Intellectual Two-sided Card Copy

Ilia Safonov[a], Hokeun Lee[b], Sangho Kim[b], Donchul Choi[b]
[a]Samsung Research Center, Moscow, Russia
[b]Samsung Electronics, Suwon, Korea
ilia dot safonov at samsung dot com

Abstract

The paper is devoted to intellectual algorithm for processing of scanned images of various two-sided cards such as ID, bank, business cards, etc. Card images significantly differ from common document images and conventional approaches developed for text document image processing do not operate well on card images. Our technique provides a sophisticated arrangement of both images of a card, front and back, for single-page and duplex printing. The method detects location, skew angle and orientation of card image. Orientation detection is based on Arabic numerals segmentation and recognition. Recognition of numerals on card images is a challenging problem, since often cards have complex color background; digits can have arbitrary color and typeface. For digits segmentation on complex background we apply labeling of connected edge regions, whereas edge pixels are labeled for dark numerals on light background and *vice versa* independently from each other. We construct a decision tree with a great generalization capability for recognition of each numeral; it improves recognition for various typefaces. We propose a final decision rule for image orientation detection based on recognition outcomes.

Our algorithm operates very fast and outperforms OCR applications ReadIRIS and FineReader in numerals recognition and orientation detection for card images. Also our algorithm works well enough for majority of originals.

Keywords: *orientation detection, segmentation, numerals recognition.*

1. INTRODUCTION

Frequently consumers copy various two-sided card-size originals such as identity cards, passports, bank cards, certificates, charge cards, business cards, etc. Samsung Electronics has implemented a special “ID Copy” function in MFP devices since 2004 year. User places one card side on scanner glass that corresponds to top half of A4 page and presses copy button. Image of card side is scanned and stored to MFP memory. Further user places another card side on scanner glass and presses copy button again. Image of another card side is scanned. Page is rendered from the two images of both card sides. The function operates well enough for all types of originals.

However existing “ID Copy” function has some limitations. For example, images of card sides are copied without any correction of skew and orientation. Each side can have a skew, and copy looks unattractive. Moreover it can appear in opposite orientations or one side in landscape and other side in portrait orientation. So, our aim is creation of an “Intellectual ID Copy” algorithm providing a sophisticated arrangement of images of both sides for single-page and duplex printing. The method should perform parallel translation and deskew depending on orientation of card image. Fig. 1 illustrates intellectual two-sided card copy. Two scanning passes provide images of front and back sides of card in arbitrary orientation. On a copy both sides are placed upright according to predefined layout.

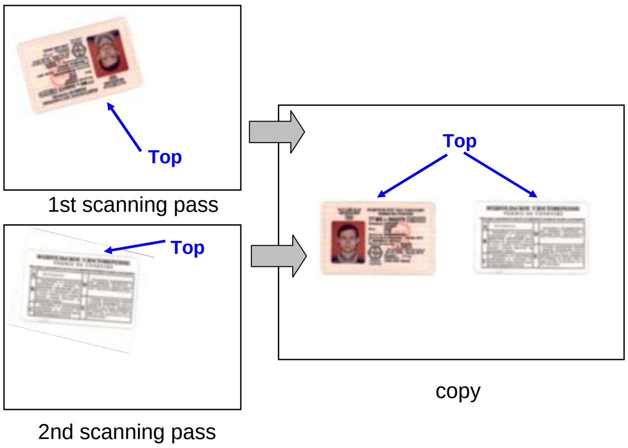

Figure 1: Illustration of *Intellectual ID Copy* function.

There are strict limitations on computational complexity of an algorithm and memory size because it is intended for embedded systems on ARM and MIPS-based SoC. Several hundreds kilobytes of RAM are available only; and all image transformation procedures should be performed *in situ*.

2. WAYS FOR ORIENTATION DETECTION

Key problem of intellectual copy is orientation detection for 4 possible orientations. There are a lot of publications devoted to this problem for various types of images. Initially we collected test set from about 300 card images and tried various approaches for orientation detection.

Some existing techniques on orientation detection, for example, [9, 10], are aimed at photographic images and strongly depend on background of the card, therefore in general case they are unfeasible for card images Another approach could use facial images for orientation detection, since almost all ID cards have face photo. However, face detector from OpenCV produced a lot of false positives in our ID cards images in various orientations. Additional conditions on skin tones and face size as in [3] allowed to decrease significantly number of FP, but several faces were detected in upright and upturned orientation identically. Fig. 2 demonstrates example of face detection for ID card rotated by 180 degrees.

Paper [1] describes computationally effective algorithm for orientation detection of text document images. The algorithm exploits an up/down asymmetry of text composed of roman letters and Arabic numerals. However, majority of card images has complex color background; therefore detection quality of the algorithm is low. In addition language origin significantly influences its detection quality even on uniform background.

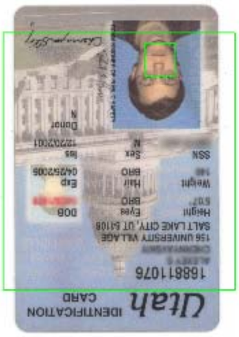

Figure 2: Example of face detection for upturned card.

OCR applications such as ReadIRIS and FineReader can detect image orientation based on results of characters recognition for all possible orientations. But presently embedding of full-functional OCR for many languages in devices for low- and middle-end markets is unfeasible. However, implementation of some reductive character recognition procedure makes sense, since, as it can be noticed, absolute majority of cards contains several Arabic numerals.

We can recognize the following numerals which are rotation noninvariant: '1', '2', '3', '4', '5', '7'. Of course, such approach has some limitations. For example, sometimes cards do not contain any numerals or have rotation invariant numerals only. Also, cards can have numerals in mutually perpendicular directions. Fortunately, these are very rare cases: in our test set only about 1% of images belong to these cases. So, we selected Arabic numerals as the main attribute for card image orientation recognition.

3. PROCESSING OF CARD IMAGES

3.1 General workflow

Front and back sides of card are processed identically. Fig. 3 demonstrates general processing pipeline for image of one side. First of all, bounding box is detected. We slightly modified traditional robust approach for RoI segmentation that employs analysis of projections on horizontal and vertical axes [6]. The next step is detection of skew-angle. Straightforward techniques for skew-angle detection is to use projection profile analysis or Hough transform. Comprehensive survey of skew-angle detection problem was presented in [4]. Unfortunately, well-known approaches do not work well enough for card images which have complex color background. So, we proposed our own skew-angle detection algorithm that is discussed in separate paper [7]. Third step is deskew and translation by means of in place affine transformations [2]. Orientation detection comprises numerals segmentation, recognition for four directions and making decision basing on recognition results. Finally, image is rotated if it is necessary.

3.2 Bounding Box detection

Projections on horizontal and vertical axes are computed according to the following formula:

$$Ph(c) = \sum_{r=1}^{Nr} BW(r,c); \quad Pv(r) = \sum_{c=1}^{Nc} BW(r,c),$$

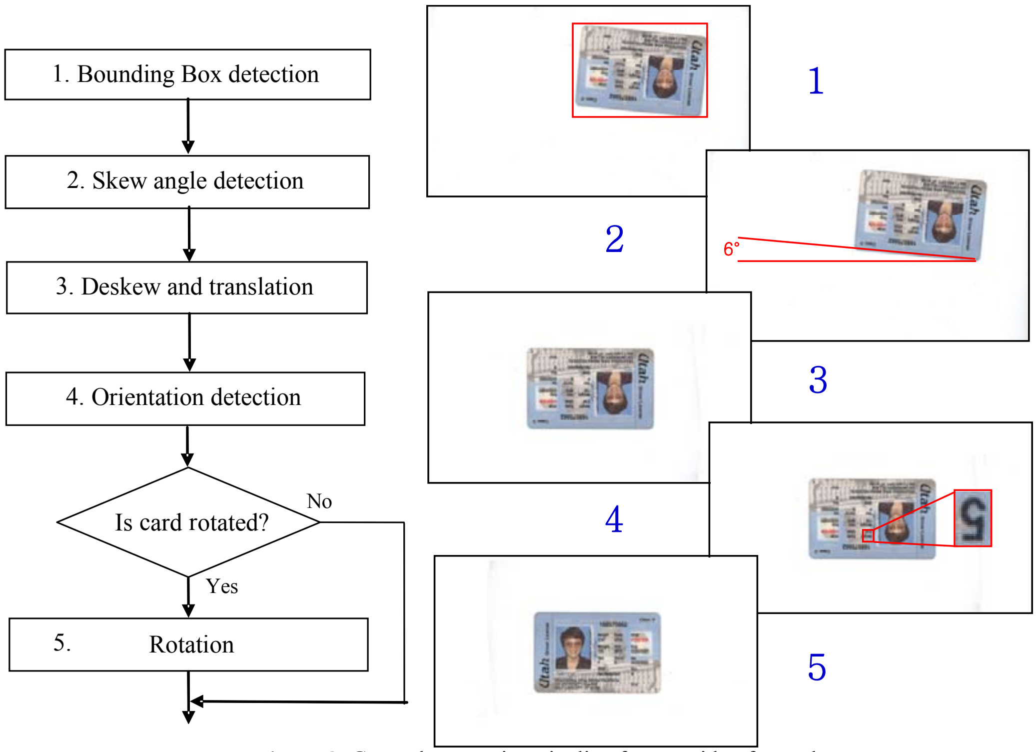

Figure 3. General processing pipeline for one side of a card.

where (r,c) are pixel coordinates, Nr is number of rows, Nc is number of columns, BW is binary image that is the result of scanned image thresholding. On this stage we use fixed threshold because we precisely know scanner characteristics. In order to suppress some noise peaks we apply 1D erosion filter to both projections:

$$Phf(c) = \min(Ph(c-2), Ph(c-1), Ph(c), Ph(c+1), Ph(c+2)),$$
$$Pvf(r) = \min(Pv(r-2), Pv(r-1), Pv(r), Pv(r+1), Pv(r+2)),$$

Bounding box coordinates are calculated as follows:

$$r_{\min} = \min_{\forall r}(Nr - h, r \mid Pvf(r) > Nc/k),$$

$$r_{\max} = \max_{\forall r}(h, r \mid Pvf(r) > Nc/k),$$

$$c_{\min} = \min_{\forall c}(Nc - h, c \mid Phf(c) > Nr/k),$$

$$c_{\max} = \max_{\forall c}(h, c \mid Phf(r) > Nr/k),$$

where coefficient k depends on scanning resolution, $h>0$ is introduced to prevent detection of possible shadow areas near platen boundaries.

We implemented two additional functions based on bounding box coordinates: blank page detection and image clipping detection: If $r_{\min} > r_{\max}$ or $c_{\min} > c_{\max}$ then image is blank page; If $r_{\max} == Nr$ then it is possible that card image is clipped in central part of scanner platen.

3.3 Numerals segmentation

There are several challenges for numerals segmentations: often cards have complex color background; numerals can be dark on light background or they can be light on dark background; typefaces can differ significantly. To segment digits on complex background we apply two labeling procedures to grayscale image that is a result of high-pass filtering of brightness channel of deskewed image.

Fig. 4 shows flow-chart of our numerals segmentation procedure. At the beginning we process image by FIR high-pass filter. Convolution kernel here looks like Laplacian with negative central element and depends on scanning resolution. Next step is labeling of connected regions for pixels of filtered image which are greater then threshold T1. These pixels correspond to inner edges of dark symbols on light background. Third step is labeling of connected regions for pixels which are less then predefined threshold -T1. These pixels correspond to inner edges of light symbols on dark background. For each connected region from both labeled images bounding box is calculated; and regions with appropriate bounding box size and aspect ratio are selected. Thresholds for bounding boxes selection criteria are chosen keeping in mind typical size of numerals on card images from 5 to 20 pt.

Fig. 5 illustrates numerals segmentation steps. To minimize memory requirements and processing time we implement two-pass labeling scheme. First pass is HPF and calculation of bounding boxes of connected regions for dark and light symbols separately. Second pass is labeling of selected appropriate regions. Such approach is different from conventional ways, which use global or local tresholding of brightness channel as, for example, in [5]. As a result of our segmentation bodies of numerals contain a lot of holes and boundary breaks. However following recognition procedure is robust to such possible defects. Great advantage of our technique is successful segmentation both dark and light symbols on complex color background.

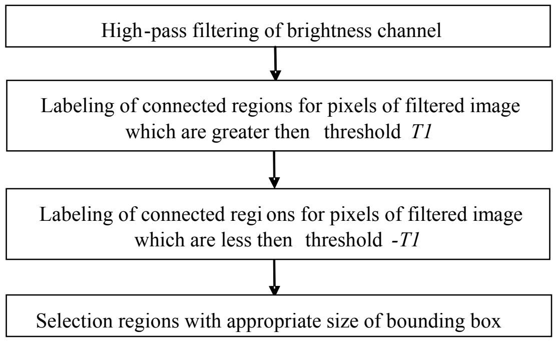

Figure 4: Flow-chart of numerals segmentation.

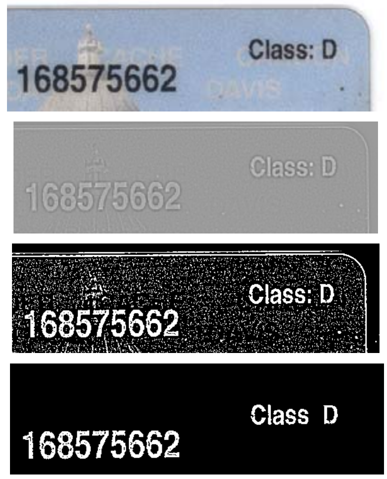

Figure 5: Illustration of numerals segmentation steps.

3.4 Arabic numerals recognition

Initially we stored about 50000 images with connected regions from our test card images by means of segmentation method as described above. Further images with numerals were selected manually. We tried several machine learning techniques for classifiers construction. However, those classifiers tend to overfitting: quite frequently symbols with typefaces absent in training set triggered false negative errors. So, for recognition of each numeral we constructed a decision tree with a great generalization capability. For some digits we created multiple trees, and it significantly improved recognition capability, allowing correct answers for numerals of completely different typefaces and even handwriting numerals.

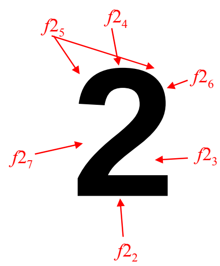

Figure 6: Features for '2' numeral.

For recognition of digit '2' we calculate the following features, which are shown on fig. 6:

$$f2_1 = Mr/Mc,\quad f2_7 = \sum_{r=Mr/2}^{2Mr/3} \sum_{c=1}^{2Mr/3-r+1} F(r,c),$$

$$f2_2 = \max\left(\sum_{r=Mr-1}^{Mr} \sum_{c=1}^{Mc} F(r,c);\ \sum_{r=Mr-2}^{Mr-1} \sum_{c=1}^{Mc} F(r,c)\right)/(2Mc),$$

$$f2_3 = \sum_{c=3Mr/5}^{3Mr/4} \sum_{r=0}^{c-3Mr/5} F(Mr-r,c),$$

$$f2_4 = \sum_{r=1}^{2} \sum_{c=1}^{2} F(r,c) + \sum_{r=1}^{2} \sum_{c=Mc-1}^{Mc} F(r,c),$$

$$f2_5 = \max\left(\sum_{r=1}^{2} \sum_{c=Mc/3}^{2Mc/3} F(r,c);\ \sum_{r=2}^{3} \sum_{c=Mc/3}^{2Mc/3} F(r,c)\right)/(2(Mc/3+1)),$$

$$f2_6 = \max\left(\sum_{r=Mr/5}^{2Mr/5} \sum_{c=Mc-1}^{Mc} F(r,c);\ \sum_{r=Mr/5}^{2Mr/5} \sum_{c=Mc-2}^{Mc-1} F(r,c)\right)/(2(Mr/5+1)),$$

where *F* is binary image of connected region, *Mr x Mc* is size of the image *F*.

If the following rule is true then connected region *F* corresponds to '2': $f2_1$>=1.1 and $f2_1$<=2.2 and $f2_2$>=0.65 and $f2_3$==0 and $f2_4$==0 and $f2_5$>0.65 and $f2_6$>0.55 and $f2_7$==0. Similar feature sets and decision trees were created for other numerals [8].

3.5 Orientation detection

We recognize numerals for each connected region for four orientations 0°, 90°, 180°, 270° and calculate number of recognized regions. Finally we make decision about card image orientation Ω based on numbers of recognized regions for each orientation according to the formula:

$$\Omega = \underset{i\in\{0,90,180,270\}}{\arg\max} \left(i\ (w_1N_1(i) + w_2N_2(i) + w_3N_3(i) + w_4N_4(i) + w_5N_5(i) + w_7N_7(i)) \times (sign(N_1(i)) + sign(N_2(i)) + sign(N_3(i)) + sign(N_4(i)) + sign(N_5(i)) + sign(N_7(i)))\right),$$

where $N_1(i)$, $N_2(i)$, $N_3(i)$, $N_4(i)$, $N_5(i)$ and $N_7(i)$ are numbers of recognized regions as numerals '1', '2', '3', '4', '5', '7' accordingly for orientation *I;* $w_{1\text{-}7}$ are weights, which are defined during testing on our set; w_1 and w_7 are smaller than other weights because probability of false positive error is higher for '1' and '7' in comparison with other digits. Detection of several various digits indicates orientation with high probability because sum of recognized regions for each digit is multiplied by quantity of recognized various numerals.

Obviously, the technique can be extended easily by means of recognition of '0', '6', '8' and '9' numerals for 90° orientation. Also there is no problem to add recognition of several characters.

4. RESULTS AND CONCLUSION

Our algorithm operates very fast. Processing time for 6.2 MPix (2500 x 2500) RGB image is in range 0.2 - 0.3 s on PC with single-core P4 3.2 GHz. Application requires about 165 Kb of additional memory only.

99% of card images with numerals from our test set are processed correctly. We compared our digits recognition, orientation detection for upturned images and deskew outcomes for 75 images with the results of well-known OCR applications ReadIRIS 11 and FineReader 10. Totally 1137 concerned digits are on the test images. In addition OCR applications recognized several English characters. Our approach detected numerals only. Table 1 contains comparison of operation quality for our algorithm, FineReader and ReadIRIS. Our method outperforms respected OCR applications. It is clear that those OCR packages have a lot of features and are not specifically intended for processing of card images. However, proposed algorithm demonstrates the same advantages in processing any originals with complex color background.

Our balanced solution allows to create intellectual two-side card copy function for modern MFP devices. Such functionality is attractive for consumers.

TABLE 1 COMPARISON OF NUMERALS DETECTION QUALITY AS WELL AS ORIENTATION DETECTION AND DESKEW

	Numerals detection, %	*Orientation detection, %*	*Deskew, %*
Proposed	87	99	99
ReadIRIS	74	72	69
FineReader	63	69	68

5. REFERENCES

[1] Caprari, R.S., *Algorithm for text page up/down orientation determination*, Pattern Recognition Letters 21, pp. 311-317, 2000.

[2] Catmull, E., Smith, A., *3-D transformations of images in scanline order*, Proc. of SIGGRAPH'80, pp. 279–285, 1980.

[3] Egorova, M.A., Murynin, A.B., Safonov, I.V., *An Improvement of face detection algorithm for color photos*, Pattern Recognition and Image Analysis, vol. 19, No. 4, pp. 634–640, 2009.

[4] Hull, J., Document image skew detection: survey and annotated bibliography, Document Analysis Systems II, pp. 40-64, 1998.

[5] Llados, J., Lumbreras, F., Chapaprieta, V., Queralt, J., ICAR: Identity Card Automatic Reader, Proc. of ICDAR, pp. 470-474, 2001.

[6] Pavlidis, T., Algorithms for graphics and image processing, Springer, 1982.

[7] Safonov, I., Kurilin, I., *Deskew for Card Image Scanning*, Graphicon - 2011.

[8] Safonov, I., Lee, H.K., Kim, S.H., US2011/0141534.

[9] Tolstaya E. *Content-based image orientation recognition,* Graphicon – 2007, pp.158-161.

[10] Wang, Y., Zhang, H. *Content-based image orientation detection*, Proc. of IEEE Workshop on content-based access of image and video libraries, pp. 17-23, 2001.

About the author

Ilia V. Safonov received his MS degree in automatic and electronic engineering from Moscow Engineering Physics Institute/University (MEPhI), Russia in 1994 and his PhD degree in computer science from MEPhI in 1997. Since 2004, Dr. I. Safonov has joint Samsung Research Center, Moscow, Russia where he is engaged in image and video processing projects.