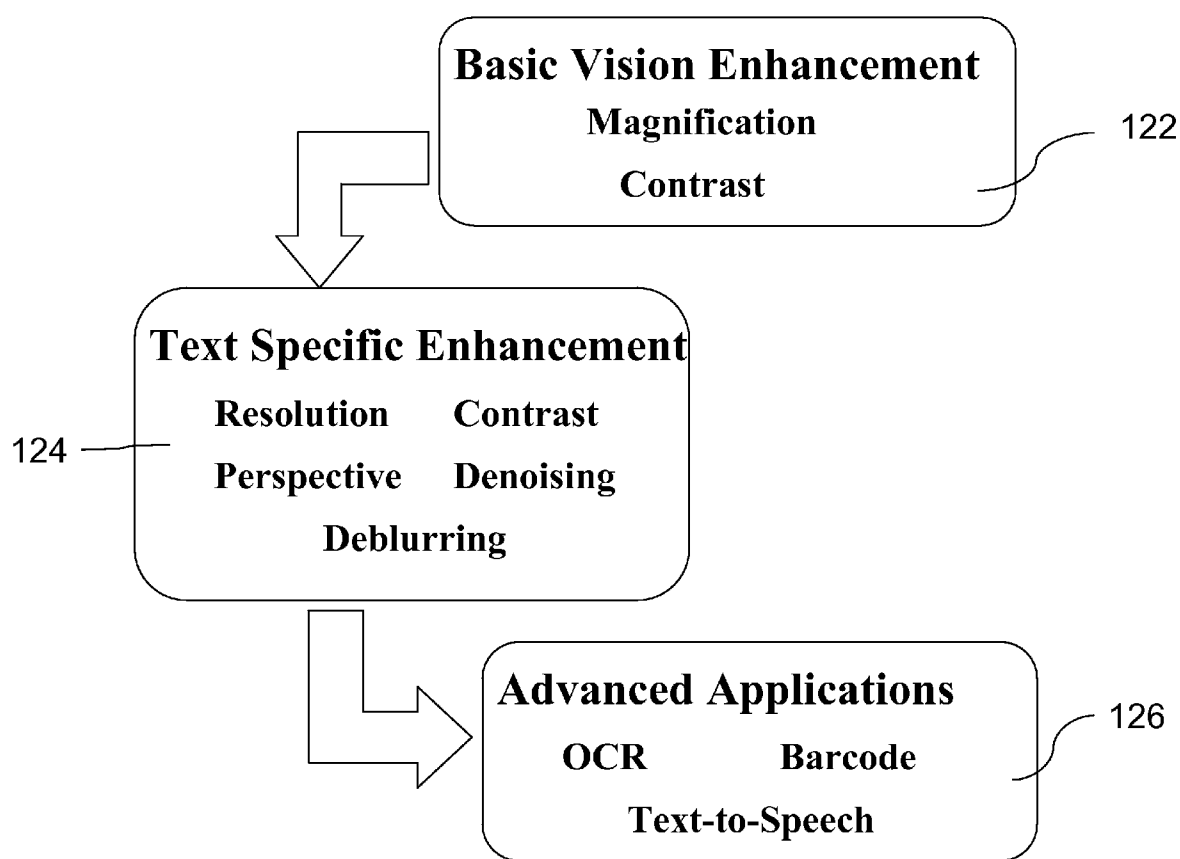


FIG. 2

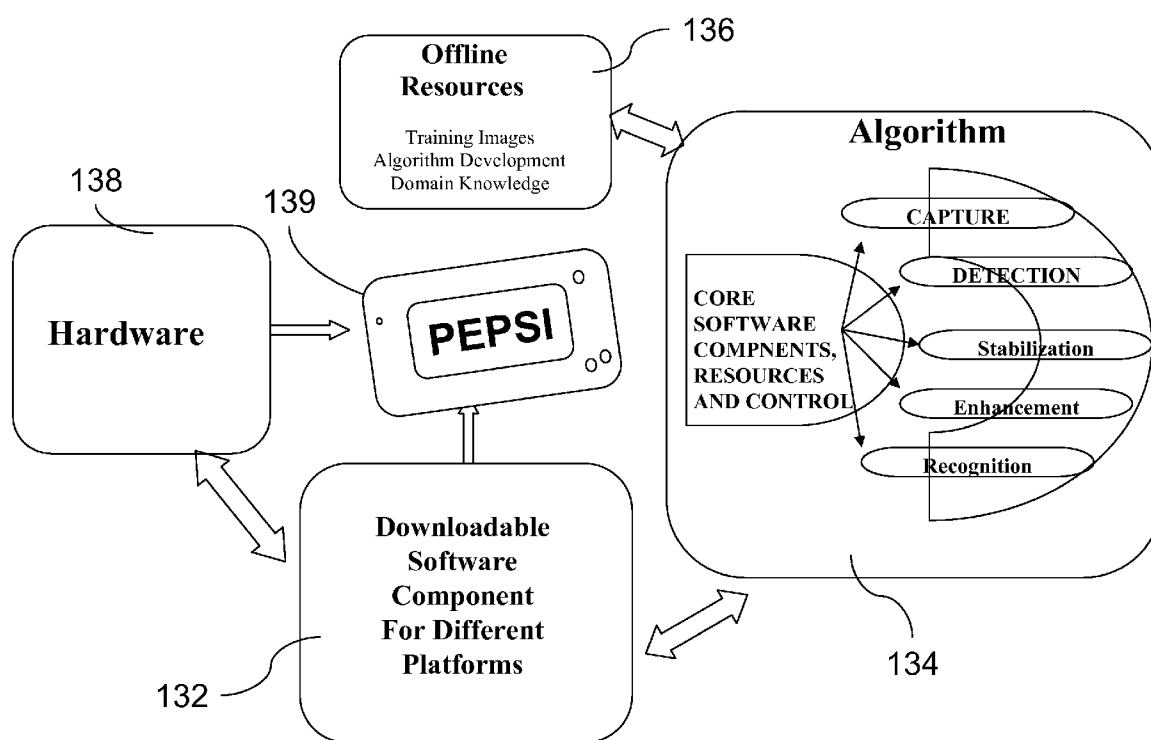


FIG. 3

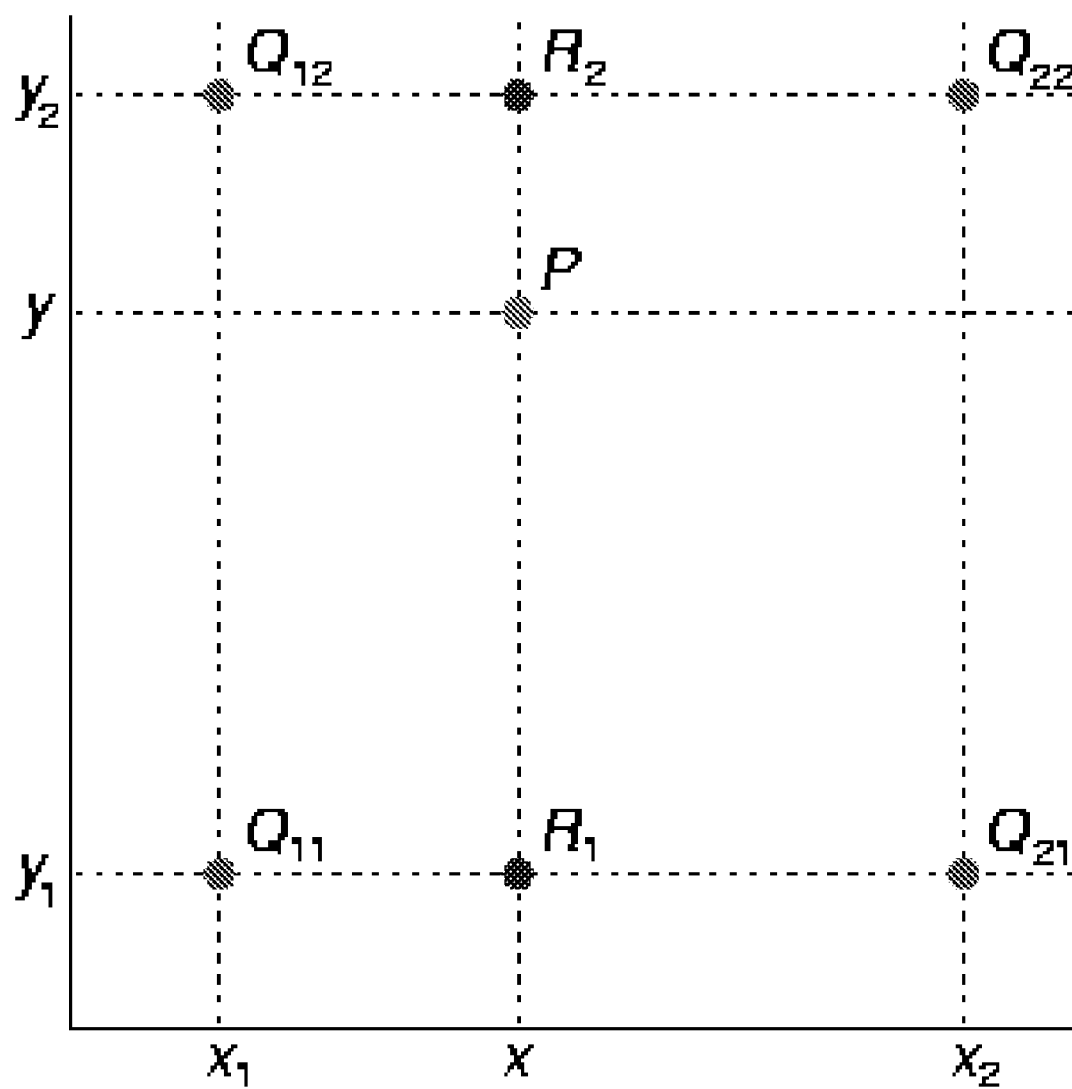
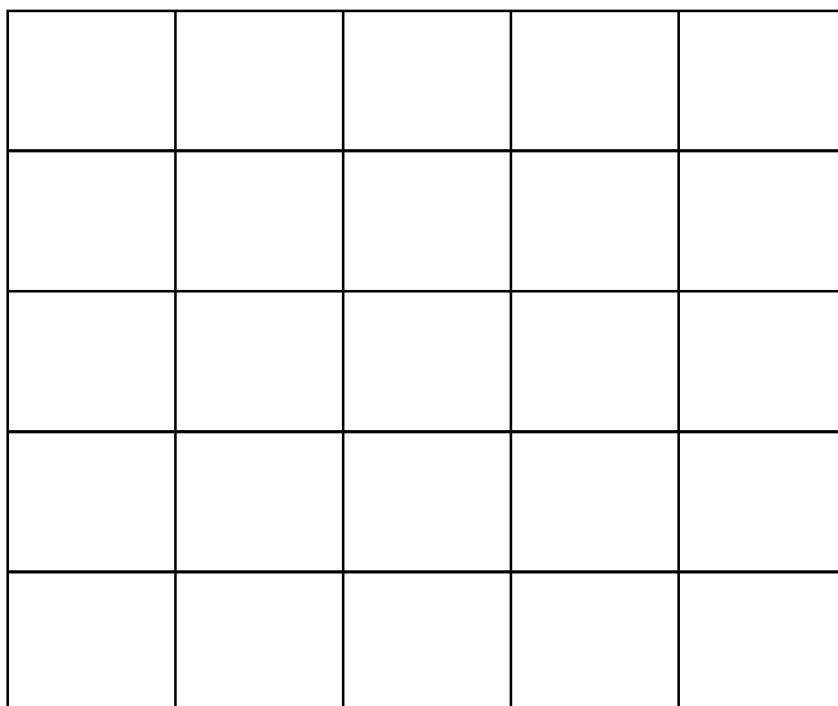


FIG. 4

X,Y

X+1,Y



X,Y+1

X+1,Y+1

FIG. 5

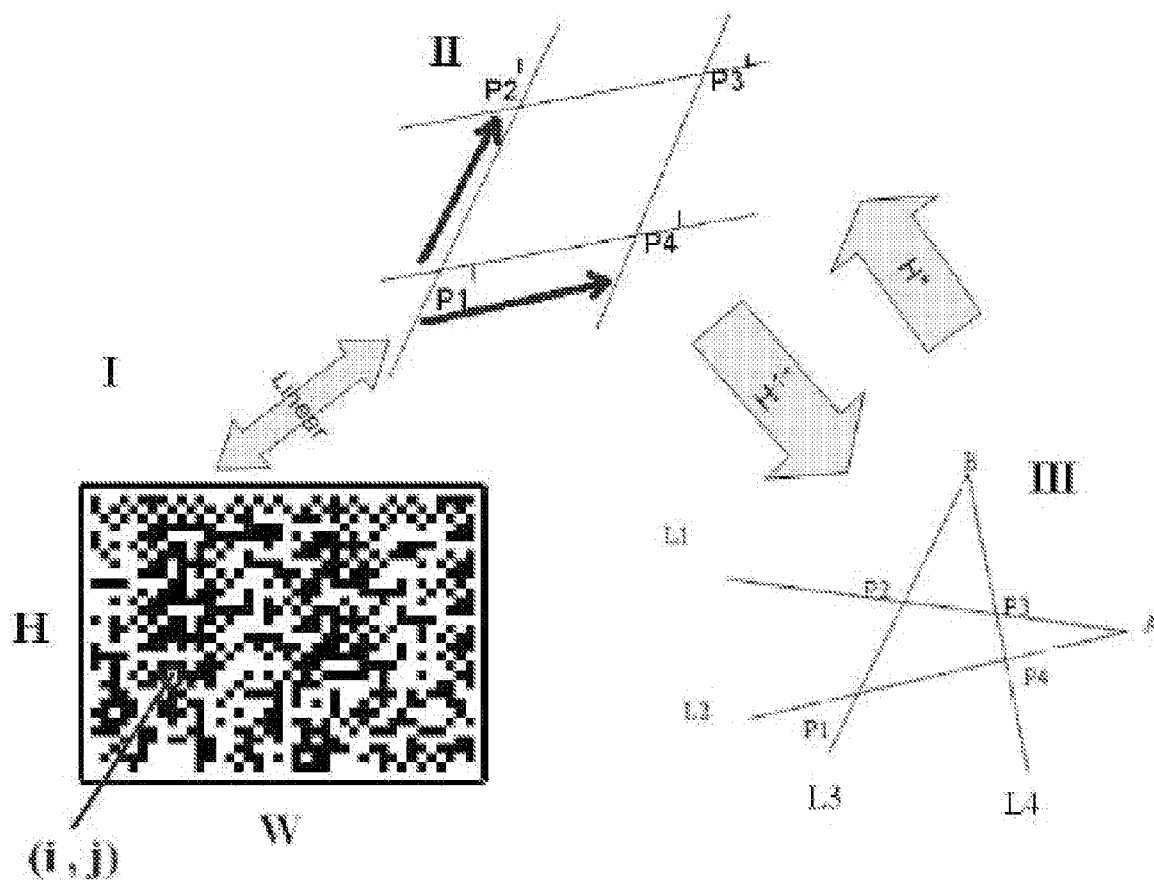


FIG. 6

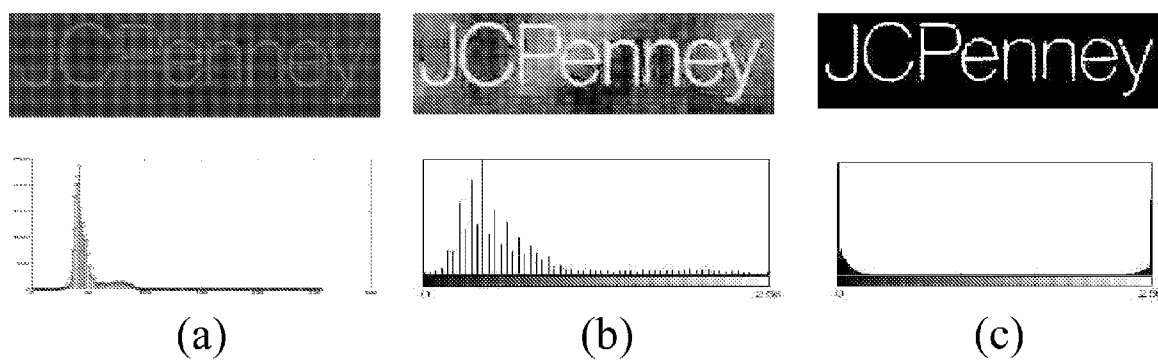


FIG. 7

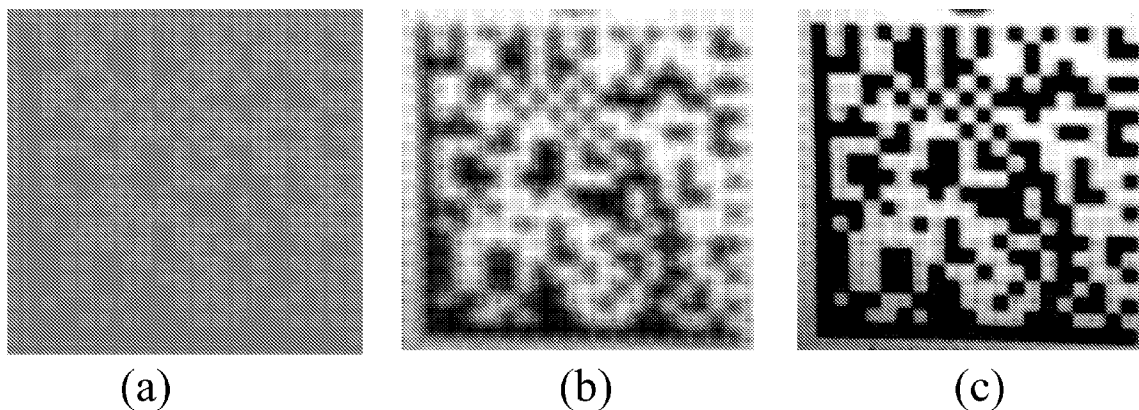


FIG. 8

950 North Glebe Road, Lower Floor • Arlington, Virginia 22203
Voice (703) 841-3017 • (703) 812-3812 • Fax (703) 812-3813

(a)

950 North Glebe Road, Lower Floor • Arlington, Virginia 22203
Voice (703) 841-3017 • (703) 812-3812 • Fax (703) 812-3813

(b)

950 North Glebe Road, Lower Floor • Arlington, Virginia 22203
Voice (703) 841-3017 • (703) 812-3812 • Fax (703) 812-3813

(c)

FIG. 9

950 North Glebe Road, Lower Floor • Arlington, Virginia 22203
Voice (703) 841-3017 • (703) 812-3812 • Fax (703) 812-3813

(a)

950 North Glebe Road, Lower Floor • Arlington, Virginia 22203
Voice (703) 841-3017 • (703) 812-3812 • Fax (703) 812-3813

(b)

950 North Glebe Road, Lower Floor • Arlington, Virginia 22203
Voice (703) 841-3017 • (703) 812-3812 • Fax (703) 812-3813

(c)

FIG. 10

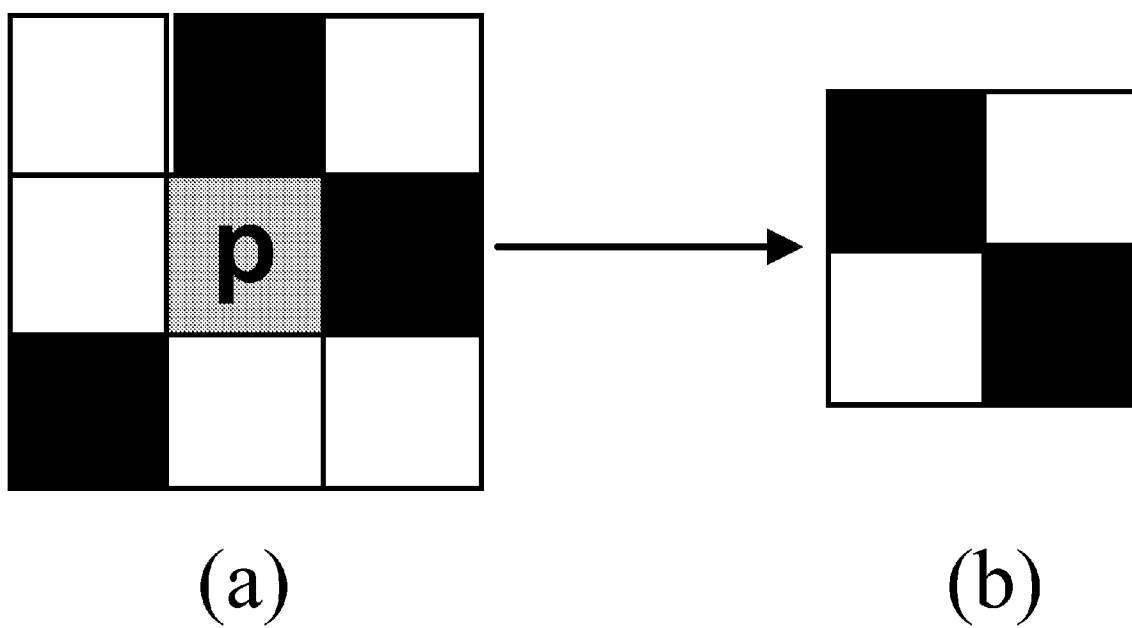


FIG. 11

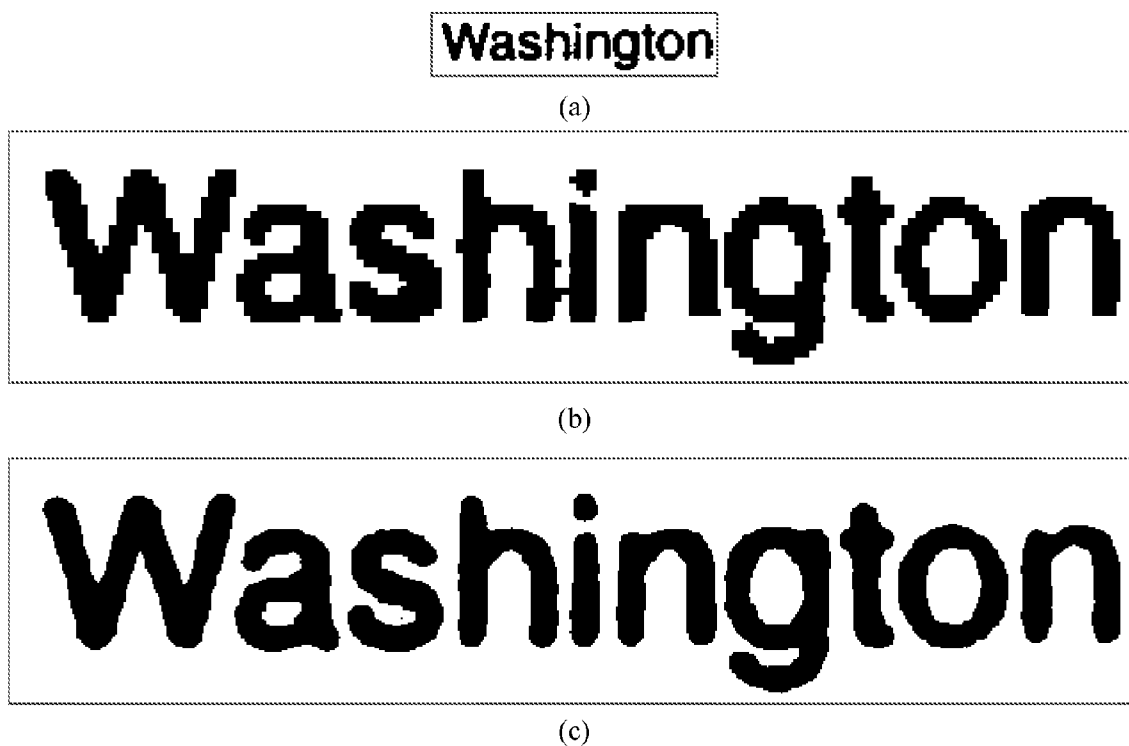
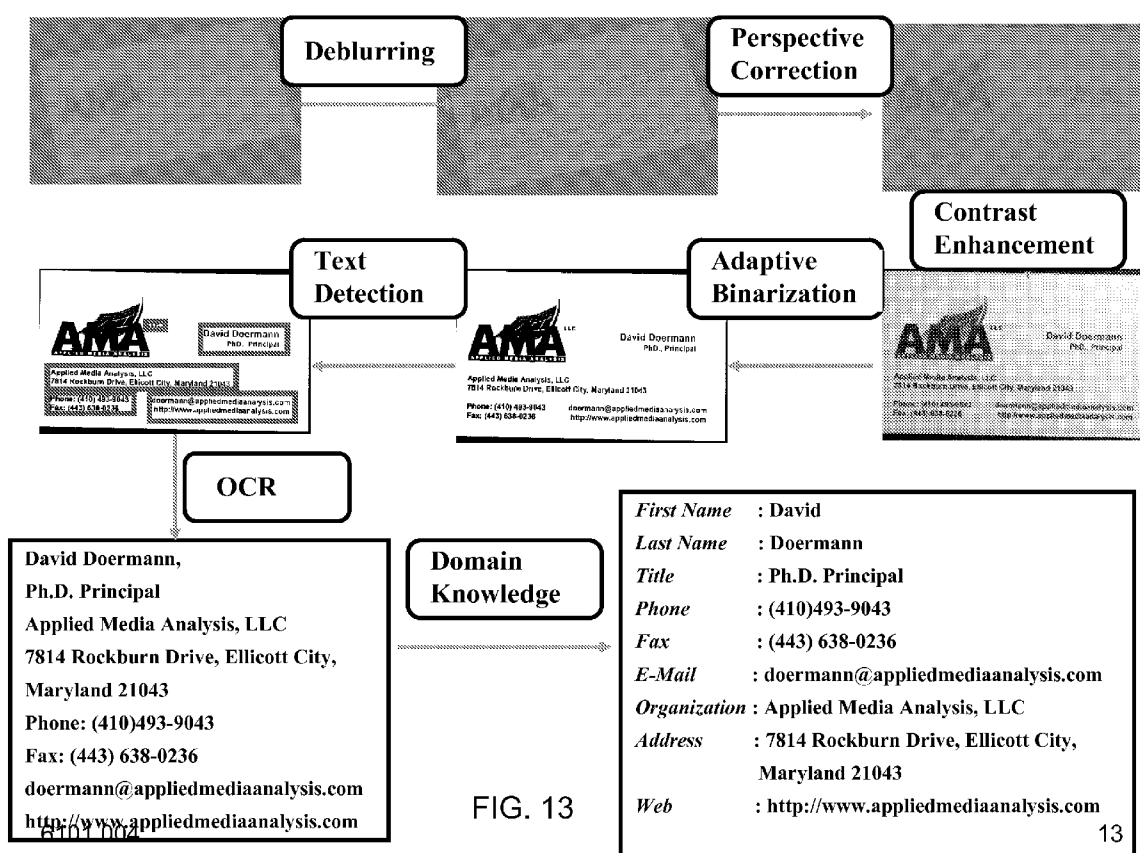


FIG. 12



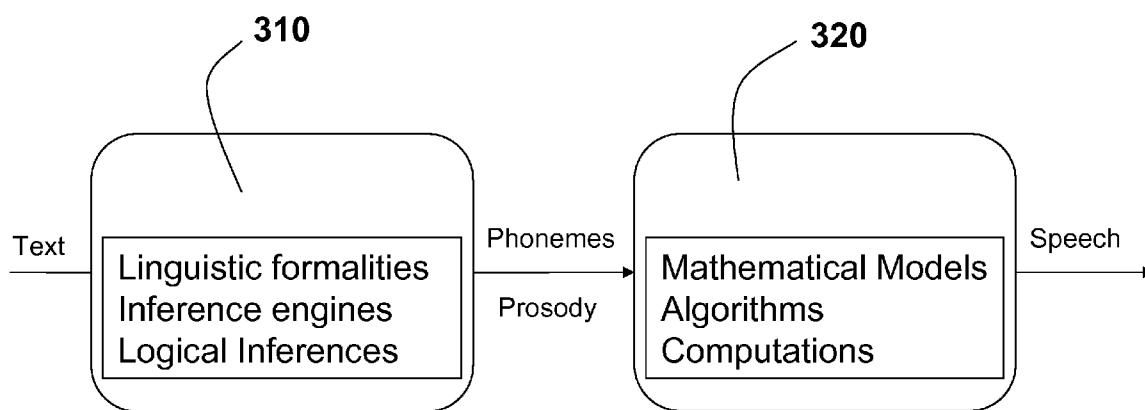


FIG. 14

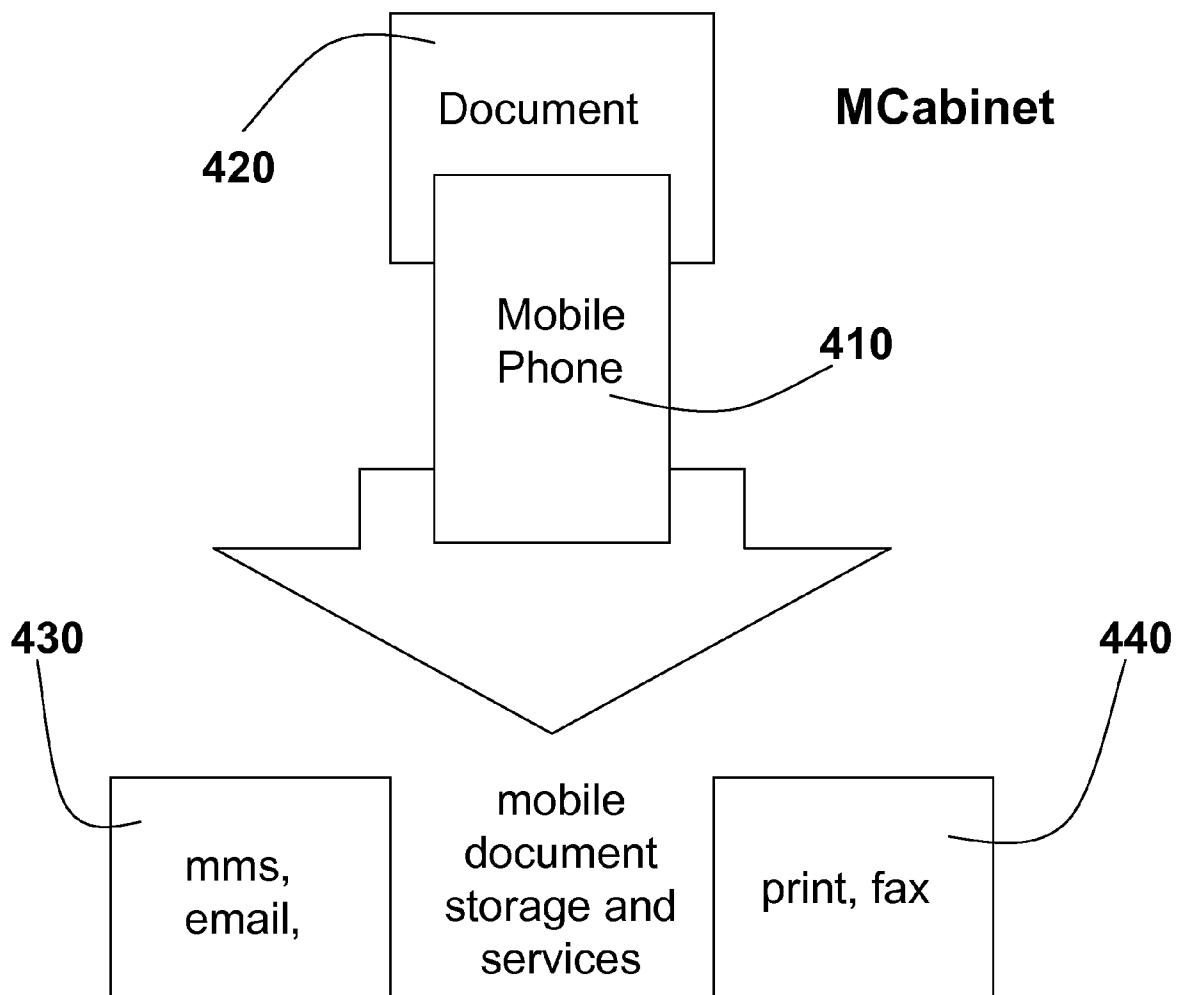


FIG. 15

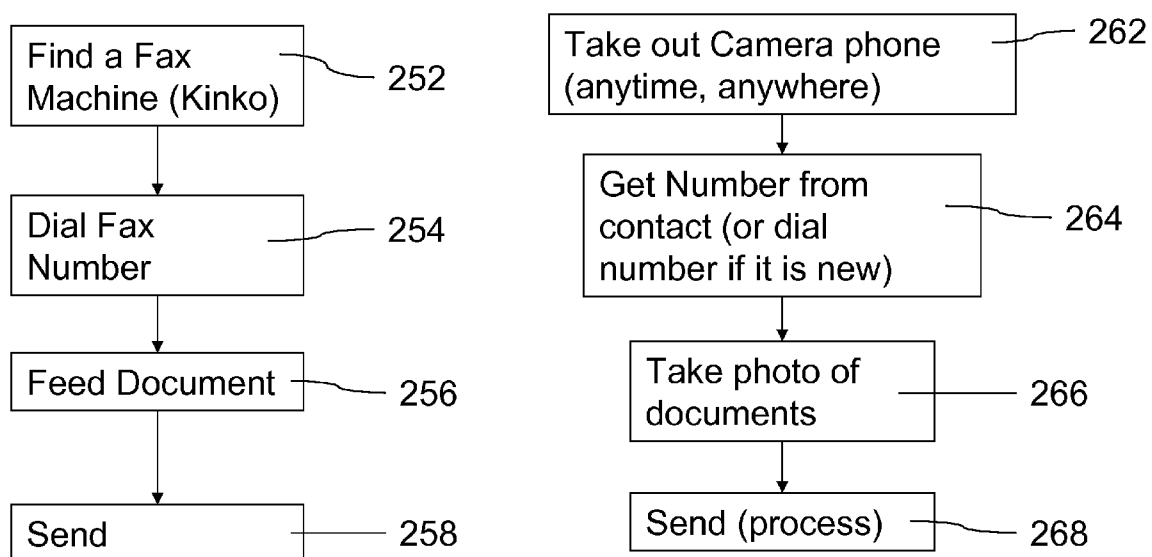


FIG. 16

THE LAFACE INTEGRAL FOR A WATERFLOODED SEGMENTATION

ASSIST

transform. It
ned lines, w
aplace value
realino

second long cigarette. While doing so, he took three papers
out of his pocket, the yellow "Lucky" and searched
for cigarettes. Of the three cigarettes, a standard size the
"Lucky" or "Havanna" or other. They are the
characteristics of all other cigarettes. Figure 10.

The above representation usually protects itself against the charge that the group is "repressed" and is conscious of a "collective unconscious" or "latent" desire. It insists that genuine or otherwise good or possible or latent intentions and wishes that it is personifying are by no means repressed. It says its thinking is controlled by forces like the average person's and somewhat larger, which are conscious to the group as well as the other groups. Thus, there are also good for the group, identification, to control or contain, identified by other nations, and to the mind, for the little understood behavior of the national inner state.

[illegible]

SECRET

Abstract

The respondent reports a total of 100 minutes of contact in December 1970. From 12/1 to 12/10, the respondent spent 10 minutes in contact with the subject. From 12/11 to 12/20, the respondent spent 10 minutes in contact with the subject. From 12/21 to 12/31, the respondent spent 80 minutes in contact with the subject.



Page 1 of 1



Figure 1

1. The first step is to identify the problem or question that needs to be answered. This involves understanding the context and the specific requirements of the task.

FIG. 17

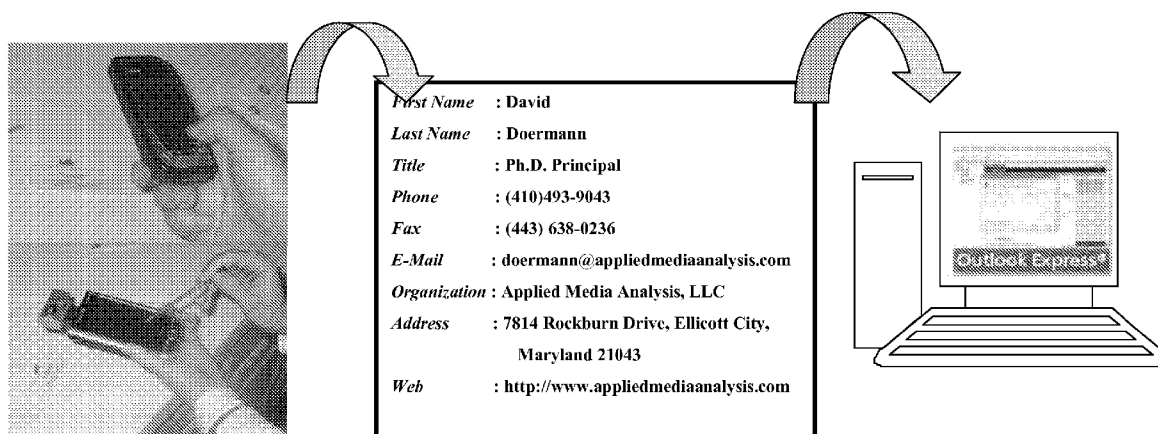


FIG. 18

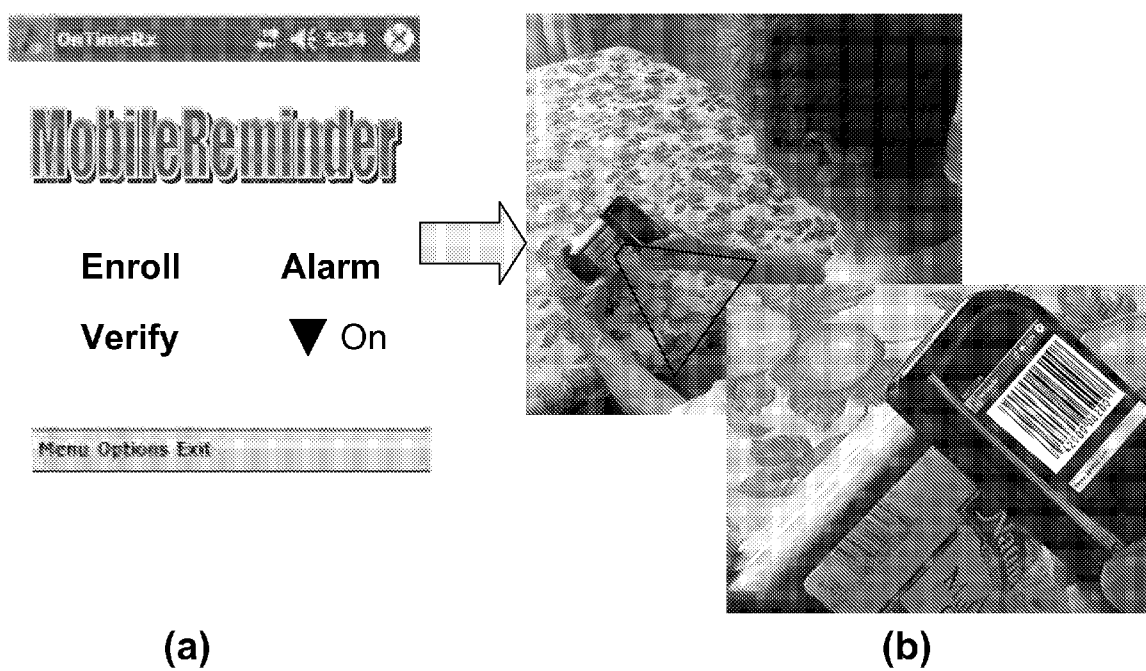


FIG. 19

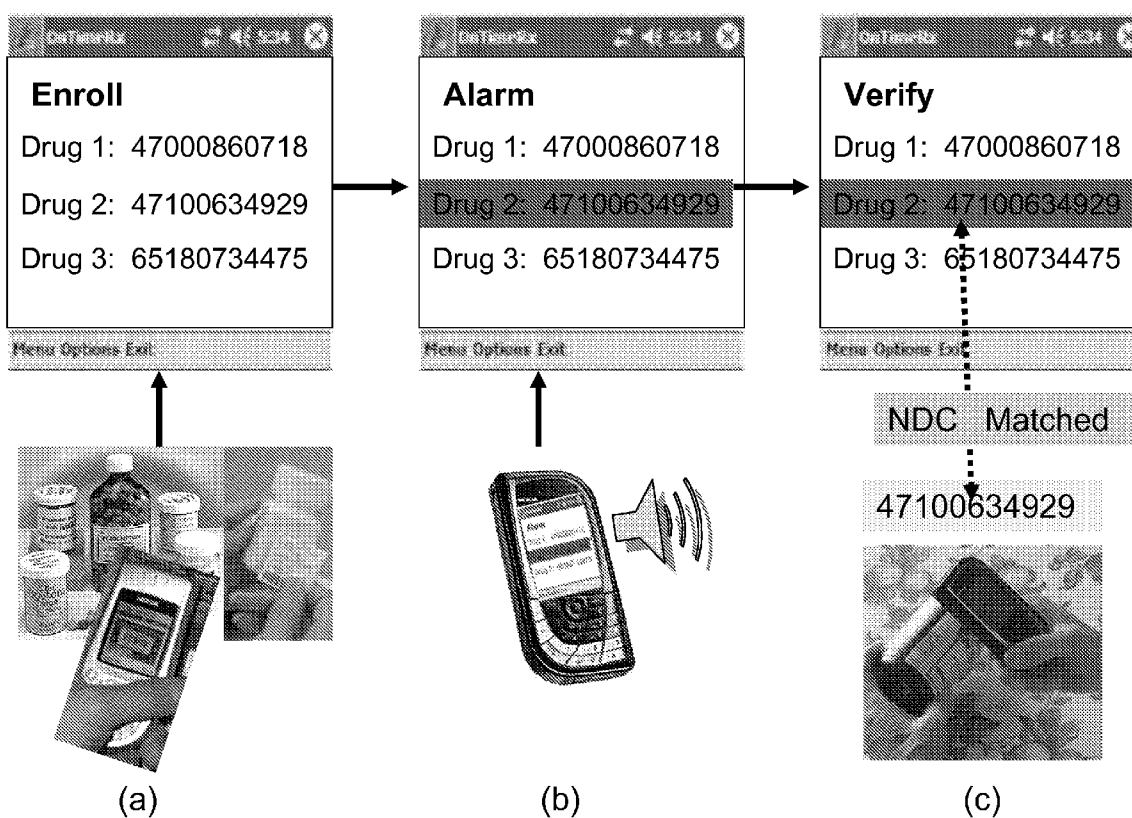


FIG. 20

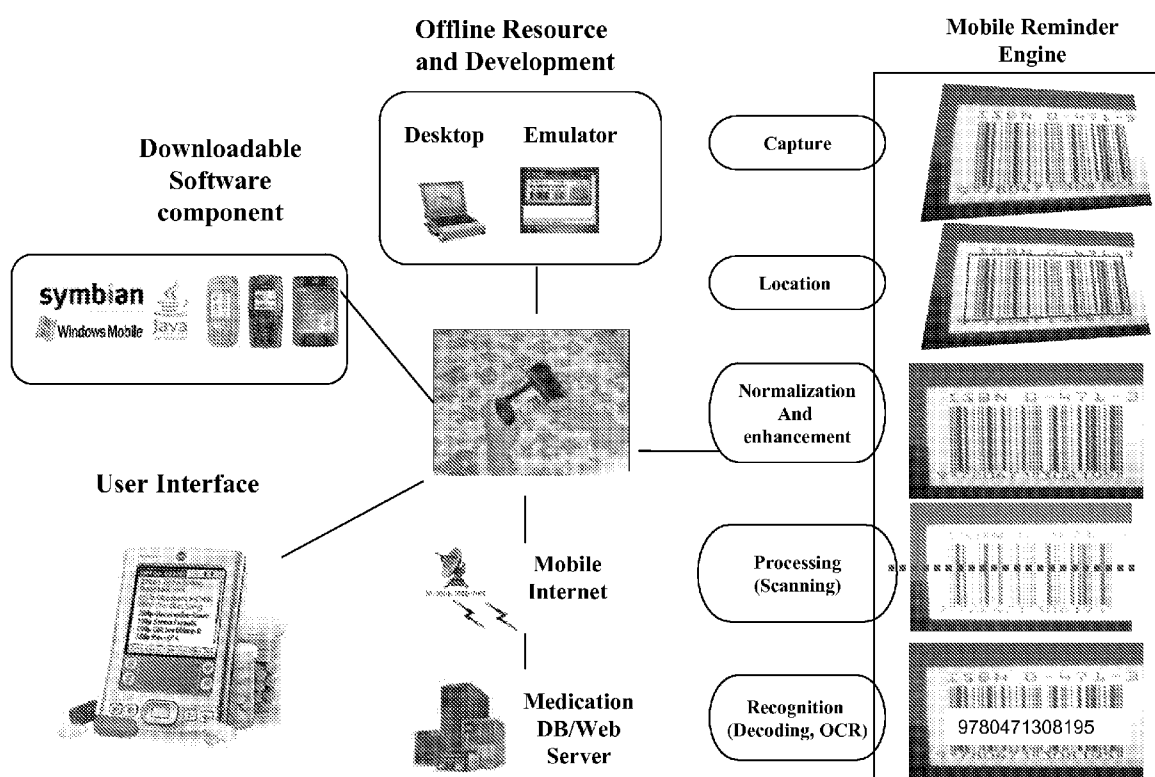
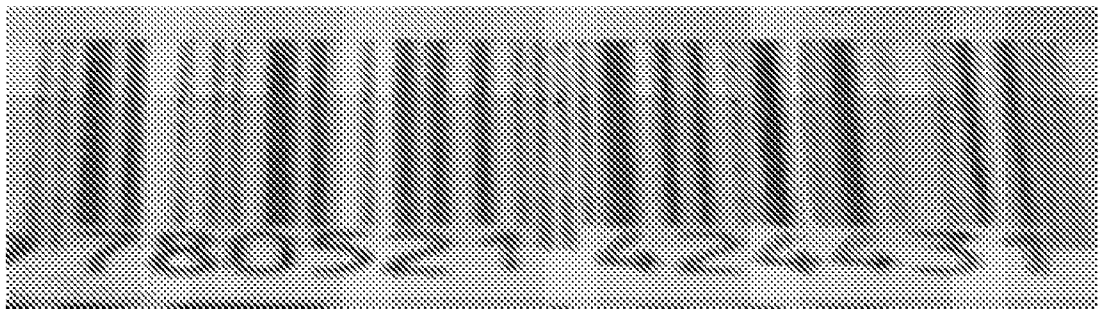
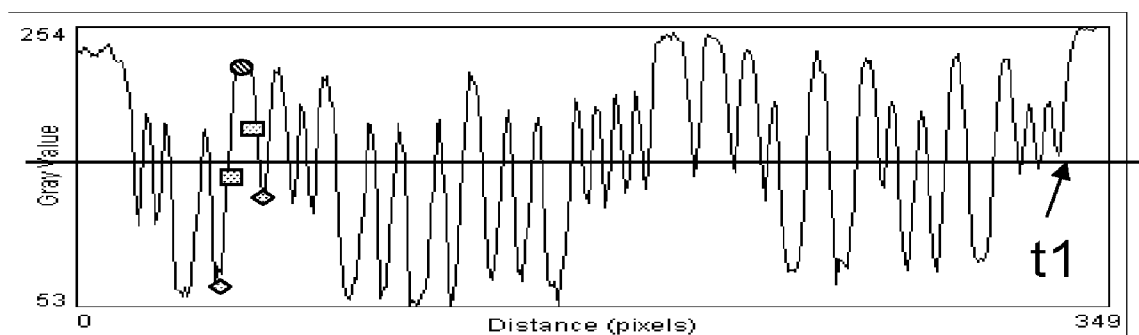


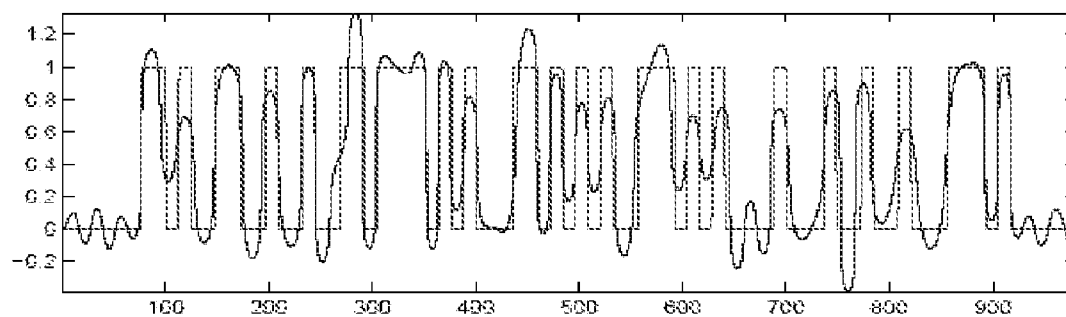
FIG. 21



(a)



(b)



(c)

FIG. 22

SYSTEM AND METHOD FOR EFFICIENT ENHANCEMENT TO ENABLE COMPUTER VISION ON MOBILE DEVICES

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] The present invention claims the benefit of the filing dates of the following U.S. Provisional Patent Applications: Ser. No. 60/806,081, entitled "Mobile Image Enhancement" and filed on Jun. 28, 2006 by David Doermann and Huiping Li; Ser. No. 60/746,752, entitled "Business Card Reader" and filed on May 8, 2006 by David Doermann and Huiping Li; Ser. No. 60/746,755, entitled "Medication Reminder" and filed on May 8, 2006 by David Doermann and Huiping Li; and Ser. No. 60/806,083, entitled "Symbol Acquisition and Recognition" and filed on Jun. 28, 2006 by David Doermann and Huiping Li.

[0002] These prior applications are hereby incorporated by reference in their entirety.

STATEMENT REGARDING FEDERALLY SPONSORED RESEARCH OR DEVELOPMENT

[0003] None.

BACKGROUND OF THE INVENTION

[0004] 1. Field Of The Invention

[0005] The present invention relates to systems and methods for enhancement and usage of enhanced images captured using mobile devices.

[0006] 2. Brief Description Of The Related Art

[0007] Previously, systems and methods have been developed for image superresolution and for text detection and tracking in digital video. See, for example, Changjiang Yang, Ramani Duraiswami, and Larry Davis, "Superresolution Using Preconditioned Conjugate Gradient Method," University of Maryland, Computer Vision Laboratory; and Huiping Li, David Doermann, and Omid Kia, "Automatic Text Detection and tracking in Digital Video," IEEE Transactions on Image Processing, Vol. 9, No. 1, January 2000. In recent years, it has become commonplace for mobile devices such as cell phones, personal digital assistants (PDA's) and mobile computers to have integrated cameras. Such cameras are commonly used for taking pictures and videos, but have not been further developed for use in various applications.

[0008] Traditional image processing and computer vision algorithms designed for desktop or laptop computers often assume sufficient memory, processor speed, image quality and battery power. Mobile devices, however, typically are resource limited and cannot be viewed as traditional computers. The cameras on such mobile devices typically are low resolution CMOS technology; processors are an order of magnitude slower than typical desktop or laptop computers and may not have floating point capability; and memory in the mobile devices typically is limited in how it can be used and is much slower than desktop or laptop memory. For these reasons, any algorithms that we have must be implemented carefully and efficiently to ensure that the performance requirements can be met for new applications.

SUMMARY OF THE INVENTION

[0009] The present invention is a system and method for enhancing images taken on a mobile camera device to enable the mobile device, for example, a personal digital assistant (PDA) or cell phone, to provide enhanced imaging capabilities.

[0010] In a preferred embodiment, the present invention is a method for enhancing an image on a mobile device. The method comprises the steps of pre-calculating a pixel value at each point on a grid and storing the pre-calculated pixel values in a lookup table, using one bit to represent each pixel in the image, quantizing the image at a small step interval such that each pixel in the image corresponds to one point on the grid, and interpolating the image through a memory-indexing process. The method may further comprise the step of performing clustering based contrast enhancement on the image prior to the step of using one bit to represent each pixel in the image.

[0011] In another preferred embodiment, the present invention is a system for enhancing an image on a mobile device. The system comprises means for pre-calculating a pixel value at each point on a grid, means for storing the pre-calculated pixel values in a lookup table, means for using one bit to represent each pixel in the image, means for quantizing the image at a small step interval such that each pixel in the image corresponds to one point on the grid, and means for interpolating the image through a memory-indexing process. The system may further comprise means for performing clustering based contrast enhancement on the image.

[0012] In another embodiment, the present invention is a method for enhancing an image on a mobile device, wherein coordinates of four corners (P_1 , P_2 , P_3 and P_4) of a bounding box in the image are known, top and bottom boundaries of the bounding box intersect at a vanishing point A and right and left boundaries of the bounding box intersect at a vanishing point B. The method comprises the steps of calculating a mapping between an ideal, non-perspective image and the image and for any matrix entry (i, j) in a $w \times h$ matrix compute its affine coordinate $i/wP_1P_4 + j/hP_1P_2$ and use H^{-1} to map this affine coordinate to the image coordinate. The calculated mapping comprises a plane-to-plane homograph matrix $H=(H_1, H_2, H_3)$, wherein the step of calculating a mapping comprises the steps of reshaping matrix H as a vector $h=(h_{11}, h_{12}, h_{13}, h_{22}, h_{23}, h_{24}, h_{31}, h_{32}, h_{33})^T$, calculating H_3 according to the equation

$$\begin{cases} H_3 \cdot A = 0 \\ H_3 \cdot B = 0 \end{cases} \Rightarrow H_3 \sim A \times B,$$

and $H_3 \sim ((P_1 \times P_2) \times (P_2 \times P_3)) \times ((P_1 \times P_2) \times (P_3 \times P_4))$, calculating H according to the equation

$$H = \begin{pmatrix} h_{33} & 0 & 0 \\ 0 & h_{33} & 0 \\ h_{31} & h_{32} & h_{33} \end{pmatrix},$$

calculating H^{-1} according to the equation

$$H^{-1} \sim \begin{pmatrix} h_{33} & 0 & 0 \\ 0 & h_{33} & 0 \\ -h_{31} & -h_{32} & h_{33} \end{pmatrix},$$

and mapping P_1, P_2, P_3 and P_4 to affine points P'_1, P'_2, P'_3, P'_4 using homography H .

[0013] In another embodiment, the present invention is a method for enhancing an image on a mobile device. The method comprises the steps of, for each pixel in the image, determining if binarization is required based upon an $N \times N$ neighborhood using a block-based approach. If binarization is not necessary for a particular neighborhood, set all pixels in the particular neighborhood to background. For each pixel requiring binarization, calculating a binarization threshold using Niblack's approach and conducting binarization, and post-processing the binary image to remove ghost objects.

[0014] In still another embodiment of the present invention, the present invention is a method for enhancing an image on a mobile device comprising the steps of representing each foreground pixel in the image with a pattern vector generated from pixel values in an $N \times N$ neighborhood of the foreground pixel and converting each foreground pixel to f^2 pixels in a higher resolution image where f is a magnification factor. How to convert a foreground pixel depends upon the pattern vector, $k-1$ other pattern vectors in the image that are similar to the pattern vector where similarity is measured by a Hamming distance of two pattern vectors, and pixels in the higher resolution image corresponding to the k pattern vectors.

[0015] In still other embodiments, the present invention is an application incorporating some or all of the image enhancement methods and systems. In one such embodiment, the present invention is a medication reminder system on a mobile device. The system comprises a means such as a camera for acquiring digital images of barcodes, means for enhancing acquired digital images, means for enrolling medication in the system using the digital images, means for scheduling medication intakes, and alarm means for notifying a user of a time to take a particular medication or medications. The system may further comprise means for verifying medication intakes.

[0016] Still other aspects, features, and advantages of the present invention are readily apparent from the following detailed description, simply by illustrating a preferable embodiments and implementations. The present invention is also capable of other and different embodiments and its several details can be modified in various obvious respects, all without departing from the spirit and scope of the present invention. Accordingly, the drawings and descriptions are to be regarded as illustrative in nature, and not as restrictive. Additional objects and advantages of the invention will be set forth in part in the description which follows and in part will be obvious from the description, or may be learned by practice of the invention.

BRIEF DESCRIPTION OF THE DRAWINGS

[0017] For a more complete understanding of the present invention and the advantages thereof, reference is now made to the following description and the accompanying drawings, in which:

[0018] FIG. 1 is a diagram of image magnification in accordance with a preferred embodiment of the present invention.

[0019] FIG. 2 is a block diagram illustrating basic and specific enhancement in accordance with a preferred embodiment of the present invention.

[0020] FIG. 3 is a block diagram illustrating the system architecture of a preferred embodiment of the present invention.

[0021] FIG. 4 is a diagram illustrating an image magnification algorithm using bilinear interpolation in accordance with a preferred embodiment of the present invention.

[0022] FIG. 5 is a diagram illustrating a grid used to store all pixel values which any pixel in the interpolated image will be projected to in accordance with a preferred embodiment of the present invention.

[0023] FIG. 6 is a diagram illustrating fast perspective distortion correction in accordance with a preferred embodiment of the present invention.

[0024] FIG. 7 illustrates contrast enhancement in accordance with a preferred embodiment of the present invention.

[0025] FIG. 8 illustrates a clustering-based contrast enhancement method compared with a histogram-stretching based method.

[0026] FIG. 9 is a diagram illustrating a modified Niblack's binarization result in accordance with a preferred embodiment of the present invention where (a) is an original image, (b) is a binary image without post-processing, and (c) is a post-processed binary image.

[0027] FIG. 10 is a diagram illustrating binary image post-processing in accordance with a preferred embodiment of the present invention where (a) is the original binary image with background noise removed, (b) is the post-processed binary image after gap filling, and (c) is the post-processed image after stroke quality improvement.

[0028] FIG. 11 is a diagram illustrating a text super-resolution method with magnification factor 2 in accordance with a preferred embodiment of the present invention. The center pixel P in (a) is magnified to four pixels in (b) based on its neighborhood, which records the statistical text shape information based on training.

[0029] FIG. 12 is a diagram illustrating a fast text super-resolution enhancement by making use of text shape patterns in accordance with a preferred embodiment of the present invention. The original image is zoomed 4 times in (b) and (c), where (a) is an original low resolution image, (b) is a bi-linear interpolation, and (c) is a result of a text super-resolution method in accordance with a preferred embodiment of the present invention.

[0030] FIG. 13 illustrates optical character recognition of degraded text in accordance with a preferred embodiment of the present invention.

[0031] FIG. 14 illustrates test-to-speech for audio feedback in accordance with a preferred embodiment of the present invention.

[0032] FIG. 15 illustrates an example of mobile file management in accordance with a preferred embodiment of the present invention.

[0033] FIG. 16 is a flow diagram illustrating mobile faxing in accordance with a preferred embodiment of the present invention.

[0034] FIG. 17 illustrates a mobile magnifying glass in accordance with a preferred embodiment of the present invention.

[0035] FIG. 18 is a diagram of a smart phone-based business card management system in accordance with a preferred embodiment of the present invention.

[0036] FIG. 19 is a diagram of a camera phone-based medication reminder system in accordance with a preferred embodiment of the present invention.

[0037] FIG. 20 is a diagram of a camera phone-based medication reminder system in accordance with a preferred embodiment of the present invention.

[0038] FIG. 21 is a diagram of a system architecture for a camera phone-based medication reminder system in accordance with a preferred embodiment of the present invention.

[0039] FIG. 22 illustrates adaptive thresholding methods for 1D and 2D barcode recognition in accordance with preferred embodiments of the present invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

[0040] In the present invention, a software based suite uses camera enabled personal digital assistant (PDA) or cell phone hardware to provide enhanced imaging capabilities. For example, when individuals with low-vision need to read material with low contrast or fine print, they may use their mobile camera device to assist them. As shown in FIG. 1, a mobile camera device 102 is pointed at material 104 a user wishes to read. The vision enhancement tool within the mobile device captures an image of the material 104 to be read and enhances the text (e.g., through deblurring, resolution and/or contrast enhancement) to produce readable text 104. The enhancement may be accomplished according to a profile, for example, set up by a user. The enhanced text may be displayed as shown in FIG. 1 or may be converted to audio and read aloud by speakers embedded in the device.

[0041] As shown in FIG. 2, the vision enhancement tool may provide basic magnification and contrast enhancement capabilities 122 for both near and far vision. Further, when text is present, the system may adapt and optimize parameters 124 such as resolution, perspective, contrast, denoising and deblurring for improved presentation by sharpening and binarization of the content. Other capabilities may include:

[0042] User-defined magnification enhancement. Digital image analysis combined with text line height estimation may scale text to a uniform size (e.g., the size may be known a priori to be of benefit to the user).

[0043] Advanced contrast enhancement. Text on some types of paper, and colored text on colored backgrounds may be difficult to read, even when magnified. Therefore, the contrast and edges may be enhanced and presented in a variety of text and background colors.

[0044] Lighting adjustment. Dark or uneven lighting conditions may make perception difficult. Therefore, adaptive enhancement may be applied to unify lighting.

[0045] Automatic deblurring. The vision enhancement tool may deblur the image so that the text can be displayed shapely.

[0046] Stabilization of images. Images may be stabilized for users with shaky hands.

[0047] Text-to-Speech (ITS). Text within images may be recognized, converted to speech using text-to-speech algorithms, and read aloud (126).

[0048] Among the many tasks where general magnification is beneficial (e.g., using appliances, reading a wrist watch, or even looking at pictures), benefits may be realized with respect to the ability to read text.

[0049] By detecting and making use of the special characteristics of text, text enhancement algorithms may be tuned to improve resolution, contrast and sharpness. For example, users may set a minimal text size displayed, and text lines may be scaled appropriately. Contrast may be customized for the user and for the content. One user with one condition may read better with a given foreground/background combination, while another may prefer the colors reversed. For non-text content, edge enhancement and histogram equalization may provide increased perception, while for text, black characters on an all white background to maximize the contrast may be more easily read. Varying and/or reversing the contrast may be provided as an individual choice.

Functional Overview

[0050] The vision enhancement tool may be software based, and may be easily downloadable as an application to a device. The system may operate in different modes. First, the magnifier mode may provide direct video magnification and contrast enhancement of the content being imaged through the camera. To process images, a process with filters and effectively a digital zoom obtained through pixel replication may be employed. In addition, simple edge enhancement and contrast enhancement filters may be employed. To obtain higher definition or text enhancement, the user may capture a still image, which may be enhanced and displayed on the device. Since the content is magnified digitally, the screen may not be able to display the entire scene at the increased resolution. Therefore, different mechanisms for navigating through the enhanced content may be provided. For example, in order to navigate through the content, the user may move the device as if he/she was panning over the enlarged scene. The process of text enhancement may be partitioned into the tasks of image stabilization and acquisition, text detection, enhancement, and recognition. The system may be based on a dynamically reconfigurable component architecture. The component architecture may make the system easily extendable so more advanced applications can be integrated. In one embodiment, the system may be configured with only the capabilities needed for a particular enhancement task, thus optimizing resources and keeping the flexibility.

[0051] A mechanism may be provided to let the user provide the context so as to accommodate variations in lighting conditions.

System Architecture

[0052] Component architecture may manage considerable resources on a small device. The system may operate in standalone mode providing an integrated capability.

[0053] The software modules may include the user-interface, image acquisition and display module, text detection module, and the enhancement module.

[0054] As shown in FIG. 3, the system may include a set of basic components that are managed by a core software control module 132. The core components may manage resources 134 needed by the analysis modules and swap them in from Microdrive storage on demand. The component architecture may be implemented in, for example, Symbian OS or Microsoft windows mobile. The detection

and enhancement components may be written in, for example, C or C++ first, and then transplanted into different embedded platforms.

[0055] Software reusability and component management may be supported. The component architecture may provide an easy way to develop and test new algorithms **136**, and may provide a basis for moving to new devices **138**, where resources may be even more limited.

[0056] The cameras attached to the camera phone may be directly used as an image capturing device.

[0057] A GUI may continue to display video sequences and capture single images at the same time. When text is shown at the center of the display **139**, the user may hit a button to capture the image, which may be passed to detection and recognition modules.

Image Acquisition and Display Components

[0058] Since captured images may be at the resolution of megapixels, which may be significantly larger than the screen resolution, the user interface may allow the user to browse images within limited screen size and resolution. Image browsers may use scroll bars to cycle through image thumbnails and locate images of interest to inspect in full resolution. Another alternative is Zoomable User Interface technology, which may allow a user to watch images with gradually improved resolutions.

[0059] Additionally or alternatively, the user may navigate larger images or document images by simply moving the device. The basic concept is that after a static image of the scene is obtained and processed, to obtain an enhanced image, the camera may be retargeted to measure relative motion of the device. When the phone is panned across the scene in a given direction, the view port over the enhanced image may be moved in the same direction, giving the user the perception of scanning across the scene. The sensitivity of the motion may be controlled so that the user gets a smooth scan.

Image Enhancement Module

[0060] Text enhancement algorithms **157**, **158** may be performed prior to display. The techniques may include, for example, perspective distortion correction, image stabilization, deblurring, contrast enhancement, noise removal and resolution enhancement.

Enhancement 1: Fast Image Magnification

[0061] The present invention implements an efficient magnification method to improve text resolutions. Generally bilinear interpolation requires floating point calculations, which make it extremely slow since there is no floating point processor in most smart phones. The real time implementation of image magnification at arbitrary scale is a fundamental requirement for lots of applications in mobile image related applications. The simple replication of pixels can satisfy the real-time requirement, the artifact, however, is very obvious.

[0062] The present invention uses a look-up table to accelerate the bi-linear interpolation to achieve real-time

performance in mobile phones. In this way, the computational speed in the embedded image processing library is accelerated.

Speedup Bi-linear Image Interpolation Algorithms.

[0063] When zooming in/out of an image, a pixel in the new image is often projected back at a point with non-integer coordinates in the original image (Point P in FIG. 4). Therefore, we need to estimate the value of Point P from its four neighboring points with integer coordinates ($Q_{11}(x_1, y_1)$, $Q_{12}(x_1, y_2)$, $Q_{21}(x_2, y_1)$ and $Q_{22}(x_2, y_2)$). To determine the pixel value at point P, first the linear interpolation in X direction is performed:

$$\left. \begin{aligned} f(R_1) &\approx \frac{x_2 - x}{x_2 - x_1} f(Q_{11}) + \frac{x - x_1}{x_2 - x_1} f(Q_{21}) \\ f(R_2) &\approx \frac{x_2 - x}{x_2 - x_1} f(Q_{12}) + \frac{x - x_1}{x_2 - x_1} f(Q_{22}) \end{aligned} \right\} \quad (1)$$

where
 $R_1 = (x, y_1)$
 $R_2 = (x, y_2)$

And then interpolation in Y direction is performed:

$$f(P) \approx \frac{y_2 - y}{y_2 - y_1} f(R_1) + \frac{y - y_1}{y_2 - y_1} f(R_2) \quad (2)$$

Substituting (1) into (2), we have

[0064]

$$f(x, y) \approx \frac{f(Q_{11})(x - x_2)(y - y_2)}{(x_1 - x_2)(y_1 - y_2)} + \frac{f(Q_{21})(x - x_1)(y - y_2)}{(x_1 - x_2)(y_1 - y_2)} + \frac{f(Q_{12})(x - x_2)(y - y_1)}{(x_1 - x_2)(y_1 - y_2)} + \frac{f(Q_{22})(x - x_1)(y - y_1)}{(x_1 - x_2)(y_1 - y_2)} \quad (3)$$

[0065] From this formula one can estimate how many floating point multiplication and addition operations are required to finish the process. Since we know that $(x_1 - x_2) = -1$ and $(y_1 - y_2) = -1$, and we need to calculate $f(Q_{11})(x - x_2)(y - y_2)$, $f(Q_{21})(x - x_1)(y - y_2)$, $f(Q_{12})(x - x_2)(y - y_1)$, and $f(Q_{22})(x - x_1)(y - y_1)$ these four formulas, each of which requires two floating point subtractions and multiplications. Therefore, each pixel in the interpolated image will requires $2 \times 4 + 3$ floating point additions (subtractions), and 2×4 floating point multiplications. This means the interpolation of an image at VGA (640×480) resolution requires $640 \times 480 \times 11 = 3379200$ floating additions, and $640 \times 480 \times 8 = 2457600$ floating multiplications. As we tested on a DELL Av50 PDA (650 MHz CPU, 64 MB memory) the interpolation of an image to VGA resolution takes almost 2 minutes. The reason is, mobile devices often use software emulation to process floating point calculation, instead of specific hardware floating point processor in PC.

[0066] Since many applications mainly handle the text, one bit can be used to represent each pixel: black for foreground and white for background, or versa visa. This is true since clustering based contrast enhancement was per-

formed first. Therefore, $[f(Q_{11}) f(Q_{12}) f(Q_{21}) f(Q_{22})]$ has at most 16 combinations. We quantize $(x-x_1)$ and $(y-y_1)$ at a small step interval t , as shown in FIG. 5. Each pixel in the interpolated image will correspond to one grid point in FIG. 5. Therefore, we only need to pre-calculate the pixel value at each grid point and store it. The smaller t is, the larger the number of grid points is, and the more memory it requires to store the values. In our case we let $t=0.01$. The size of the look-up table which stores all the pixel values is $100 \times 100 \times 16=160$ KB.

[0067] After pre-calculating the look-up table, the image interpolation becomes a memory-indexing process without any floating calculation. As we tested on Dell Av50 PDA, it takes only 10 milliseconds to interpolate an image at VGA resolution. This means we achieved over 200 times acceleration in PDA. The acceleration comes from: 1) The elimination of all floating point calculation, and 2) look-up table. When we move the same experimental protocol to the desktop PC, however, we only observed around $5 \times$ acceleration, since there is floating point processor in Desktop PC. The $5 \times$ acceleration is mainly achieved through the look-up table. The cost of this acceleration is 160 KB extra memory.

Enhancement 2: Perspective Distortion Correction

[0068] Users may capture the text image from any arbitrary angle. To read the text we need to correct the perspective distortion first. The first step is to calculate the mapping between the ideal, non-perspective image and the real-captured image, which can be described as a plane-to-plane homograph matrix H . For any matrix entry (i, j) , $\hat{H}$ maps homogeneous coordinate $x=(i, j, 1)$ to its image coordinate $X=\hat{H}x$. Suppose we know n matrix entries $(x_i, y_i, 1)$, and their corresponding image points $(X_i, Y_i, 1)$, where $i=1, 2, \dots, n$. The classical way of computing $\hat{H}$ is the homogeneous estimation method: First, reshape matrix H as a vector $h=(h_{11}, h_{12}, h_{13}, h_{22}, h_{23}, h_{24}, h_{31}, h_{32}, h_{33})^T$, and then solve for $Mh=0$.

[0069] When $n=4$, h is the null vector of M and we have a unique solution for h (assume $h_1=1$ or $h_{33}=1$). This means what we only need the coordinates of four corners (P_1, P_2, P_3, P_4) in FIG. 6 to compute the homography $\hat{H}$. It is very expensive, however, to solve the equation in camera phones. It usually requires LU decomposition with pivoting, which often involves significant amount of floating point calculation which is not supported by mobile phones at the hardware level. Instead, the operating systems (Symbian, Windows Mobile) provides a software emulation of IEEE-754 64 bit floating point which is much slower than integer operations. Other platforms, such as Java(J2ME), provide no floating point capabilities. This motivates us to design a simpler/faster algorithms without floating point operations, and we map out a very promising approach.

[0070] As shown in FIG. 6, an affine transformation is performed first and then a perspective transformation. Suppose we know the coordinates of four corners (P_1, P_2, P_3, P_4) in the image plane and the top and bottom boundaries of the bounding box intersect at vanishing point A . Then under homogeneous coordinates $A=L_1 \times L_2 + (P_1 \times P_4) \times (P_2 \times P_3)$. Similarly the left and right boundaries intersect at $B=L_3 \times L_4 + (P_1 \times P_2) \times (P_3 \times P_4)$. A and B are infinite points in the original plane. The third element of A and B under homogeneous coordinates should be 0 in the affine image. Any

homography $H=(H_1, H_2, H_3)$ that maps the perspective image back into affine image should map A and B to infinity, which implies

$$\begin{cases} H_3 \cdot A = 0 \\ H_3 \cdot B = 0 \end{cases} \Rightarrow H_3 \sim A \times B, \text{ and} \quad (1)$$

$$H_3 \sim ((P_1 \times P_2) \times (P_2 \times P_3)) \times ((P_1 \times P_2) \times (P_3 \times P_4))$$

[0071] This indicates we can calculate H_3 using only seven cross products. Any homography H with the third row H_3 calculated by (1) maps the perspective image to an affine image. The next task is to fill the first and second row of Matrix H . The reason we calculate this homography H is: Given any matrix coordinate we can quickly tell its pixel coordinate in the image. From the matrix coordinate (I) to the affine image (II), the transformation is linear and can be easily computer by transforming the base of the coordinate system. In final step we need to transform the affine image (II) to the perspective image (III) by computing H_1 . We choose the first and second row of H so that it has a near inverse. We have

$$H = \begin{pmatrix} h_{33} & 0 & 0 \\ 0 & h_{33} & 0 \\ h_{31} & h_{32} & h_{33} \end{pmatrix} \quad \text{and} \quad H^{-1} \sim \begin{pmatrix} h_{33} & 0 & 0 \\ 0 & h_{33} & 0 \\ -h_{31} & -h_{32} & h_{33} \end{pmatrix} \quad (2), (3)$$

[0072] This inverse only requires the reverse of two signs in the third row of H . In this way it simplifies the coordinate transformation with numerical stability. Normally the numerical inverse often suffers from "division by zero" when H is nearly singular. In summary, we computer the coordinate transformation in the following way:

[0073] Calculate H_2 using Equation 1

[0074] Calculate H and H^{-1} using Equation (2) and (3)

[0075] Map P_1, P_2, P_3, P_4 to affine points P'_1, P'_2, P'_3, P'_4 using H

[0076] For any entry (i, j) in the $w \times h$ matrix compute its affine coordinate

$$\frac{i}{w} P'_1 P'_4 + \frac{j}{h} P'_1 P'_2$$

and use H^{-1} to map this affine coordinate to the image coordinate.

No floating point computation is required in the above procedures.

Enhancement 3: Contrast Enhancement

[0077] Under some adverse imaging conditions the majority of the pixel values may lie in a narrow range, potentially making them more difficult to discriminate. One technology that may make it easier to discern subtle contrasts is called contrast enhancement which may stretch the values in the range where the majority of the pixels lie. Mathematically, contrast enhancement may be described as $s=T(r)$, where r is the original pixel value, T is the transformation, and s is the transformed value. T may be linear or non-linear, depending on the practical imaging conditions. The prin-

ciple is to make light colors (or intensity) lighter and dark colors darker at the same time, so the total contrast of an image can be increased. FIG. 7 illustrates an original low-contrast image (FIG. 7(a)) and improved images using a histogram-stretching based method (FIG. 7(b)) and a cluster based contrast enhancement (FIG. 7(c)).

Cluster Based Contrast Enhancement

[0078] The text and background pixels may form two clusters. When image contrast is high, the distance between two cluster centers may be larger. Therefore, a clustering based contrast enhancement method that uses this unique feature of text images for contrast enhancement, may be used to achieve contrast enhancement. First, the two clusters may be found, and then the contrast may be enhanced based on the two clusters.

[0079] Histogram stretching may be a very common and effective approach to general contrast enhancement. However, it may not be the ideal technique when the content is pseudo binary. This is illustrated in FIG. 8. FIG. 8(a) is an original low-contrast image, and FIG. 8(b) is the contrast enhanced image after performing a histogram stretching technique. Although the contrast may be increased, some background pixel values also may be stretched, making the black cell units hard to separate from background.

[0080] The black block and background pixels form two clusters. When image contrast is high, the distance between two cluster centers is larger, and vice versa. Therefore, the two clusters may be found and used to enhance the contrast. The algorithm is described as:

[0081] 1. Initialization. Choose two initial cluster centers $C1(0)$ and $C2(0)$ representing the black clock and background pixels, which can be random values between, for example, 0 and 255 for gray scale images. Practically the Convergence may be accelerated if $C1(0)$ and $C2(0)$ are selected as values between the minimum, maximum, and the mean of the image pixel values.

[0082] 2. Pixel Clustering: For each pixel in text image $I(i,j)$ at iteration n , calculate the minimum distance: $d(i,j) = \arg \min_k \|I(i,j) - C_k(n)\|, k=1,2$. The pixels then may be allocated to the cluster with the minimum distance. In this way, the pixels may be partitioned into two clusters $C1$ and $C2$ based on this distance measure. The error at iteration n may be calculated as:

$$e(n) = \frac{1}{M \times N} \sum_{i=0}^M \sum_{j=0}^N d(i, j)$$

where $M \times N$ is the size of the image. The iteration may stop when $e(n)$ is smaller than a preset threshold.

[0083] 3. Updating: Generate the new location of the center by averaging the pixel values in each cluster:

$$C1(n) = \frac{1}{N_{C1}} \sum C1(i, j); \quad C2(n) = \frac{1}{N_{C2}} \sum C2(i, j);$$

where N_{C1} and N_{C2} are the number of pixels in $C1$ and $C2$ respectively. The iteration stops where $e(n)$ does not decrease.

4. Smart Stretching

[0084] After two cluster centers are determined, one center may be put at a small value (0, for example), and another may be put at a large value (255, for example), and the histogram may be stretched based on these two centers.

[0085] FIG. 7(c) is an example of a contrast enhancement result based on the cluster based contrast enhancement approach.

Enhancement 4: Denoising

[0086] To make the algorithm be able to applied on mobile devices, a novel binarization method is used which combines the Niblack's and block-based binarization approaches. The approach consists of the following three steps: (i) For each pixel, determine if binarization is required based on a $N \times N$ neighborhood using the block-based approach. If binarization is unnecessary, then all pixels inside this neighborhood are set to background and skipped. (ii) For each pixel requiring binarization, calculate the binarization threshold using Niblack's approach and conduct binarization. (iii) Post-process the binary image to remove 'ghost' objects.

[0087] A special implementation of the computation of sample mean and standard deviation significantly improves the speed of binarization. Given the neighborhood size 5×5 , for a pixel at position (i, j) , we compute the standard deviation of pixel values in its neighborhood and then decide if the whole block need binarization based on a predefined threshold T_b . If no binarization is required for this block, then we mark all pixels inside this block background and move to the next undecided pixel, which is $(i, j+2)$ in the example. In this way, we can remove all the computation for pixels that don't need binarization. The implementation of this approach is described in details as follows.

[0088] To save the computation time, for each image, we pre-compute the accumulated sum AS and square sum ASQ as, where $p(i, j)$ is the pixel value at position (i, j) :

$$AS(i, j) =$$

$$\begin{cases} p(i, j) & \text{if } i = 0, j = 0 \\ AS(i, j-1) + p(i, j) & \text{if } i = 0, j > 0 \\ AS(i-1, j) + p(i, j) & \text{if } i > 0, j = 0 \\ AS(i-1, j) + AS(i, j-1) - AS(i-1, j-1) + p(i, j) & \text{otherwise} \end{cases}$$

$$ASQ(i, j) = \begin{cases} p(i, j) * p(i, j) & \text{if } i = 0, j = 0 \\ ASQ(i, j-1) + p(i, j) * p(i, j) & \text{if } i = 0, j > 0 \\ ASQ(i-1, j) + p(i, j) * p(i, j) & \text{if } i > 0, j = 0 \\ ASQ(i-1, j) + ASQ(i, j-1) - ASQ(i-1, j-1) + p(i, j) * p(i, j) & \text{otherwise} \end{cases}$$

[0089] After AS and ASQ are obtained, the sample mean m and standard deviation s in a block with left-top corner (i, j) and right-bottom corner (k, l) are computed as:

$m =$

$$m = \begin{cases} AS(k, l) / K & \text{if } i = 0, j = 0 \\ (AS(k, l) - AS(i, j - 1)) / K & \text{if } i = 0, j > 0 \\ (AS(k, l) - AS(i - 1, j)) / K & \text{if } i > 0, j = 0 \\ (AS(k, l) - AS(i, j - 1) - AS(i - 1, j) + AS(i - 1, j - 1)) / K & \text{otherwise} \end{cases}$$

$s = \sqrt{ss - m \cdot m}$, where K is the number of pixels in this block and ss is computed as

$$ss = \begin{cases} ASQ(k, l) / K & \text{if } i = 0, j = 0 \\ (ASQ(k, l) - ASQ(i, j - 1)) / K & \text{if } i = 0, j > 0 \\ (ASQ(k, l) - ASQ(i - 1, j)) / K & \text{if } i > 0, j = 0 \\ \left(\frac{ASQ(k, l) - ASQ(i, j - 1) - ASQ(i - 1, j) + ASQ(i - 1, j - 1)}{K} \right) & \text{otherwise} \end{cases}$$

[0090] To save memory which is critical for mobile devices, the above-mentioned operations are conducted on a image strip with size $N \times W$, where W is the image width and N is the block height. Each time, only values for the middle row pixels in this strip are computed. Once the calculation is done and results are stored, the first row data in the strip is discarded and a new row will be computed based on the previous rows and added to the end of the strip. The process continues until the whole image is finished. By this implementation, we not only saved the computation time, but also saved the intermediate memory/space usage.

Binary Image Post-processing to Remove Ghost Object

[0091] To remove the 'ghost' objects generated by Niblack's binarization approach, the post-processing step used in Yanowitz and Bruckstein's method (see S. D. Yanowitz and A. M. Bruckstein, "A New Method for Image Segmentation", *Computer Vision, Graphics and Image Processing*, Vol. 46, No. 1, pp. 82-95, April 1989) is selected to improve the binarization result. In this step, the average gradient value at the edge of each foreground object is calculated. Objects having an average gradient below a threshold T_p are labeled as misclassified and removed. The detailed procedure is described as follows:

[0092] Smooth the input image by averaging the image in a 3×3 neighborhood.

[0093] Compute the gradient magnitude image G of the smoothed image using Sobel's edge operator.

[0094] For all connected foreground object, compute the average gradient of the edge pixels that are defined to be pixels connected to the background. Remove those objects having an average edge gradient value below threshold T_p .

[0095] After this post-processing step, most background noise introduced by Niblack's method will be removed. However, the binarized text image is not smooth, especially at the edge of each character, there exist many small spurs which reduced the readability of text. Another observation about this approach is it might introduce broken strokes or holes in the binary image. Further post-processing steps are required to improve the text stroke quality. FIG. 9 shows an example of the binarization result. In FIG. 9, (a) is the original image; (b) is the binarized image before post-

processing; and (c) is the binarized image after post-processing. We can see the image shown in FIG. 3(c) is much cleaner than the original image.

Binary Image Post-processing to Improve Text Quality

[0096] To improve the text quality, a swell filter described in R. J. Schilling, "Fundamentals of Robotics Analysis and Control", Prentice-Hall, Englewood Cliffs, N.J., 1990, is selected to fill the possible breaks, gaps or holes, and improve the text stroke quality. The procedure is described as follows:

Scan the entire binary image with a sliding window having size $N \times N$.

[0097] Suppose the central pixel at (x, y) in the sliding window is a background pixel, and the average coordinates of the foreground pixels inside this window is (x_a, y_a) .

[0098] Then the central pixel is changed to foreground if $P_{sw} > k_{sw}$ and $|x - x_a| < dx$ and $|y - y_a| < dy$, where P_{sw} is the number of foreground pixels in the window, $k_{sw} = 0.05N^2$, and $dx = dy = 0.25N$.

[0099] An extension of the above conditions is applied to improve the text stroke quality. Scan the entire binary image with a sliding window having the same size $N \times N$. Whenever the central pixel of the window is a background pixel, count the number of foreground pixels P_{sw1} inside this window and change the central background pixel to foreground if $P_{sw1} > k_{sw1}$, where $k_{sw1} = 0.35N^2$.

[0100] Applying this approach to real mobile-device-captured images, we obtained promising results, which are shown in FIG. 10(f), where (a) is the original binary image with background noise removed; (b) is the post-processed image after break, gap and hole filling; and (c) is the post-processed image after the text stroke quality improvement.

Enhancement 5: Text Resolution Enhancement

[0101] The general text resolution enhancement method does not make use of specific information about text shape.

[0102] To reduce the magnifying artifacts we need to make use of text shapes, which can provide information to maintain the high fidelity of the image even when image is magnified at large times.

[0103] The present invention uses a text super-resolution enhancement approach based on text shape training. The original method is proposed in H. Y. Kim, "Binary Operator Design by k-Nearest Neighbor Learning with Applications to Image Resolution Increasing.", *International Journal Imaging Systems and Technology*, Vol. 11, pp. 331-339, 2000, but is very expensive. We will optimize the algorithm to make it be able to run on mobile devices with limited resources and computation capability.

[0104] The basic idea of the method we propose to implement is as follows. Each foreground pixel in the low resolution image is represented with a pattern vector which is generated from pixel values in a $N \times N$ neighborhood of this pixel. FIG. 11(a) shows a foreground pixel with value P in the low resolution image and its neighbors in a 3×3 neighborhood. These 9 pixels will join together to generate a pattern vector reflecting the central pixel. The vector that represents this pattern is $[p_0 p_1 p_2 p_3 p_4 p_5 p_6 p_7 p_8]$, where p_i ($i=0, 1, \dots, 8$) are values of pixels in this neighborhood (0

or 1 for binary image). The magnification of low resolution image is to convert each foreground pixel in the low resolution image to f^2 (f is the magnification factor) pixels in the high resolution image. FIG. 11(b) shows the foreground pixel p is converted to four pixels with values P_0, P_1, P_2 and P_3 in the high resolution image when $f=2$. How to convert a foreground pixel in the low resolution image is determined by: (i) The pattern that describes the foreground pixels; (ii) $k-1$ other patterns in the low resolution image that are similar to the pattern in (i), where the similarity between patterns is measured by the Hamming distance of two pattern vectors; and (iii) The f^2 pixel values in the high resolution training image corresponding with these k patterns. In the following, we use magnification factor 2 and neighborhood size 3×3 as an example to describe the detailed training phase. Assuming the training data consists of only two noiseless (ideal) images with the same text content. One image (labeled as I_1) has the low resolution 200 dpi, the other one (labeled as I_2) has the high resolution 400 dpi.

Training Procedures:

[0105] Find all different patterns (different pattern number is 2^9 for a 3×3 neighborhood) representing all foreground pixels in I_1 . For each appeared pattern instance, find four possible corresponding pixel values in image I_2 . A voting vector for each foreground pixel pattern in I_1 is computed. Assume a single pattern appears M times in I_1 , the corresponding magnification pixel values in I_2 are $P^{(j)} = [P_0^{(j)} P_1^{(j)} P_2^{(j)} P_3^{(j)}]$, where $j=1, \dots, M$, then the voting vector $C = [C_0 C_1 C_2 C_3]$ for this foreground pixel pattern in I_1 , is computed as follows:

$$C_i = \sum_{j=1}^M P_i^{(j)} \quad i = 0, \dots, 3$$

[0106] For each pixel pattern in I_1 , search for the k nearest patterns measured using Hamming distance, and the corresponding voting vectors $C^{(l)}$, $l=1, \dots, k$. Based on all voting vectors of these patterns, the trained magnification output $[P_0 P_1 P_2 P_3]$ for this pattern is defined as follows:

$$P_i = \begin{cases} 0 & \text{if } \sum_{l=1}^k C_i^{(l)} < C_h \\ 1 & \text{otherwise} \end{cases} \quad i = 0, 1, 2, 3$$

where C_h is half of the total number of pixels of the k patterns that attend the voting.

[0107] The training results are put into a look-up table which has 2^9 rows and four columns. Four cells in each row store four output pixel values in image I_2 . The binary string of the index of the table represents the pixel patterns in I_1 . For example, if a foreground pixel in I_1 has the pixel pattern $[0 \ 1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ 1]$ and the four pixels it corresponds in I_2 have value $[0 \ 1 \ 1 \ 1]$. Then the 185th ($010111001=185$) row of the look-up table has values $[0 \ 1 \ 1 \ 1]$.

[0108] The training phase is finished once the look-up table is created. When magnifying a given binary image I_u by factor 2, we only examine the foreground pixel, find the

corresponding pattern and convert it to four pixels in the magnified image. For example, if a foreground pixel at position (x, y) of I_u has pattern $[0 \ 1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ 1]$, then the pixels at positions $(2x, 2y)$, $(2x+1, 2y)$, $(2x, 2y+1)$ and $(2x+1, 2y+1)$ of the magnified image have values 0, 1, 1 and 1, respectively. Since the training operations can be performed offline, the magnification operation is converted to the searching of pixel pattern and looking-up in a look-up table. By this modification, text super-resolution operation can be finished in a short time. Experimental results during the writing of this proposal show that this approach can be made to run very fast on mobile devices. FIG. 12(c) shows a magnified result using this approach, which is much better than a simple magnification and magnified result using the bi-linear approach, as shown in FIG. 12(b).

Speed and Memory Requirement Analysis:

[0109] Most super-resolution algorithms are extremely expensive and can not be embedded in the phone. For the present invention, the memory required is the size of the look-up table addressed above. It increases exponentially with respect to the neighborhood size, and linearly with the square of magnification factor f i.e. f^2 . Given a neighborhood size $N \times N$, and magnification factor f the look-up table size will be $2^{N \times N} \cdot f^2$ bits, if we use one bit to represent a pixel (which is true in the binary image case). For instance, if the neighborhood size is 3×3 , and the magnification factor $f=2$, then the look-up table only occupies 2048 ($2^9 \cdot 4$) bits or 512 bytes with half of bits unused. If the neighborhood size is 4×4 , and the magnification factor is also 4, then the look-up table occupies 1048576 ($2^{16} \cdot 16$) bits or 131072 bytes. In our initial experiments we found even with $N=3$, the result is significantly improved. This will leave the magnification factor as high as possible.

[0110] The most time will be spent on off-line training. After training the magnification is just the memory access process of the look-up table.

[0111] The following applications are based on the enhancement techniques described above:

Application 1: Mobile OCR and Text-to-Speech

[0112] The device may also be used to read and present textual information, as illustrated in FIG. 13. For example, when an individual with low-vision needs to read a sign, he/she may take out his/her device and point it at the sign, hit a button, and the recognized text in the sign may be read out through text-to-speech (TTS).

[0113] Exemplary uses:

[0114] 1. An elderly person with low vision does not know if he or she should make a turn on the next street. They take out their camera phone and aim it toward the street panel. The street name is recognized and read out from the speaker of the camera phone.

[0115] 2. A visually impaired grandmother wants to purchase over the counter medication and she doesn't know if the product in her hand is what she is looking for or not. Rather than seeking assistance, she turns on the device and it reads the labels on the bottle and converts the product information to speech.

[0116] 3. A grandfather receives a business card from a friend or an appointment card from a doctor's office. He wants to save the contact number into his cell phone or add the appointment to a reminder service, but finds it

is really hard to input through the small keypad in the cell phone. He then captures an image of the business card; immediately text reading software converts the physical card into tagged electronic contact info, and automatically adds it to his contact list in the cell phone.

Optical Character Recognition (OCR) from Degraded Text HMM Model for OCR from Degraded Text

[0117] Text captured from camera phones may be degraded even after enhancement. For example, touching and/or broken characters may be common. Once the text region is segmented, a hidden Markov model (HMM) approach may be used to handle the touching or broken characters. In this approach a statistical language model may be created in terms of bi-gram co-occurrence probabilities of symbols and models for individual characters. This method may simultaneously segment and recognize characters based on a statistical model.

[0118] In the HMM approach, each character may be represented using a discrete HMM. HMM is a generative model: at each discrete time the system may be in a particular state. In this state it throws out one of the allowed symbols. The symbol picking process may be random and may depend on the probability of each symbol in that state. After a symbol is thrown out, the system may jump to another state according to a state transition probability. The HMM parameters may be for example: symbol probability within each state, bi-gram state transition probability, and initial state probability. Model training may be performed as follows: each text line image may be broken into a left-to-right sequence of overlapping sub-images; each sub-image then may be converted into a discrete observation symbol by using a vector quantization scheme; the observation sequence and the corresponding transcription (ASCII groundtruth text) then may be used to estimate the model parameters.

[0119] The recognition process may split the text line image into a sequence of sub-images and convert the sub-image sequence into a sequence of discrete symbols. A dynamic programming algorithm may be used to segment the symbol sequence into a sequence of recognized characters.

Knowledge-Driven OCR

[0120] To further improve OCR accuracy, knowledge in a specific domain may be used to refine the OCR result from the recognition engine. A database consisting of digitized samples of reading material for each task may be developed and used to characterize the distributions of print parameters (e.g., size, font, contrast, color, background pattern, etc.) for each task. The system parameters may be specifically selected based on these specific application domains.

Contextual Dictionaries

[0121] The words that appear in the list of ingredients in a product may be from a very restricted vocabulary. In fact, once the generic category of a product is known, the words that may appear in the contents may be further restricted. Domain knowledge may be used to improve the recognition accuracy of the OCR subsystem. The knowledge may be represented as, for example, dictionaries and/or thesauri.

Furthermore the consumer may add words that they encounter in their daily living and create their own user dictionaries.

Query Driven Recognition

[0122] The system may allow users to spot keywords in large document repositories or in isolated documents in the field. At times, the consumer may be searching for the existence or absence of certain ingredients in a product. For example, an asthma patient might want to confirm that a bottle of wine does not contain sulfites. In such a scenario, words other than "sulfites" (and various orthographic renditions thereof) may not be important. A user-interface may be provided so the user can specify the word.

Output and Presentation

[0123] As described above, both audio and visual feedback of recognized text may be provided to users. For visual feedback, the enhanced text may be overlaid on the display of the camera phone.

[0124] Audio feedback may be provided by a Text-To-Speech (TTS) synthesizing technology that reads text out through speakers attached to the camera phone.

[0125] FIG. 14 is an example of a functional diagram of a TIS synthesizer. The synthesizer may include, for example, a Natural Language Processing module (NLP) 310 capable of producing a phonetic transcription of the text read, together with the desired intonation and rhythm (often termed as prosody), and a Digital Signal Processing module (DSP) 320, which may transform the symbolic information it receives into speech.

Application 2: Using Camera Phone as a Photo Copy and Faxing Machine

[0126] A camera equipped mobile phone having image enhancement capabilities may allow the capture and transmission of full page documents, as shown in FIG. 15. When a camera equipped mobile phone 410 is used to capture images of a document 420 to be transferred, the image quality of the captured documents may be degraded due to effects such as, for example, radial distortion, and lighting. However, such degradations may be overcome using the image enhancement techniques described above. In addition, compression and transmission methods may be used to allow a user to acquire, manipulate, store and retrieve documents through the user's mobile device. As such, a camera enabled mobile phone may be used to accomplish the equivalent of office copying, faxing, filing, e-mailing, etc. 430, 440.

[0127] Image processing capabilities may dynamically enhance images captured by a camera equipped mobile phone thereby providing copy quality suitable for reproduction, storage or faxing. In addition, captured documents may be stored automatically and mirrored on a server so as to not overwhelm the limited memory available on a mobile device. After a document has been mirrored on a server, a compact signature identifying the document may remain on the mobile device to facilitate document retrieval. Complex document search capabilities may be available on the server side. For example, documents mirrored on the server may be enhanced with OCR metadata and converted to PDF format to enable complex search capabilities. Acquired images may be faxed, emailed and/or shared with other users from the mobile telephone and/or the server. The process for selecting

which documents should be mirrored on a server and when documents should be mirrored on a server may be tailored according to a user's preferences.

[0128] According to this example, users may fax or email documents acquired by camera phones anywhere and any-time. The unique document enhancement capabilities may remove excess background and provide a readable document image.

[0129] FIG. 16 shows a comparison of conventional faxing 252, 254, 256, 258 versus mobile faxing 262, 264, 266, 268. When a user desires to send a fax, the user may do so using his/her camera phone. The user can either select a fax number directly from a local or hosted contact, or the user can input the fax number using a keypad on the mobile phone (step 264). Then the user may acquire images of the documents to be faxed using the camera equipped mobile phone (step 266). The documents may be one page or multiple pages. The image processing software may process the acquired document(s) to, for example, improve image quality, compress them so they can be sent quickly, and add digital watermarking for security purposes. The enhanced document images may be sent to the fax number through, for example, telecommunication network (step 268).

EXAMPLE 2

Mobile Magnifying Glass

[0130] The techniques and systems disclosed herein may allow a camera equipped mobile phone to be used as a mobile magnifying glass. Two modes may be provided. (1) Continuous video mode. In this mode, the camera phone may be used just like an optical magnifying glass. For example, the user may move the camera phone around a document (or scene) he wants to read (or view), and an image is captured, enhanced and magnified, as shown in FIG. 17. The user may then use the camera phone to scan the document (or scene) and captured images will be magnified continuously just as if the user were scanning the document or scene with an optical magnifying glass. (2) In a second mode, a user may capture a still image, enhance it, and then browse and navigate the image using technology described in U.S. Provisional Application 60/748,615 entitled "Camera Motion Estimation."

EXAMPLE 3

Business Card Reader

[0131] This description discloses processing techniques for enabling a resource constrained device (e.g., a mobile telephone equipped with a camera) to be used as a business card reader and contact management tool. For example, a smart-phone based business card reader enables a user to turn the user's camera-enabled mobile phone or PDA (personal Digital Assistant) into a powerful contact management tool (FIG. 18).

[0132] Smart phones are equipped with a robust business card reading capability. As a result, smart phones can be used to read business cards and manage contact information. This capability can be integrated with various devices through wireless connections. In one implementation, a user who receives a business card from a colleague at a conference may find it inconvenient to enter information through the small keypad in a mobile phone. The user captures an image of the business card with the user's mobile phone; text

reading software converts the physical card into tagged electronic contact info which can later be synchronized with the information in the user's smart phone or with the contacts in other devices and or applications including, for example, Pocket PC, Outlook, PalmOS, Lotus Notes, and GoldMine, through Bluetooth or other wireless or wired connections.

[0133] Field Analysis Using: Contextual Dictionaries

[0134] The OCR will use the technology we presented in previous claims. After OCR we use contextual dictionary to refine the OCR result. Words that appear in business cards can form a very restricted lexicon. Some examples are email, com, net, CEO, etc. The domain knowledge can be used to improve the recognition accuracy and conduct the field analysis.

[0135] Text extracted from a business card may include, for example, strings of digits, sequences of words, or a combination of strings of digits and sequences of words. A digital string may indicate, for example, a telephone number, a fax number, a zip code, or a street address. A sequence of words may include one or more keywords. A keyword in a sequence of words may identify a particular field to which a portion of the extracted text should be associated. For example, a key word "email" may indicate that a line of extracted text represents an email address. Similarly, a key word "President" may indicate that a line of extracted text represents a person's title.

[0136] Extracted text may be searched for digital strings or keywords. Recognition of particular digital strings or key words may be used to associate a portion (e.g., a line) of the extracted text with a particular field. In some instances, it may not be possible to identify one or more fields by digital string or keyword. In such cases, heuristics may be used to identify the one or more fields. For example, a person's name is often found in the same block as the person's title. Typically, a title field is easily identified using a keyword search. Therefore, the person's name may be identified after the person's title has been identified.

[0137] Users may be allowed to add words that they encounter in business cards and create their own user dictionaries.

EXAMPLE 4

Medication Reminder

[0138] Patients may find it difficult to remember when to take which medication, and in what quantities. These problems may be addressed by a non-intrusive, compact, inexpensive, lightweight and portable solution, which may integrate multiple medication reminder services.

[0139] The medication reminder may include software and a camera enabled smart phone to provide enhanced medication reminding and verification capabilities (FIG. 19). Computer vision technology enables camera-enabled handheld devices to read and see. A camera phone equipped with computer vision may be used as a personal barcode scanner that allows patients to enroll and verify medication-taking by a simple barcode scanning. The approach therefore may avoid manual entry which may be extremely challenging for some patients (e.g., older patients or patients with low vision). A camera-phone based medication reminder may include the following advantages:

[0140] The camera-phone based solution may provide alarms with audio, visual and vibration.

- [0141] Camera phones may be easily programmable, providing the flexibility to easily adapt to patients' specific needs
- [0142] Integrated networking capabilities allow remote monitoring and configuration options via existing protocols.
- [0143] The camera-phone based solution may be integrated with barcode reading capabilities to allow patients to enroll, remove and verify the medication with a simple barcode scan.
- [0144] Tracking may be set up so that users know when medication was taken, or adherence can be monitored at visits to the doctors office or remotely by family through network connections (note a continuous network connection is not required for this feature, only a periodic synchronization).
- [0145] Powerful image processing capabilities may be embedded into smart phones which may help improve medication adherence of patients especially if they have low vision and decreased memory. Image based barcode reading software uses cameras mounted in the smart phone to directly decode barcodes.
- [0146] Some medication barcodes may include Lot/control/batch number and expiration date to protect a patient from receiving a medication that is beyond its expiration date. 1 D Barcodes are symbols consisting of horizontal lines and spaces and are widely used on consumer goods. In retail settings, barcodes may be used to link the product to price and other inventory-related information. Medication barcodes may be designed for tracking medication errors associated with drug products. 2D barcodes and other symbols also may be used to provide information relevant to a patients medicine or retail products.
- [0147] A smart-phone based medication reminder may include smart-phone based barcode and symbol reading technology which may enroll and verify the medication simply through scanning.
- [0148] Enrolling Medications
- [0149] Consider this example. A 73-year-old man takes several different medications. He uses his smart phone preloaded with the medication reminder software as a medication reminding device. Some of these medications are packaged in rectangular boxes, and others in plastic cylindrical bottles. In order to enroll all of the medications into the device, he simply scans the barcode printed on the label either in a plane or in a cylinder. If he has difficulty in aiming the camera phone at the barcode, he merely moves or rotates the boxes or bottles around the camera until the smart phone beeps to indicate it detects and recognizes the barcode. He then sets the daily frequencies by pressing a numeric key (for example, number 2 to indicate to take it twice a day). This completes the enrollment process. The lot number and/or expiration date also may be decoded and saved into the system if the barcode includes them. FIG. 20(a) illustrates the enrollment process.
- [0150] Reminder Alarms
- [0151] Depending on the patient's needs and preferences, when it is time to take medication, the medication reminder phone may ring or present some other sound or even speech signal, vibrate, or both. If desired, this may be followed by a flashing screen, and then speech or text information to provide additional information such as the number of pills to take, as shown in FIG. 20(b).
- [0152] Verification
- [0153] After the reminder alarm (and perhaps the informational message), the patient may choose the medication container(s) needed. If he wants to verify that he has chosen the correct one, he may then scan the barcode, and the smart phone may compare the decoded National Drug Code (NDC) number with the one which was previously enrolled and is reminded to take, as shown in FIG. 20(c). Whether the container is correct or incorrect may be indicated by a sound, a speech message, and/or a text message.
- [0154] Self-monitoring
- [0155] Because adults of all ages often complain they can't recall whether or not they have taken their pills, a self-monitoring option may be included. For example, a graphic showing a day's pill schedule may be displayed. Once a pill is taken, the consumer may be able to indicate that on the graphic by pressing a key.
- [0156] Functional Overview of Technology
- [0157] The components of the medication reminder may be loosely partitioned into image acquisition, barcode detection, recognition, interface, alarm, Text-to-Speech (TTS), and the implementation of all of these on mobile devices. The system may be based on a dynamically reconfigurable component architecture so that it may be easily plugged into various mobile devices (cell phones, PDAs, etc.).
- [0158] System Architecture
- [0159] The component architecture manages a large number of resources on a small device. Physical storage for resources, memory for processing, and power consumption are all considerations. The system may operate in standalone mode providing an integrated capability. Dynamic management is also possible.
- [0160] The software modules include the user-interface; medication enrollment; a removal and verification module; and a barcode detection, enhancement and recognition module.
- [0161] As shown in FIG. 21, the system may include a set of basic components that may be managed by a core software control module. The core components may manage resources needed by the analysis modules and may swap them in from the Microdrive storage on demand. The component architecture may be implemented, for example, in Symbian OS or Microsoft Windows Mobile. The detection and enhancement components may be written in C or C++ first, and then transplanted into different embedded platforms.
- [0162] Software reusability and component management may be supported. The component architecture may provide an easy way to develop and test new algorithms, and it may provide a basis for moving to new devices, where resources may be even more limited.
- [0163] Interface for the Functionalities
- [0164] The interface may include, for example, the following functionalities:
- [0165] Drug information enrollment: The interface may allow users to enter a New Drug record. Users may type the information through the popup keyboard in the smart phone. However, many patients may not be able to do this. Therefore, a barcode reading capability may be provided which allows users to enroll the new drug through a simple scan.
- [0166] Select Frequency and Time of Doses: The interface may allow users to set the frequency of dose they want to take. For example, they may select from 1 to 6 doses per day, or select hour-based dosing alarm times. After selecting the

frequency, they may be able to adjust the alarm time by, for example, using the up and down arrows.

[0167] Setting supply reminders: It may be important to maintain an adequate supply of all of medications at all times. Missing doses of certain types of drugs may be very serious, even life threatening. The interface may allow the users to input the total supply and count the actual number of pills if some doses have been taken.

[0168] Deleting Drug Items: When patients want to delete an item, they may simply scan the barcode of that drug and select "delete" from the menu in the interface.

[0169] Verification: Verification may be accomplished simply through a barcode scanning.

[0170] Customization: The interface may be customizable in terms of functionality, so that users with different physical requirements can use the system effectively, allowing for different alarms, vibrations or visual displays.

[0171] Summary: The present invention provides robust algorithms for detection and rectification of barcodes on planes and generalized cylinders subject to perspective distortions and logging features allow adherence monitoring by the user, family or medical personnel. The fact that these devices have inherent connectivity (they are networked devices) allows for further remote setup, and monitoring. A cross-platform software architecture so the software based solution can be easily embedded into smart phones with different operating systems. Further, the systems may incorporate combined visual/audio/vibration outputs for alarms.

Scanning the Barcode

[0172] AMA has developed a technology called Mobile IBARS, which converts users' camera phones into personal data scanners. AMA is commercializing Mobile IBARS technology in the healthcare market with potential licensing deals with a commercial company providing nutrition/diet service for the customers. The principle of Mobile IBARS simulates the general optical barcode scanner by scanning the image line by line and decoding the barcode from the generated 0 and 1 sequence. The steps can be briefly described as:

[0173] Image Capture: In this process, the barcode images are captured using the interface customized for a variety of applications (FIG. 22(a)). The user simply starts the application and points the camera at the barcode.

[0174] Scanning: The software scans the image from a starting point and gets a waveform. FIG. 22(b) shows the waveform of a scanned line.

[0175] Thresholding: Binarization converts the waveform into a rectangular series of pulses (FIG. 22(c)).

[0176] Sequence Generation: After generating the rectangular waveform, each bar (or space) can be converted to the count of 1s or 0s by $n_i = w_i / w_b$, where w_i is the width of the i th bar (or space), and w_b is the module width. The module width can be directly estimated from guard bars.

[0177] Decoding: After generating a binary sequence, the decoding is straightforward. For example, in EAN-13 each character is encoded by seven digits, and consists of two bars and two spaces. The check digit can be used to further improve the identification result.

[0178] Verification: A barcode often contains a check digit to verify if a barcode is correctly decoded. For example, the last digit of EAN-13 is a check digit which satisfies

$$\left(\sum_{i=1}^{12} w_i c_i + c_{13} \right) \% 10 = 0,$$

where % denotes the mod operation, $c_1, c_2, \dots, c_{12}$ is the digit sequence, c_{13} is the check digit, $w_i = 1$ if $i \% 2 = 1$, and $w_i = 3$, if $i \% 2 = 0$. If the verification passes, then the decoded character sequence is correct; otherwise, the program scans the next line until the decoding and verification pass.

[0179] The foregoing description of the preferred embodiment of the invention has been presented for purposes of illustration and description. It is not intended to be exhaustive or to limit the invention to the precise form disclosed, and modifications and variations are possible in light of the above teachings or may be acquired from practice of the invention. The embodiment was chosen and described in order to explain the principles of the invention and its practical application to enable one skilled in the art to utilize the invention in various embodiments as are suited to the particular use contemplated. It is intended that the scope of the invention be defined by the claims appended hereto, and their equivalents. The entirety of each of the aforementioned documents is incorporated by reference herein.

What is claimed is:

1. A method for enhancing an image on a mobile device comprising the steps of:

pre-calculating a pixel value at each point on a grid and storing said pre-calculated pixel values in a lookup table;

using one bit to represent each pixel in said image;

quantizing said image at a small step interval such that each pixel in the image corresponds to one point on said grid; and

interpolating said image through a memory-indexing process.

2. A method for enhancing an image on a mobile device according to claim 1 further comprising the step of performing clustering based contrast enhancement on said image prior to said step of using one bit to represent each pixel in said image.

3. A method for enhancing an image on a mobile device, wherein coordinates of four corners (P_1, P_2, P_3 and P_4) of a bounding box in said image are known, top and bottom boundaries of said bounding box intersect at a vanishing point A and right and left boundaries of said bounding box intersect at a vanishing point B, comprising the steps of:

calculating a mapping between an ideal, non-perspective image and said image, wherein said calculated mapping comprises a plane-to-plane homograph matrix $H = (H_1, H_2, H_3)$, wherein said calculating a mapping comprises the steps of:

reshaping matrix H as a vector $h = (h_{11}, h_{12}, h_{13}, h_{22}, h_{23}, h_{24}, h_{31}, h_{32}, h_{33})^T$;

calculating H_3 according to the equation

$$\begin{cases} H_3 \cdot A = 0 \\ H_3 \cdot B = 0 \end{cases} \Rightarrow H_3 \sim A \times B, \text{ and}$$

$$H_3 \sim ((P_1 \times P_2) \times (P_2 \times P_3)) \times ((P_1 \times P_2) \times (P_3 \times P_4)),$$

calculating H according to the equation

$$H = \begin{pmatrix} h_{33} & 0 & 0 \\ 0 & h_{33} & 0 \\ h_{31} & h_{32} & h_{33} \end{pmatrix},$$

calculating H^{-1} according to the equation

$$H^{-1} \sim \begin{pmatrix} h_{33} & 0 & 0 \\ 0 & h_{33} & 0 \\ -h_{31} & -h_{32} & h_{33} \end{pmatrix},$$

mapping P_1, P_2, P_3 and P_4 to affine points P'_1, P'_2, P'_3, P'_4 using homography H; and

for any matrix entry (i, j) in a $w \times h$ matrix compute its affine coordinate

$$\frac{i}{w} P'_1 P'_4 + \frac{j}{h} P'_1 P'_2$$

and use H^{-1} to map this affine coordinate to the image coordinate.

4. A method for enhancing an image on a mobile device comprising the steps of:

for each pixel in said image, determining if binarization is required based upon an $N \times N$ neighborhood using a block-based approach;

if binarization is not necessary, for a particular neighborhood, set all pixels in said particular neighborhood to background;

for each pixel requiring binarization, calculating a binarization threshold using Niblack's approach and conducting binarization; and

post-processing said binary image to remove ghost objects.

5. A method for enhancing an image on a mobile device comprising the steps of:

representing each foreground pixel in said image with a pattern vector generated from pixel values in an $N \times N$ neighborhood of said foreground pixel; and

converting each foreground pixel to f^2 pixels in a higher resolution image where f is a magnification factor.

6. A method for enhancing an image on a mobile device according to claim 5, wherein how to convert a foreground pixel depends upon said pattern vector, $k-1$ other pattern vectors in said image that are similar to said pattern vector where similarity is measured by a Hamming distance of two pattern vectors, and pixels in said higher resolution image corresponding to said k pattern vectors.

* * * * *