Vision-based continuous Graffiti™-like text entry system

İ. Aykut Erdem
M. Erkut Erdem
Volkan Atalay
Middle East Technical University
Department of Computer Engineering
Ankara, Turkey
E-mail: aykut@ceng.metu.edu.tr

A. Enis Çetin
Bilkent University
Department of Electrical and Electronics Engineering
Ankara, Turkey

Abstract. It is now possible to design real-time, low-cost computer vision systems even in personal computers due to the recent advances in electronics and the computer industry. Due to this reason, it is feasible to develop computer-vision-based human-computer interaction systems. A vision-based continuous Graffiti™-like text entry system is presented. The user sketches characters in a Graffiti™-like alphabet in a continuous manner on a flat surface using a laser pointer. The beam of the laser pointer is tracked on the image sequences captured by a camera, and the corresponding written word is recognized from the extracted trace of the laser beam.
[DOI: 10.1117/1.1645257]

Subject terms: computer vision; human-computer interaction; wearable computing; Graffiti recognition; tracking.

Paper 030371 received Jul. 30, 2003; revised manuscript received Sep. 24, 2003; accepted for publication Sep. 25, 2003.

1 Introduction

We address the problem of entering ASCII text into a wearable computer or a mobile communication device. Mobile communication and computing devices currently have tiny keyboards that are not easy to use. Furthermore, such keyboards occupy a large part of the screen in tablet computers and touch screen systems. Computer vision may provide alternative, flexible, and versatile ways for humans to communicate with computers. In this approach, the key idea is to monitor the actions of the user by a camera and interpret them in real time. For example, character recognition techniques developed in document analysis[1–3] can be used to recognize handwriting or sketching. In a previous study by Ozer et al.,[1] a vision-based system for recognizing isolated characters is developed, where users draw with a pointer or a stylus on a flat surface or the forearm of a person. The user's actions are captured by a head-mounted camera. To achieve very high recognition rates, characters are restricted to a single-stroke alphabet, like the Graffiti™ alphabet. The Graffiti™ alphabet was first developed by Xerox Corp. and nowadays its variants are used in many handheld computers.

We develop a vision-based continuous Graffiti™-like text entry system as an extension of Ref. 1. In this system, instead of drawing isolated characters, the user sketches the Graffiti™ alphabet in a continuous manner on his or her left arm or on a flat surface using a pointer, stylus, or a finger. In this approach, the alphabet is also based on the Graffiti™ alphabet. However, some letters of the Graffiti™ alphabet have to be modified to increase recognition accuracy. By restricting the alphabet to Graffiti™-like characters, very high recognition rates can be achieved.

The proposed continuous Graffiti™ recognition system can be incorporated into a presentation system as well. In many large auditoriums, the computer containing the presentation material is not on the stage. It is usually very difficult for the speaker to jump to previous or future slides or to extract another document from the computer. The user can mark some keywords or slides before the presentation. During the presentation, he or she can write the keyword on the screen using the laser pointer, and then the system brings the premarked slide or the requested document to the screen.

The organization of the work is as follows: In Sec. 2, the basics of the overall text entry system are presented. The details of tracking and recognition phases are described in Secs. 3 and 4, respectively. The experimental results are given in Sec. 5. The work concludes with Sec. 6, in which the presented study is discussed and future work is stated.

2 Vision-Based Continuous Graffiti™-Like Text Entry System

Unistroke isolated character recognition systems are successfully used in personal digital assistants, in which people feel it is easier to write rather than type on a small-size keyboard.[4,5] In this approach, it is assumed that each character is drawn by a single stroke as an isolated character. One of the alphabets that has this property is the Graffiti™ alphabet. In a study by Ozer et al.,[1] a vision-based system for recognizing isolated Graffiti™ characters is proposed. In this system, the user draws characters by a pointer or a stylus on a flat surface or the forearm of a person. In our study, we extend the work of the isolated Graffiti™ recognition problem, to continuous Graffiti™ recognition. To increase the recognition accuracy of the system, we have modified the original Graffiti™ alphabet. The original Graffiti™ and our modified alphabets can be seen in Figs. 1(a) and 1(b), respectively.

In this handwriting method, the transitions from a character to another are also restricted to the three possible strokes shown in Fig. 2(a). Transition from one character to another can be done with a horizontal line segment, a

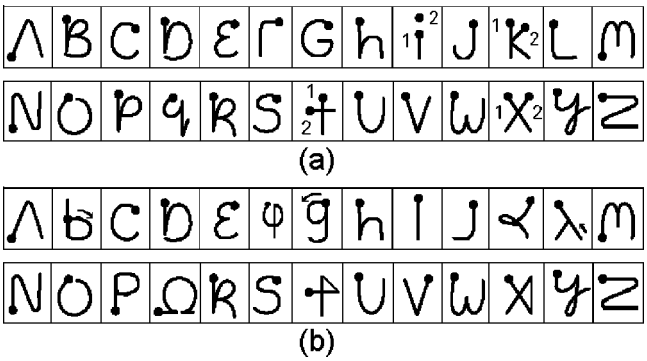


Fig. 1 (a) Original Graffiti™ alphabet and (b) modified alphabet. Heavy dots indicate the starting point.

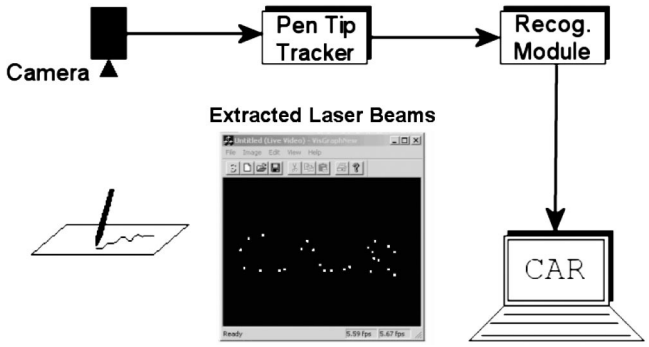


Fig. 3 Overall system architecture of vision-based continuous Graffiti™-like text entry system.

monotonically increasing convex curve, or a monotonically decreasing convex curve. An example word ''team'' is written in continuous Graffiti™ in Fig. 2(b).

In the current system, the user writes in continuous Graffiti™ using a laser pointer on the forearm, captured by a camera mounted on the forehead or a shirt pocket. The video is segmented to image sequences corresponding to each written word. The image sequence starts with a laser pointer turn-on action, and terminates when the user turns off the laser pointer. In each image in this sequence, the beam of the laser pointer is located by the tracker module, and after obtaining these sample points, the recognition module outputs the recognized word. As the overall system architecture shows in Fig. 3, the system is composed of tracking and recognition phases.

The advantages of our vision-based text entry system compared to other vision-based systems[6–8] are as follows.

- The background is controlled by the forearm of the user. Furthermore, if the user wears a unicolor fabric, then the tip of the finger or the beam of the pointer can be detected in each image of the video by a simple image processing operation, such as thresholding.
- It is very easy to learn a Graffiti™-like alphabet. Only a few characters are different from the regular Latin alphabet. Although it may be easy to learn other text entry systems, such as those in Refs. 6, 7, and 9, some people are reluctant to spend a few hours to learn unconventional text entry systems. Furthermore, in addition to the regular characters, other single-stroke characters can be defined by the user to be used as bookmarks, pointers to databases, etc.
- Computationally efficient, low-power-consuming algorithms exist for the recognition of unistroke characters, and they can be implemented in real time with very high recognition accuracy. After a few minutes of studying the Graffiti™-like alphabet, recognition accuracy is very high compared to the regular handwriting recognition method developed by Fink, Wienecke, and Sagerer.[8]
- Computer-vision-based text entry systems are almost weightless.

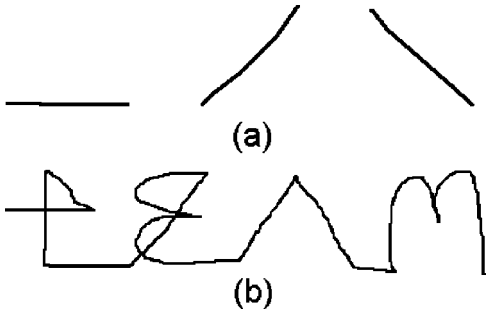


Fig. 2 (a) Character to character transition strokes and (b) word ''team'' written in continuous Graffiti™-like alphabet.

3 Tracking

The beam of the laser pointer is located by detecting the moving pixels in the current image of the video and from the color information. Moving pixels are estimated by taking the image difference of two consecutive image frames. Then by using the fact that the beam of the laser pointer is brighter than its neighbor pixels, the tracking process can be performed in a robust way. By calculating the center of the mass of the bright red pixels among the moving pixels, the position of the beam of the laser pointer is determined. The overall process is shown in Algorithm 1.

Algorithm 1: Finding the position of the beam of the laser pointer. Given two consecutive camera images I_j and I_{j-1}, proceed with the following.

1. Determine the binary difference image I_{diff} between I_j and I_{j-1}.
2. By masking I_{diff} over I_j, form the image I_{mask}.
3. Determine the maximum intensity value $i_{\max}$ over the pixels in I_{mask}.
4. Set the intensity threshold t to $0.9\times i_{\max}$.
5. For all pixels p_j, where $i_{p_j}>t$, calculate the position of the beam of the laser pointer by taking the center of the mass as follows:

$$c_x=\frac{1}{n}\sum_{j=0}^{n} p_{j_x},\quad c_y=\frac{1}{n}\sum_{j=0}^{n} p_{j_y}.$$

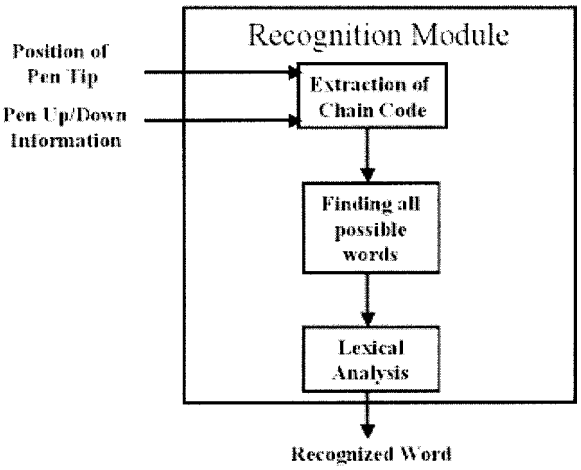


Fig. 4 The inner structure of the recognition module.

Table 1 FSMs for each character in the alphabet.

Character	Corresponding FSM	Character	Corresponding FSM
A	12 76	N	12 7 12
B	107 654 32	O	45 67 012 34
C	345 67	P	2 01 76 54
D	67 12 076 54	Q	234 10 765
E	345 670 345 670	R	6 12 076 54 076
F	456 07 12 345 6	S	345 607 543
G	345 67 012 654 3210	T	0 23 56
H	6 6 210 76	U	6 071 2
I	6 6	V	65 32
J	6 54	W	67 012 670 12
K	65 4321 07	X	7 12 56
L	7 23 5	Y	67 012 654 321
M	21 67 210 76	Z	0 5 0

4 Recognition

As shown in Fig. 4, the position of the pen tip and pen up/down information extracted in the tracking phase is applied as an input to the recognition system. First, the chain code is extracted from the relative motion of the beam of the laser pointer between consecutive camera images. Then, the extracted chain code of the word is analyzed and all possible words conforming the extracted chain code are determined. At the end, by performing a lexical analysis, the recognized word(s) are displayed on the screen.

4.1 *Extraction of Chain Code*

In our system, the unistroke characters are described using a chain code, which is a sequence of numbers between 0 and 7 obtained from the quantized angle of the beam of the laser pointer in an equally time-sampled manner, as shown in Fig. 5(a). A chain-coded representation of characters are generated according to the angle between two consecutive positions of the beam of the laser pointer. A sample chain-coded representation of the character N is shown in Fig. 5(b).

4.2 *Finding All Possible Words*

Each character in the alphabet and transition strokes are all represented by a distinct finite state machine (FSM) (see Table 1). If we have an extracted chain code of a character, we can recognize that character by examining it according to the FSMs representing each character in the alphabet. As an example, in Fig. 5(b), the character N is characterized by the chain code [2,2,2,1,7,7,6,2,2,2], where the finite state machine for the character N is shown in Fig. 6. The first four inputs, 2,2,2, and 1, do not produce any error when applied to the first state of the FSM representing the character N. The next input, 7, makes the FSM to go to the next state and the subsequent 7 lets the machine remain there. The next number of the chain code, 6, leads to an error and an increase in the error counter by 1. Whenever the input becomes 2, the FSM moves to the third state. The machine stays in this state until the end of the chain code, and the FSM terminates with an error value of 1. When we extend this analysis over all FSMs, we come up with the character recognition algorithm shown in Algorithm 2.

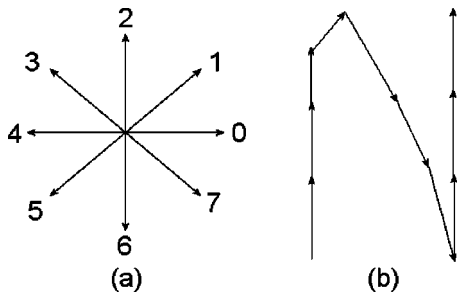


Fig. 5 (a) Chain code values for the angles. (b) A sample chain-coded representation of the character N is [2,2,2,1,7,7,6,2,2,2].

Algorithm 2: Character recognition algorithm based on analysis using FSMs. Given the extracted chain code of a character, proceed the following.

1. The chain code is applied as input to all FSMs representing each character.
2. State changes are determined, and additionally, an error counter is increased by 1 if a change is not possible according to the current FSM.
3. If a chain code does not terminate in the final state, the corresponding character is eliminated.
4. Errors in each state are added up to find the final error for each character.
5. Character with the minimum error is the recognized one.

As can be observed from Table 1, FSMs are different for each character in the alphabet. However, for some extracted

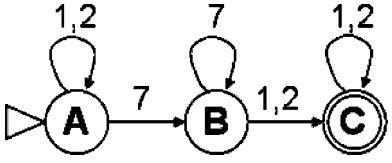


Fig. 6 Finite state machine for the character N.

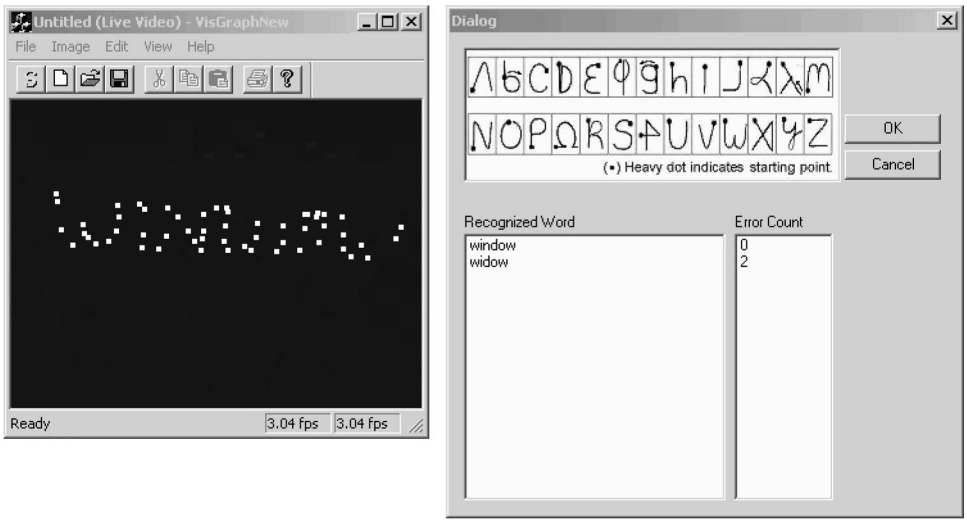


Fig. 7 The result of lexical analysis for the written word "window."

chain codes of the written characters, some FSMs can output close error counts. For example, for the input chain code [6,6,4], while the FSM for character J outputs an error count 0, the FSM for character I outputs an error count 1. This may generate a confusion between characters J and I. Similarly, while writing the character E, the FSM for the character G outputs a low error count. This is also the case for the characters J and I, S and C, U and W, W and Y, and X and L. The main reason for these confusions is that the FSMs are constructed to be tolerant enough of different user writings for alternative chain codes. However, this is corrected by introducing a lexical analysis step at the end.

It is preferred that a word be segmented into characters by examining the transition strokes. But in general, this may not be possible, since these detected transition strokes can also be a substroke of a character. Therefore, our recognition module works in a recursive manner and outputs all possible words of the extracted chain code. As described before, each FSM representing a character returns an error value: the ones having minimum errors are selected, and for each one, the next chain-code inputs will be passed to all the FSMs for the next character. This process continues until the end of the chain code is reached. The segmentation problem can also be solved at the lexical analysis step similar to the confusion issue discussed in the previous paragraph.

It is observed that the FSM-based recognition algorithm is robust as long as the user does not move his arm or the camera during the writing process of a letter. Characters can be also modeled by hidden Markov models, which are stochastic FSMs instead of deterministic FSMs, to further increase the robustness of the system at the expense of higher computational cost. In addition, to prevent noisy state changes, look-ahead tokens can be used that act as a smoothing filter on the chain code.

4.3 *Lexical Analysis*

At the end of the step described in Sec. 4.2, a list of all possible words is obtained. In the lexical analysis step, the meaningless words are eliminated by looking up a 18,000 word dictionary, which is composed of the most common English words. In the end, only the words found in the dictionary are displayed as the recognized ones in sorted order, according to their total error count. This can be seen in Fig. 7.

5 Experimental Results

In our experiments, we have a computer with an Intel Pentium IV 1.7-Ghz processor with 512-GB memory, a webcam producing 320×240-pixel color images at 13.3 frames/s, and an ordinary laser pointer. The user draws continuous Graffiti™ characters using the laser pointer on the dark background material. In Graffiti™-like recognition systems, very high recognition rates are possible.[5]

To examine the performance of our system, the system is tested with a word dataset consisting of 30 words in various lengths. These words are written at least 15 times by different people. In our system, in spite of the existence

Table 2 The words in the test set and corresponding recognition rates.

Word	Recognition rate(%)	Word	Recognition rate(%)
she	100.00	agree	94.44
car	100.00	queue	93.75
tin	100.00	three	90.00
road	100.00	money	92.00
kind	90.00	model	84.00
bird	100.00	future	95.00
them	89.47	vision	85.00
word	100.00	window	100.00
book	93.75	liquid	100.00
sand	100.00	engine	100.00
jazz	93.75	desire	100.00
nine	100.00	problem	75.00
find	93.75	science	80.00
twin	100.00	subject	80.00
crazy	85.00	lexical	90.00

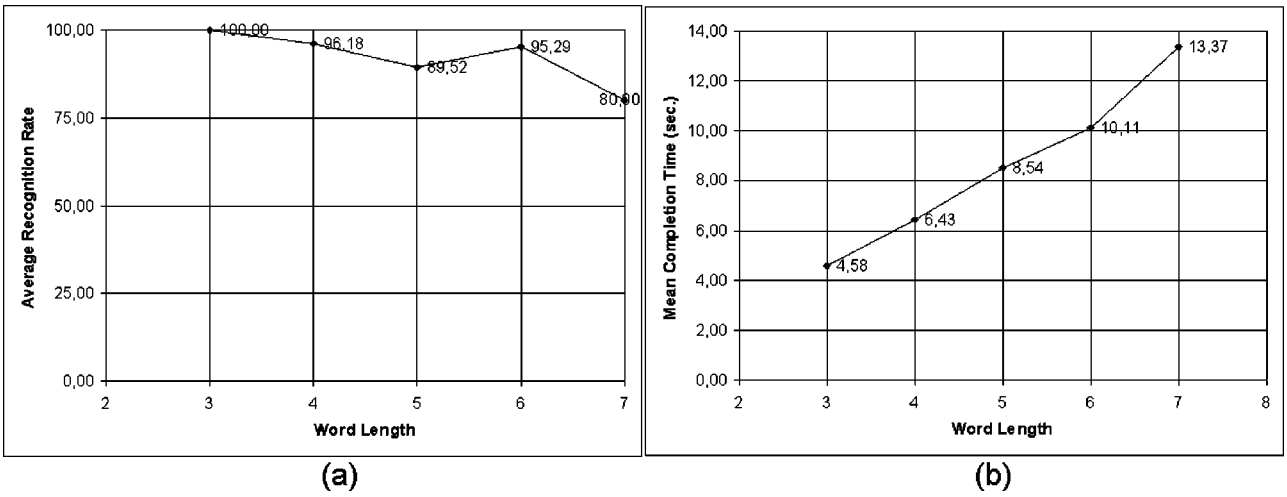


Fig. 8 (a) Recognition rate versus word length. (b) Mean completion time versus word length.

of perspective distortion, it is possible to attain a recognition rate of 93% at the word level. The words in the test set and corresponding recognition rates are listed in Table 2. Additionally, according to experiments, the average writing speed is calculated as 8 words per minute (wpm). Actually, there is a trade-off between writing speed and the recognition rate. Since the whole process depends on the CPU power of the computer and the frame rate of the webcam, if the user writes quickly, the extracted chain code may not be fully correct due to the frame losses, and consequently, this directly affects the recognition. Due to this trade-off, the size of written characters, and therefore the written word, must be big enough. In this case, only two to three words can be written in the viewing area of the camera. However, we believe that the effect of this trade-off can be minimized with the improvements in current hardware.

In addition, when we examine the recognition rate versus word length graph shown in Fig. 8(a), we can infer that although the word length has an importance, the recognition rate is not directly related with word length. Furthermore, the mean completion time of the written word versus the word length graph, which is given in Fig. 8(b), shows that the writing time increases linearly with the increase in word length.

It is also observed that the recognition process is writer independent with little training, and we believe that we can achieve higher writing speed rates with advances in digital camera and wearable computer technology. The perspective distortion plays some role in the recognition accuracy of the system. In our experiments, we have observed that the degradation in recognition is at most 10% around 30 deg differences between the plane on the which writing is performed and the camera.

Several tests are also carried out under different lighting conditions. In day/incandescent/fluorescent light, the average intensity of the background is about 50/180/100, whereas the intensity value of the beam of the laser pointer is about 240/250/240. In all cases, the beam of the laser pointer can be easily identified from the dark background.

6 Conclusion

In this study, we present a vision-based continuous Graffiti™-like text entry system. A Graffiti™-like alphabet is developed, where the users can write characters in a continuous manner on a flat surface using the laser pointer. This alphabet can be easily extended by defining finite state machines for each newly added character. The video is segmented to image sequences corresponding to each written word. Every image sequence starts with a laser pointer turn-on action, and ends when the user turns off the laser pointer. In each image in this sequence, the beam of the laser pointer is tracked, and the written word is recognized from the extracted trace of the laser beam. Recognition is based on finite state machine representations of characters in the alphabet.

According to the experiments, the recognition rate of our vision-based Graffiti™-like text entry system is measured as 93% at the word level, and the writing speed as around 8 wpm. It is also observed that the system is writer independent and requires little training for learning the alphabet. Also, the writing time increases linearly with the increase in word length.

Since we use the laser pointer as the pointing device, tracking the beam in real time is not a complicated process. As future work, the possibility of using some other pointing devices (e.g., finger, ordinary pen, etc.) can be investigated. But at this time, to track the tips of these pointers, some complex feature trackers (e.g., Kanade-Lucas-Tomasi (KLT) point-based feature tracker[10]) in combination with a Kalman filter[11] can be used.

Acknowledgment

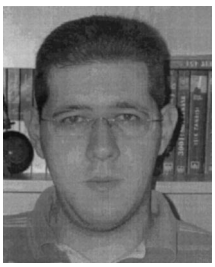
A. Enis Çetin's research is supported in part by the Turkish Academy of Sciences.

References

1. O. F. Ozer, O. Ozun, V. Atalay, and A. E. Cetin, "Visgraph: Vision based single stroke character recognition for wearable computing," *IEEE Intell. Syst. Appl.* **16**, 33–37 (May–June 2001).
2. O. Gerek, A. Cetin, A. Tewfik, and V. Atalay, "Subband domain cod-

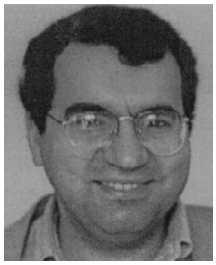
ing of binary textual images for document archiving,'' *IEEE Trans. Image Process.* **8**, 1438–1446 (Oct. 1999).
3. M. Munich and P. Perona, ''Visual input for pen-based computers,'' *Proc. 13th Intl. Conf. Patt. Recog.*, pp. 33–37 (1996).
4. D. Goldberg and C. Richardson, ''Touch-typing with a stylus,'' *Proc. INTERCHI'93 Conf. Human Factors Computing Syst.*, pp. 80–87 (1993).
5. I. MacKenzie and S. Zhang, ''The immediate usability of graffiti,'' *Proc. Graphics Interface'97*, pp. 129–137 (1997).
6. J. A. R. A. Vardy and L. T. Cheng, ''The wristcam as input device,'' *Proc. 3rd Intl. Symp. Wearable Comput.*, pp. 199–202 (Oct. 1999).
7. T. Starner, J. Weaver, and A. Pentland, ''A wearable computing based american sign language recognizer,'' *Proc. 1st Intl. Symp. Wearable Comput.* (Oct. 1997).
8. G. A. Fink, M. Wienecke, and G. Sagerer, ''Video-based on-line handwriting recognition,'' *IEEE Proc. Intl. Conf. Document Anal. Recog.*, pp. 226–230 (2001).
9. See http://www.handykey.com as accessed on.
10. C. Tomasi and T. Kanade, ''Detection and tracking of point features,'' Tech. Rep. CMU-CS-91132, Carnegie Mellon Univ. School of Computer Sci., Pittsburgh, PA (1991).
11. R. E. Kalman, ''A new approach to linear filtering and prediction problems,'' *Trans. ASME J. Basic Eng.* **82** (Series D), 35–45 (1960).

İ. Aykut Erdem is currently a PhD student and a research assistant in the Department of Computer Engineering at Middle East Technical University, Ankara, Turkey. He received his BSc and MSc degrees in computer engineering from Middle East Technical University in 2001 and 2003, respectively. His research interests include computer vision, computer graphics, and pattern recognition. He is a member of the IEEE Computer Society and the Turkish Pattern Recognition and Image Analysis Society.



M. Erkut Erdem is a doctoral candidate and a research assistant in the Department of Computer Engineering at Middle East Technical University, Ankara, Turkey. He received his BSc and MSc degrees in computer engineering from Middle East Technical University in 2001 and 2003, respectively. His research interests include computer vision, computer graphics, and pattern recognition. He is a member of the IEEE Computer Society and the Turkish Pattern Recognition and Image Analysis Society.

Volkan Atalay is an associate professor of computer engineering at the Middle East Technical University. Previously, he was a visiting scholar at the New Jersey Institute of Technology. He received a BSc and MSc in electrical engineering from Middle East Technical University, and a PhD in computer science from the Université René Paris, France. His research interests include computer vision, document analysis, and applications of neural networks. He is a member of the IEEE Computer Society and the Turkish Pattern Recognition and Image Analysis Society.



A. Enis Çetin studied electrical engineering at the Middle East Technical University. After getting his BSc degree, he got his MSE and PhD degrees in systems engineering from the Moore School of Electrical Engineering at the University of Pennsylvania in Philadelphia. Between 1987 to 1989, he was an assistant professor of electrical engineering at the University of Toronto, Canada. Since then he has been with Bilkent University, Ankara, Turkey. Currently he is a full professor. During the summers of 1988, 1991, and 1992 he was with Bell Communications Research (Bellcore), New Jersey. He spent the 1994 to 1995 academic year at Koc University in Istanbul, and the 1996 to 1997 academic year at the University of Minnesota, Minneapolis, as a visiting associate professor. He is an Associate Editor of the *IEEE Transactions on Image Processing*, and a member of the DSP technical committee of the IEEE Circuits and Systems Society. He founded the Turkish Chapter of the IEEE Signal Processing Society in 1991. He is currently Signal Processing and AES Chapter Coordinator in IEEE Region 8. He is a senior member of IEEE and EURASIP. He received the young scientist award of the Turkish Scientific and Technical Research Council in 1993. He was the chair of the IEEE-EURASIP Nonlinear Signal and Image Processing Workshop (NSIP'99), which was held in June 1999 in Antalya, Turkey.