

METHODS AND APPARATUS FOR COUNTING OBJECTS

FIELD OF THE INVENTION

The present invention relates generally to methods and apparatus for counting objects and more particularly to methods and apparatus for counting stacked flat objects.

BACKGROUND OF THE INVENTION

U.S. Reissue Patent 27,869 to Willits et al describes apparatus for counting stacked sheets having no sheet separation requirements. The active area of a sensor array is matched to the width of a sheet and the sensor array traverses the stack. The signal output of the sensor array is stripped of unwanted components in a high gain, diode clamped capacitive input operation amplifier whose square wave output is processed and counted by a counting circuit. U.S. Patent 5,005,192 to Duss describes a system for counting flat objects in a stream of partially overlapping objects which are conveyed past a locus of impingement of ultrasonic waves.

U.S. Patent 4,694,474 to Dorman et al describes a device for counting a stack of thin objects in which light is directed at the stack and a light sensor generates a signal proportional to the light reflected by the stack.

U.S. Patent 5,040,196 to Woodward describes an instrument for counting stacked elements which images a portion of the side of the stack and then autocorrelates the image, while the instrument is stationary, and then cross-correlates the image as the instrument is moved. The result is a time varying signal whose repeating cycles, when counted, indicate the number of elements in

the stack.

U.S. Patent 3,971,918 to Saito counts stacked corrugated cardboards by scanning an end of the stack horizontally and vertically, using an array of photodiodes switched in turn by electric pulses. The outputs of the photodiodes are counted and compared to successively detect flat and corrugated sheets.

U.S. Patent 4,912,317 to Mohan et al describes apparatus for counting stacked sheets whose apparent brightness is not uniform. The Mohan et al system normalizes the phase polarity of the sensor signal differential output, thereby avoiding the effects of brightness polarity reversals in the sensor output data. Mohan et al employs sensors whose effective imaged width on the stacked objects is very narrow relative to the individual objects. The data is differentially summed, then rectified to normalize phase polarity.

None of the above U.S. Patents teaches that the devices described therein are suitable for counting banknotes.

U.S. Patent 5,324,921 describes a conventional sheet counting machine in which a photosensor is disposed across a bill passage downstream of a pulley. Emitted light is interrupted by each bill passing through the light path and therefore the number of bills can be counted by counting the number of intervals during which light is not received by the light receiver.

A general text on image processing is Pratt, W. K, Digital image processing, Second Ed., Wiley 1991, New York.

The disclosures of all of the above publications and of the references cited therein are hereby incorporated by reference.

Brandt, Inc. of Bensalem, PA 19020, USA, markets a Model 8640D Note Counter accommodating notes of at least a minimum note size and thickness and no more than

a maximum note size and thickness. The 8640D leafs through the banknotes in order to determine the number of banknotes.

SUMMARY OF THE INVENTION

The present invention seeks to provide an improved method and apparatus for rapidly, accurately and inexpensively counting stacked objects, preferably by imaging, from below, a stack of flat objects which is standing on its side, preferably on its long side. The objects need not be identical in surface appearance or in configuration. The objects preferably may be of substantially any size or thickness and need not be less than some maximum size or within some narrow range of thicknesses.

Preferably, the objects are not leafed through or otherwise moved while being imaged, in contrast to conventional devices for counting banknotes and documents such as the counting device described in U.S.P. 5,324,921 or the Brandt Note Counter.

This feature allows a loose or fastened together stack of objects, such as a stapled-together stack of papers, a rubber-banded stack of bills, or the pages of a bound volume, to be counted without being dismantled.

A stack preferably includes a plurality of objects which are generally pairwise adjacent, although not necessarily touching, wherein the edges of pairwise adjacent objects in the stack are at least roughly aligned. One example of a stack is a vertical stack which preferably includes a plurality of objects which are stacked one on top of another. Another example of a stack is a horizontal stack which preferably includes a plurality of objects standing one next to the other. Stacked flat objects may be disposed perpendicular to the ground or at any other orientation relative to the ground and may or may not be parallel to one another.

Preferably, the stacked objects are imaged by a matrix-CCD, and neither the CCD nor the stack of objects

is moved during imaging. An advantage of this embodiment is that the counting apparatus may have no moving parts and therefore may be simple to manufacture, operate and maintain.

Alternatively, the stack may be manually or automatically caused to slide over the field of view of the optical sensor which images the stack or a moving line-CCD may replace the matrix-CCD. The motion may be provided specifically to facilitate counting or alternatively, objects in motion may be counted, utilizing the existing path of motion of the objects.

Optionally, a laser emitting device such as a laser diode or a He-Ne laser may provide light and an optical sensor suitable for sensing laser rays may be employed. The laser beam may travel along the side of the stack or alternatively, the stack may be slid manually or automatically relative to the stationary laser beam so as to enable the laser beam to scan a portion of each edge of each object and/or of each gap between each two adjacent objects. The reflected or transmitted beam is then processed in order to discern the number of objects in the stack.

In the present specification and claims, the surface area of a flat object is regarded as including two "surfaces" and at least one "edge", where each edge is a nearly one-dimensional face of the object. If the object is rectangular, it has two surfaces and four edges. For example, a piece of paper has front and back surfaces and four edges.

The "edge" of an object within a stack is used herein to refer to a face of the stacked object which is parallel to the axis of the stack.

More generally, the term "edge" is employed herein to refer to a portion of an object which is imaged in order to count the number of objects.

The term "side of a stack", pertaining to a

stack of flat objects, refers to one of the four faces of the stack which are formed of the edges of the stacked objects and not to the remaining two faces of the stack which are formed of a surface of the first object in the stack and a surface of the last object in the stack, respectively.

It is believed that the present invention is applicable to counting of flat round or curved objects. In this case, the "side of the stack" refers to a face of the stack which is formed of the edges of the stacked round objects.

According to a preferred embodiment of the present invention, counting is effected by imaging a side of the stack. In the resulting images, particularly if the objects are sheets of paper, the sheet edges are seen to be non-uniform, due to material wear, bent sheets, torn sheets, folded sheets and the tendency of paper to adopt a wave-like configuration.

There is thus provided in accordance with a preferred embodiment of the present invention a method for counting banknotes including providing a stack of banknotes and estimating the number of banknotes in the stack wherein the estimation process is characterized in that the mutual orientation of the banknotes is substantially maintained.

Also provided is apparatus for counting stacked objects including at least one optical sensor for simultaneously viewing a plurality of locations along a side of a stack of objects, the locations being arranged along the edges of the objects which form the side of the stack and image processing apparatus receiving an output from the optical sensor and providing an output indication of a number of objects in the stack. Further in accordance with a preferred embodiment of the present invention, the optical sensor includes a plurality of sensing elements respectively viewing the plurality of

locations along the side of the stack.

Still further in accordance with a preferred embodiment of the present invention, the optical sensor has a two-dimensional field of view.

Further in accordance with one preferred embodiment of the present invention, apparatus is provided for varying the position of the stack relative to the optical sensor.

Still further in accordance with one preferred embodiment of the present invention, the apparatus for varying includes apparatus for moving the stack.

Additionally in accordance with one preferred embodiment of the present invention, the apparatus for varying includes apparatus for moving the optical sensor relative to the stack.

Further in accordance with one preferred embodiment of the present invention, the optical sensor is operative to repeatedly view at least one location along the stack of objects.

Also provided, in accordance with one preferred embodiment of the present invention, is a method for counting stacked objects including viewing at least a portion of a side of a stack of objects at least under first illumination conditions and under second illumination conditions, and image processing apparatus receiving an output from the optical sensor including a first image of at least a portion of the stack under the first illumination conditions and a second image of at least a portion of the stack under the second illumination conditions, and operative to compare the two images and to provide an output indication of a number of objects in the stack.

Additionally provided, in accordance with a preferred embodiment of the present invention, is apparatus for counting stacked objects including at least one support for at least one stack of objects, at least one

optical sensor disposed behind the at least one support for viewing at least a portion of a side of a stack of objects through the support, and image processing apparatus receiving an output from the optical sensor and providing an output indication of a number of objects in the stack.

Further in accordance with a preferred embodiment of the present invention, the support is transparent.

Still further in accordance with a preferred embodiment of the present invention, the support has at least one window formed therein.

Additionally in accordance with a preferred embodiment of the present invention, there is provided a method for counting banknotes including imaging a stack of banknotes from the side, and image-processing the resulting image in order to compute the number of banknotes in the stack.

Further in accordance with a preferred embodiment of the present invention, the apparatus also includes an object separator operative to separate objects in the stack from one another to facilitate counting thereof.

Further in accordance with a preferred embodiment of the present invention, the method also includes separating the banknotes in the stack from one another to facilitate counting thereof.

Additionally in accordance with a preferred embodiment of the present invention, the at least one optical sensor includes a plurality of optical sensors each of which is operative to view a plurality of locations along a side of a different stack.

Further in accordance with a preferred embodiment of the present invention, the at least one optical sensor includes a plurality of optical sensors each of which is operative to view at least a portion of a side

of a different stack of objects.

Still further in accordance with a preferred embodiment of the present invention, a plurality of light sources illuminates the stacked objects.

Further in accordance with a preferred embodiment of the present invention, the first illumination conditions include ambient illumination.

BRIEF DESCRIPTION OF THE DRAWINGS AND APPENDICES

The present invention will be understood and appreciated from the following detailed description, taken in conjunction with the drawings in which:

Fig. 1 is a simplified block diagram of sheet counting apparatus constructed and operative in accordance with a preferred embodiment of the present invention;

Fig. 2 is an example of a negative image of stacked sheet portions;

Fig. 3 is a logic diagram of the operation of the image processing and counting computer of Fig. 1;

Fig. 4 is a flowchart illustration of a method for implementing the image processing step of Fig. 3 based on selection of an appropriate sequence of image processing operations;

Fig. 5 is a flowchart illustration of a preferred method for implementing the sheet counting step of Fig. 3; and

Fig. 6 is a simplified block diagram of a modification of the sheet counting apparatus of Fig. 1 which is operative to count a plurality of stacks of objects.

Attached herewith are the following appendices which aid in the understanding and appreciation of one preferred embodiment of the invention shown and described herein:

Appendix A is a computer listing of a program entitled EZ_MONEY.PAS, a program which implements a banknote counting method operative in accordance with a preferred embodiment of the present invention; and Appendix B is a description of the code MODEX.ASM which resides in the computer program MODEX.ASM. MODEX.ASM is a public domain software package available from Matt Pritchard, P.O.Box 140264, Irving, TX 75014, USA.

DETAILED DESCRIPTION OF PREFERRED EMBODIMENTS

A portion of the disclosure of this patent document contains material which is subject to copyright protection. The copyright owner has no objection to the facsimile reproduction by anyone of the patent document or the patent disclosure, as it appears in the Patent and Trademark Office patent file or records, but otherwise reserves all copyright rights whatsoever.

Fig. 1 is a simplified block diagram of apparatus for counting stacked objects. The apparatus includes a support 10 for the stack of objects 20 to be counted, at least one light source 30, and a light sensor 40, such as a matrix-CCD or a line-CCD, operatively associated with a lens 50, for converting the image of the stack into electric signals. The optical apparatus may, optionally, include mirrors (not shown) for such functions as enlargement, focussing and/or changing direction.

The axis of the stack is indicated by reference number 54.

Alternatively, the support 10 may be omitted. The apparatus may optionally be portable such that counting of objects takes place by transporting the counting apparatus to the objects rather than by transporting the objects to the counting apparatus.

It is appreciated, however, that the support, if provided, may perform one or more of the following functions:

- a. Alignment of the stack.
- b. Separation of the stack, e.g. by providing a diagonally oriented support on which the stack is placed on its side such that the edges of the stack become separated due to the diagonal.
- c. The support may serve as a track along which the stack is moved.

d. The support may be operative to electrostatically charge the stack, thereby to enhance separation of the objects. For example, the support may comprise a capacitor.

Depending on the optical characteristics of the lens and the CCD elements, magnification may be provided, so as to provide a suitable picture resolution, such as at least 5 pixels for the shortest dimension of the object and for the average gap between objects. One suitable depth of field value is about 5 mm. A suitable linear resolution is at least 500 dots per half-inch. The above numerical values are suitable for the specific equipment detailed below and are not intending to be limiting.

It is appreciated that a laser beam emitting device such as a laser diode or a He-Ne laser may be employed for light source 30 and an optical sensor suitable for sensing laser rays may be employed for sensor 40.

Preferably, the sensor and lens are disposed below the support 10 and the support 10 includes a transparent window 60 or a slit (not shown) through which the stack 20 can be imaged from below. The stack is placed on its side, preferably on its long side, and may optionally be manually guided along the long dimension of the transparent window 60, as indicated by arrow 100. In some applications, motion along arrow 100 may not require manual guidance since the stack is in motion, e.g. is traveling along a conveyor belt, due to processes other than counting which are being performed on the stack or with the aid of the stack.

Alternatively, the CCD comprises a line-CCD which can be moved parallel, or at any other suitable angle, to the long dimension of the transparent window. Preferably, however, the CCD comprises a matrix-CCD and neither the stack nor the matrix-CCD are moved during

imaging.

The output of the sensor is fed to an image capturing unit 80 which transforms the analog data captured by the light sensor 40 in digital form to a RAM 68. An image processing and counting computer 70, associated with a conventional control device 84, analyzes the picture stored in the RAM in order to discern or "count" the number of objects in the stack. The counting capability may be implemented in software which is held in a ROM 94.

The result of "counting" the number of objects in the stack is displayed on a display device 90 such as an LCD. Optionally, diagnostic statistics or warning indications may also be displayed.

It is appreciated that information related to the counting process other than the number of objects may be derived and displayed. For example, it may be desirable to provide an indication of poor quality objects, such as bills.

In Fig. 1, illumination is provided, however, alternatively, only natural illumination may be employed. Furthermore, any suitable type of artificial illumination may be employed. Optionally, if artificial illumination is employed, the natural illumination is blocked out as by opaque blocking screens.

One or more light sources may be employed. Each of the one or more beams provided by the one or more light sources may be any color of light, or may have a selectable plurality of colors as by provision of a plurality of filters. Each beam may be focussed or divergent. The angle of each beam relative to the stack may be any fixed angle or may be varied by the user. The light itself may be coherent or non-coherent. Filters may be employed to control the wavelength of the light and/or the polarization of the light.

Optionally, the objects in the stack are

processed so as to minimize the probability that two objects overlies one another and are consequently perceived as being a single object. For example, a plurality of apertures may be provided in the window 60 through which airflows or air jets access the objects in order to enhance the separation thereof. Alternatively or in addition, the objects may be electrostatically charged such that they tend to repel one another and become separated from one another. Alternatively or in addition, a mechanical device may be provided to grip one side of the stack, typically the side opposite the side which is to be imaged, which has the effect of separating the edges of the objects which lie along the side of the stack which is to be imaged.

It is appreciated that the above two examples of how to minimize the probability of overlying objects are only examples and are not intended to be limiting.

Fig. 2 is an example of a negative image of stacked sheet portions.

As seen, the sheet edges are non-uniform, which may be due to material wear, bent sheets, torn sheets, folded sheets, the tendency of paper to adopt a wave-like configuration, and other factors. Therefore, different lines drawn perpendicular to the imaged edges create different sequences of intersection points with the images of the sheets. The sequences may differ as to the distances between corresponding intersection points and/or even as to the number of intersection points. For example, the bottom two intersection points on line A in Fig. 2 would probably correspond to a single intersection point on line B due to the lack of distance between the bottom two sheets in Fig. 2, at the location of line B.

For this reason, according to a preferred embodiment of the present invention, a two dimensional image of the stack is provided, or alternatively the stack is imaged with a linear sensor at a plurality of

locations along the sheets, such as more than 400 locations. For example, the stack of Fig. 2 may be imaged at a plurality of locations including line A and line B.

Fig. 3 is a logic diagram of the operation of the comparing and counting computer of Fig. 1, which includes image processing and counting.

Image processing typically includes noise removal, sharpening, edge enhancement, filtering, and/or threshold limiting, any or all of which may be based on conventional methods such as those described in Pratt, W. K, Digital image processing, Second Ed., Wiley 1991, New York. A preferred image processing method is described below with reference to Fig. 4.

A preferred counting method is described below with reference to Fig. 5.

Fig. 4 is a flowchart illustration of a method for implementing the image processing step of Fig. 3 based on selection of an appropriate sequence of image processing operations from among a set of image processing "primitives". The set of image processing "primitives" illustrated in Fig. 4 includes:

- a. a negative imaging operation N,
- b. a differential operation D along columns to emphasize changes between bills and background,
- c. a static cut-off operation C which reduces noise using a threshold value set according to image brightness and contrast,
- d. a dynamic cut-off operation X to reduce noise along rows (banknotes),
- e. a dynamic cut-off operation Y to reduce noise between rows (banknotes),
- f. a binarization operation B,
- g. a smoothing operation S to reduce high-frequency noise,
- h. a sharpening edge-enhancing operation P,
- i. a hi-pass filtering operation H,

- j. a thick line detecting filtering operation I for emphasizing banknote images; and
- k. a line-detecting filtering operation L.

Suitable sequences of these image processing operations include: SSCDBS, SCPS, SIY, SIX, or simply C.

It is appreciated that a suitable image processing sequence need not be composed only of operations S, C, D, B, P, I, Y. A suitable image processing sequence may include other conventional image processing operations and/or the remaining image processing operations referred to in Appendix A and in Fig. 4, namely H (high pass filter), L (line detection filter), B (image binarization), N (negating of image).

Fig. 5 is a flowchart illustration of a preferred method for implementing the sheet counting step of Fig. 3. Each column is searched for sequences of non-zero pixels. The number of such sequences is termed "bills" in Fig. 5. A histogram is constructed for "bills". The output of the process is an indication of the central tendency of the histogram such as the modal value (peak) thereof and/or the mean value thereof.

Fig. 6 is a simplified block diagram of a modification of the sheet counting apparatus of Fig. 1 which is operative to count a plurality of stacks of objects, even simultaneously. As shown, the apparatus of Fig. 6 is similar to the apparatus of Fig. 1 except that image processing and counting computer 70, image capturing unit 80 and control unit 84 are associated with a plurality of stack inspecting subunits 110, only two of which are illustrated. Each stack inspecting subunit typically comprises a support 10, a light source 30, a light sensor 40, a lens 50, and a display device 90.

Appendix A is a computer listing of a program entitled EZ_MONEY.PAS, a program which implements a banknote counting method operative in accordance with a preferred embodiment of the present invention. The

program employs several image processing methods to count banknotes in a picture file.

The picture file is an image which may be captured using a CORTEX frame grabber. The frame resolution is 512 x 512 pixels x 256 gray levels/pixel. The program uses MODEX, a public domain software package written by Matt Pritchard, P.O.Box 140264, Irving, TX 75014, USA. A description of the program MODEX, entitled MODEX.ASM, is appended hereto and is referenced Appendix B. MODEX is employed as a graphics package, in order to process and display a 256 gray level picture, since this ability is not supported by the Turbo Pascal 6.0 Graphics Unit.

The program uses a subset of the MODEX graphics routines to handle two VGA pages, one being the source of the image processing operation and the other being the destination thereof. The program sets and gets pixel values and prints text.

The program uses the MODEX screen resolution, 320H x 400V, which is smaller than the CORTEX image resolution but is sufficient in order to display the essential part of the image which stores the image of the banknotes to be counted.

To use the program of Appendix A to count a stack of banknotes, such as a stack of approximately one dozen Bank of Israel 20 New Shekel denomination notes, the following equipment may be employed:

Hardware:

Computer - PC 386DX (40Mhz, 128K Cache, 4MB RAM, 340MB hard disk, SVGA monitor).

Graphics card - Trident 8900CL (SVGA), 1MB RAM on-board (manufactured by JUKO Electronics Industrial Co. Ltd. 208-770000-00A, Taiwan).

Frame grabber card - CORTEX-I, 256 Gray levels, 512H x 512V resolution in CCIR/PAL mode (manu-

factured by Imagenation Corp., P.O. BOX 84568, Vancouver WA 98684-0568, USA).

Video camera - JAVELIN JE-7442 Hi-Resolution 2/3" CCD camera (manufactured by JAVELIN Electronics, 19831 Magellan Dr., Torrance CA 90502-1188, USA).

Lens - Micro-Nikkor 55mm Macro lens (manufactured by NIKON Corp., Fuji Bldg., 2-3, Marunouchi 3-chome, Chiyoda-ku, Tokyo 100, JAPAN).

Camera accessories - Cosmincar x2 C-Mount lens TV Extender, Video Camera tripod.

Software:

MS-DOS 6.2 (by MicroSoft Corp.). Turbo Assembler 3.0 (by Borland International, Inc.)

Turbo Pascal 6.0 (by Borland International, Inc).

CORTEX frame grabber software (by Imagenation Corp).

MODEX SVGA graphics library (author: Matt Pritchard, P. O. B. 140264, Irving, TX 75014-0264, USA; on Fido NET ECHO Conference: 80xxx), a description of which is provided herein as Appendix B;

EZ_Money - TurboPascal version counting program whose listing is appended hereto as Appendix A.

Bills-counting processes, the text files of which are set forth within the above description under the captions COUNT_1.OPR, COUNT_5.OPR.

A preferred method for counting notes, using the above equipment, is as follows:

1. Install the CORTEX frame grabber card inside the computer.
2. Install CORTEX software in C:\BANKNOTE directory.
3. Generate a digital file whose contents are identical to the computer listing of Appendix A and name this file EZ_MONEY.PAS. Additionally, generate a digital file of the computer program MODEX.ASM which is described in Appendix B. The code MODEX.ASM is available from Matt Pritchard. P.O. Box 140264, Irving, TX 75014, USA. Put

EZ_MONEY.PAS and MODEX.ASM into C:\BANKNOTE directory.

4. Compile MODEX.ASM using Turbo Assembler 3.0 in order to create MODEX.OBJ.

5. Compile EZ_MONEY.PAS and link it to MODEX.OBJ using Turbo Pascal 6.0.

6. Mount the Micro Nikkor lens onto the Javelin camera with the Cosmimar TV Extender.

7. Attach the Javelin camera to the tripod and connect the camera video output to the CORTEX card input.

8. Place the stack of banknotes such that the stack's side (the edges of the bills) is in the viewing field of the camera.

9. Focus the lens on the bills' edges: change aperture opening to match the environment luminance which may, for example, be ambient room light.

10. Run CORTEX utility program to grab the banknotes image to a CORTEX image file format, using the command C:\BANKNOTE> UTILITY\GRAB.COM BANKNOTE.PIC.

11. Run EZ_MONEY counting program on the default BANKNOTE.PIC image file by:

a. Interactive running (i.e. C:\BANKNOTE) EZ_MONEY); or

b. Running using any one of the counting processes, COUNT_1.OPR to COUNT_5.OPR, which are as follows:

COUNT_1.OPR:

BANKNOTE.PIC

SSCDBS#

COUNT_2.OPR:

BANKNOTE.PIC

SCPS#

COUNT_3.OPR:

BANKNOTE.PIC

SIY#COUNT_4.OPR:

BANKNOTE.PIC

SIX#

COUNT_5.OPR:

BANKNOTE.PIC

C#

For example, to run the EZ_MONEY counting program using the first counting process, type:

C:\BANKNOTE> EZ_MONEY COUNT_1.OPR. The five
counting processes listed above are sequences including
one or more image processing operations, referred to in
Appendix A and in Figs. 3 and 4 as S, I, X, Y, C, P and
D, and also including a counting process # which is
operative to count banknotes in each column and give, as
a result, the most frequent count.

It is appreciated that the above image processing operations can be combined into counting processes other than COUNT_1.OPR, ..., COUNT_5.OPR. It is also appreciated that the above set of image processing combinations may be augmented by other conventional image processing operations such as but not limited to the following image processing operations which are referred to in Appendix A and in Fig. 4:

H (high pass filter), L (line detection filter), B (image binarization), N (negating of image).

Preferably, at least one of the image processing operations employed operates on a multipixel area such as a 3 x 3 pixel matrix or a 3 x 5 pixel matrix, rather than operating on one pixel at a time.

Optionally, a neural network or other learning mechanism may be employed such that the counting apparatus shown and described herein may be trained to count correctly.

Alternatively, all five of the counting processes may be employed and the results thereof combined, as by a weighted average, to determine a final result.

The number of banknotes in the stack is displayed on the screen or is recorded on the counting-algorithm file, if supplied. The result is the 'peak' value; in addition, the 'average' value is written.

For example, when the negative of the banknote stack image of Fig. 2 was processed, the result was found to be 12.

The present invention is described herein in the context of a banknote counting application as for a cash register, automatic cash withdrawal device or other banknote handling device, in a bank, postal facility, supermarket, casino, transportation facility or household use. However, it is appreciated that the embodiments shown and described herein may also be useful for counting other objects, and particularly flat, stacked objects such as stacks of cardboard sheets, forms, bills, films, plates, metal foils, cards, and pages photocopied or to be photocopied by a photocopier. The counting device may, optionally, be portable and may be either battery-powered or powered by connection to an electric outlet.

It is appreciated that the software components of the present invention may, if desired, be implemented in ROM (read-only memory) form. The software components may, generally, be implemented in hardware, if desired, using conventional techniques.

It is appreciated that the particular embodiment described in the Appendices is intended only to provide an extremely detailed disclosure of the present invention and is not intended to be limiting. It is appreciated that various features of the invention which are, for clarity, described in the contexts of separate embodiments may also be provided in combination in a single embodiment. Conversely, various features of the invention which are, for brevity, described in the context of a single embodiment may also be provided separately or in any suitable subcombination.

It will be appreciated by persons skilled in the art that the present invention is not limited to what has been particularly shown and described hereinabove. Rather, the scope of the present invention is defined only by the claims that follow:

WO 96/22553

PCT/US96/00658

23

APPENDIX A

{ COPYRIGHT © 1994, by: Charlie S. Antebi & Lior Goldenberg }

Program EZ_Money(input,output);
Uses Crt;

{\$L modex.obj} { This file is the external ModeX Library .OBJ }
{\$F+}

{ Mode Setting Routines }

Function SET_MODEX (Mode:integer) : Integer; external;

{ Graphics Primitives }

Procedure CLEAR_VGA_SCREEN (Color:integer); external;
Procedure SET_POINT (Xpos,Ypos,Color : integer); external;
Function READ_POINT (Xpos,Ypos:integer) : integer; external;
Procedure DRAW_LINE (Xpos1,Ypos1,Xpos2,Ypos2,Color:integer); external;

{ VGA DAC Routines }

Procedure SET_DAC_REGISTER (RegNo,Red,Green,Blue:integer); external;

{ Text Display Routines }

Procedure PRINT_STR (Var Text;MaxLen,Xpos,Ypos,ColorF,ColorB:integer);
external;

{ Page and Window Control Routines }

Procedure SET_ACTIVE_PAGE (PageNo:integer); external;
Procedure SET_DISPLAY_PAGE (PageNo:integer); external;

{ Sprite and VGA memory -> Vga memory Copy Routines }

Procedure COPY_PAGE (SourcePage,DestPage:integer); external;

{\$F-}

Const

```
CR = Chr(13);
ESC = Chr(27);
FRAME_Y = 512;
FRAME_X = 512;
FILTER_SIZE = 20;
COMMAND_LENGTH = 20;
MAX_BILLS = 256;
XMAX = 320;
YMAX = 400;
DISPLAY_MODE = 1; { 320H x 400V }
DATA_FRAME = 0;
WORK_FRAME = 1;
```

Type

```
Filter_Matrix= Array[0..FILTER_SIZE-1,0..FILTER_SIZE-1] of Integer;
```

Var

```
command_string: String[COMMAND_LENGTH];
command_index: Integer;
peak: Integer;
```

```
{ Error Handler - Returns to Text Mode & Displays Error }
```

Procedure MESSAGE(s : string);

Begin

asm

mov ah,0

mov al,3

int 10h

end;

WriteLn(s);

Halt(0);

END;

```
{ MAIN ROUTINE - Run Through Counting and Exit }
```

Procedure Beep;

Var

i: Integer;

Begin

```
Sound(1000);  
For i:=0 to 16000 Do;  
  NoSound;  
End;
```

```
Procedure Gray_Scale;  
Var  
  i: Integer;  
Begin  
  For i:=0 to 255 do  
    SET_DAC_REGISTER (i,i div 4,i div 4,i div 4);  
  End;
```

```
Procedure Display_Frame(filename: String; skip_lines: Integer);  
Var  
  frame_file: Text;  
  x,y: Integer;  
  c: Char;  
  
Begin  
  Assign(frame_file,filename);  
  Reset(frame_file);  
  
  For y:=0 to FRAME_Y-1 do  
    For x:=0 to FRAME_X-1 do  
      Begin  
        Read(frame_file,c);  
        If y>=skip_lines Then  
          SET_POINT(x,y-skip_lines,Ord(c));  
        End;  
      End;  
    End;  
  
  Close(frame_file);  
End;
```

```
Procedure Negate_Frame;  
Var  
  x,y,n: Integer;  
Begin  
  SET_DISPLAY_PAGE(WORK_FRAME);  
  For y:=0 to YMAX-1 do
```

```
For x:=0 to XMAX-1 do
Begin
  n:=255-READ_POINT(x,y);
  SET_ACTIVE_PAGE(WORK_FRAME);
  SET_POINT(x,y,n);
  SET_ACTIVE_PAGE(DATA_FRAME);
End;
End;

Procedure Cutoff_Frame(n: Integer);
Var
  x,y,v: Integer;
Begin
  SET_DISPLAY_PAGE(WORK_FRAME);
  For y:=0 to YMAX-1 do
    For x:=0 to XMAX-1 do
      Begin
        v:=READ_POINT(x,y);
        If v<=n Then
          v:=0;
          SET_ACTIVE_PAGE(WORK_FRAME);
          SET_POINT(x,y,v);
          SET_ACTIVE_PAGE(DATA_FRAME);
        End;
      End;
    End;
  End;

Procedure Dynamic_Y_Cutoff_Frame(r: Real);
Var
  x,y,v,n: Integer;
Begin
  SET_DISPLAY_PAGE(WORK_FRAME);
  For y:=0 to YMAX-1 do
    Begin
      v:=0;
      For x:=0 to XMAX-1 do
        If v<READ_POINT(x,y) Then
          v:=READ_POINT(x,y);
      n:=Round(v*r);
      For x:=0 to XMAX-1 do
        Begin
```

```
v:=READ_POINT(x,y);
If v<=n Then
  v:=0;
  SET_ACTIVE_PAGE(WORK_FRAME);
  SET_POINT(x,y,v);
  SET_ACTIVE_PAGE(DATA_FRAME);
End;
End;
End;
```

```
Procedure Dynamic_X_Cutoff_Frame(r: Real);
Var
  x,y,v,n: Integer;
Begin
  SET_DISPLAY_PAGE(WORK_FRAME);
  For x:=0 to XMAX-1 do
    Begin
      v:=0;
      For y:=0 to YMAX-1 do
        If v<READ_POINT(x,y) Then
          v:=READ_POINT(x,y);
      n:=Round(v*r);
      For y:=0 to YMAX-1 do
        Begin
          v:=READ_POINT(x,y);
          If v<=n Then
            v:=0;
            SET_ACTIVE_PAGE(WORK_FRAME);
            SET_POINT(x,y,v);
            SET_ACTIVE_PAGE(DATA_FRAME);
          End;
        End;
      End;
    End;
```

```
Procedure Bin_Frame(n: Integer);
Var
  x,y,v: Integer;
Begin
  SET_DISPLAY_PAGE(WORK_FRAME);
  For y:=0 to YMAX-1 do
```

```
For x:=0 to XMAX-1 do
Begin
  v:=READ_POINT(x,y);
  If v<=n Then
    v:=0
  Else
    v:=255;
  SET_ACTIVE_PAGE(WORK_FRAME);
  SET_POINT(x,y,v);
  SET_ACTIVE_PAGE(DATA_FRAME);
End;
End;

Procedure Diff_Frame;
Var
  x,y,n: Integer;
Begin
  SET_DISPLAY_PAGE(WORK_FRAME);
  For y:=0 to YMAX-1 do
    For x:=0 to XMAX-1 do
      Begin
        n:=(READ_POINT(x,y+1)-READ_POINT(x,y-1)+255) div 2;
        SET_ACTIVE_PAGE(WORK_FRAME);
        SET_POINT(x,y,n);
        SET_ACTIVE_PAGE(DATA_FRAME);
      End;
    End;
  End;

Function Byte_Bound(v: Integer): Byte;
Begin
  Byte_Bound:=v;
  If v<0 Then
    Byte_Bound:=0;
  If v>255 Then
    Byte_Bound:=255;
End;

Procedure Filter_Frame(devider: Integer; f: Filter_Matrix; m,n: Integer);
Var
  x,y,i,j: Integer;
```

```

v: Integer;
Begin
  SET_DISPLAY_PAGE(WORK_FRAME);
  For y:=0 to YMAX-m do
    For x:=0 to XMAX-n do
      Begin
        v:=0;
        For i:=0 to m-1 do
          For j:=0 to n-1 do
            v:=v+f[i,j] * READ_POINT(x+j,y+i);
          v:=Byte_Bound(v div divider);
          SET_ACTIVE_PAGE(WORK_FRAME);
          SET_POINT(x+(n div 2),y+(m div 2),v);
          SET_ACTIVE_PAGE(DATA_FRAME);
        End
      End;
End;

```

```

Procedure Smooth_Frame;
Var
  f: Filter_Matrix;
  i,j: Integer;
Begin
  For i:=0 to FILTER_SIZE-1 do
    For j:=0 to FILTER_SIZE-1 do
      f[i,j]:=1;
    Filter_Frame(9,f,3,3);
  End;

```

```

Procedure Sharp_Frame;
Var
  f: Filter_Matrix;
  i,j: Integer;
Begin
  f[0,0]:= 1; f[0,1]:= 1; f[0,2]:= 1;
  f[1,0]:= 1; f[1,1]:=-2; f[1,2]:= 1;
  f[2,0]:=-1; f[2,1]:=-1; f[2,2]:=-1;
  Filter_Frame(1,f,3,3);
End;

```

```

Procedure Line_Detection_Frame;

```

```
Var
  f: Filter_Matrix;
  i,j: Integer;
Begin
  f[0,0]:=-1; f[0,1]:=-1; f[0,2]:=-1;
  f[1,0]:= 2; f[1,1]:= 2; f[1,2]:= 2;
  f[2,0]:=-1; f[2,1]:=-1; f[2,2]:=-1;
  Filter_Frame(1,f,3,3);
End;

Procedure Hi_Pass_Frame;
Var
  f: Filter_Matrix;
  i,j: Integer;
Begin
  f[0,0]:= 0; f[0,1]:=-1; f[0,2]:= 0;
  f[1,0]:=-1; f[1,1]:= 5; f[1,2]:=-1;
  f[2,0]:= 0; f[2,1]:=-1; f[2,2]:= 0;
  Filter_Frame(1,f,3,3);
End;

Procedure Bill_Detection_Frame;
Var
  f: Filter_Matrix;
  i,j: Integer;
Begin
  f[0,0]:=-2; f[0,1]:=-3; f[0,2]:=-2;
  f[1,0]:=-1; f[1,1]:=-1; f[1,2]:=-1;
  f[2,0]:=10; f[2,1]:=15; f[2,2]:=10;
  f[3,0]:=-1; f[3,1]:=-1; f[3,2]:=-1;
  f[4,0]:=-2; f[4,1]:=-3; f[4,2]:=-2;
  Filter_Frame(10,f,5,3);
End;

Function Select_Process: Char;
Var
  c: Char;
  menu_line: Packed Array [1..40] of Char;
Begin
  Copy_Page(DATA_FRAME,WORK_FRAME);
```

```

SET_ACTIVE_PAGE(WORK_FRAME);
SET_DISPLAY_PAGE(WORK_FRAME);
if command_string=" Then
Begin
    menu_line:='Neg Diff Cut-off X-cut Y-cut Bin Hi-pass';
    Print_Str(menu_line,40,0,0,255,0);
    menu_line:='Smooth sharP Line-det cOl-det bIll-det #';
    Print_Str(menu_line,40,0,8,255,0);
    c:=ReadKey;
End
Else
Begin
    command_index:=command_index+1;
    c:=command_string[command_index];
    If (c='#') Or (c=' ') Or (c='"') Or (c=CR) Then
        c:=ESC;
    End;
    If c='#' Then
        c:=ESC;
    Select_Process:=c;
    SET_ACTIVE_PAGE(DATA_FRAME);
    SET_DISPLAY_PAGE(DATA_FRAME);
End;

```

```

Procedure Process_Frame(s: Char);

```

```

Begin
    Case s of
        'N','n': Negate_Frame;
        'D','d': Diff_Frame;
        'C','c': Cutoff_Frame(60);
        'Y','y': Dynamic_Y_Cutoff_Frame(0.45);
        'X','x': Dynamic_X_Cutoff_Frame(0.45);
        'B','b': Bin_Frame(128);
        'S','s': Smooth_Frame;
        'P','p': Sharp_Frame;
        'L','l': Line_Detection_Frame;
        'H','h': Hi_Pass_Frame;
        'T','t': Bill_Detection_Frame;
        '#':;
    Else

```

```
    Beep;
End;
If (command_string<>"") Then
    COPY_PAGE(WORK_FRAME,DATA_FRAME)
Else
    If (ReadKey=CR) Then
        COPY_PAGE(WORK_FRAME,DATA_FRAME);

    SET_DISPLAY_PAGE(DATA_FRAME);
End;

Function Count_Bills: Real;
Var
    x,y,i,j: Integer;
    bills,ave,sum: Integer;
    count: Array [0..MAX_BILLS] of Integer;
Begin
    For i:=0 to MAX_BILLS do
        count[i]:=0;
    For x:=FILTER_SIZE div 2 to XMAX-(FILTER_SIZE div 2) do
        Begin
            bills:=0;
            For y:=FILTER_SIZE div 2 to YMAX-(FILTER_SIZE div 2) do
                If (READ_POINT(x,y+1)>0) and (READ_POINT(x,y)=0) Then
                    bills:=bills+1;
                count[bills]:=count[bills]+1;
            End;
            CLEAR_VGA_SCREEN(0);
            peak:=0;
            ave:=0;
            sum:=0;
            For i:=0 to MAX_BILLS do
                Begin
                    DRAW_LINE(i+i,YMAX,i+i,YMAX-count[i]-1,64);
                    If (i mod 10)=0 Then
                        SET_POINT(i+i,YMAX-1,255);

                    If count[i]>count[peak] Then
                        peak:=i;
                        ave:=ave+i*count[i];
```

```
    sum:=sum+count[i];
End;
Count_Bills:=ave / sum;
End;

Var
  frame_file: String;
  s: Char;
  i,j,k: integer;
  peaks,bills: String;
  command: Text;

Begin
  command_index:=0;

  If ParamCount=0 Then
    Begin
      Write('Frame File < BANKNOTE.PIC > ');
      Readln(frame_file);
      If frame_file="" Then
        frame_file:='BANKNOTE.PIC';
      Write('Command String ? ');
      Readln(command_string);
    End
  Else
    Begin
      Assign(command,ParamStr(1));
      Reset(command);
      Readln(command,frame_file);
      Readln(command,command_string);
    End;
  If command_string<>" Then
    command_string[Length(command_string)+1]:='#';

  If SET_MODEX(DISPLAY_MODE) = 0 Then
    MESSAGE('Unable to SET_MODEX ');
    CLEAR_VGA_SCREEN(0);
    Gray_Scale;

  Display_Frame(FRAME_FILE,80);
```

```
s:=Select_Process;
While s<>ESC Do
Begin
  Process_Frame(s);
  s:=Select_Process;
End;

Str(Count_Bills:10:5,bills);
Str(peak,peaks);
If ParamCount<>0 Then
Begin
  Close(command);
  Append(command);
  Writeln(command,bills);
  Writeln(command,peak);
  Close(command);
End
Else
  s:=ReadKey;
MESSAGE('EZ_Money IS FINISHED: '+ peaks + ' bills counted.');
```

End.

WO 96/22553

PCT/US96/00658

36

APPENDIX B

```
; MODEX.ASM - A Complete Mode X Library
;
; Version 1.04 Release, 3 May 1993, By Matt Pritchard
; With considerable input from Michael Abrash
;
; The following information is donated to the public domain in
; the hopes that save other programmers much frustration.
;
; If you do use this code in a product, it would be nice if
; you include a line like "Mode X routines by Matt Pritchard"
; in the credits.
;
;
; All of this code is designed to be assembled with MASM 5.10a
; but TASM 3.0 could be used as well.
;
; The routines contained are designed for use in a MEDIUM model
; program. All Routines are FAR, and is assumed that a DGROUP
; data segment exists and that DS will point to it on entry.
;
; For all routines, the AX, BX, CX, DX, ES and FLAGS registers
; will not be preserved, while the DS, BP, SI and DI registers
; will be preserved.
;
; Unless specifically noted, All Parameters are assumed to be
; "PASSED BY VALUE". That is, the actual value is placed on
; the stack. When a reference is passed it is assumed to be
; a near pointer to a variable in the DGROUP segment.
;
; Routines that return a single 16-Bit integer value will
; return that value in the AX register.
;
; This code will *NOT* run on an 8086/8088 because 80286+
; specific instructions are used. If you have an 8088/86
; and VGA, you can buy an 80386-40 motherboard for about
; $160 and move into the 90's.
;
; This code is reasonably optimized: Most drawing loops have
```

; been unrolled once and memory references are minimized by
; keeping stuff in registers when possible.

; Error Trapping varies by Routine. No Clipping is performed
; so the caller should verify that all coordinates are valid.

; Several Macros are used to simplify common 2 or 3 instruction
; sequences. Several Single letter Text Constants also
; simplify common assembler expressions like "WORD PTR".

----- Mode X Variations -----

Mode #	Screen Size	Max Pages	Aspect Ratio (X:Y)
0	320 x 200	4 Pages	1.2:1
1	320 x 400	2 Pages	2.4:1
2	360 x 200	3 Pages	1.35:1
3	360 x 400	1 Page	2.7:1
4	320 x 240	3 Pages	1:1
5	320 x 480	1 Page	2:1
6	360 x 240	3 Pages	1.125:1
7	360 x 480	1 Page	2.25:1

----- The Legal Stuff -----

; No warranty, either written or implied, is made as to
; the accuracy and usability of this code product. Use
; at your own risk. Batteries not included. Pepperoni
; and extra cheese available for an additional charge.

----- The Author -----

; Matt Pritchard is a paid programmer who'd rather be
; writing games. He can be reached at: P.O. Box 140264,
; Irving, TX 75014 USA. Michael Abrash is a living
; god, who now works for Bill Gates (Microsoft).

CLAIMS

1. A method for counting banknotes comprising:
providing a stack of banknotes including at least one surface defined by edges of the banknotes; and
counting the number of banknotes in the stack using at least one optical sensor characterized in that the mutual orientation of the banknotes is substantially maintained relative to said at least one optical sensor, the counting step including:
employing said at least one optical sensor for generally simultaneously viewing at least two separate columns along said at least one surface; and
receiving an output from said at least one optical sensor and providing an output indication of a number of banknotes in the stack.
2. Apparatus for counting stacked sheets defining at least one surface defined by edges of the sheets, the apparatus comprising:
at least one optical sensor for generally simultaneously viewing at least two separate columns along said at least one surface; and
image processing apparatus receiving an output from said at least one optical sensor and providing an output indication of a number of sheets in the stack.
3. Apparatus according to claim 2 wherein the optical sensor comprises a plurality of sensing elements respectively viewing the plurality of side-by-side locations along the side of the stack.
4. Apparatus according to claim 2 wherein the optical sensor has a two-dimensional field of view.
5. Apparatus according to claim 2 and also com-



prising apparatus for varying the position of the stack relative to the optical sensor.

6. Apparatus according to claim 5 wherein said apparatus for varying comprises apparatus for moving the stack.

7. Apparatus according to claim 2 wherein said optical sensor is operative to repeatedly view at least one location along the stack of objects.

8. A method for counting stacked objects comprising:

viewing at least a portion of a side of a stack of objects first at least thereafter under first illumination conditions and under second illumination conditions; and

image processing apparatus receiving an output from said optical sensor comprising a first image of at least a portion of the stack under the first illumination conditions and a second image of at least a portion of the stack under the second illumination conditions, and operative to compare the two images and to provide an output indication of a number of objects in the stack.

9. Apparatus for counting stacked objects comprising:

at least one support for supporting at least one stack of objects such that the mutual orientation of the objects relative to the support is maintained;

at least one optical sensor disposed behind the at least one support for viewing at least a portion of a side of said at least one stack of objects through the support; and

image processing apparatus receiving an output from said optical sensor and providing an output indica-



tion of the number of said objects in said at least one stack.

10. Apparatus according to claim 9 wherein the support is transparent.

11. Apparatus according to claim 9 wherein the support has at least one window formed therein.

12. Apparatus according to claim 2 wherein said at least one optical sensor comprises a plurality of optical sensors each of which is operative to view a plurality of locations along a side of a different stack.

13. Apparatus according to claim 2 wherein said at least one optical sensor comprises a plurality of optical sensors each of which is operative to view at least a portion of a side of a different stack of objects.

14. Apparatus according to claim 2 and also comprising a plurality of light sources illuminating the stacked objects.

15. Apparatus according to claim 9 and also comprising a plurality of light sources illuminating the stacked objects.

16. Apparatus according to claim 2 and also comprising at least one support for supporting at least one stack of objects and wherein the at least one optical sensor is disposed behind the at least one support for viewing at least a portion of a side of a stack of objects through the support.

17. A method according to claim 8 wherein the stack portion is viewed from the side.

