Clearspeech: A Display Reader for the Visually Handicapped

Tim Morris, *Member, IEEE*, Paul Blenkhorn, Luke Crossey, Quang Ngo, Martin Ross, David Werner, and Christina Wong

***Abstract*—Many domestic appliances and much office equipment is controlled using a keypad and a small digital display. Programming such devices is problematical for the blind and visually handicapped. In this paper, we describe a device that may be used to read the displays on these devices. The device is designed to accept a description of the display being read, which specifies the types and locations of elements of the display. Images are captured using a handheld webcam. Images are processed to remove the distortions due to the camera orientation. The elements of the screen are interpreted and a suitable audio output is generated. In suitably illuminated scenes, the display data is interpreted correctly in approximately 90% of the cases investigated.**

***Index Terms*—Assistive devices, audio interpretation, display reader.**

I. Introduction

MANY consumer white goods now utilize a display to indicate their status and to provide feedback to the user as they are being programmed. For example, a microwave oven can be programmed by the user setting the power level and cooking time, the display would echo these settings, and during cooking, it would indicate the remaining cooking time. While these displays are a boon to the able bodied, they offer no advantage to the blind and partially sighted population.

A survey of consumer products currently on sale revealed the distribution of displays and display types summarized in Table I. The columns indicate the types of products that were investigated and the numbers found, and the numbers of the products without any display, with liquid crystal displays (LCD) and with light emitting diode displays (LED). The type of display is also shown: whether it was a dot matrix, seven- or sixteen-segment display or whether the display included any iconic symbols. There were no displays that included both LED and LCD, but both types of display could have dot matrix, seven- or sixteen-segment or symbolic regions. Since the displays often included areas of different types, the sum of the final four columns does not correspond to the number of products with displays (the sum of the columns headed LED and LCD). Overall, 58% of the products surveyed had some form of electronic display. It is expected that this proportion will, in time, increase.

Manuscript received March 23, 2005; revised May 24, 2006; accepted June 6, 2006.

T. Morris and P. Blenkorn are with the School of Informatics, The University of Manchester, Manchester, M60 1QD, U.K. (e-mail: tim.morris@manchester.ac.uk).

L. Crossey, Q. Ngo, M. Ross, D. Werner and C. Wong were with the Department of Computation, UMIST, Manchester, M60 1QD, U.K.

Digital Object Identifier 10.1109/TNSRE.2006.881538

According to the U.K.'s Department of Health [1], 30 480 people in the 18–49-year-old age group were registered as blind or partially sighted in March 2000. Approximately 1.8% of the total population is visually impaired. This represents a significant number of people who are unable to take advantage of the information presented by the displays on these consumer products. Anecdotal evidence suggest that blind or partially sighted people use consumer products either by fixing Braille labels to the switches, by memorizing the commonly used settings, or by using older, less sophisticated models (e.g., using one power setting on the microwave oven and one cooking time). In the latter two cases, the sophisticated functionality offered by modern products is unusable or unavailable. In some cases, this will make the complete product unusable. In this paper, we report an investigation into a method of alleviating this problem; we describe a device that is able to "read" the display. Our device uses a handheld webcam to capture images of the display, these are processed to recognize the text, which is then spoken. A webcam was chosen for cost reasons: many home computer systems are supplied with one, hence there is no cost implication; otherwise the purchase cost is minimal. Contrast this with the alternative of purchasing a video camera and framegrabber.

Several fundamental problems must be addressed in the development of this device. First, can the blind user align the webcam to the display such that suitable data can be captured for the transcription process? Second, how can the device be made compatible with the wide and increasing range of consumer products? Third, what is the appropriate output that the device should provide?

In the remainder of this paper, we give a brief review of relevant vision aids, then examine the requirements of our device and describe how these were realized. The device is then evaluated, according to the accuracy of the results it gives, its speed of operation, and its flexibility with respect to interpreting the displays of different devices and user acceptance. We conclude by reviewing the achievements of this work and suggesting the directions for future research and development.

II. Low Vision Aids

Most visually impaired people have sufficient residual sight to accomplish everyday tasks but will experience difficulty in reading, writing and tasks that require a higher level of visual acuity. Low vision aids are used to overcome these difficulties and can range from simple magnifiers to sophisticated electronic devices, depending on the task to be performed and the degree of visual loss.

TABLE I
INCIDENCE OF DISPLAY TYPES IN PRODUCT DISPLAYS

Product	Number of products	No display	LED	LCD	Dot matrix	7-segment	16-segment	Symbolic
Breadmaker	8	0	2	6	0	8	0	
Clocks and clock radios	18	4	8	6	1	14	0	3
Digital phones	23	0	8	15	8	15	0	4
Dishwashers	17	12	4	1	0	5	0	
DVD players	22	6	13	3	5	9	7	7
Electric cookers	43	20	23	0	0	23	0	
Fax machine	9	0	0	9	9	7	2	
Gas cookers	20	10	10	0	0	10	0	
Hi-fi	23	1	18	4	20	7	15	18
Hobs	25	25	0	0	0	0	0	
Microwave	27	3	10	14	0	24	0	
Ovens	23	9	14	0	0	14	0	
Steamer	7	5	2	0	0	2	0	
Video players	16	1	13	2	0	3	12	10
Washing machines/driers	76	56	14	6	0	20	0	
Totals	357	152	139	66	43	161	36	42

The most well-known vision aids are tactile. Braille is used for printed materials while tactile clocks and watches are available that are "read" by feeling the hands.

Talking products have become more popular, especially among the older visually impaired. Speech capabilities have been added to clocks, watches, calculators, and some kitchen equipment. The market for such devices will necessarily be small.

Video magnifiers have been in use for many years. A video camera is used to view an object and the magnified image displayed on a monitor. Magnification ratios up to 60 times may be achieved.

Screen readers are programmes that convert on-screen text to speech, while text readers include optical character recognition and can, therefore, read whatever text is imaged. They can, therefore, be viewed as a combination of video magnifier and screen reader. The device we are proposing is similar in concept to the text reader, but must allow for uncontrolled positioning of the camera with respect to the text to be read and uncontrolled illumination; two factors that are well controlled by the text reader.

Display readers have been considered by several organizations (Schepens Eye Research Institute,[1] American Foundation for the Blind,[2] Tiresias[3]). Their potential usefulness has been noted: "this has the potential to allow totally blind consumers access to home appliances…".[1] So have the problems that must be addressed when implementing them: "if the device is handheld, the system must allow for a blind user holding the device at an angle to the display."[4]

Only one device has been actually implemented, and that as an evaluation, by Blindsight[4]. Blindsight claim 95% accuracy for their display reader, but do not give any of the test details: types and complexities of displays, lighting conditions, and so on.

The Section III will define the functional requirements of the device and describe the characteristics of the images that are input.

III. FUNCTIONAL REQUIREMENTS

The proposed device is to read the LCD or LED displays that convey information from a range of consumer devices that are used in the home or office environment. Images of the display will be captured using a handheld webcam. The text in the image will be interpreted and the significant portions will be spoken.

The display could be composed of alphanumeric characters and icons. The characters could be composed of seven- or sixteen-segment or dot matrices. The device will be required to interpret all of these various components.

Three modes of output are proposed. In the first, the device will translate all the components of the display. In the second, the device will translate what has previously been determined to be the significant portions of the display. And in the third, the device will recognize and translate only those portions of the display that have changed since it was last read.

Note that these modes of operation imply that a description of the display is available to the system. This description would indicate the components of the display and their locations, and the "significant" portions of the display. Note also that the meaning of the strings of characters on the display is context dependent: normally, a microwave oven's display would show the current time, during programming it might show the total cooking time, during cooking it would show the remaining time. This information must be available to the device as it translates the image. We have termed the description of the display a "profile." It is described in more detail below.

It is critically important that the visually impaired users are able to position the webcam to successfully capture a useable image of the display. In this context, "useable" implies that the

[1]http://www.eri.harvard.edu/faculty/peli/index.html

[2]http://www.afb.org

[3]http://www.tiresias.org

[4]http://www.blindsight.com/projects.html

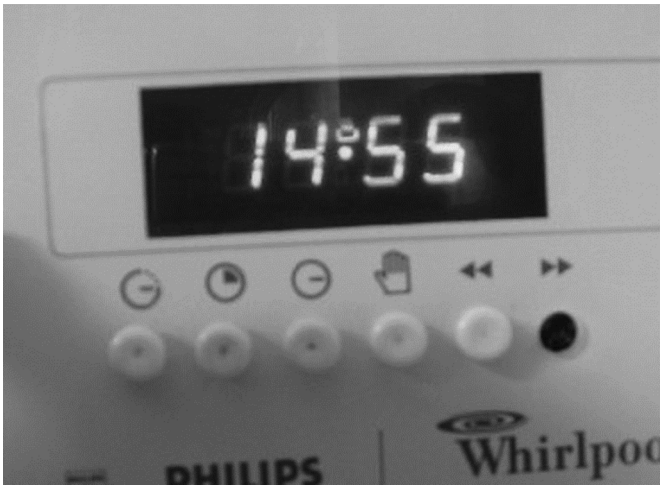

Fig. 1. Typical image captured by a visually handicapped user.

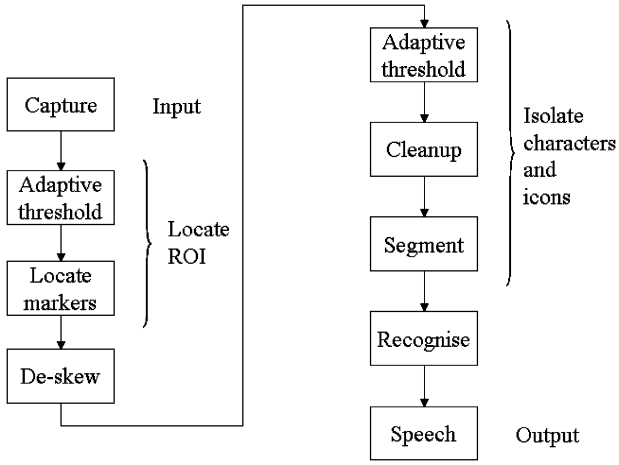

Fig. 2. System's processing pipeline.

display's characters are visible and approximately correctly orientated, i.e., the characters read almost horizontally and are not too skewed. We cannot expect the users to achieve any degree of accuracy, and any errors must be corrected by the system. In a preliminary study, we asked a number of potential users of the system to photograph displays on consumer products. They were able to do this without difficulty. Having located the display by touch, the camera was positioned with the aid of an outstretched hand and the photograph taken (the little finger was placed adjacent to the display and the camera was held just touching the outstretched thumb). In all cases, the display was clearly visible in the image, although it was often skewed and sometimes blurred. Fig. 1 presents a typical example of one of these images. The deficiencies of the image that must be addressed are clear. The camera is not directly facing the display, resulting is the rectangular display having a trapezoidal appearance. The camera is also rotated in the plane of the display, resulting in the major axis of the display not being parallel with the sides of the image.

Finally, we require the device to give a response every second, that is, an image must be captured and processed, and text prepared for speaking within this time. This time is dictated by the shortest event that must be recognized: the change in a clock's display.

The Section IV describes the design of the system, and the sections that follow examine the major components of the device.

IV. Design

The system was designed to repeatedly execute the processing pipeline of Fig. 2. The pipeline specifies modules to capture images, to perform some initial processing that identifies the region of the image containing the display, to deskew the images (that is to manipulate the image such that rectangular objects appear rectangular), to threshold the data, isolate the characters and icons, recognize them, and finally prepare the text for speaking.

A product's profile is used to store all the product specific information required in the processing pipeline.

A. Image Capture

Having chosen to use a webcam for cost reasons, various makes were investigated for their suitability. It was found that they captured poor quality images: the resolution was poor, the signal-to-noise ratios were worse than for video cameras and colors were rendered inconsistently. These problems were consequences of the small image detector and the image coding used to allow transmission to the USB port of the host.

Nevertheless, we elected to use a QuickCam Pro 3000. This had the additional advantage of having manual focussing, which allowed the camera's focus to be set at a suitable range for this task. Our experience was that autofocussing cameras tended to focus on the scene reflected in the display, the display was, therefore, too blurred to be readable.

The characteristics of the images dictate the stages of the pipeline that follow. We cannot guarantee that the image is orientated uniformly; even if the operator were able to see the display it would not be possible to orientate a handheld camera correctly. Therefore, the first stage in the pipeline is to identify the region of interest in the image, i.e., the quadrilateral defining the display, and to resample the image making it rectangular.

B. Locate Region of Interest and De-skewing

Locating the region of the image that contains the display is a significant problem, since the displays that are in use in these products have such a large range of appearances. A pure computer vision solution could be formulated. This would involve searching for text in the captured image [2], [3] using the text's appearance as cues for locating it. This process would simultaneously locate and recognize the text. But it would yield no information regarding the location and status of the symbolic portions of the display. Instead, we have elected to take a computationally simpler approach and place markers near the four corners of the display. Although this will require some initial setting up and calibration by a seeing assistant, it allows a much simpler, and, consequently, faster and more robust, method of identifying the region of interest. In a release version of the system, the markers would be printed on a transparent overlay,

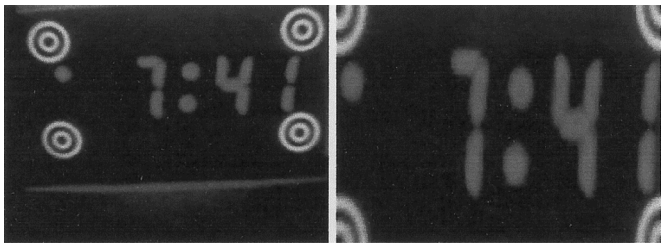

Fig. 3. Left: Captured image. Right: De-skewed image with the marker centres at the four corners.

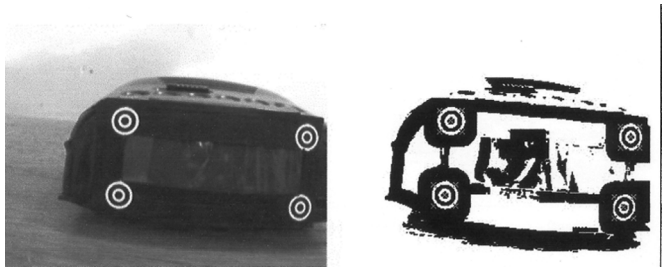

Fig. 4. Left: Captured image. Right: Image thresholded by the locally adaptive algorithm.

unique to the product being processed. The overlay would be affixed by a seeing assistant who would also probably be responsible for installing the profile for that product and calibrating the device: this involves simply aligning the display to the overlay. Profiles would be built, probably on an ad-hoc basis by some charitable organization.

Our first attempt used colored squares as the markers. Our belief was that we would be able to identify them by their hue values, which would be largely independent of ambient illumination [4], compare similar work in identifying faces in images [5], [6]. Provided that the colors were chosen carefully, there was no confusion with any similarly colored region in the image. However, the color values captured by the webcam were not sufficiently stable to allow reliable identification of these colored regions.

We, therefore, elected to use uniquely shaped markers. These were composed of a series of concentric black and white circles (Fig. 3). The markers were found by searching for alternating bands of maximum and minimum values in a thresholded version of the image. At this point, it is appropriate to discuss the thresholding methods.

Thresholding simply converts a greyscale image to a binary one by comparing the value of each pixel against a threshold. In our case we use the intensity of each pixel, computed as the average of the red, green and blue values. Using a fixed threshold is inappropriate in this and indeed in most applications, neither is a global threshold: the first method does not give results that are robust against changes in illumination, the second method does not give results robust against local variations in illumination. Instead we have used an adaptive thresholding method; the average of the surrounding 196 pixels (a 13×13 pixel region) is used as each pixel's threshold. The method was optimized to improve performance in two ways. As the neighborhood window was moved across the image, the values of the pixels removed were subtracted from the total and the values of the newly included pixels were added. Therefore, the new sum of pixels in the neighborhood was computed by 28 additions and subtractions. And rather than divide the sum by 196 to compute the average, the pixel under consideration had its value multiplied by 196.

The resulting image (Fig. 4) clearly indicates the display-corner markers. They were located in the image by systematically searching for the correct alternating sequence of black and white pixels. For example, the marker at the top left was found by searching from the top left corner of the image along each line in turn. A limited amount of each line was searched. Having found the correct sequence of black and white pixels, the target was verified by sampling radially from the supposed centre. If the location corresponded to the center of the marker, then the sampled values along each radius would be equal (Fig. 5). Similar search patterns were used for the other three markers.

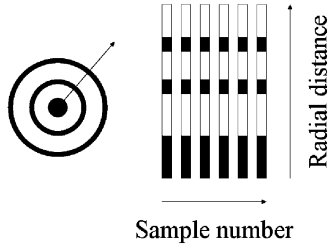

Fig. 5. Method of locating the centre of a marker.

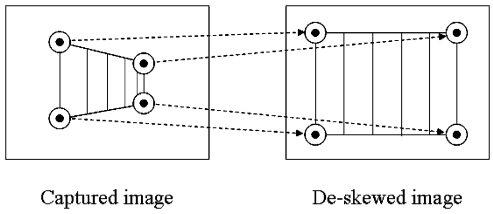

Fig. 6. De-skewing algorithm: image is de-warped and resampled to take perspective effects into account.

Having identified the region of interest, and defined it by its corners, the final preprocessing step was to de-skew it. This resulted in a rectangular image of a standard size, equivalent to what would have been observed had the camera being facing the display correctly. The output image was derived using the data of the original color image, and the region of interest coordinates obtained from the processed image.

The standard de-skewing process would compute the equivalent skewed image coordinates of every pixel in the de-skewed image, assuming that the de-skewed coordinates are uniformly spaced. The value to be stored in the de-skewed location would be computed from the value of the skewed image at that location. Since the skewed coordinates are likely to be nonintegral values, the de-skewed values are computed by interpolation. However, this algorithm did not remove any perspective effects, and the supposedly de-skewed image retained significant distortions. Instead, the de-skewed coordinates were not uniformly spaced, rather the spacing between coordinates changed uniformly, being closer together further from the camera (Fig. 6).

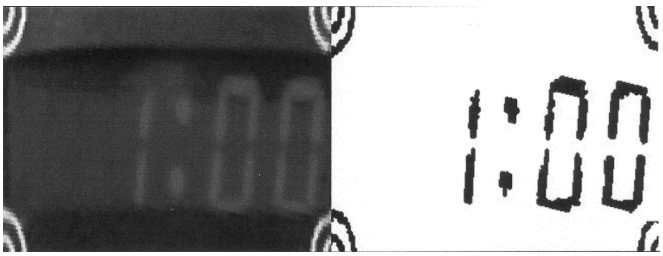

Fig. 7. Left: Typical de-skewed image. Right: Image after thresholding.

At this point, we had a color image corresponding to the view that would have been seen had the camera been directly facing the display. The image was cropped to include only the display (see Fig. 3). The following stage of the pipeline was concerned with locating the individual components of the display: whether they were characters or icons, and determining whether the icons were illuminated or not and what characters were being displayed.

C. Character Identification and Recognition

The resampled image was again adaptively thresholded. It was found that the original data had to be processed instead of continuing to process the already thresholded image, as by this stage too many artefacts had been introduced to allow accurate recognition by the subsequent stages. It was found that many pixels were incorrectly thresholded, mainly in uniformly darker colored regions. The false positive results in the darker regions of the image were corrected by thresholding the image with a fixed threshold set at the twenty-fifth percentile of the grey values. The final part of the thresholding stage was to remove all small false positive regions, which was achieved by two iterations of erosion. Fig. 7 is typical of the results obtained.

The display's profile, amongst other information, specified the locations of the components of the display. For example, a digital clock's profile would specify the regions of the image that contained each of the four digits, the colon between the hours and minutes pairs of digits, and any icons indicating AM or PM or that the alarm was set. Since the resampled image is a standard size, these coordinates are absolute values.

The regions corresponding to the different components were again resampled to a standard size. Nearest neighbor interpolation was used. Icons and character regions were treated differently; since we wish to recognize a character, but merely recognize the presence or absence of an icon.

An icon was recognized by summing the number of pixels set in the thresholded image and comparing this to a threshold defined in the profile. If the number exceeded the threshold, then the icon was illuminated; otherwise, it was not. The threshold was determined as a proportion of the maximum number of pixels that could be set, should the icon be present.

Two methods were investigated for recognizing characters. One we termed the method, the other was termed the eigenimage method.

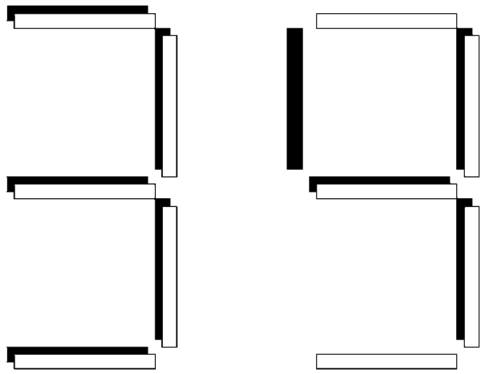

Fig. 8. XOR method of identifying illuminated components of a display.

D. XOR Character Recognition

This was designed as a simple and rapid method of recognizing characters. It computed the exclusive OR (XOR) of the extracted image and a template of the character to be matched. The number of TRUE pixels in the XOR image was a measure of the mismatch between the image and template: a value of zero indicated a perfect match between the two while a value equal to the number of pixels indicated a perfect mismatch. A zero result could also have been generated if both image and template were zero, a case that did not occur. Intermediate values were indicative of a certain similarity, for example a "3" and a "4" in a seven-segment display would share four segments (three on and one off), and would differ in three, their XOR score would, therefore, reflect this. Fig. 8 presents examples of a template and matching and nonmatching images.

While template matching is not often used to solve image recognition problems as many templates are usually required to allow for image variability, in this application we could define what characters were allowed in each region of the image and store this information in the profile. Therefore, the recognition module could be provided with a set of templates.

E. Eigenimage Recognition

A more sophisticated recognition algorithm was also implemented, using the eigenimage technique suggested by Turk and Pentland for face recognition [7]. The eigenimage representation is simply an alternative set of basis functions for representing a dataset. Thus, any image D_i can be represented by a weighted sum of the basis functions B_k

$$D_i = \sum_k B_k \beta_{ik}. \tag{1}$$

The basis functions are an orthogonal weighted sum of pixel values. The power of the method is that the basis functions may be ordered according to the variance of the data measured along the functions

$$Var(D_i) \geq Var(D_j) \,\forall\, i > j. \tag{2}$$

The number of basis functions required to completely represent the data is equal to the number of pixels in an image. However, for most practical purposes, we may approximate the data using a much smaller set of functions. In this investigation it was found that a 128 × 128 pixel image could be approximated with sufficient accuracy by eight basis functions. Therefore, each image was represented by eight coefficients. Recognition was achieved by computing the eight coefficients for each input image and comparing them with coefficients computed offline for each different character, a simple modification to the nearest neighbour rule was used for the assignment

$$D_i = \sum_k \frac{|\hat{\beta}_k - \beta_k|}{\beta_k^{\max} - \beta_k^{\min}}. \tag{3}$$

The absolute difference between each sample mean weight and the image's weight was divided by the range of the samples' weights and summed. The sample was assigned to the class having the minimum distance.

Both methods were found to give acceptable recognition rates.

F. Speech Output

Three modes of output were built into the system. The user might require all the display contents to be spoken, only the significant elements to be spoken, or the changes that have been made to the display, either since it was last viewed or as it is being viewed.

Speech was generated using the Microsoft Speech API, a text to speech engine. Therefore, the aim of processing the image was to determine the appropriate text string.

The display's profile was vital for determining the speech string, since it defined the speech associated with each component of the display, the ordering of the speech and the significance of each component. Thus, if the system was in the "speakall" mode, all components of the display would be interrogated: recognition of icons would return the associated text string or a null string, recognition of characters or numerals would return the appropriate character or number. The ordering of the text in the output was determined by the significance of the component which was implied by the ordering of components in the profile, as will be described in Section V.

It should be noted that many displays do not have text fields, only numerical fields. For example, a digital clock has numerical fields to indicate the time and icons to indicate other information such as AM or PM or whether the alarm has been set. Other products have more sophisticated displays, but have the same components. Therefore, no word recognition capability is required.

The status of the device also influenced the speech string. For example, a microwave oven might display the time and the cooking time as a pair of two digit numbers. In the first case, the numbers are hours and minutes, in the second, they are minutes and seconds. The profile contains information to determine how to interpret the data.

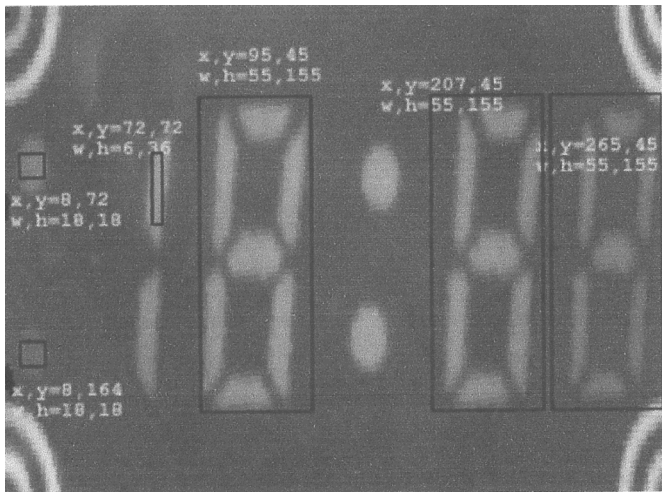

Fig. 9. Profile information superimposed on the image to which it refers.

V. PROFILES

In this section, we describe the syntax of the profile. In practice, a device's profile is a text file that is loaded once when the particular device is to be interrogated. In effect, the profile contained all of the parameters particular to a display. It is best explained by a simple example. Essentially, the profile defined the regions of the image that corresponded to the symbols and icons to be identified, it defined the text to be spoken and what text and in what order for the three modes of operation. It also contained various rules dealing with exceptions.

Consider the image of Fig. 9. It shows a deskewed image of an alarm clock. Superimposed on it are the boundaries of the regions of interest in the image that are defined by the profile. In the profile, the regions of interest were termed sectors corresponding to regions that contained numbers, and symbols corresponding to regions containing icons. In this example, the icons are the AM/PM symbol, the alarm ON/OFF symbol and the tens of hours digit. These were specified in the profile as follows:

$$\begin{aligned}
\text{device} &= \text{alarm clock}\\
\text{display_type} &= \text{LED}\\
\text{sector} &= \text{Hrs_2}, 95, 45, 55, 155, x\\
\text{sector} &= \text{Min_1}, 207, 45, 55, 155, x\\
\text{sector} &= \text{Min_2}, 265, 45, 55, 155, x\\
\text{symbol} &= \text{Hrs_1}, 1, 72, 72, 6, 36\\
\text{symbol} &= \text{PM}, \text{P.M.}, \text{A.M.}, 8, 72, 18, 18\\
\text{symbol} &= \text{Alarm}, \text{Alarm set}, \text{Alarm off}, 8, 164, 18, 18\\
\text{symbol} &= \text{colon}, :, :, 0, 0, 1, 1.
\end{aligned}$$

The device field was used to give the profile an identification that could be spoken on loading to provide confirmation that the correct profile had been loaded.

The display type could be LED or LCD. Slightly different processing methods were used for the two types.

The sector field specified a name for the sector, the x and y coordinates of the top-left corner of the region of interest, the width and height of the region and the recognition method to be used. Coordinates were measured from the origin of the image at

its top-left corner. Absolute coordinates were used since images were resampled to a standard size. Thus, the Hrs_2 sector had its origin at (95, 45), was 55 × 155 pixels in extent and the XOR method was used for recognition.

The symbol field specified a name for the symbol, the text to be spoken if the symbol was found to be on or off and the location of the symbol (x, y, width, height). The colon symbol was used to ensure correct output of the speech.

The profile then defined the sectors and symbols to be spoken, and their order, in the various operating modes as follows:

$$\begin{aligned}\text{main} &= \text{Hrs\_1}, \text{Hrs\_2}, \text{colon}, \text{Min\_1}, \text{Min\_2}\\ \text{change} &= \text{Hrs\_1}, \text{Hrs}_2, \text{colon}, \text{Min\_1}, \text{Min\_2}, \text{Alarm}\\ \text{all} &= \text{Hrs\_1}, \text{Hrs\_2}, \text{colon}, \text{Min\_1}, \text{Min\_2}, \text{PM}, \text{Alarm}\end{aligned}$$

Therefore, the mode also defined the sectors and symbols to be identified. The penultimate part of the profile defined the sectors or symbols to be spoken and their formats when the system was in the various modes as follows:

$$\begin{aligned}\text{speech\_main} &= \text{Hrs\_1} + \text{Hrs\_2} > \%d, \quad \text{colon} > \%s,\\ &\quad \text{Min\_1} + \text{Min\_2} > \%d\\ \text{speech\_change} &= \&\text{Hrs\_1} + \text{Hrs\_2} > \%d, \quad \&\text{colon} > \%s,\\ &\quad \text{Min\_1} + \text{Min\_2} > \%d, \quad \text{Alarm} > \%s\\ \text{speech\_all} &= \text{Hrs\_1} + \text{Hrs\_2}\%d, \quad \text{colon} > \%s,\\ &\quad \text{Min\_1} + \text{Min\_2} > \%d,\\ &\quad \text{PM} > \%s, \quad \text{Alarm} > \%s\end{aligned}$$

Symbols can be concatenated to form more complex structures, thus the pairs of single digits were combined to give numbers in the correct ranges. The %d and %s are formatting symbols. The colon symbol was included to indicate to the speech engine that a string representing a time was being synthesized, the speech engine would treat strings of the form "$dd : dd$" as a time and "dd dd" as two pairs of digits.

Finally, the ampersands were used to control the symbols spoken when the system was in the speaking change mode, the ampersands delineated symbols that were to be spoken regardless of whether they had changed or not. Without them, only the changed digits of the time would be spoken.

The final set of rules defined the set of values that could appear in a sector. Symbols did not appear in this portion of the profile as they could only be on or off. In some instances, the allowed values of a sector were dependent on the values of other regions of the screen. Continuing with the clock example, if the first hours digit of the clock is on, i.e., is a one, then the second digit could only be 0, 1, or 2. Therefore, the rules specifying the allowed values of the second hour's digits were as follows:

$$\begin{aligned}\text{rule} &= \text{Hrs\_2} > \text{Hrs\_1(ON)}, 0, 1, 2\\ \text{rule} &= \text{Hrs\_2} > \text{Hrs\_1(OFF)}, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9\end{aligned}$$

and the first minutes digit is specified by

$$\text{rule} = \text{Min\_1}, 0, 1, 2, 3, 4, 5$$

```
device = device_name
display_type = LED | LCD

sector = region_ name, X_pos, Y_pos, width, height, recognition_type
symbol = region_name, on_text, off_text, X_pos, Y_pos, width, height
Recognition_type = x | e

mode → main | change | all
mode = region_name*

speech_mode → speech_main | speech_change | speech_all
speech_mode = segment*
segment = &region_name+region_name>format | region_name>format |
region_name+region_name>format

rule = region name, allowed value*
rule = region name > dependent region (allowed value), allowed value*

–
```

Fig. 10. Backus–Naur form (BNF) [8] definition of the device profile.

The interpretation of the minutes rule was that this field could take the set of values listed. The two hours rules indicated the set of allowed values of the Hrs_2 digit as they were affected by the Hrs_1 symbol. These rules ensured that the system would only speak valid times, but they did not ensure that the time would be correct: digits could still be misread and yield a valid time.

These rules defined the placing of symbols and characters on the display and the allowed values that characters could take. Formally, the rules were defined by the syntax shown in Fig. 10. The profile also included the data necessary for recognition: the templates that would be used in the XOR recognition and/or the eigenimage data.

VI. Evaluation

Our initial study indicated that visually handicapped persons were able to capture images suitable for processing by this system. The evaluation, therefore, concentrated on the system's ability to identify the region of interest markers, and symbols and characters in the display under varying lighting conditions. How quickly the system responded to changes was also evaluated. Finally, the system was evaluated by a number of blind and visually handicapped people.

A. Accuracy

The ability of the system to detect all four markers under varying lighting conditions and varying camera orientations with respect to the display plane was investigated. It was found that the markers were found reliably under all combinations of natural sunlight, incandescent, halogen, and fluorescent bulbs when the camera was directly facing the display, and at 45° and 135° to the normal of the display's plane. At an angle of 160° to the normal, only two markers were located, however, this is an extreme angle and we would not expect images to be taken at this orientation in practice. It could be that the users of this system fail to locate the display accurately, in which case fewer markers would be detected, then an error message is output (orally) stating the number of markers that are detected. This implies that the camera should be repositioned. Note that it is

TABLE II
SUCCESS RATES OF ANALYZING LED AND LCD DISPLAYS

Illumination	Display Type	
	LED	LCD
Good	97%	88%
Moderate	66%	57%
Poor	22%	3%

not always possible to identify which markers have not been located.

The marker detection algorithm was sensitive to the apparent size of the markers. It was found to fail at a range greater than approximately 25 cm. We do not consider this a problem; in fact, it could be construed as an advantage, since at greater ranges than this the character detection would also probably fail. Further, we observed that visually handicapped people spontaneously positioned the camera using an outstretched hand with the little finger adjacent to the display and the camera touching the thumb. This aided the camera orientation and also set the range at a consistent distance of approximately 20 cm.

It was also observed that the marker detection algorithm failed when the camera was rotated to such a degree that one of the bottom pair of markers was nearer the top of the image than one of the upper pair of markers. Again, given the methods used to capture the data, this is an unlikely occurrence.

Having located the markers, the image was always de-skewed satisfactorily. The thresholding process was successful, provided that the objects in the display were not so small as to be removed by the noise reduction process. In general, this was not the case. The profiles were used successfully to abstract the different regions of the image containing icons and characters.

Recognizing the presence of an icon was successful due to the trivial nature of this operation. Recognition of the characters using both methods was evaluated and both were found to be acceptably accurate.

Finally, the complete system was tested under varying lighting conditions that were intended to span the range of lighting conditions that users might encounter in practice. Three lighting conditions were investigated: good, moderate, and poor, as assessed by a normally sighted person. Good lighting conditions corresponded to the room being well lit by natural daylight. Moderate lighting corresponded to the natural light being partially excluded, but artificial light (fluorescent and halogen) being used to compensate. Poor lighting corresponded to the absence of artificial lighting and the partial exclusion of natural light. The success rates summarized in Table II were obtained. Under good lighting conditions, the system read an LED display correctly in 97% of the attempts. The LCD display was read slightly less accurately, 88% of trials were successful. As the illumination was degraded, the success rates declined, as would be expected, reaching 22% and 3%, respectively, for the LED and LCD displays when they were poorly lit.

Under poor lighting conditions, the automatic gain control of the camera overcompensated for the lighting, adversely affecting the image and resulting in erroneous thresholding. In these conditions it was also found that the human eye could not detect the characters on the LCD display. Extremely bright illumination could cause the screen to become unreadable due to reflections and glare.

B. Speed

We required that the system should respond to events within one second, since this is likely to be the shortest time interval between changes to a typical display. Using a desktop PC with a 1-GHz clock speed we observed that the image processing and character recognition took approximately 0.3 s, the speech engine (which was outside of our control) took approximately 0.5 s between receiving a text string and beginning its utterance. The total time taken is well within our target response time.

C. User Testing

Finally, five potential users tested the system, four were completely blind, and one was partially sighted. The group was selected by a charity known to us.[6] Although not intended to be representative of the blind population, the group's members were typical of the target users of the system. They were asked to use the system to interact with a digital clock to:

- determine the current alarm time;
- advance the clock time to the next hour;
- reset the alarm to a specified time;
- turn on the alarm.

All the test subjects managed the tasks. Some initially obscured the webcam's field of view, but this was soon realized and corrected. They reported that the error messages were particularly effective in ensuring that the correct image data was captured.

All users also commented that the device would be potentially useful, given the proliferation of products having visual displays.

VII. CONCLUSION

In this paper, we have reported on a prototype screen reader that is intended to vocalize the information displayed on the LCD or LED screen of home or office equipment. Three modes of operation were defined in which the screen reader spoke all the information, the significant portions, or the changes that appeared on the screen.

The properties of each screen were specified in a profile, a text file that is to be loaded prior to reading the equipment's screen. The components of the display were specified in the profile. Components were described as a symbol or a text region and the limits of the region were specified. The characters that could be recognized in each region were also specified, to aid error reduction. The profile also specified what regions of the screen were to be investigated in each of the operating modes.

Image data is captured using a cheap webcam. We have specified this input as it is a low-cost device and we require the cost of the complete system to be minimized. Despite the low quality of the images that are captured, low signal-to-noise and imperfect alignment with respect to the display, we are able to identify the display in the image with the aid of specific markers fixed

[5]http://www.screenreader.co.uk

to the display, and identify the symbols and characters in it. In good lighting conditions, we are able to correctly read an LED display in 97% of the cases investigated and an LCD display in 88% of the cases. The success rates are lower in poorer quality illumination.

Several issues have arisen from this research and merit further investigation. First, how should the markers be placed on the device? It is our intention that an able bodied assistant would place the markers in the correct location. Since it is unlikely that this will be accurate to within one pixel, as the profile requires, we anticipate that an automatic calibration would be required, this would locate some of the display's fields and determine the error between the published profile and the true location of the field.

Second, certain types of display were not addressed in this investigation, namely scrolling displays such as are seen on audio devices such as DAB radios. These may be processed by adding additional region types to the profile and incorporating further functionality into the display reader software. This is the subject of further research.

Third, it is not likely that a device hosted on a PC would be acceptable to the majority of our intended users. It is, therefore, imperative that a future version of the software be ported to a PDA, mobile phone, or other camera-equipped portable device. This should confer the additional advantage of requiring the users to be less dextrous in holding the camera correctly while simultaneously pressing the appropriate buttons.

REFERENCES

[1] *"Registered Blind and Partially Sighted, Year Ending 31 March 2001,"* Dept. Health, London, U.K., 2002.

[2] J. Gao and J. Yang, "An adaptive algorithm for text detection from natural scenes," in *Proc. IEEE Compt. Soc. Conf. Vision Pattern Recognnition*, 2001, vol. 2, pp. 84–89.

[3] H. Li, D. Doermann, and O. Kia, "Automatic text detection and tracking in digital videos," *IEEE Trans. Image Process.*, vol. 9, no. 1, pp. 147–156, Jan. 2000.

[4] P. Blenkhorn and J. McCollin, Controlling the cursor by tracking a coloured bead, School Informatics, Univ. Manchester, Manchester, U.K..

[5] M. Störring, H. J. Andersen, and E. Granum, "Skin colour detection under changing lighting conditions," in *Proc. 7th Int. Symp. Intell. Robot. Syst.*, Coimbra, Portugal, 1999, pp. 187–195.

[6] M. M. Fleck, D. A. Forsyth, and C. Bregler, "Finding naked people," in *Proceedings of the 4th European Conference on Computer Vision-Volume II*, ser. Lecture Notes Comput. Sci.. London, U.K.: Springer-Verlag, vol. 1065, pp. 593–602.

[7] M. A. Turk and A. P. Pentland, "Face recognition using eigenfaces," in *Proc. IEEE Conf. Comput. Vision Pattern Recogniton*, 1992, pp. 586–590.

[8] P. Naur, "Revised report on the algorithmic language ALGOL 60," *Commun. ACM*, vol. 3, no. 5, pp. 299–314, May 1960.

Tim Morris (M'95) was born in Cardiff, U.K. He received the B.Sc. degree in physics from the University of Southampton, Southampton, U.K., in 1981 and the Ph.D. degree in medical physics from the University of Sheffield, Sheffield, U.K., in 1995.

He is currently a Lecturer in the School of Informatics, the University of Manchester, Manchester, U.K., having been a Lecturer at UMIST and a Researcher in the British Glass Industry Research Association, Sheffield, U.K. He is the author of Multimedia Systems (London, U.K.: Springer, 2000) and Computer Vision and Image Processing (Basingstoke, U.K.: Palgrave, 2004). He is interested in applications of computer vision, especially in developing aids for the visually handicapped, in ophthalmology, and in metrology.

Dr. Morris is a member of the British Machine Vision Association.

Paul Blenkhorn researches into speech and sight rehabilitation engineering needs for people with disabilities. Assistive devices have been developed including hand-held and head-controlled systems. He was an invited participant for the EPSRC/AgeNet/Department of Health workshop to develop the EPSRC Rehabilitation Engineering programme 2000. He is a member of the Royal National Institute for the Blind's Technical Advisory Group. He was a keynote speaker at the Second International Congress on Special Needs Education in Argentina, September 2000. He has acted as a reviewer for research proposals submitted to the Guide Dogs for the Blind Association. He is Joint Technology Editor of the Royal National Institute for the Blind's magazines *Eye Contact* and *Visibility*. In collaboration with Microsoft, he developed the Narrator screen reader that is supplied with Windows 2000.

Prof. Blenkhorn is an Assistant Editor of the IEEE TRANSACTIONS ON REHABILITATION ENGINEERING. He is co-developer of the head-operated mouse that won the Innovation in Education Award at the Education Show 2000.

Luke Crossey received the M.Eng. degree from the Department of Computation, University of Manchester Institute of Science and Technology, Manchester, U.K., in 2003.

Quang Ngo received the M.Eng. degree from the Department of Computation, University of Manchester Institute of Science and Technology, Manchester, U.K., in 2003.

Martin Ross received the M.Eng. degree from the Department of Computation, University of Manchester Institute of Science and Technology, Manchester, U.K., in 2003.

David Werner received the M.Eng. degree from the Department of Computation, University of Manchester Institute of Science and Technology, Manchester, U.K., in 2003.

Christina Wong received the M.Eng. degree from the Department of Computation, University of Manchester Institute of Science and Technology, Manchester, U.K., in 2003.