Boise State University
ScholarWorks

Electrical and Computer Engineering Faculty Publications and Presentations

Department of Electrical and Computer Engineering

1-26-2011

Characterizing Challenged Minnesota Ballots

George Nagy
Rensselaer Polytechnic Institute

Daniel Lopresti
Lehigh University

Elisa H. Barney Smith
Boise State University

Ziyan Wu
Rensselaer Polytechnic Institute

Copyright 2011 Society of Photo-Optical Instrumentation Engineers. One print or electronic copy may be made for personal use only. Systematic reproduction and distribution, duplication of any material in this paper for a fee or for commercial purposes, or modification of the content of the paper are prohibited. DOI: 10.1117/12.876558

Characterizing Challenged Minnesota Ballots

George Nagy[1], Daniel Lopresti[2], Elisa H. Barney Smith[3], Ziyan Wu[1]
[1]Rensselaer Polytechnic Institute, [2]Lehigh University, [3]Boise State University
nagy@ecse.rpi.edu, lopresti@cse.lehigh.edu, EBarneySmith@boisestate.edu, wuz5@rpi.edu

ABSTRACT

Photocopies of the ballots challenged in the 2008 Minnesota elections, which constitute a public record, were scanned on a high-speed scanner and made available on a public radio website. The PDF files were downloaded, converted to TIF images, and posted on the PERFECT website. Based on a review of relevant image-processing aspects of paper-based election machinery and on additional statistics and observations on the posted sample data, robust tools were developed for determining the underlying grid of the targets on these ballots regardless of skew, clipping, and other degradations caused by high-speed copying and digitization. The accuracy and robustness of a method based on both index-marks and oval targets are demonstrated on 13,435 challenged ballot page images.

Keywords: ballot image processing, election technologies, challenged Minnesota ballots, paper trail

1. INTRODUCTION

The design, generation, use, preservation, and format of election ballots differ markedly from the characteristics of other types of document images, such as technical articles, books, business letters and forms, technical drawings and schematic diagrams that have attracted research on image processing and pattern recognition[1]. We report here our on-going image processing studies within a larger endeavor on paper based election technologies, *Paper and Electronic Records for Elections: Cultivating Trust* (PERFECT), initiated in 2007 [2, 3, 4, 5, 6]. This work is based on the following fortuitous sequence of events.

In the 2008 general elections, the Minnesota state legislature decreed the use op-scan ballots. In the initial tally, the number of votes for Republican Norm Coleman and for Democrat Al Franken in the U.S. Senate race differed by less than 0.01%. This triggered a recount[7] . Photocopies of the 6737 challenged ballots, which constitute a public record, were scanned on a high-speed scanner and made available on a public radio website as PDF files. We wrote a simple program to download the files, extract the TIF images, and then we posted them on the PERFECT website[6]. The laws that govern the validity of the cast votes, directives for recounts, and the challenge process, were explicated at DAS 2010[5]. Our DAS'10 paper also shows many ambiguous examples that divided voting officials and lay voters alike. About three months after the election the senate race was settled in favor of Al Franken.

Many commercial ballot readers, like the older mechanical lever machines, retain only the totals for each race and candidate, and the number of processed ballots. An interactive system for analyzing these ballots, Ballot Tool, based on our previous research[8], was presented at DAS 2010[5]. Ballot Tool allows an operator to record every significant aspect of each ballot: the type and position of the vote marks (including overvotes and undervotes), scribbles (possibly involuntary) outside the targets which may attempt to contravene anonymity, official stamps and notes by inspection officials, and even skew. The operator can also enter his or her judgment of the validity of each vote. All of the information is saved in text files associated with the ballot image.

Document Recognition and Retrieval XVIII, edited by Gady Agam, Christian Viard-Gaudin,
Proc. of SPIE-IS&T Electronic Imaging, Vol. 7874, 787413 · © 2011 SPIE-IS&T
CCC code: 0277-786X/11/$18 · doi: 10.1117/12.876558

Downloaded from SPIE Digital Library on 03 Jun 2011 to 132.177.228.79. Terms of Use: http://spiedl.org/terms

The collection of detailed information about the challenged Minnesota ballots was undertaken by Professor Elisa Barney Smith's group at Boise State University. Over 800 ballots have already been annotated using Ballot Tool. Some of the methods proposed below aim to speed up this process.

The challenged Minnesota ballot images were observed and analyzed in a Spring 2010 graduate picture processing seminar at Rensselaer. Experiments addressed the location of the index marks, registration in the absence of index marks based on image correlation, detection or rulings and target ovals, and determination of the presence or absence of marks within the ovals. We review below some relevant aspects of paper-based election machinery and provide additional statistics and observations on the posted sample data. We also present some robust tools that we subsequently developed to determine the underlying grid of the targets regardless of skew, clipping, and other degradations caused by high-speed copying and digitization. These tools are specific to ballot images and are not directly applicable to other tasks.

Most op-scan ballots, including the ones used in Minnesota, are still designed to be readable by mark-sense machines introduced more than fifty years ago. These machines had a single photocell aligned with each column of oval, circular, rectangular or "broken arrow" targets (Fig. 1). The photocell for the row *index marks* tracked the relative vertical position of the ballot. The photocells for the columns recorded a mark when the reflected light reaching the photocell at the appropriate vertical position fell below a preset threshold.

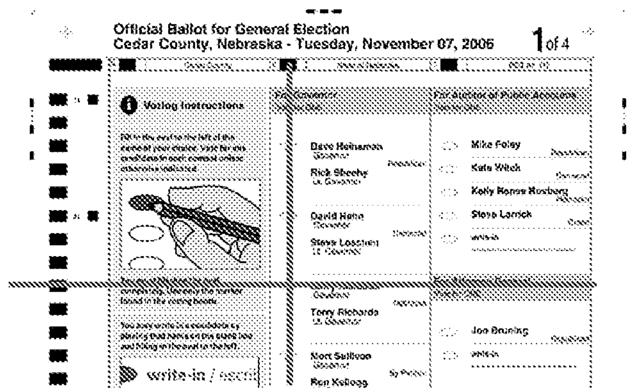

Figure 1. Grid on op-scan ballot.

In the 2008 Minnesota elections 2,885,555 ballots were cast. Minnesota has 98 counties and 4130 election precincts (including split precincts that accommodate several school districts). Challenges occurred in most but not all precincts. The layout of the ballots is each county's prerogative. Although all the ballots are laid out in either two or three columns, there are differences in the size and configuration of the headers in addition to the layout of the local races that vary from precinct to precinct. The mandated minimum print size is 6 points for instructions but the candidates' names must be set in at least 10-point type. The races and the candidates or propositions within the races, from Federal Offices to Judicial Offices, must be listed in a specified order. In a typical precinct, the voter must consider over 75 choices and mark 25 targets for a complete ballot.

Most of the Minnesota ballot templates were provided by two different companies. One template has fiducial marks in the corners (a circle with a cross), index bars along the left margin on the face of the ballot and along the right margin on the verso, and position-coded and numbered identification bars next to the index marks. The other type has smaller index bars along the top and both sides, and position-coded identification bars along the bottom. Many index bars are partially or completely lost in scanning or copying. Although dedicated ballot scanners may have more reliable paper transports, if the supply at a polling station runs out, ballots can be photocopied. This may add to distortion and degradation introduced by the scanning process. Most of the ballots were apparently scanned at 300 dpi, but some of the longer ballots must have been scanned at about 240 dpi (this assessment is based on the assumption that these ballots were

Downloaded from SPIE Digital Library on 03 Jun 2011 to 132.177.228.79. Terms of Use: http://spiedl.org/terms

printed or copied on 8.5" wide paper). Almost all of the ballots consist of two pages, front and back. Readers can view the original PDF files and the TIF versions at http://perfect.cse.lehigh.edu/BallotData/MNChallengedBallots/ .

In the next section we summarize the operations necessary for complete analysis of paper ballots. This paper addresses only the initial document image processing step which locates the targets for the actual votes. The degradation introduced by high speed copying and scanning in the above collection of scanned photocopies of challenged ballots is likely to be more severe than it would be on original ballots digitized by current ballot scanners.

Sections 3 and 4 describe the combination of image processing techniques that we developed to overcome uneven skew, variable scanning resolution, missing corner fiducial marks, sheared off columns or rows of index marks, and poor binarization resulting in broken and barely-visible line art and oval targets. Section 5 reports the results of applying these techniques to process over 13,000 challenged ballot images (~ 2GB of data). Section 6 is a brief discussion of the range of applicability of our methods, and possible further contributions to paper-based election technology by the document-image-processing community.

2. COMPLETE BALLOT ANALYSIS

Ballot counters must provide only a (statistically) sound determination of the popular vote where offsetting errors are tolerated[9,10,11,12]. The purpose of ballot image analysis is evaluating, standardizing, or improving election technologies. It may also characterize failure modes and attempt to mitigate them through improved ballot, hardware, and software design. Both ballot analysis and ballot counting require the following five essential steps.

1. Location of the targets on each ballot relative to a fixed reference frame. The reference frame can be determined from the fiducial marks in the corner of the ballot, the index bars along the top and sides, the location of the targets themselves, or from the rulings and printed text.

2. Detection and location in the same reference frame of marks made by the voter. The marks are typically partially or completely filled in ovals, X's, or checkmarks. Image processing problems may be caused by voters' attempts to erase or cross out mistaken marks.

3. Association of the marks with targets. Each mark must be assigned to at most one target. Marks not associated with a target have no effect unless they identify the voter. In Minnesota, marks outside but sufficiently near a target to reveal the voter's intention are accepted even though the instructions on the ballot request "filling the oval". Check marks and X's are also acceptable. Write-in votes are accepted even if the voter forgot to mark the relevant target.

4. Determination of the validity of the vote for each race. Validity depends on the rules of the particular election and this step does not involve image processing. In Minnesota, an *overvote* (voting for four candidates in a race where only three are allowed) invalidates all the votes for that race. An *undervote* (voting for only two) does not. Cross-party votes invalidate primary ballots but have no effect in general elections.

5. Checking the anonymity of the ballot. Marks deemed *intended* (as opposed to being merely *sufficient*) to identify the voter invalidate the entire ballot (to forestall possible trafficking of votes). At least one ballot shows a fingerprint, others have initials or messages.

Steps 1-3 can be performed without knowledge of each race in every precinct, but only Step 1, which is the focus of this paper, can be evaluated without detailed ground truth indicating the location of every mark. In addition to reporting precinct-by-precinct election results, document image processing may contribute to vote recounts and election audits[3].

Next we describe our experimentation with several image processing methods to establish the target grid.

Downloaded from SPIE Digital Library on 03 Jun 2011 to 132.177.228.79. Terms of Use: http://spiedl.org/terms

3. DETERMINATION OF THE TARGET LAYOUT GRID

In traditional forms processing, *rulings* are the main clue for determining translation, scale and rotation (preprinted text often appears in a drop-out color). In the following five methods, we also explored image processing techniques based on the additional image features found on the Minnesota ballots.

1. *Full page registration.* Each ballot is correlated against an unmarked prototype ballot of the same type. The voter marks are then obtained by differencing [BLN08, BLN09]. The disadvantage of this method is that reference ballots must be provided for each of several thousand layouts.

2. *Coordinate transformation based on fiducial marks.* The distance of the fiducial marks from the edges of the ballots varies by as much as 10 mm due to copying or scanning, but a single mark suffices to determine translation. Locating a second mark allows correcting also for rotation and scale (10° rotations are common in this data). Three and four marks can cope with affine and perspective transformations respectively. However, on many samples several fiducial marks were clipped because of skew during copying or digitization. Furthermore, the spacing of the targets cannot be predicted by interpolation from the fiducial marks because the spacing differs between ballot types.

3. *Rulings.* The detection of rulings is often an important element of forms processing. The principal rulings on the ballots were detected using morphological structuring elements and the Hough Transform. The column division lines are always evenly spaced, but the rulings between races vary and detecting them is more difficult when the division titles are shaded. As with fiducial marks, the geometrical relationship between targets and rulings is not fixed. On 12 *same-precinct* ballots, 84% of all the ruling lines were detected. This sufficed to find the target grid of each ballot in this restrictive scenario. However, not enough rulings could be detected on light and dark scans (which affect rulings more than index marks), so we did not pursue this method further.

4. *Direct location of four oval targets.* This four-step method requires as input data only the values of the approximate vertical spacing between index marks for all ballot types and scanning resolutions, and a sample of the two types of oval targets, one from each vendor. (We could have added samples of lightly and darkly scanned targets, but one of the two prototype targets sufficed to find at least four targets among the several dozen on most ballots.) Although Steps i and ii of Method #4 locate most of the targets, they are used only to establish the complete underlying grid in Steps iii-iv.

(i) *Find the Connected Components.* Extract the black 4-connected components of each ballot.

(ii) *Classify.* Classify every connected component as *target* or *non-target* according to the proximity of its feature vector (area of the CC and size of its bounding box) to the feature vector of the nearest prototype target.

(iii) *Select target pairs.* Among the CCs classified with high confidence (low distance), select one or more pairs of targets aligned approximately horizontally, and one or more pairs aligned vertically. Estimate the number of grid steps between each pair by dividing the distance by the known horizontal and vertical intervals. The correct spacing yields a near-zero remainder for any pair. This defines the *vertical vector* and the *horizontal vector* displacement of adjacent targets.

(iv) *Form Target Grid.* Starting at the centroid of the CC with the highest confidence, form the entire target grid. Estimate the vertical and horizontal slopes independently. Since the number of rows and columns in each ballot is unknown, extend the grid beyond the largest possible ballot size.

This method is vulnerable to light scans that break almost all of the oval targets on a ballot into several connected components. Except for this case, it yields precise locations of the grid underlying the oval targets (Fig. 2). It was therefore retained as an exception-processing method for the cases where the primary method, described next, fails.

Downloaded from SPIE Digital Library on 03 Jun 2011 to 132.177.228.79. Terms of Use: http://spiedl.org/terms

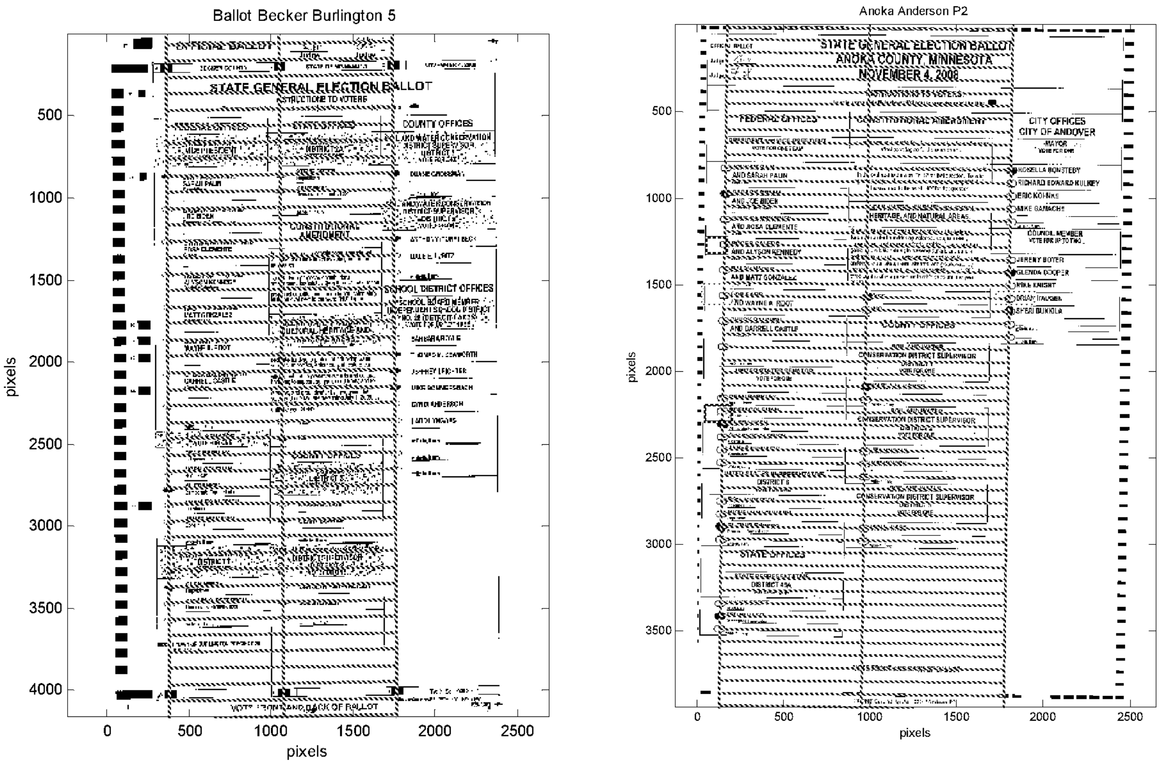

Figure 2. Method #4 is robust, as indicated by the exact placement of the grid on two ballots with different layouts and skew. The vertical and horizontal pairs of targets selected to determine the vectors that determine the grid are enclosed by rectangles on the right hand ("Anoka Anderson") ballot.

4. TARGET DETECTION BASED ON INDEX BARS

Not surprisingly, the best method that we have found locates the oval targets at the intersection of an orthogonal (but not isothetic, i.e., not parallel to the pixel array coordinates) grid defined by the index marks (sometimes called *Scantron marks*). As seen in Figure 2, one type of ballot has index bars in two columns on the left, and in rows at top and bottom, while the other has smaller index bars along both sides and along the bottom. Since this is the primary method that we used in the experiment reported in Section 5, we describe the steps in some detail.

1.1 Preprocessing

Preprocessing in the context of ballots eliminates most foreground elements without affecting the elements used for constructing the target grid. Reducing quickly the number of foreground components accelerates the subsequent extraction and analysis of connected components.

One-pixel holes in the foreground are filtered out. The resulting image is morphologically *opened* with a 10x5 pixel structuring element that removes most of the characters without altering the index bars and targets. (These filters are also applied before the oval detection method described in the previous section.)

Rulings, stamps, and large print are also filtered out. This is carried out by successive *open* operations with a disk structuring element whose diameter is reduced by steps of 2 pixels until at least 20 connected components are left. The initial diameter of the disk, designed to be slightly smaller than the height of the largest index bar, is set to $\sqrt[3]{A/2}$, where A is the median area of the components left after noise filtering. Adaptive processing is necessary here because the optimal size of the structuring element depends both on the style of the ballot (which have different sized index bars) and

Downloaded from SPIE Digital Library on 03 Jun 2011 to 132.177.228.79. Terms of Use: http://spiedl.org/terms

on the scanning resolution (which varies according to the physical size of the ballot). If there are fewer than 20 connected components left after this step, then most of the index bars must have been clipped, and the program switches to the oval detection method.

1.2 Location of the index bars and computation of the grid parameters.

Not all the elements left after preprocessing are index bars. The index bars are found from the peaks of the histograms of the eccentricity and area of the connected components (Fig 3.). Among the patterns left after filtering (like filled in ovals), only the index bars have consistent area and eccentricity.

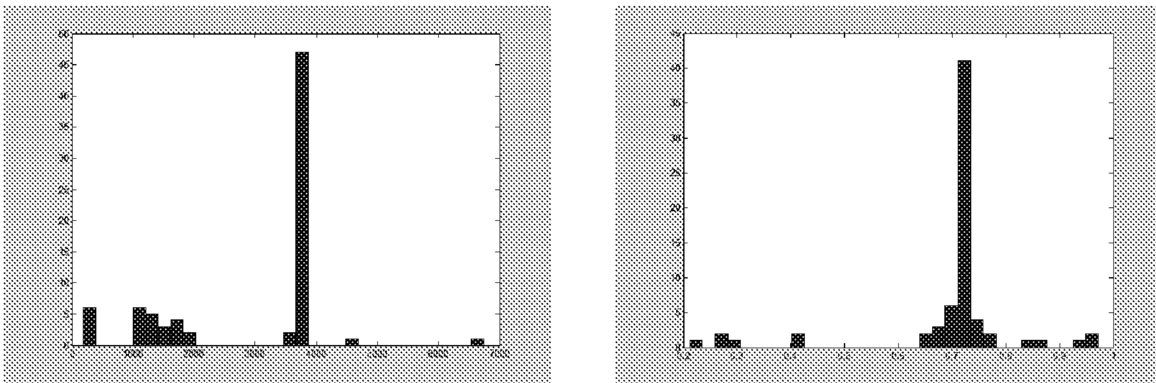

Figure 3. Histograms of the area (left) and eccentricity (right) of the connected components left after adaptively filtering out most much smaller and much larger objects. It is easy to distinguish the index bars.

Some index bars are spaced by the manufacturer as a kind of barcode to identify the specific election district and link the ballot to the database of candidates for each race. Therefore the next step finds the candidate index bars that actually define the target grid. Aligned bars are found via the Hough Transform on the centers of the connected components (Figure 4). The straight line segments with the most frequent near-vertical and near-horizontal orientations are selected as axis candidates. Among these, the segments spanning the components with the least variance in the distance among adjacent components are selected as the horizontal and vertical axes. (If there are two line segments with very small variance, the one with more components wins.) Since there are far more rows than columns, the vertical slope computation is more accurate than the horizontal slope computation. When the two differ from 90° by more than a small threshold value, the horizontal axis is set perpendicular to the vertical axis.

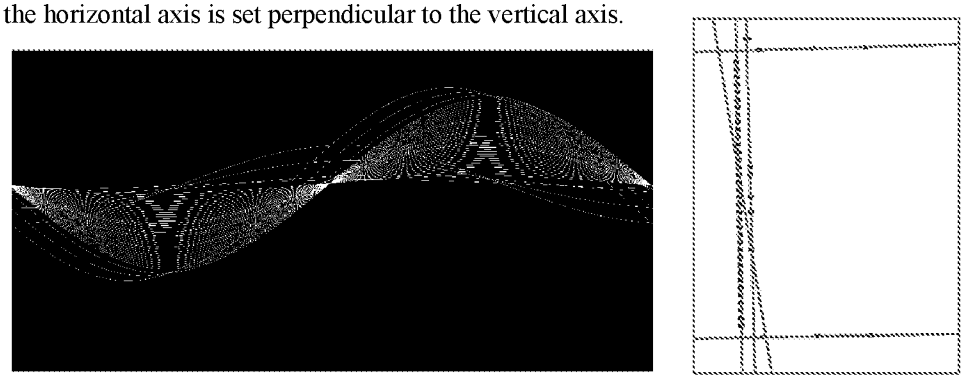

Figure 4. Hough Transform. Left: in parameter space with distance on y-axis and angle on the x-axis. Right: Examples of detected alignments of centroids of connected components.

Downloaded from SPIE Digital Library on 03 Jun 2011 to 132.177.228.79. Terms of Use: http://spiedl.org/terms

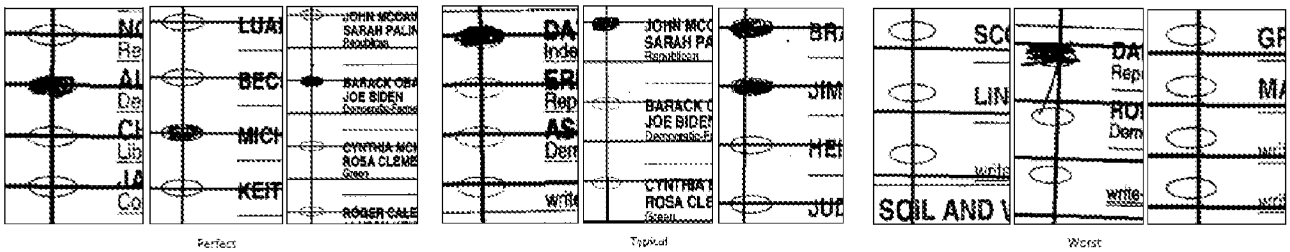

Figure 5. Three examples each of the best, typical, and worst results index bars.

The spacing is computed from the number of times the smallest interval between bars divides into the distance between the extremal bars. The slope values and spacing are checked for consistency. If the standard deviation among the slopes of the populated line segments is greater than 10°, then the result is deemed unreliable and the program switches to direct oval detection. Index bar detection also yields to oval detection if the vertical axis is too far from the vertical, and if the vertical spacing is larger than 10% of the ballot height or the horizontal spacing is larger than 50% of the ballot width.

The above procedures result in estimates of the five parameters that define the target grid used to locate the ovals. These parameters are: Origin (x_0, y_0), Slope θ, and Spacing (Δ_x, Δ_y). Examples of the local fit of the grid are shown in Figure 5.

5. EXPERIMENTAL RESULTS

The methods described above were applied without any program failure to a set of 13,345 ballot pages from the PERFECT website. The program failed to generate a grid on less than 1% of the pages. Best, typical, and worst results are shown in Figures 5 and 6. Since no ground truth is available for this data, 1031 randomly-selected pages (10%) were visually examined to determine the accuracy of the target grids. On nine pages among these the program failed to generate a grid. Six of these were anomalous broken-arrow ballots from the town of White in St. Louis County. The other three were absentee ballots intended for manual vote counting.

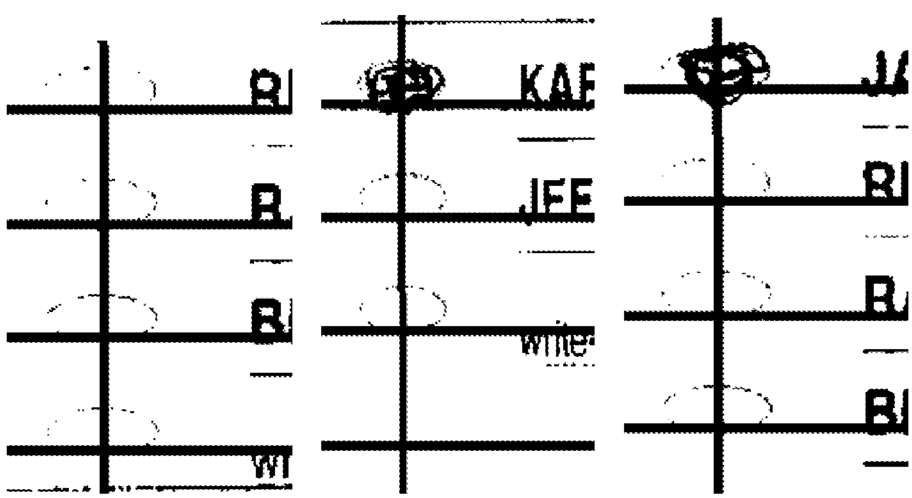

Figure 6. Three examples of very light scans where the oval targets could not be found directly.

Downloaded from SPIE Digital Library on 03 Jun 2011 to 132.177.228.79. Terms of Use: http://spiedl.org/terms

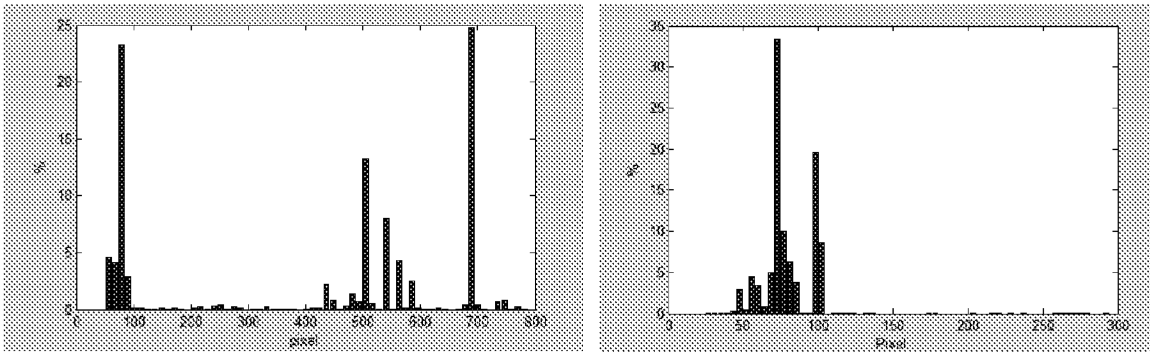

Figure 7. Detected horizontal (left) and vertical (right) spacing of the index bars.

The width of the ballots ranged from 1136 to 4127 pixels, and their height from 1904 to 5056 pixels. The mean widths and heights were 2476 and 4127 pixels. Morphological filtering with fixed-sized structuring elements typically yielded 70 connected components. The program found enough index bars on 95.8% of the pages. On many of these direct oval location would fail (Figure 6). The maximum slope was 22° and the detected spacings are shown in Figure 7.

The average ballot-maximum error (i.e., the maximum distance on each a ballot between the centroid of the ovals and the intersections of the horizontal and vertical grid line, averaged over all ballots) was 7.8 pixels. The median ballot-maximum error was 8.1 pixels, indicating relatively few outliers. The maximum error over all ballots was 21. By way of comparison, the oval targets are 40 to 70 pixels wide. It is likely, of course, that the maximum error on the entire set is higher than the maximum observed error on the 10% sample. We are nevertheless pleased that the combination of significant clipping and broken ovals (that would defeat the combined algorithm), is rare.

The experiment was run on a 32GB Dell PE2950 server with an Intel XEON E5440 2.83 GHz processor. The unoptimized Matlab 2009b program ran in 8.3 seconds per page on average. There are several options for significant speed-up, including reprogramming the modules in C or C++, but even the current program can process all 6,727 ballots overnight.

6. CONCLUSIONS

Ballots processing, which is a specialized kind of forms processing, requires the detection of the presence or absence of marks at specific locations. The principal tasks are the association of each mark with a candidate, and the detection of extraneous marks that may invalidate the ballot. Character recognition is seldom needed because the number of write-in votes is insignificant in most elections and can therefore be quickly tallied by election officials. (It is, however, important to *detect* the presence of write-in votes)

An early and essential step in ballot processing is the mapping of the possible location of the targets and voter marks regardless of scale, skew, clipping, and defective binarization. We have reported above our studies of this task. We used only standard image processing and pattern recognition methods, including static and adaptive morphological filters, Hough transform, connected component analysis, and supervised (ovals) and unsupervised (bars) classification based on simple image features. Unlike methods based on image registration and resampling, these algorithms perform intensive processing only on very limited regions of the image. The results demonstrate the advantages of simple adaptive processing and of integrating two independently developed methods based on different image features.

More specifically, the results of our experiment demonstrate that the combination of index bar and target oval based processing is adequate to anchor the complete analysis of the posted 2008 Minnesota challenged ballots. Although all these ballots fall into only a few types, they represent several hundred different layouts. It is important to remember that

Downloaded from SPIE Digital Library on 03 Jun 2011 to 132.177.228.79. Terms of Use: http://spiedl.org/terms

the photocopying and digitization of these ballots was carried out without any intention of using the resulting image files in a ballot reader. The original objective of posting the ballots was to call public attention to the nature of the challenges made by the two major political parties in order to swing the senate race. Therefore no attempt was made to adjust the copier and scanner to retain the index marks or to yield machine-recognizable targets. Nevertheless our combination of methods proved adequate to determine accurately the location of the target grid on every "regular" ballot page in the 10% sample. Such performance is likely to provide an additional margin of robustness for processing carefully scanned original ballots in a real election scenario.

We are aware that the format of paper ballots in many elections differs from that of the 2010 Minnesota ballots. Even in Minnesota, the town of White (pop. ~3500) did not conform. Ballots from other states exhibit significant differences in paper size, number of pages per ballot, type and layout of the targets, rulings, and print density. Nevertheless, the targets on all ballots intended for machine reading are arranged on an orthogonal grid, and every ballot design provides some kind of fiducial, registration or index marks to establish the corresponding coordinate system. There are only half-a-dozen popular target designs. Furthermore, most states have laws to ensure some statewide uniformity in ballot layout. We therefore believe that our methods can be adapted to other US elections with relatively little effort.

The five parameters of the target grids for 9000 pages of the challenged Minnesota ballots are posted on the PERFECT website for use for downstream experiments by interested researchers. Possible follow-up work includes determining whether there is a target and a voter mark at every intersection of the target grid, tallying the counts race by race to flag overvotes and undervotes, and identifying possible violations of voter anonymity.

ACKNOWLEDGMENT

We gratefully acknowledge the contributions of recent Rensselaer MS graduates Sam Gilbert, Alex Franz, and Bryan Winrow. This work was supported in part by the National Science Foundation under awards NSF-0716368, NSF-0716393, NSF-0716647, and NSF-0716543. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the National Science Foundation.

Downloaded from SPIE Digital Library on 03 Jun 2011 to 132.177.228.79. Terms of Use: http://spiedl.org/terms

REFERENCES

[1] G. Nagy, Twenty Years of Document Image Analysis in PAMI, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, #1, 20th Anniversary Issue, pp. 38-62, January 2000.

[2] D. Lopresti, G. Nagy, and E. Barney Smith. A document analysis system for supporting electronic voting research. In Proceedings of the Eighth IAPR Workshop on Document Analysis Systems, pages 167-174, Nara, Japan, September 2008.

[3] E. H. Barney Smith, D. Lopresti, and G. Nagy, Ballot Mark Detection, Proceedings of International Conference on Pattern Recognition XIX, Tampa, FL December 2008.

[4] E. H. Barney Smith, D. Lopresti, and G. Nagy, Mark Detection from Scanned Ballots, *Proc. Document Recognition and Retrieval XVI,* San Jose, CA, January 2009, paper #72470P.

[5] D. Lopresti, G. Nagy, and E. Barney Smith. Document Analysis Issues in Reading Optical Scan Ballots, Proceedings *of the Ninth IAPR Workshop on Document Analysis Systems*, Boston, June 2010.

[6] Paper and Electronic Records for Elections: Cultivating Trust (PERFECT), 2008. http://perfect.cse.lehigh.edu/ .

[7] United States Senate election in Minnesota, 2008. Wikipedia, December 2009. http://en.wikipedia.org/wiki/United_States_Senate_election_in_Minnesota_2008.

[8] D. Lopresti and G. Nagy, Issues in ground-truthing graphic documents, *Proceedings of the Fourth IAPR International Workshop on Graphics Recognition*, pages 59-72, Kingston, Ontario, Canada, September 2001.

[9] Brennan Center for Justice, The Machinery of Democracy: Voting System Security, Accessibility, Usability, and Cost, Technical report, The Brennan Center Task Force on Voting System Security, June 27 2006, http://brennan.3cdn.net/cb325689a9bbe2930e_0am6b09p4.pdf.

[10] M. Lausen, Design for Democracy: Ballot + Election Design, U. of Chicago Press, 2007.

[11] Ohio Evaluation & Validation of Election-Related Equipment, Standards & Testing (EVEREST), March 2008. http://www.sos.state.oh.us/sos/info/everest.aspx .

[12] P.S. Hernson, R.G> Niemi, M.j. Hammer, B.G. Bederson, F.C. Conrad, and M.W. Traugott, Voting Technology, Brookings Institution Press, Washington, D.C., 2008.

Downloaded from SPIE Digital Library on 03 Jun 2011 to 132.177.228.79. Terms of Use: http://spiedl.org/terms