Improving the Accuracy of Ballot Scanners Using Supervised Learning

Sameer Barretto, William Chown, David Meyer, Aditya Soni(✉), Atreya Tata, and J. Alex Halderman

University of Michigan, Ann Arbor, USA
{sambarr,chownwil,davidmey,adisoni,artata,jhalderm}@umich.edu

Abstract. Most U.S. voters cast hand-marked paper ballots that are counted by optical scanners. Deployed ballot scanners typically utilize simplistic mark-detection methods, based on comparing the measured intensity of target areas to preset thresholds, but this technique is known to sometimes misread "marginal" marks that deviate from ballot instructions. We investigate the feasibility of improving scanner accuracy using supervised learning. We train a convolutional neural network to classify various styles of marks extracted from a large corpus of voted ballots. This approach achieves higher accuracy than a naive intensity threshold while requiring far fewer ballots to undergo manual adjudication. It is robust to imperfect feature extraction, as may be experienced in ballots that lack timing marks, and efficient enough to be performed in real time using contemporary central-count scanner hardware.

1 Introduction

Hand-marked paper ballots counted by optical scanners are the most popular voting method in the United States, used by jurisdictions home to about 70% of registered voters [29], and they are becoming even more prominent due to the rapid expansion of postal voting spurred by the COVID-19 pandemic [13]. Despite its importance, optical scan voting faces two significant integrity challenges. First, deployed scanners suffer from a host of well-documented vulnerabilities (e.g., [2,11,14,15,18]). Second, and the focus of this study, even in the absence of an attack, traditional scanning techniques sometimes fail to accurately count some voter marks [12]. In principle, risk-limiting audits can address both problems by ensuring that any fraud or error sufficient to change the outcome of a contest is likely to be detected [17,24], but widespread adoption of RLAs, even for Federal contests, may be a decade or more in the future. Given that many major contests will not be subject to rigorous audits anytime soon, it is important to ensure that scanners themselves count ballots as accurately as practically possible.

Today's ballot scanners typically employ variations of a relatively simplistic technique [12,27]. After creating a digital image of the ballot, they identify the voting targets and calculate the average shading within each target area, s_i. For a predefined threshold α, target i is treated as marked whenever $s_i \geq \alpha$. Some

R. Krimmer et al. (Eds.): E-Vote-ID 2021, LNCS 12900, pp. 17–32, 2021.
https://doi.org/10.1007/978-3-030-86942-7_2

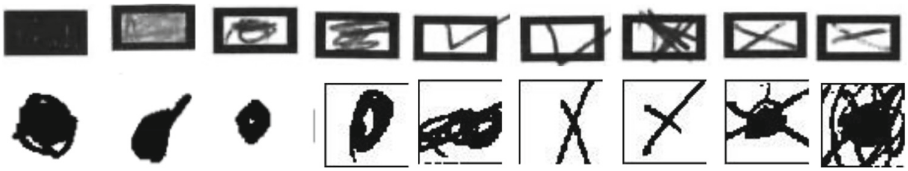

Fig. 1. Voted targets from Humboldt (*top*) and Pueblo (*bottom*) datasets. These scans originate from Hart InterCivic and Dominion scanners, respectively. This difference is reflected in the style of the targets and the quality of the scans.

modern scanners make use of a second threshold, β. If $\beta \leq s_i < \alpha$, the target is treated as an ambiguous or *marginal* mark, and the ballot is set aside for officials to manually determine the voter's intent, in a process known as *adjudication*.

This technique performs well on ballots that have been properly marked, but it sometimes falls short when handling ballots where the voter has not followed the instructions precisely [12], as in many of the samples in Fig. 1. Often, voters disregard ballot instructions and use other marks such as X-marks or check marks to indicate their intent. As discussed in Sect. 3.1, we found that roughly 8.5% of marks in one large corpus of voted ballots were not filled as directed. While humans can easily identify these "marginal" marks and typically interpret them correctly, they may be challenging for current optical scanning systems to process accurately. If marks are not dark enough, they may not meet either threshold will therefore be ignored by current systems. Even in the case where marks fall within the adjudication range, tabulating them imposes increased labor costs for resource-constrained voting jurisdictions.

We investigate the feasibility of improving scanner accuracy and reducing adjudication costs by applying supervised learning techniques. Using real voted ballots, we train a convolutional neural network to classify a variety of mark styles, including both properly marked targets and common marginal marks. Compared to a generic implementation of mark recognition based on intensity thresholds, our model achieves more accurate classification and lower rates of adjudication. We further validate our technique using a second real-world ballot corpus for which we have the results of scanning and adjudication reported in the election, and achieve identical results in every case. These findings suggest that our approach could improve scanner accuracy while reducing election costs.

2 Related Work

The challenging nature of ballot mark recognition has long been recognized and is discussed at length by Jones [12] and Toledo et al. [27].

A number of previous studies have investigated methods for improving ballot scanning. Several groups have approached the problem by combining computer vision for feature extraction with human judgement for checking the interpretation of marks. In 2010, Cordero et al. proposed a method for efficiently verifying the scanner's mark interpretations by having humans review batches of ballot images automatically superimposed on each other [6]. Wang et al. later developed

OpenCount, a system that similarly automated feature extraction and provided interactive tools for classifying voter marks [30]. Although our goal is to improve automatic mark recognition and reduce reliance on operator input, these earlier works could complement our techniques and result in further efficiency gains, if applied to the ballots that our approach determines require manual adjudication.

More closely related to our approach, other prior work has applied supervised learning to mark recognition. In 2009, Xiu et al. briefly investigated a classification approach generally similar to ours, but based on modified quadratic discriminant functions (MQDFs) instead of convolutional neural networks (CNNs) [31]. Although they reported strong performance, their dataset consisted of only a few hundred ballots, making comparisons with real-world scanner performance difficult. A 2015 NIST study further benchmarked several ML-based approaches for categorizing marginal marks [1], but their primary goal was to improve testing of optical scanners rather than to surpass intensity-based mark detection.

3 Methods

In recent years, convolutional neural networks (CNNs) have become the industry standard for image classification [26]. CNNs use a divide and conquer strategy to classify images, attempting to gain a localized understanding of an image's structure to identify key characteristics which are then used to classify the image as a whole. For instance, in classifying marks on a ballot, one feature a CNN might identify is lines at a 45° angle, corresponding to X-marks. We chose to use a two-dimensional CNN, since it allows for the detection of multidimensional structures, in contrast to a one-dimensional fully-connected network which would immediately flatten the image, losing the ability for the network to extract this type of structural feature from the data. Another advantage of CNNs is that they use comparatively fewer parameters than fully connected networks, since they reuse their parameters several times. This means that the model is easier to train because it requires less data to achieve a higher accuracy and takes less time.

We developed our own CNN model and then tested it on ballot scans collected from actual elections, evaluating its performance relative to a simple threshold-based approach. It was not possible to obtain a currently marketed optical scanner to use as a baseline for comparison, so we wrote our own implementation closely modeled on the Dominion ImageCast scanner system, as described in patents and court documents [7,22]. The Dominion system, which is used in parts of 28 states [29], defaults to $\alpha = 35\%$ and $\beta = 12\%$, which we adopted for our implementation. One advantage of using this baseline model rather than an actual optical scanner was that both models used the same extracted features, allowing for a truer comparison of their mark detection methods.

We decided to build a model that would classify individual targets as features rather than examining entire pages. This way our model generalizes well across different contests and page types, so long as the targets are the same shape and size. Since the two datasets we used (described below) had differently shaped targets, we used a separate model for each. Both models used the same CNN architecture, but each was trained on different data.

3.1 Data

The ballot scans we used came from two datasets: the November 2009 election in Humboldt County, California and the November 2020 election in Pueblo County, Colorado [23]. Initially, we used a representative subset of the Humboldt data, consisting of 23,846 out of the 28,383 non-blank pages, which contained 149,394 voting targets. Later, to validate our approach, we used a subset of the Pueblo dataset, which provided ballot scans as well as the official interpretation of each target resulting from the real scanners and adjudication process. This allowed us to directly compare the CNN model's output to real election practice. From the 89,098 Pueblo County ballots, we used a representative subset of 1,719 ballots that contained 147,121 voting targets. Each ballot consisted of multiple contests. Some Humboldt contests allowed for only one vote while others allowed multiple choices to be selected. Additionally, the ballots in both datasets did not have a straight-ticket option, so most contests contained marked targets.

Labeling. To provide ground truth for the Humboldt data, we manually labeled all of the targets in our subset. We started by labeling each ballot page type; for purposes of this study, a page type is defined as a set of scans that contain the same contests in the same relative locations on each page. We then labeled the individual targets in two passes, according to two labeling schemes. In the first pass, we labeled targets by the mark type, and in the second, by perceived voter intent (0 for no vote, 1 for vote). The first schema is presented in Fig. 2, along with a summary of the first pass of labeling. Approximately 69% of targets were unmarked, 29% were properly marked, and 2.7% contained a marginal mark.

We verified our labels by comparing the election results published by Humboldt County [10]. There was near perfect agreement for contests that had been labeled completely, with the maximum difference being 15 out of 6529 votes (or 0.2%). Most contests were either in complete agreement or differed by only 1 or 2 votes. In all the contests where there was a mismatch, our vote totals were less than the official counts. Upon investigation, most of the discrepancies were due to malformed or flipped scans, which we did not label. The small residual disagreement could be due to inaccuracies in the original count or our own human error.

Unlike the Humboldt scans, which were stored as grayscale images, the Pueblo scans were 1-bit black and white. There may have been some faint marginal marks on ballots that were undetected by the optical scanners and were also missed by our manual labeling. In this case, all models would have misclassified this type of mark, since it was lost at the scanning stage rather than the interpretation stage.

Feature Extraction. The ballots in the Humboldt dataset lack timing marks, and we found that the position and orientation of the ballot relative to the scanned image was inconsistent across scans. To overcome this, we created a template for each page type that indicated the location of each voting target relative to the top-left corner of a rectangular printed border that surrounds the ballot. We used OpenCV's contour detection algorithm [3] to obtain the coordinates of the corners of the border in each scan, then aligned the appropriate

Mark Type	**Count**
No Mark	100,999
Properly Marked	42,165
Marginal Mark:	
X-Marked	1,115
Check-Marked	93
Lightly Marked	1,903
Partially Marked	489
Marked and Crossed Out	316
Bad Scan / Wrong contest	2,242
Other	72
Total	149,394

Fig. 2. Number of marks of different types in Humboldt dataset, as determined by manual classification. 8.5% of the marks in this dataset were marginal marks.

template to extract all of the voting targets. This method accounted for the common case of vertical and horizontal shifts of the ballot within the scans. However, this method is not able to account for other kinds of scanning artifacts, including ballots with nonlinear distortions due to misfeeding.

For the Pueblo dataset, the ballots contained timing marks, which provided four points of reference for each individual target, giving us an extremely accurate position for extraction. For each page type, we used OpenCV to identify the timing marks corresponding to each target and used them to extract the target regions. This was highly resilient to rotations and other scanner distortions.

Partitioning into Training and Test Data. We used a subset of the labeled targets from each dataset for training and the remainder for testing. Of labeled targets from the Humboldt dataset, 54% (corresponding to 12 out of 17 page types) were used for training. For the Pueblo dataset, 75% were used for training. These differing splits were a matter of convenience. Both models exhibited excellent performance, but we note that the larger amount of training data may have benefited the performance of the Pueblo model relative to the Humboldt model.

3.2 Baseline Model

We sought to compare our methods to the commonly used intensity-threshold technique. Since we did not have access to a deployed ballot scanner, we created our own implementation modeled after the Dominion system described in Sect. 2. For each ballot, our baseline model considers all the extracted targets in a given contest and predicts each target as either no vote, vote, or adjudicate. In practice, a single adjudicated mark will result in the entire contest on that ballot being reviewed by humans, so if any mark was predicted as adjudicate, we labeled all the targets in that contest the same way.

Dominion's scanners create 1-bit-per-pixel bitmaps, as shown in Fig. 1. In order to replicate this behavior using the grayscale Humboldt scans, we applied Floyd-Steinberg dithering (a common graphics algorithm provided by the imaging library we used [4]) to reduce the grayscale images to black and white while approximately maintaining the average intensity within local regions.

The next step was to calculate the number of marked pixels inside the target area. However, each feature consisted of not only the voter's mark (inside the target), but also the pre-printed target border and the area immediately outside it. To account for this, we first converted our thresholds into raw pixel counts, leveraging the fact that all targets had the same dimensions. Then we subtracted the average number of black pixels occupied by the unmarked target border.

To allow for imperfect feature extraction, our baseline implementation considered a target area that is somewhat larger than the printed targets. Some fielded scanners are known to do so as well, but to our knowledge the specifics of this behavior are not well documented by any manufacturer. We note this as a limitation of our baseline model. It is possible that real scanners differ in such aspects and so would sometimes produce different results; however, we expect variations based on marks outside the printed target to be uncommon. In our datasets, such marks rarely occurred except in cases where the shading within the printed target alone would have clearly been an intended mark.

3.3 CNN Model

Preprocessing. Before we could train our model, we needed to transform our dataset. In order to decrease computational costs, we resized the cropped target areas to 28 × 28 pixels with 8-bits-per-pixel of depth. We then stored them in a three-dimensional array, X, parallel to their associated labels, y. Finally, we normalized the pixel values in X to a 0–1 scale.

Our manual classification rubric included "lightly", "partially", and "properly" marked labels, but we later realized that the distinction between these classes varied depending on who was assigning the label. Due to the subjectivity, we merged these classes prior to training. All three labels indicated that the voter intended a mark; we reviewed the entire contest when making these classifications, and in each case the voter's intent was clear.

Finally, we made a second partition of the targets from those that were set aside for training, reserving 85% for training and a standard 15% for validation. This allowed us to train our model using various parameter combinations and determine which were best by examining performance on the validation set. We followed this process for both datasets independently.

Model Structure. The model we chose consisted of a single convolutional layer with 25 filters of kernel size 3 × 3, stride 1, and no padding. The output was passed through a ReLU nonlinearity, followed by a fully connected layer with ReLU, and finally a second fully connected layer that culminated in seven neurons. We used the softmax function to create a probability distribution from the final layer weights and outputted our prediction as the class with the highest probability.

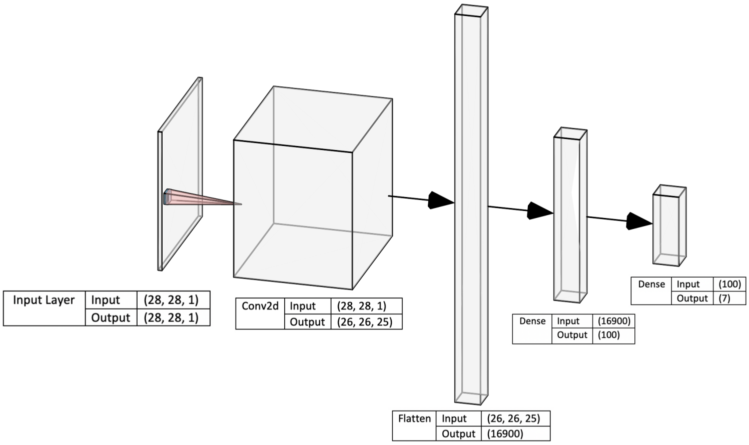

Fig. 3. The CNN architecture we used. Pictured layers appear from left to right in the order they were applied. (*Image generated using* [16].)

A primary consideration while designing the model was the number of convolutional layers. Models today can have upwards of 50 layers [9], but excess layers can cause overfitting. Our dataset was relatively uncomplicated, with X-marks, check marks, and marked and crossed-out marks being the most complicated structures. We wanted a model capable of learning these structures but also general enough to categorize all X-marks, regardless of their shape, size or orientation, as an X-mark. We initially made the assumption that more layers would result in higher accuracy, but in evaluating our model, we noticed that our training loss was significantly lower than testing loss, and our validation accuracy was low, which suggested that the design was overfitting. This led us to use a shallower model, reduced to one convolutional layer and with an increased number of convolutional filters. We observed that this approach reduced overfitting and significantly increased validation accuracy (Fig. 3).

Before trying a shallower model, we experimented with hyperparameter tuning, as well as regularization methods such as dropout. We also attempted to add a pooling layer, to downsample, and to reduce the number of parameters, but found that these features were unnecessary due to the already low spatial size of our images. The shallower model we settled on also had the side benefit of faster training, allowing more iteration in our model development process.

We implemented our model using Keras and TensorFlow. We were able to take advantage of the built-in convolutional and fully-connected layers while having the flexibility to write our own evaluation metrics.

Hyperparameter Selection. One important hyperparameter was the evaluation metric. Our ultimate goal is to produce a vote tally that comes as close as possible to the collective will of the voters, and our model also should be intelligible to voters, allowing people to understand how their votes are counted. With these criteria in mind, accuracy is the most logical evaluation metric. For training the model, however, simply trying to optimize for accuracy has its drawbacks.

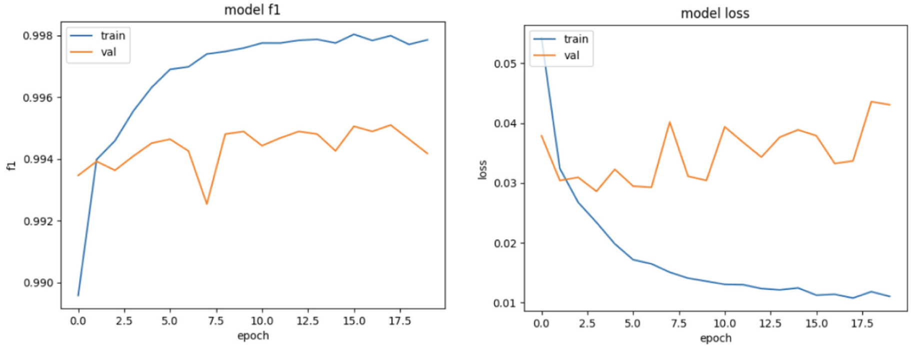

Fig. 4. Using 17 epochs optimizes validation F1 score while retaining low loss.

Since marginal marks account for such a small percentage of the data relative to properly marked and blank marks, a model trained for accuracy would not learn to classify these marks as well as their more prevalent counterparts. To address this, we chose to use a model that optimizes F1 score, the harmonic mean of precision and recall, which puts more weight on correctly classifying these marginal marks. By optimizing for F1 score, we were able to produce a model that had a higher overall accuracy compared to one that optimized for accuracy directly.

The other traditional hyperparameters we selected were batch size and the number of epochs. Based on a number of trial runs, we expect that a fairly wide range of batch sizes would be appropriate; we chose 32. For the number of epochs, we chose 17, which testing determined was past the point of diminishing marginal returns for the F1 score while maintaining low loss, as shown in Fig. 4.

The final important hyperparameter was a threshold for confidence, which we used to apply our trained model to an entire contest rather than individual targets. That is, how confident did we need to be that all the targets in a contest were classified correctly in order to not designate that ballot for adjudication? To utilize this threshold, we first obtained the product of the label probabilities for each of those targets, and then compared that value to the threshold. Similarly to the baseline model, if this value was lower than the threshold then we would send the entire contest for adjudication. We tested several threshold values and obtained the best results with a threshold of 0.95 combined with adjudicating any contest in which the classifier found three or more different types of marks.

3.4 Differences for Pueblo Dataset

Although the model structure for the Pueblo dataset was broadly similar to the Humboldt model, we did not use the baseline model to evaluate it since we had the scanner's actual interpretation as ground truth. Each ballot in the dataset included the officially counted votes (and the results of adjudication, if applicable) as a final page in the scan, a feature that Dominion calls AuditMark [8]. We extracted these results using the Pytesseract optical character recognition library (Fig. 5).

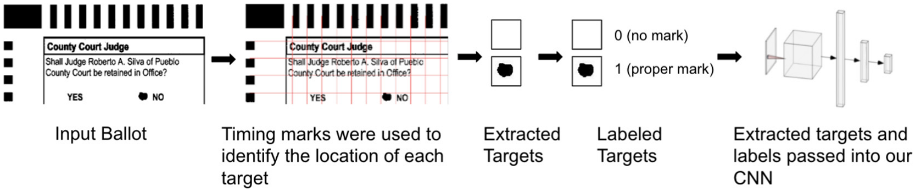
County Court Judge
Shall Judge Roberto A. Silva of Pueblo County Court be retained in Office?
YES
NO
County Court Judge
Shall Judge Roberto A. Silva of Pueblo County Court be retained in Office?
YES
NO
0 (no mark)
1 (proper mark)
Input Ballot
Timing marks were used to identify the location of each target
Extracted Targets
Labeled Targets
Extracted targets and labels passed into our CNN

Fig. 5. For Pueblo ballots, we used timing marks to extract targets, manually labeled them, and passed these features to our CNN model.

Through manual and automated inspections of the Pueblo dataset, we established that it contains far fewer marginal marks than the Humboldt data. This may be due in part to Pueblo County acting to protect voter privacy by removing ballots with unusual styles of marks that were flagged for adjudication. For this reason, we used the Pueblo dataset to test how a CNN model would perform compared to current scanning systems under "ideal" ballot conditions—i.e., post adjudication, limited marginal marks, and clear ballot instructions. Our goal was to establish whether a CNN-based system would perform as well as current systems even under the circumstances where current systems are most accurate.

4 Evaluation and Results

To compare ballot scanning models, there are three distinct metrics to consider: classification accuracy, number of ballots that require adjudication, and computational cost. First, it is important that a model is as accurate as possible because it is vital that the tabulated results match the intent of the voters. Second, it is important to minimize ballots that require adjudication. In many states such as Colorado, where the Pueblo dataset originated, ballots that are "kicked" by scanners must be adjudicated by a bipartisan team of election judges who determine how the vote should be counted by a set of criteria [5]. This process is slow and potentially subjective. If a ballot scanner kicks too many ballots, counting will be cost prohibitive. Finally, if the model is too slow, using it in practice (such as in real-time as ballots are scanned) may be difficult.

Before accuracy could be computed, we needed to determine how targets labeled as adjudicated should be handled when calculating accuracy. Our models assigned each target one of three labels—vote, no vote, or adjudicate. By contrast, each target in the dataset was labeled as either a vote or a no vote. When computing accuracy, we assumed all adjudicated ballots would be correctly classified by the adjudication process. We separately evaluated the number of ballots that required adjudication. We show results for these metrics in Fig. 6.

4.1 Baseline Model Performance

The baseline model performed better than we anticipated; however, it still struggled where we expected. First, it sometimes classified targets with small or light

Model	Targets Accurately Classified	Flagged for Adjudication
Baseline	68,540 (99.895%)	2,181 (3.179%)
CNN	68,588 (99.965%)	1,465 (2.135%)
Hybrid #1	68,597 (99.978%)	3,242 (4.725%)
Hybrid #2	68,557 (99.920%)	430 (0.627%)

Fig. 6. Performance of each model on the Humboldt dataset. The CNN misclassifies 67% fewer targets and flags 33% fewer ballots for adjudication versus the baseline.

marks as no votes because these marks did not contain enough dark pixels to pass either threshold and be classified as a vote or flagged for adjudication. Second, the model often classified targets with marks that were filled in and crossed out as votes, because these targets contained a higher percent than the second threshold of dark pixels. Figure 7 shows examples of misclassified targets.

4.2 CNN Model Performance

By comparison, the CNN model outperformed the baseline model in both overall accuracy and number of ballots sent to a human. It had 66.7% fewer misclassifications and 32.8% fewer ballots flagged for adjudication versus the baseline.

The cases where the CNN model produced inaccurate classifications fell into a few categories. First, it appeared to be more sensitive than the baseline model to poor feature extraction and struggled off center targets. Fortunately, there exist more sophisticated techniques for ballot feature extraction than was used in this study [19]. Second, our model struggled with some of the X-marked targets. The CNN model occasionally labeled these targets as empty, causing it to predict no vote where a vote should have been. Figure 7 shows examples where the CNN model failed, but we emphasize that its overall performance was clearly superior to the baseline's when comparing accuracy or adjudications.

Figure 8 shows how each model performed on targets the other classified correctly, incorrectly, or adjudicated. Notably, all marks that the CNN model misclassified were also misclassified or flagged for adjudication by the baseline model.

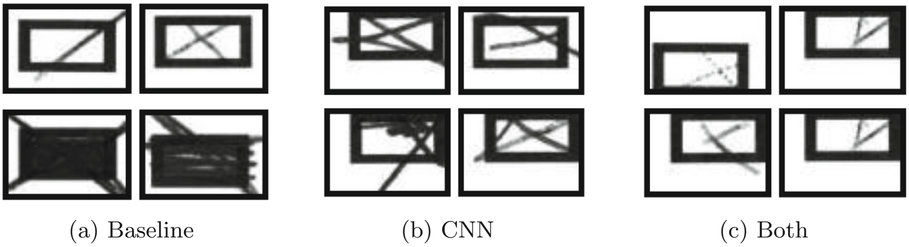
(a) Baseline (b) CNN (c) Both

Fig. 7. Examples of misclassified targets from Humboldt ballots. The CNN performed better overall, but it failed in some cases with X marks or off-center targets.

		Baseline		
		Correct	Adjudicate	Incorrect
CNN	Correct	65,355	1,742	26
	Adjudicate	1,004	430	31
	Incorrect	0	9	15

Fig. 8. Overlapping performance of each model on 68,612 Humboldt targets.

4.3 Computational Costs

An additional metric to consider is computational cost. For the CNN, the most computationally expensive step was training the model. However, training need only be done once for each type of scanner hardware and style of voting target.

Ideally, a pre-trained model can predict labels for ballots at least as fast as they are scanned in, ensuring that the model is not a limiting component of the device as a whole. Today, a typical speed rating for a high-speed central-count optical scanner is on the order of 300 ballots per minute [28]. Different ballots contain vastly different numbers of targets, but an upper bound estimate for a traditional-style ballot might be 128 targets per page. With double-sided ballots, the high-speed scanner would need to process 1280 targets per second to keep up. Both the pre-fitted CNN and the baseline model far exceeded this rate, taking less than a second on a mid-line laptop to label the extracted, preprocessed features from the 68,612 targets in our test dataset. (Although feature extraction adds additional costs, these are the same with both models.) This indicates that the CNN approach can outperform the baseline in both accuracy and adjudication frequency while performing fast enough to keep pace with modern scanners.

4.4 Hybrid Models

After examining the results from the baseline and CNN models, we considered additional models that involved combining the two. We optimized the first of these hybrid models for accuracy. In this model, we flagged a contest's targets for adjudication if either the baseline model or the CNN model labeled any of that contest's targets as adjudicate, or if the two models disagreed on their predictions. This hybrid achieved a higher overall accuracy than either model alone. However, it also required adjudicating significantly more targets than either model alone did. Since this model was better than the CNN model by accuracy but worse by number of ballots adjudicated, it is not clearly an improvement. It is also worth noting that similar results might be possible from the CNN alone by increasing the confidence threshold at which the CNN model flags ballots for adjudication.

The second combined model we considered strove to maintain accuracy while reducing the number of ballots adjudicated. In this hybrid, we used the CNN model as a primary classifier, and when the CNN model chose to adjudicate, we used the baseline model to try to classify the target first. By accuracy, this model was still better than the baseline model but not as good as the CNN alone. By number of adjudications, this method was highly effective. It would be interesting to investigate if one could increase the accuracy of this type of hybrid model by increasing the confidence threshold of the CNN. Like the first hybrid model, since this model was better than the CNN in one aspect but worse in the other, we cannot conclude which is decisively better. Figure 6 shows results for both hybrids.

4.5 Optimized Baseline Model

In addition to the performances of combined CNN and baseline models, we also investigated how a baseline model with different thresholds would have performed compared to the CNN. By starting with the Humboldt voting results and working backwards, it was possible to use a brute force approach to calculate which thresholds would produce the optimal results for this specific dataset given either a minimum accuracy or maximum adjudication rate condition.

If we insist that the baseline model achieves a lower adjudication rate than the CNN, $\alpha = 13.2\%$ and $\beta = 8.1\%$ maximized accuracy. This modified baseline achieved an accuracy of 99.862%—worse than even the original baseline model—and an adjudication rate of 2.035%. Likewise, if we modify the baseline model to have an accuracy higher than the CNN model, $\alpha = 99.8\%$ and $\beta = 1.7\%$ minimized adjudication. While this model had an accuracy of 99.968%, it would be virtually pointless as 97.042% of all contests required adjudication. This strongly suggests that there are types of marks, such as those marked and then crossed out, that simply cannot be correctly identified by a model that only looks at the shading of the target area.

4.6 Pueblo Test Results

We used the Pueblo dataset to more directly compare the CNN model to a deployed election system and to address concerns about whether a CNN could sometimes harm results. That is, in elections where current scanners perform well, would a CNN achieve a comparable accuracy? Once we had determined the efficacy of a CNN for the relatively messy Humboldt dataset, we retrained our model on the comparatively clean Pueblo dataset. Retraining was necessary, because the datasets use different styles of voting targets, and the raw scans, which were captured on different types of hardware, have vastly different intensity response characteristics. We used the same model architecture, only changing input/output sizes for the model layers. This model achieved similar training accuracy and loss to the Humboldt CNN model.

The Pueblo CNN model found 36 contests on 24 ballots with an overvote. When combined with 161 targets where feature extraction failed, this amounted to 0.0067% of the targets in the test dataset, and after accounting for the overvotes, the Pueblo model agreed with the post-adjudication ballot interpretations from the real election for every target in the test dataset of about 35,000 targets. This suggests that a CNN can produce accuracy as good as state-of-the-art deployed systems, while potentially requiring fewer ballots to be adjudicated.

Since the Pueblo dataset had extremely few marginal marks, the baseline also had a very high accuracy and made almost no mistakes, leaving little room to improve upon the accuracy. However, our previous experiments showed that on datasets with a larger variety of marks, such as the Humboldt ballots, our CNN approach can achieve significant improvements to accuracy.

5 Discussion

We trained CNN models on the Humboldt dataset and the Pueblo dataset and found that they match or outperform the baseline threshold-intensity approach in terms of the number of correctly labeled targets and the number of ballots that require adjudication by election officials. A similar approach could be implemented in future elections. Scanner manufacturers could each train a model once on ballots that reflect their particular style of voting targets (e.g., ovals or rectangles) and hardware imaging characteristics (e.g., grayscale or one-bit black and white), then implement the model in a software update for their machines. This would potentially benefit future elections in multiple ways.

The benefits to increased labeling accuracy are clear. Better target classifications mean election results will better match voter intent. Demonstrated accuracy improvements may also increase public trust in the election process. Additionally, despite the expert consensus regarding the importance of rigorous post-election audits as a defense against both fraud and error [20], many states still do not require any form of tabulation audit, and very few perform risk-limiting audits [21]. As a result, the outcomes of the vast majority of contests currently depend on the accuracy of ballot scanners. Even when audits or manual recounts are applied, it is important for initial machine counts to be accurate, because if the audit or recount shows different counts, public confidence is likely to be eroded.

One of the biggest benefits of adjudicating fewer ballots is the time saved. When an absentee ballot is sent for review, election officials need to analyze it in the presence of multiple observers, determine voter intent, and then (for manual adjudication processes) copy the voter intent onto a new ballot and scan it. Reducing adjudication will save administrative costs and improve the speed at which election results are tabulated—which may help further increase voter confidence. Moreover, reducing the number of times voter intent needs to be determined by humans will reduce the potential for bias, subjectivity, and disputes.

5.1 Future Work

Our results suggest that application of machine learning techniques can achieve substantial improvements for the ballot scanning process, but we emphasize that far more work is possible. While our model was able classify targets correctly with greater than 99.9% accuracy, outperforming the baseline model, there are numerous improvements that can be made to further enhance the performance of supervised learning techniques and better understand voter intent.

First, although our CNN model matched the performance of an actual scanner for the Pueblo dataset, which had very few marginal marks, further work is needed to more rigorously quantify the gains from CNN techniques against actual deployed scanners when marginal marks are more common. The baseline model we implemented may be more capable towards marginal marks than some currently deployed tabulators, since it considers intensity within a fairly large region around the voting target, and so may underestimate the potential improvements.

Second, performance can very likely be enhanced by improving on the rather basic feature extraction methods that we used in the bulk of our experiments. Most of the mistakes in the Humboldt model originated from our target crops not being centered. A model trained on more structurally uniform, less variable data should better classify targets. In the Pueblo dataset, our feature extraction used timing marks and was more accurate than the Humboldt extraction. However, not all ballots utilize timing marks, and those that do not would benefit from the application of more sophisticated existing target extraction techniques (e.g., [30]).

Third, the performance of the CNN model can likely be greatly improved by training on a larger corpus of marginal marks, particular X-marks, check marks, and marked-and-crossed-out marks. With more data from these classes, models will be even better equipped to correctly classify these less common marks. Election officials could help accelerate this process by making larger and more complete datasets of scanned ballots available for research.

Fourth, more research is needed to investigate how ML techniques might provide even greater flexibility in understanding voter intent, such as by recognizing and processing marks that are not in the voting targets or in the small area around them. We found several examples of voters making marks and even writing in the margins of the ballots. These marks get ignored by both the current system and by our model. Scanners could potentially make better use of these marks for deciphering voter intent, whether by intelligently processing them or merely recognizing when they call for adjudication.

Finally, there is some evidence that demographic disparities exist in the rate of voter error when using existing ballot scanners [25, p. 19]. Since CNN models perform better when interpreting marginal marks, they might help reduce this bias. Research is needed to fully understand the causes and extent of bias in existing systems and to test how adopting a CNN model would affect it.

6 Conclusion

Marginal marks are a common feature on hand marked paper ballots, and current ballot scanning systems do not adequately account for them. In one dataset, we found that 8.5% of marked targets were not filled in completely, but rather consisted of X-marks, check marks, lightly filled targets, partially filled targets, and various forms of crossed-out targets. While traditional intensity-threshold methods are often able to classify such marginal marks correctly, we identified numerous cases where they either fail or require unnecessary human intervention.

By accounting for different kinds of marks and using a CNN trained to identify them, we were able to make ballot scanning more accurate. Compared to the baseline, we found that our model correctly classifies more targets and reduces the number of ballots sent to humans for review. While additional work is needed, our research indicates that supervised learning has the potential to make ballot scanning smarter by counting ballots both faster and more accurately.

Acknowledgments. The authors thank Marilyn Marks and Harvie Branscomb for assistance acquiring ballot images and the students of EECS 498.5: Election Cybersecurity (Fall 2020) for their suggestions and feedback. We thank the Humboldt County Election Transparency Project and Pueblo County Elections for making ballot images available. We also thank our anonymous reviewers and our shepherd, Catalin Dragan. This material is based upon work supported by the Andrew Carnegie Fellowship, the U.S. National Science Foundation under grant number CNS-1518888, and a gift from Microsoft.

References

1. Bajcsy, A., Li-Baboud, Y.-S., Brady, M.: Systematic measurement of marginal mark types on voting ballots. Technical report (2015). https://nvlpubs.nist.gov/nistpubs/ir/2015/NIST.IR.8069.pdf. NIST
2. Bowen, D.: Top-to-Bottom Review of voting machines certified for use in California. Technical report (2007). https://www.sos.ca.gov/elections/voting-systems/oversight/top-bottom-review/. California Secretary of State
3. Bradski, G.: The OpenCV library. Dr. Dobb's J. Softw. Tools (2000)
4. Clark, A.: Pillow (PIL Fork) Documentation (2015). https://buildmedia.readthedocs.org/media/pdf/pillow/latest/pillow.pdf
5. Colorado Secretary of State Elections Division: Voter Intent: Determination of Voter Intent for Colorado Elections (2013). https://www.broomfield.org/DocumentCenter/View/11702/Voter-Intent-Guide
6. Cordero, A., Ji, T., Tsai, A., Mowery, K., Wagner, D.: Efficient user-guided ballot image verification. In: Electronic Voting Technology/Workshop on Trustworthy Elections. EVT/WOTE (2010)
7. Curling v. Raffensperger, Civil Action No. 1:17-cv-2989-AT (N.D. Ga. Oct. 11, 2020)
8. Dominion Voting Systems: AuditMark. https://www.dominionvoting.com/democracy-suite-ems/
9. He, K., Zhang, X., Ren, S., Sun, J.: Deep Residual Learning for Image Recognition. CoRR abs/1512.03385 (2015). arXiv:1512.03385
10. Humboldt County: November 3, 2009 UDEL Election: Official Canvass Precinct Report. https://humboldtgov.org/DocumentCenter/View/3941/November-3-2009-UDEL-Election-Official-Canvass-Precinct-Report-PDF
11. Hursti, H.: Critical Security Issues with Diebold Optical Scan Design, The Black Box Report (2005)
12. Jones, D.W.: On optical mark-sense scanning. In: Chaum, D., et al. (eds.) Towards Trustworthy Elections. LNCS, vol. 6000, pp. 175–190. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-12980-3_10

13. Kamarck, E., Ibreak, Y., Powers, A., Stewart, C.: Voting by mail in a pandemic: a state-by-state scorecard, Brookings Institution (2020). https://www.brookings.edu/research/voting-by-mail-in-a-pandemic-a-state-by-state-scorecard/
14. Kiayias, A., Michel, L., Russell, A., Shvartsman, A.: Security assessment of the Diebold optical scan voting terminal (2006). https://voter.engr.uconn.edu/voter/wp-content/uploads/uconnreport-os.pdf
15. Kiayias, A., Michel, L., Russell, A., Shashidhar, N., See, A., Shvartsman, A.: An authentication and ballot layout attack against an optical scan voting terminal. In: USENIX/ACCURATE Electronic Voting Technology Workshop (EVT) (2007)
16. LeNail, A.: NN-SVG: publication-ready neural network architecture schematics. J. Open Source Softw. **4**(33), 747 (2019). https://doi.org/10.21105/joss.00747
17. Lindeman, M., Stark, P.: A gentle introduction to risk-limiting audits. IEEE Secur. Priv. **10**, 42–49 (2012)
18. McDaniel, P., Blaze, M., Vigna, G.: EVEREST: evaluation and validation of election-related equipment, standards and testing. Technical report (2007). http://siis.cse.psu.edu/everest.html. Ohio Secretary of State
19. Nagy, G., Lopresti, D., Smith, E.H.B., Wu, Z.: Characterizing challenged Minnesota ballots. In: 18th Document Recognition and Retrieval Conference (2011)
20. National Academies of Sciences, Engineering, and Medicine: Securing the Vote: Protecting American Democracy. The National Academies Press, Washington, DC (2018). https://www.nap.edu/catalog/25120/securing-the-vote-protecting-american-democracy
21. National Conference of State Legislatures: Post-Election Audits (2019). http://www.ncsl.org/research/elections-and-campaigns/post-election-audits635926066.aspx
22. Poulos, J., Hoover, J., Ikonomakis, N., Obradovic, G.: Marginal Marks with Pixel Count, U.S. Patent 9,710,988 B2 (2012)
23. Pueblo County Elections: Ballot Images, November 2020 Election. https://county.pueblo.org/clerk-and-recorder/ballot-images
24. Rivest, R.: On the notion of 'software independence' in voting systems. Phil. Trans. R. Soc. A **366**(1881), 3759–3767 (2008)
25. State of Georgia: Report of The 21st Century Voting Commission, (2001). https://voterga.files.wordpress.com/2014/11/21st_century_report.pdf
26. Sultana, F., Sufian, A., Dutta, P.: Advancements in image classification using convolutional neural network. ICRCICN (2018). https://doi.org/10.1109/icrcicn.2018.8718718
27. Toledo, J.I., Cucurull, J., Puiggalí, J., Fornés, A., Lladós, J.: Document analysis techniques for automatic electoral document processing: a survey. In: Haenni, R., Koenig, R.E., Wikström, D. (eds.) VOTELID 2015. LNCS, vol. 9269, pp. 129–141. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-22270-7_8
28. U.S. Election Assistance Commission: Central Count Optical Scan Ballots (2008). https://www.eac.gov/sites/default/files/documentlibrary/files/Quick_Start_Guide_-_Central_Count_Optical_Scan_Ballots.pdf
29. Verified Voting: The Verifier: Polling Place Equipment (2021). https://www.verifiedvoting.org/verifier/
30. Wang, K., Kim, E., Carlini, N., Motyashov, I., Nguyen, D., Wagner, D.: Operator-assisted tabulation of optical scan ballots. In: Electronic Voting Technology/Workshop on Trustworthy Elections. EVT/WOTE (2012)
31. Xiu, P., Lopresti, D., Baird, H., Nagy, G., Smith, E.B.: Style-based ballot mark recognition. In: 10th International Conference on Document Analysis and Recognition (2009)