

Text-Tracking Wearable Camera System for the Blind

Hideaki Goto
Cyberscience Center
Tohoku University, Japan
hgot @ isc.tohoku.ac.jp

Makoto Tanaka
Graduate School of Information Sciences
Tohoku University, Japan

Abstract

Disability of visual text reading has a huge impact on the quality of life for visually disabled people. One of the most anticipated devices is a wearable camera capable of finding text regions in natural scenes and translating the text into another representation such as speech or braille. In order to develop such a device, text tracking in video sequences is required as well as text detection. The device needs to group homogeneous text regions to avoid multiple and redundant speech syntheses or braille conversions. An automatic text image selection is also required for better character recognition and timely text message presentation.

We have developed a prototype system equipped with a head-mounted video camera. Particle filter is employed for fast and robust text tracking. We have tested the performance of our system using 1,730 video frames of hall ways with 27 signboards. The number of text candidate regions is reduced to 1.47%.

1. Introduction

We human beings make the most of text information in surrounding scenes in our daily lives. Disability of visual text reading has a huge impact on the quality of life for visually disabled people. Although there have been several devices designed for helping visually-impaired people to “see” objects using an alternative sense such as sound and touch, the development of text reading devices is still at an early stage. One of the most anticipated devices is probably a wearable camera capable of finding text regions in natural scenes and translating the text into another representation such as speech or braille.

Another device is a helper robot that can recognize characters in human living spaces and read out the text information for the user. Some robots with character recognition capability have been proposed so far [2, 3, 4, 6, 8]. Iwatsuka et al. proposed a guide dog system for blind people [2]. We developed a text capturing robot equipped with an active

camera [8]. In the robot applications, camera movement is constrained by some simple and steady robot movements. On the other hand, the movement of a wearable camera is basically unpredictable. A robust text tracking method is required to develop a text capturing device.

Some duplicate text strings may appear in the consecutive video frames. Recognizing all the text strings in the images is computationally impractical. More importantly, the camera user would not want to hear repeatedly a synthesized voice originated from the same text. Merino and Mirmehdi presented a framework for real-time text detection and tracking and demonstrated a system [5]. We also presented a wearable camera system with text tracking capability [9]. The text tracking performance was not so satisfactory and a lot of improvements are still needed. In addition, the timing of the text message presentation is very important on a wearable system. In many situations, the camera user want to hear the text messages while he/she is close enough to a signboard and before passing by it. The system should have a capability of selecting a text image good enough for character recognition at an appropriate moment.

In this paper, we present an improved wearable camera system with automatic text image selection. The system is equipped with a head-mounted video camera. The text strings are extracted using the revised DCT-based method [1]. The text regions are then grouped into image chains by a text tracking method based on particle filter [9].

In Section 2, we present the overview of our camera system and the algorithms used. In Section 3, the text tracking algorithm is given. The text image selection and filtering methods are described in Section 4. Section 5 describes experimental results and performance evaluations.

2. Wearable Camera System

Figure 1 shows the prototype of the wearable camera system which we have constructed. The system consists of a head-mount camera (380k-pixel color CCD), an NTSC-DV converter, and a laptop PC with Core2Duo 1.06GHz proces-



Figure 1. Wearable Camera System.

sor running Linux. The NTSC video signal is converted to DV stream and the video frames are captured at 352×224 -pixel resolution. The processes in the proposed system are as follows.

1. Partition each frame into 16×16 -pixel blocks.
2. Extract text blocks using revised DCT feature.
3. Generate text candidate regions by merging adjoining text blocks.
4. Text tracking : Create chains (groups) of text regions by finding the correspondences of the text regions between the $(n - 1)$ th frame and the n th frame using particle filter.
5. Filter out non-text regions in the chains.
6. Select some text images that appear to be good for character recognition in each chain.

Our current system is not equipped with a character recognition process, since we are concentrating only on text detection, tracking, and image selection.

We have employed the revised DCT-based feature proposed in [1] for text extraction from scene images. High-wide (HWide) frequency band of the DCT coefficient matrix is used. The discriminant analysis-based thresholding is also used. The blocks in the image are classified into two classes; “text blocks” whose feature values are greater than the automatically found threshold, and “non-text blocks.”

After extracting the text blocks, connected components of the blocks are generated. The bounding box of each connected component is regarded as a text region. The text regions whose area is smaller than four blocks are discarded, because they are considered to be too small for legible text strings.

3. Text Tracking Using Particle Filter

3.1. Extraction of text region chains

Text region chain is a group of text regions that seem to be originated from the same text. A new chain is created

when a text candidate region without any correspondence appears in the current frame. A new label is assigned to both the chain and the region. When a chain of the text candidate regions is broken in a frame, the chain is terminated and its validity is examined. If the chain is shorter than 3 frames, it is regarded as noise and discarded.

3.2. Block-based text region matching

We have employed particle filter for the tracking of text candidate regions.

The particles are scattered around the predicted center point of text block in the current frame from the center of each text block in the previous frame as shown in Figure 2. The particles that fall outside any of the text blocks have their weight set to zero. If a particle falls into a text block, its weight is set to the similarity value calculated between the previous and the current text blocks. The text region inherits the label from the text image chain which the region belongs to. In the same way, the source text block inherits the label from the text region which the block belongs to. All the scattered particles will have the same label as that of the source text block.

The similarity $s_{1,2}$ between text blocks 1 and 2 is defined by

$$s_{1,2} = \frac{1}{d_{1,2} + \varepsilon}, \quad (1)$$

where $d_{1,2}$ is the distance defined as follows and ε is a small value ($\varepsilon = 1$) to avoid divergence.

Cumulative histogram is used as a measure to evaluate the dissimilarity between text blocks, since it can represent color distribution with small computational cost. The cumulative histogram $H(z)$ is given by

$$H(z) = \sum_{i=0}^z h(i) \quad (2)$$

where $h(i)$ denotes the normal histogram and i denotes the intensity. For comparing two cumulative histograms $H_{1,c}(z)$ and $H_{2,c}(z)$, where c denotes one of RGB color channels, the following city block distance is used.

$$d_{1,2} = \sum_c \sum_{z=0}^{255} |H_{1,c}(z) - H_{2,c}(z)| \quad (3)$$

Weighted centers of the particles are found in the current frame. The label of the current text block is determined by finding the weighted center nearest to the center of the block.

The region label in the current frame is determined by voting as shown in Figure 3. The most popular label is found and assigned to the text region. Thus, the chain is extended.

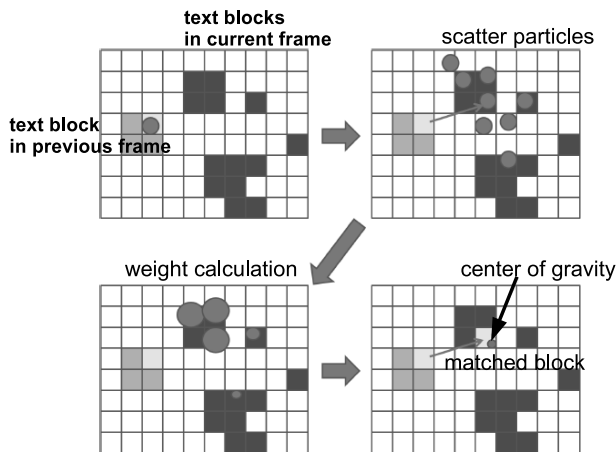


Figure 2. Block-based text tracking.

	2	2	1	1
2	2	2	2	1
3	2	2		2
	3	2	2	2

region label = 2

Figure 3. Label selection by voting in block-based text tracking.

The variance of particle’s random walk is set to 13.0. The number of particles per block is 500. Only the velocity is taken into account in the particle filter.

3.3. Region merging process

In real applications, we often see undesirable splits of text blocks as shown in Figure 4. If such a temporal split occurs, the text region chain breaks up and more than one chains are detected for the same text.

We have introduced a simple region merging process to reduce the splits of chains. If the side-to-side distance between the text regions is lower than a pre-determined threshold $\alpha = 1$ (block) or the regions overlap each other, they are merged together. Otherwise a new label is assigned to the split region which is the most distant from the corresponding region in the previous frame. This simple region merging process makes the text tracking more robust to the noisy output of the text detector.

4. Text image selection and filtering

All the text candidate regions belonging to the chains are examined and the images that seem to be non-text are re-

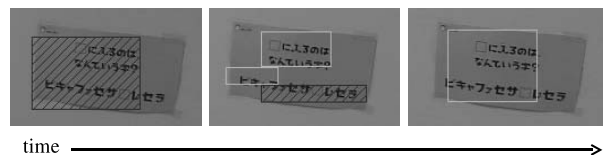


Figure 4. Temporary split of text regions.

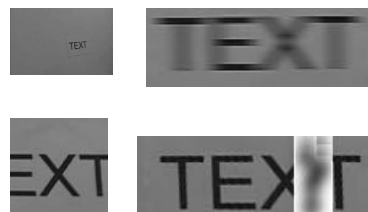


Figure 5. Inappropriate text images.

moved using the edge counts method described in [7].

The captured image sequence may contain a lot of text images inappropriate for character recognition as shown in Figure 5. Some characters are too small because the camera is not yet close enough. Some text regions may be too small due to a failure of text region detection. A part of text image may be hidden by obstacles. Some images may be blurred by camera motion.

It is necessary to pick up text images good enough for character recognition. A good text image is the one that is large and clear enough, without any obstacle, and taken in as fronto-parallel as possible. At least one text image should be extracted from each chain, and fed to the character recognition stage. We examine the following six features to evaluate the goodness of text images.

- F_{area} : Area of text region.
- F_{width} : Width of text region. (Used in our previous work [9].)
- F_{FDR} : FDR (Fisher’s Discriminant Ratio) calculated when Otsu’s binarization is applied to the text region image. This value is expected to reflect the clearness of the edges.
- F_{Sobel} : Sum of the absolute values of the vertical components of Sobel edge detector divided by the image area. This value shows the average intensity of the vertical edges.
- $F_{\text{edgecount}}$: The number of vertical edges in the binarized image. This value is expected to drop when the text image is too small or unclear.
- $F_{\text{edgeintensity}}$: Sum of the absolute values of the vertical edge intensity defined as follows. Let W and H

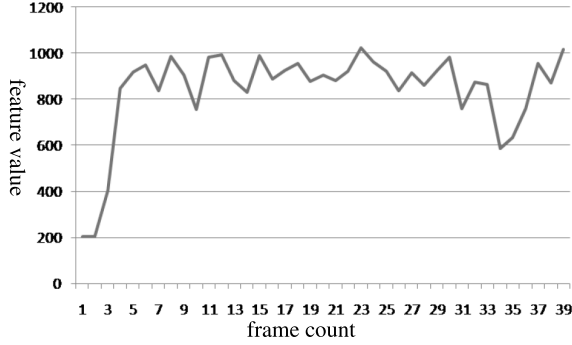


Figure 6. Temporal change of a feature value.

be the width and the height of the image, respectively.

$$F_{\text{edgeintensity}} = \begin{cases} \sum_x \sum_y E_H(x, y) & \text{if } W \geq H \\ \sum_x \sum_y E_V(x, y) & \text{otherwise} \end{cases} \quad (4)$$

where

$$E_H(x, y) = \begin{cases} 0 & \dots \text{ if } b(x, y) = b(x-1, y) \\ |p(x, y) - p(x-1, y)| & \dots \text{ otherwise} \end{cases} \quad (5)$$

$$E_V(x, y) = \begin{cases} 0 & \dots \text{ if } b(x, y) = b(x, y-1) \\ |p(x, y) - p(x, y-1)| & \dots \text{ otherwise} \end{cases} \quad (6)$$

where $p(x, y)$ is the pixel value of the image and $b(x, y)$ is the pixel value of the binarized image.

Since this feature is not normalized by the image area, it reflects the image size as well as the edge intensity.

Figure 6 shows an example of the temporal change of a feature value. In our previous system, the image with the highest F_{width} is detected after the chain has been terminated, i.e. the text has disappeared from the camera's view. There is a delay in message presentation in this case, and the user will not hear the text message as long as a signboard stays in the camera view. We have changed the image selection algorithm to avoid such a situation. The text image is picked up and passed to the next process stage when the feature value is at a local maximum and it is the highest among the past peaks in a chain.

There could still be a long delay while the user is steadily approaching to a signboard. To deal with the problem, the system picks up the current text image at every two seconds in addition to the above mentioned image selection process.



Figure 7. Results of text region tracking.

5. Experimental Results and Discussions

5.1. Text tracking performance

Experimental images were taken at the hall ways in our building. Eight signboards exist on the walls in Scene 1. In Scene 2, there are 19 signboards. A sighted person wore the camera, and approached every signboard. 1,000 video frames were obtained in total in Scene 1, and 730 frames in Scene 2.

Consequently, 5,192 text candidate regions were found in Scene 1 using the DCT feature. After the text region tracking, the total number of text images to be passed to the character recognition stage has been significantly reduced to 66. Thus, the number has been cut down to 1.27%. In Scene 2, the total number of regions is 4,503. Table 1 shows the numbers of extracted text chains and average precessing times per frame. The region merging process works effectively and the proposed method outperforms our previous method [9]. Some text tracking results are shown in Figure 7. The text regions are successfully detected and tracked well.

The average processing speed in Scene 2 without DV decoding is 9.3fps (= 1/108msec), which is near real-time. Some improvements in the implementation may be possible.

Although the proposed method performs the best so far,

Table 1. Numbers of detected text chains and processing times.

	Scene 1		Scene 2	
	chains	time	chains	time
Previous [9]	86	81	85	105
Proposed method	66 (1.27%)	82	77 (1.71%)	108

msec

Table 2. OCR accuracy (%) for each text selection feature.

	linear scan	distance change	pan
F_{area}	90	66	50
F_{width}	100	66	70
F_{FDR}	70	66	30
F_{Sobel}	30	66	20
$F_{\text{edgecount}}$	90	100	70
$F_{\text{edgeintensity}}$	90	100	70

a lot of duplicate or non-text images can be seen. The value 66 is about 8.3 times of 8 which is the ideal number of text images. A lot of splits of text region chains can be still observed. Further improvements will be needed.

5.2. Text image selection performance

To find the best text selection feature, we compared the character recognition accuracy. In this experiment, we put a signboard on the wall and captured the images in three different camera movements. As the first case, the camera is linearly moved from the left of the signboard to the right. As the second case, the camera is moved forward and backward in front of the signboard. We examine whether the feature is useful for preventing the system from taking small text images by mistake. As the third case, the camera is horizontally panned. This movement is for introducing some motion blur. The automatically selected text images are fed to a commercial OCR engine, Panasonic Yomitori-Kakumei 9.

Table 2 shows the results. Although F_{width} in [9] is the best in the linear scan, the feature allows the system to pick up a lot of small text images. $F_{\text{edgecount}}$ and $F_{\text{edgeintensity}}$ are the best among the six features. Further analysis is needed to see the difference between these two features.

6. Conclusions

We have presented a wearable camera system capable of automatically finding and tracking text regions in surrounding scenes in near real-time. The proposed text tracking method is based on particle filter and can effectively reduce the number of text images to be recognized. In the experiment in indoor scenes, our system could reduce the number of text candidate images down to 1.47% on average. The text image selection method with an edge-based feature works fine.

Making the text detection and tracking more robust to quick camera movements, and finding an efficient and user-friendly method for message presentation are included in our future work.

References

- [1] H. Goto. Redefining the DCT-based feature for scene text detection — Analysis and comparison of spatial frequency-based features. *International Journal on Document Analysis and Recognition (IJ DAR)*, 11(1):1–8, 2008.
- [2] K. Iwatsuka, K. Yamamoto, and K. Kato. Development of a guide dog system for the blind people with character recognition ability. *Proceedings 17th International Conference on Pattern Recognition*, pages 683–686, 2004.
- [3] D. Létourneau, F. Michaud, and J.-M. Valin. Autonomous Mobile Robot That Can Read. *EURASIP Journal on Applied Signal Processing*, 17:2650–2662, 2004.
- [4] D. Létourneau, F. Michaud, J.-M. Valin, and C. Proulx. Textual message read by a mobile robot. *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2724–2729, 2003.
- [5] C. Merino and M. Mirmehdi. A Framework Towards Real-time Detection and Tracking of Text. *Second International Workshop on Camera-Based Document Analysis and Recognition (CBDAR2007)*, pages 10–17, 2007.
- [6] J. Samarabandu and X. Liu. An Edge-based Text Region Extraction Algorithm for Indoor Mobile Robot Navigation. *International Journal of Signal Processing*, 3(4):273–280, 2006.
- [7] H. Shiratori, H. Goto, and H. Kobayashi. An Efficient Text Capture Method for Moving Robots Using DCT Feature and Text Tracking. *Proceedings 18th International Conference on Pattern Recognition (ICPR2006)*, 2:1050–1053, 2006.
- [8] M. Tanaka and H. Goto. Autonomous Text Capturing Robot Using Improved DCT Feature and Text Tracking. *9th International Conference on Document Analysis and Recognition (ICDAR2007) Volume II*, pages 1178–1182, 2007.
- [9] M. Tanaka and H. Goto. Text-Tracking Wearable Camera System for Visually-Impaired People. *Proceedings 19th International Conference on Pattern Recognition (ICPR2008)*, 2008.