Implementing an audio side channel for paper voting

Kristjan Krips[1], Jan Willemson[1,2], and Sebastian Värv[1]

[1] Cybernetica AS
Ülikooli 2, 51003 Tartu, Estonia
{krisjan.krips,jan.willemson,sebastian.varv}@cyber.ee
[2] Software Technology and Applications Competence Center
Ülikooli 2, 51003 Tartu, Estonia

Abstract. In the ongoing debate between the proponents of electronic and paper voting, a frequently used argument is that electronic voting is susceptible to electronic attacks, and those are less detectable by a human than physical ones. This paper contributes to the research of electronic attacks against paper voting by building a proof-of-concept classifier for audio samples recorded while writing numbers. Such a classifier can be used to break the privacy, for example, in case of preferential voting ballot sheets, or voting systems where the voter must fill in the candidate number. We estimate the quality of the classifier and discuss its implications to the physical security measures of polling stations and ballot design.

1 Introduction

Voting is a form of public opinion polling used when a group of people needs to take a common decision. The size of the group may vary from just a few persons to whole societies, and the decisions may vary from selecting a beauty queen to determining who is going to rule the country for the next 5 years.

The bigger implications the decision has, the more critical role is played by the actual voting and vote counting processes. There are a number of requirements set to contemporary voting systems, and thick rule books describing how to enforce them.

Unfortunately, these rules can be contradictory. In order to gain public acceptance of an election result, all the processes should be fully auditable, ideally by everyone. On the other hand, to prevent coercion and vote-buying, the actual votes should remain secret, introducing an inherently non-auditable component into the system.

It is also the case that important elections tend to have a large voter set easily reaching millions of people. This has implications on the vote counting. A single person is unable to count millions of votes in a reasonable time frame, so this work has to be distributed between many people, not all of whom are equally careful or trustworthy. If a physical medium like paper is used for voting, there can also be ambiguous markings that need interpretation, and this interpretation

may depend on the interpreter. And last-but-not-least, organizing voting based on physical carriers is a huge logistical challenge, requiring all of these millions of people to go to polling stations and collecting the ballots later.

These problems have motivated research in alternative vote casting mechanisms, including electronic ones. Starting from T.A.Edison's "Electrographic Vote Recorder and Register"[3], various methods including voting machines and remote vote casting over Internet have been proposed and tried out.

While helping to ease some of the inherent difficulties of elections, electronic means can bring up new concerns. Humans can not control digital environments directly and need to rely on imperfect interfaces. Also, it is hard to be sure that a digital device acts according to its specification and does not include anything extra, like malware.

Another example of out-of-specification behaviour is the existence of side channels threatening vote privacy. Perhaps one of the most notorious examples of potential implications of such problems was observed in the Netherlands. As those events greatly inspired our current research, we will make a short recap here.

1.1 The rise and fall of electronic voting in the Netherlands

Netherlands has been a true pioneer of electronic voting. Legislation allowing machine voting was put in place already in 1965, and the first voting machines appeared in 1966 [8]. The first attempts to automate counting were done in late 1980s. From 1994, the government actively promoted the usage of electronic apparatus in voting [6]. By 2005, the Dutch market had been divided by two bigger suppliers of the voting machines – Nedap and Sdu [8]. There had been a few complaints e.g. favouring a candidate with number 31 due to his/her name being displayed on top of the second column of candidates [6], but in general the public trust in voting machines seems to have been rather high.

However, in 2006, a series of events took place that changed the situation drastically. First, during 2006 elections a fraud suspicion was raised in one of the districts where Nedap voting machines were used. After repeated shadow elections and several rounds in court, this led to a conviction [8].

As a reaction to this (and probably also earlier complaints), a civil activist and hacker Rop Gonggrijp initiated a movement called "Wij vertrouwen stemcomputers niet" ("We don't trust voting computers"). He got access to some of the Nedap machines, managed to reverse engineer the source code and demonstrated the ease of maliciously replacing the onboard chips [6].

The other major problem Gonggrijp and his collaborator Maurice Wessling discovered was the possibility to eavesdrop electromagnetic emanations (called a TEMPEST attack) which, under certain circumstances, revealed the voter's party preferences. More precisely, the name of one of the parties (Christen-Democratisch Appèl) contained a diacritic letter (è) and in order to display this, the voting machine screen had to be switched to a different mode. It was

[3] US patent no. 90,646, patented June 1st, 1869

this switch that could be detected from a distance using rather standard radio equipment [5].

The fix for this problem was straightforward (just use e instead of è), but the authorities also looked at the Sdu machines and the electromagnetic emanation problem was much worse there. In the beginning of 2007, Sdu attempted to re-certify its machines, but they managed to deliver a device for testing that did not pass other requirements, so this attempt eventually failed. As a result, in October 2007, the existing regulation allowing voting machines was withdrawn [6]. The Netherlands has been using 19th century paper voting ever since.

1.2 Side channel attacks on voting

As mentioned above, the TEMPEST exploit implemented by Gonggrijp and Wessling falls into the category of side channel attacks. These sorts of attacks are in general relatively difficult to prevent since, by definition, they make use of some out-of-system-model feature like power consumption, message timing, etc.

Electromagnetic emanation leakage is not the first side channel vulnerability considered for voting. Taking a photo of the ballot with a phone or some other device is a well-known privacy problem [2]. Moran and Naor note that in case Direct Recording Electronic (DRE) equipment posts encrypted votes on a bulletin board, posting timing can be used by a compromised DRE machine to reveal the voter preference [9].

An interesting side channel attack (called Three-Pattern) against the Three-Ballot optical scan voting system was described by the original author Ronald Rivest himself [11]. As the voter in this system has exponentially many choices for encoding her vote on the ballot, the coercer may convince her to do so in a predefined pattern, checking later from the public bulletin board that the pattern has been followed. This leakage is actually so severe that, according to Rivest, "...it makes ThreeBallot much less attractive than I had originally hoped for" [11].

Recently, Toreini *et al.* have improved paper fingerprinting techniques. Their approach allows to create short fingerprints of physical paper sheets using off-the shelf apparatus like overhead projector and photo camera with a sufficiently good resolution. As a result, this makes the vote privacy violation attack proposed by Calandrino *et al.* [3] more accessible to a moderately-resourced attacker. This example demonstrates clearly how advancement of technology also makes paper voting more insecure.

In this paper, we will be considering another type of emanation occurring during paper voting, namely the sound that the pen makes while marking the ballot.

The feasibility of extracting (capital) letters from the audio recording was studied by Yu *et al.* in 2016 [13]. Their results are encouraging, but also show significant challenges. If the training data from the attack subjects can be collected in advance and the position of the microphone can be well predicted, the letter recognition precision can achieve almost 65%. However, if the subjects' handwriting can not be studied beforehand, precision drops below 27%. The

authors of [13] also extend their attack to recognising words from a predefined dictionary and achieve the best case accuracy of 50-60%.

We will concentrate our efforts on a smaller set of glyphs to recognise, namely Arabic numerals. We will study how well decimal digits can be recognised from the audio samples of writing them, and discuss the implications to voting privacy and ballot design.

The rest of the paper is organised as follows. In Section 2 we will discuss different types of ballot designs and their implications on the vulnerability to audio side channel attacks. Section 3 describes audio sample classification and Section 4 discusses its implications on security of various election settings. Finally, Section 5 draws some conclusions and sets directions for future work.

2 Types of ballots

The primary sources of requirements for the ballot sheet design are local voting traditions and the implied legal requirements. Susceptibility to audio side channels has most likely not been taken into account as a concern. Hence we start our discussion by reviewing some of the typical ballot designs from this viewpoint.

A frequently used ballot type lists a number of candidates and requires marking one or several of them somehow (writing "X" marks next to one's preferences, crossing some candidates out, etc.). Even though audio side channels against such ballot designs are still possible (e.g., the attacker may draw conclusions based on the timings between writing several "X"-s), they require development effort that remains outside of the scope of the current paper.

Good detection accuracy can potentially be obtained for the ballots allowing write-ins, e.g. leaving an empty slot on the ballot sheet to allow voting for an unlisted candidate.[4] As the voters are not forced to write the names in capital letters, recognising each person's handwriting becomes a major problem, and without reliable personalised training data the results can be expected to be considerably worse than those of Yu *et al.* [13].

Still, we can consider a subset of the handwriting recognition problem. For example, in a referendum the participant might be asked to make a binary decision by writing either "Yes" or "No" to the referendum sheet. Such ballots have been previously used e.g. in Australian constitutional referendums and are currently used e.g. in Swiss referendums. We can see that the corresponding ballot design leaks information that can be classified as Yu *et al.* have already shown. Due to the uniqueness of letters and the lengths of the words it should be easy to distinguish between the two cases.

However, there is a specific type of write-ins that has not yet been considered, namely numbers. This is the most promising target of attack for an audio side channel, because the amount of decimal digits is limited to 10, and the variance

[4] This option has been used to cast protest votes. For example, in 1985, Donald Duck received 291 votes in Sweden. As a result, voting for non-existing candidates was prohibited in Sweden starting from 2006: `https://abcnews.go.com/Entertainment/WolfFiles/story?id=91051&page=1`.

of handwritten numbers between different individuals can be expected to be smaller compared to the variance of handwritten letters.

The most common types of ballots where the voter is expected to fill in some numbers come from preferential voting, e.g. single transferable vote (STV) systems (see an example ballot from the Tasmanian House of Representatives elections in Figure 1). Similar kinds of ballots are used, for instance, in:

- Ireland for municipal, parliamentary and European Parliament elections,
- Malta for municipal, parliamentary and European Parliament elections,
- Northern Ireland for European Parliament elections,
- Scotland for municipal elections,
- Austria for European Parliament elections (preference number is optional),
- Australia for electing the Senat and for electing the House of Representatives.

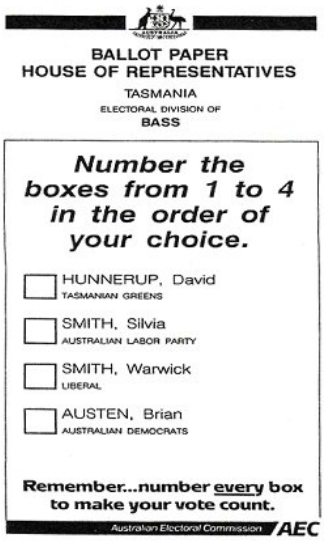


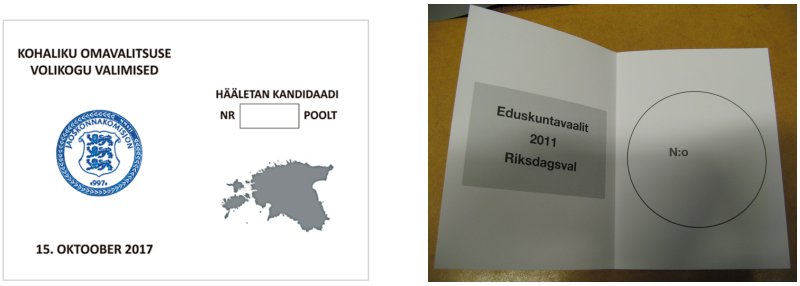
Fig. 1: An example of the Tasmanian election ballot.[5]

When implementing an audio side channel attack against a preferential ballot, we can largely expect to detect two kinds of patterns. First, when we hear the numbers written in the order 1-2-3-4-..., the voter is probably filling her preferences in in the ascending order and finding the correct slots on the fly. Without looking at the timings between the numbers, this pattern does not reveal the voter preferences.

[5] Australian electoral systems, `https://www.aph.gov.au/About_Parliament/Parliamentary_Departments/Parliamentary_Library/pubs/rp/RP0708/08rp05`

However, if the voter uses some other order of the numbers, she can be conjectured to fill the ballot from start till the end of the slot sequence, and her preferences leak. This may be expected to be the case with higher probability when the number of slots to fill is smaller.

There are also some countries (e.g. Estonia and Finland) where the voter is expected to write the candidate number on the ballot (see Figure 2). In these cases the audio side channel has the potential of completely breaking the vote privacy.



(a) Ballot used in Estonia for the municipal council elections in 2017 [1]

(b) Ballot used in Finland for the parliamentary elections in 2011. The same ballot design was also used in the 2015 elections.

Fig. 2: Examples of ballots that are designed to be filled with numbers.

The core contribution of this paper is studying the feasibility of identifying the digits by the sound of handwriting. We have created a proof-of-concept implementation that takes an audio sample, splits it into digits and then tries to recognize them. We also created a classifier which performs this task.

The following Section will describe our results in more detail.

3 Audio sample preprocessing and classification

By looking at the waveforms of recordings that correspond to the writing of different digits, it can be observed that the representations of digits are more or less unique. Thus building a good automatic classifier should at least theoretically be possible.

To verify this hypothesis, we conducted several experiments. First, we collected a number of writing samples from volunteers (see Section 3.1 for more details).

Next we tried the standard step of converting the samples into the frequency domain by using fast Fourier transform (FFT). However, if we would only apply FFT, we would get the frequency distribution for the sample, but lose the time

dimension. On the other hand, time dimension carries useful information about the digits following the movement of the pen or pencil on the paper. Therefore, we decided to transform the samples into spectrograms. Spectrograms are created by moving a window over the audio sample and applying FFT to the corresponding audio fragments. This gives a representation of the sample where one dimension represents frequency and the other represents time. An example of the result is shown in Figure 3.

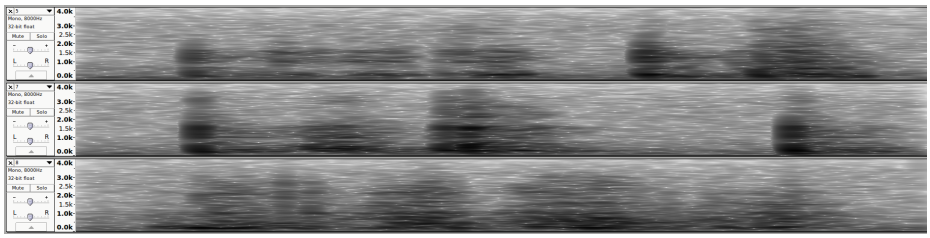


Fig. 3: Spectrogram representations of numbers five, seven and eight.

3.1 Recording and preprocessing

We tested several microphones to find out which one is best suited for the task. The following devices were used: HP laptop, iPhone SE, Jabra Speak 410 and Rode VideoMic Pro. The first three devices had omnidirectional microphones, while Rode VideoMic Pro was a directional cardioid microphone. Comparison of the technical parameters of the microphones is given in Table 1.

Table 1: Comparison of tested recording devices. There was no technical specification available for the microphones in HP laptop and iPhone.

	number of microphones	**type**	**range**	**sensitivity**
HP laptop	2	omni-directional	N/A	N/A
iPhone SE	3	omni-directional	N/A	N/A
Jabra 410 Speak	1	omni-directional	100 Hz - 10 kHz	N/A
Rode VideoMic Pro	1	directional	40 Hz - 20 kHz	-38dB re 1V/Pa ± 2dB @ 1kHz

Testing showed that the laptop microphone was not able to capture handwriting as it could not distinguish the signal from background noise. Rode VideoMic

Pro and the microphone of iPhone SE were able to capture the signal, but the quality was not as good as we got from Jabra Speak 410. It was a bit surprising that the more expensive Rode VideoMic Pro was not able to capture the signal as well as a common conference call device. Therefore, we decided to use Jabra Speak 410 for collecting the training data.

We prepared a sheet of square cells for collecting the samples in order to make the process as uniform as possible. The recording was performed in a closed office room which blocked most of the outside noise. Each volunteer was asked to fill in at least one sheet of ten rows, such that each row would contain all the digits from 0 to 9 once. In addition, the volunteers were asked to leave a small pause after writing each digit to make automatic labelling of the samples easier. The same room and the same table were used for all the samples. The locations of the microphone and the sheet were kept the same throughout the sample collection, with the microphone placed in about 15cm from the edge of the sheet.

Once we had the samples, the next task was to label them to prepare training data for the automatic classifier. As the samples were written on the sheet in a predefined order, we were able to create a script to extract and label the samples. However, manual review of the samples was still necessary to ensure correct operation of the script.

Now that the labelled samples were ready, they had to be prepared for analysis. For that, we converted stereo recording to mono and normalized the tempo. We used WSOLA algorithm [12] to transform the samples such that all of them would have the length of 0.55 seconds. It is important to note that WSOLA does not change the pitch of the sound, otherwise the change of tempo could distort the representation of the digit.

3.2 Building the classifier

We used the k-nearest neighbors algorithm (k-NN) [4] for the classification task. One of the reasons to prefer this method is its capability of producing good results with a small training set. The method works by calculating distance between all samples and then uses majority vote on k nearest samples to determine the class. This was also one of the reasons for normalizing the tempo of the samples as it allowed us to represent the samples as arrays of the same length and therefore align the corresponding frequencies. We pre-processed the data by creating a spectrogram representation from each sample and flattened the output (an array or arrays) to get a one-dimensional array.

We used scikit-learn [10] implementation of the k-NN method to build the model. To use it, the dataset was split into training and testing sets using the `train_test_split` function of scikit-learn. This method allowed us to make sure that the labels would be uniformly distributed in the output sets. The dataset was randomly split into training and test sets so that 10 percent of the samples were used for testing. As the splitting was done on the whole dataset, the ratio of training data to test data did not necessarily hold for the samples belonging to one individual. Thus, individuals might have been over- or under-represented in the training set and test set.

We tested multiple distance metrics to find the one that is most suitable for the representation of the audio data. The results showed that Canberra distance [7] gave significantly better results compared to other distance metrics.

Finally, we used cross-validation for parameter tuning in order to obtain the optimal value of k. We created a list of odd integers as the candidates, fitted a model for each value of k and used cross-validation to determine the k value which gave the best out-of-sample accuracy. In our case, the optimal value for k turned out to be 7.

3.3 Classification results

We used cross-validation to measure the out-of-sample accuracy of the model. Cross-validation partitions the dataset into n equally sized non-overlapping sets, $n-1$ sets are used for training and the n-th set is used for validation. This process is repeated n times, so that each set is validated once. Overall result is calculated by averaging accuracy over all partitions.

Our dataset consisted of 1676 samples and contained recordings from 11 volunteers. Some of the volunteers contributed more than one data sheet and in one case only part of the data sheet recording was usable due to the corruption of data.

We used scikit-learn implementation of 10-fold cross-validation which uses stratified KFold partitioning strategy. This method provided that uniform number of labels was assigned into each subset. For the classification we used aforementioned k-NN classifier with hyperparameter $k = 7$ as it was previously found to be best suited for our dataset by producing best out-of-sample accuracy. The 10-fold cross validation with the given configuration produced an accuracy of 60.14%. The corresponding confusion matrix can be seen in Figure 4.

We can see from the confusion matrix that the digits 8 and 9 have lower detection accuracy compared to others. One of the reasons for this might be the way how the implementation of scikit-learn breaks ties. Namely, in case of a tie the winner is picked according to the ordering of the classes. Thus, when there is a tie between, say, digits 3 and 8, the first one would win, causing 8 to be determined less.

The low accuracy of 8 and 9 might also be caused by their placement on the data sheet with respect to the microphone. The data sheet was in landscape mode during the recording and the microphone was placed close to the top middle part of the sheet. Therefore, the recorded signal of the digits that were written to the middle of the sheet should have slightly better quality compared to the digits on the sides of the sheet. This reasoning seems not to hold for 0 and 1, but this might be explained by their rather unique audio fingerprint.

Next, we ran a test to find the accuracy for the case when training data is available for the test subject. We took datasets from eleven volunteers and split them into test sets and training sets so that every person contributed 10% of their stratified data points to the test set and the remainder was used for training. Each person had 100 labelled data points and thus 1000 samples were used for training and 100 for testing. Results showed that by using such data on

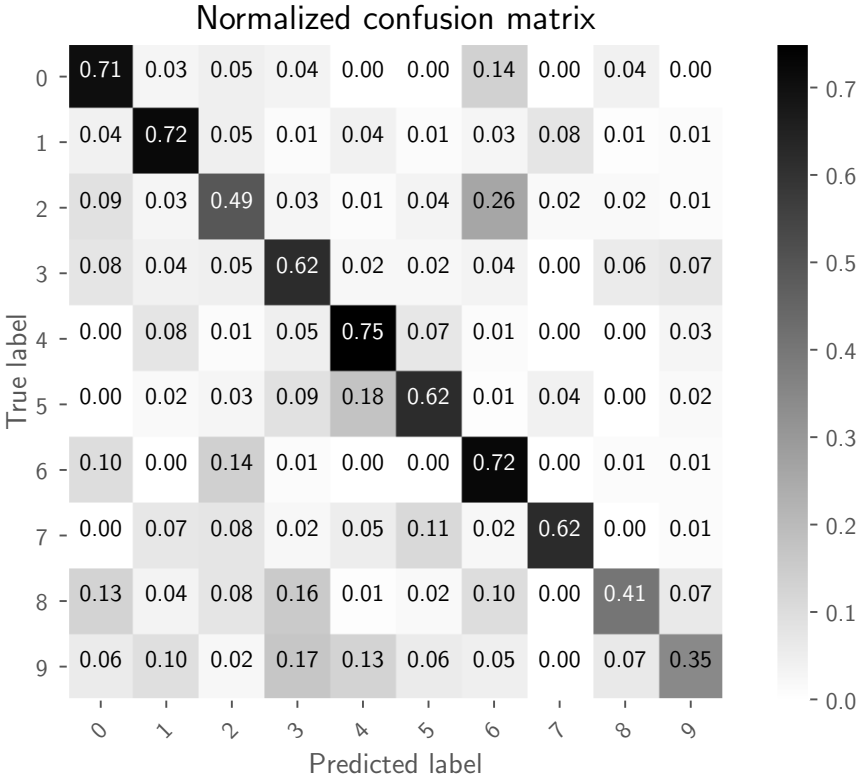


Fig. 4: A confusion matrix that was created from the output of cross-validation. The accuracy of cross-validation was 60.14%.

average 70.6% of digit predictions were accurate. This result loosely corresponds to the 65% outcome of the experiment by Yu *et al.* [13].

However, the more interesting question concerns usefulness of the classifier when the subject's training data is unavailable. We simulated this situation by selecting one data sheet recording from each of the eleven volunteers. Then we ran eleven tests so that in each test the datasets of ten volunteers was used for training and the data of the volunteer was used for testing the model. Again, training was performed with 1000 samples and the remaining 100 samples were used for validation. The results showed an average accuracy of 49%, with the minimal accuracy of 37% and maximum 65%, respectively. This accuracy can probably improved by collecting more training data.

We also observed an interesting phenomenon during our tests. There was one potential volunteer coming from a completely different cultural background, and the audio samples extracted from his recordings were classified with significantly lower probability.

Visual inspection of his handwriting revealed that this person had a completely different style of writing the numbers, most probably originating from the way numbers are taught in the schools of his country of origin. Thus, in order to achieve good detection accuracy, the volunteers who contribute to the training data should represent the cultural background of the test subjects.

4 Discussion

As expected, detecting digits from audio samples can give better results than that of letters. Compared to about 27% average accuracy of letter detection reported by Yu *et al.* [13], we were able to achieve 49% in the setting where samples from the subject are not available for training.

In the context of elections, the attacker is not typically interested in just one digit, but the whole composition of the ballot. Making use of the fact that several digits need to be written, the attacker may be able to compensate for poor detection of some of them.

For example, in the case of a preferential ballot it is known that all the numbers 1-2-3-... should occur, so if there is one sample that can be interpreted either as 2 or 6 and another one that is definitely 6, we know that the first one must be 2.

Similar reasoning applies for the ballots where the voter needs to write the candidate number. For example in case of Estonia, the candidate number consists of three digits, so the expected correct detection probability is $0.49^3 \approx 0.118$, but not all of the possible triplets correspond to existing candidate numbers. Note that the audio side channel can also be used to detect which candidates the voter *did not* select with high probability. This information may be of equal interest for the attacker in the coercion setting.

Success of the audio side channel attack in the setting of paper voting directly depends on the quality of the audio samples the attacker is able to capture. This quality in turn depends on several aspects: amount of background noise, quality of the microphone and the ability to place the microphone into a good location.

Adding more noise in the polling station does not work as a good countermeasure, since it may have a general irritating effect on the voters. In case the level of the background noise is low, our experiments show that already a mid-class microphone can get relatively good results.

Hence, the main success factor that both the attacker and defender can influence is the microphone placement.

We have conducted no research on the physical protection measures of polling stations, but we conjecture that these measures mostly do not take the threat of audio surveillance into account. There are several strategies the attacker may use to plant the microphones into the voting booths. He may try to access the booth tables in the storage before elections, or assume the role of a voter himself, entering the booth to both mark his own ballot and to leave a microphone there.

Assuming physical access to the voting booths, a similar attack of planting video recording equipment is conceivable. Contemporary cameras also have

miniature size; however, they require a direct line of sight, restricting the choice of potential locations. We have not studied the effect of microphone placement extensively, but our testing shows that the signal one gets when attaching a microphone under a wooden table is actually pretty strong and clear.

The only reasonable countermeasure against audio side channel attacks is regular inspection of the voting booths during the elections to detect illegitimate recording equipment. In principle, changing the ballot designs to avoid write-ins could also help, but this may require changing the whole voting tradition and may hence not work in practice. Also, alternative designs (like marking some candidates with "X"-s) may be vulnerable to other side channel attacks of timing, triangulating the locations of the marks, etc. Studying such side channels is an interesting avenue for future research.

And last-but-not-least we would like to emphasize that the privacy-leaking side channel is inherently an issue of paper-based elections, and, to an extent, less so in case of remote electronic voting. Of course, one can imagine video recording equipment installed in someone's home, but such an attack would scale much worse than planting a microphone in a polling booth.

Thus, the main wide-scale privacy attack vector against Internet voting would still require using specially crafted malware.

Note that just an attack against vote privacy is not very interesting on its own, it becomes a real problem in conjunction with coercion. Coercion, in turn, implies the need to target specific voters.

The ease of installing malware on the computers of a particular set of target persons may depend on many aspects like physical security of their homes and general level of digital hygiene. However, we argue that determining the polling station where the target group goes voting and planting microphones there is an attack of lower technical complexity.

Planting the recording equipment can be performed by a corrupt voter (who may be the attacker himself or a voter bribed by the attacker). The attacker may then remain in the polling station observing the times when the voters enter the booth. The recording equipment, in turn, may save time stamps of the collected writing samples, and the time stamps can later be cross-referenced with the times recorded by the observing attacker. Alternatively, the recording equipment may have radio communication capability, reporting the recordings as soon as they have been detected.

Note that this attack requires significant human involvement as the attacker would need to visually identify the voters who enter the booth. However, this step can also be automated by using facial recognition software together with a corresponding personalized facial features database. At the time of this writing (summer 2018), such databases are probably not yet available for medium-level attackers, but they are being built by intelligence organizations based on vast amount of personal images available via social networks.[6] It is only a matter of time when such databases can be bought on black markets.

[6] `https://www.forbes.com/sites/thomasbrewster/2018/04/16/huge-facebook-facial-recognition-database-built-by-ex-israeli-spies/`

We stress again that our final argument is made only about vote privacy violations via side channel leakages, and does not seek to compare security of paper and remote electronic voting otherwise.

5 Conclusions and further work

There are entire communities devoting their efforts to proving superiority of paper voting over its electronic counterpart (like `https://www.verifiedvoting.org/` and `http://handcountedpaperballots.org/`). An important argument used in such efforts is that high-tech solutions are vulnerable to high-tech attacks, and the latter ones are not yet understood well enough to provide satisfactory mitigation measures.

What proponents of such arguments often do not mention is that high-tech methods can also be used against low-tech elections. The current paper stressed this point by presenting an audio side channel attack against the form of paper voting where the voter is expected to fill in the ballot by writing some numbers.

Success of such an attack in practice depends on many aspects like noisiness of the polling station and the ability to place microphones well enough to capture good-quality audio samples. However, we argue that the resulting leakage is considerably more severe than that of the TEMPEST attack by Gonggrijp and Wessling that forced all electronic voting initiatives in the Netherlands to halt in 2007. Our attack has the potential of revealing the exact voter preference, whereas the attack by Gonggrijp and Wessling only leaked whether the vote was given to one specific party (CDA) or not.

We are not claiming that all the paper voting should be discontinued, but we do advocate for balancing the criticism against electronic voting based on the problems that actually exist in the case of paper voting as well. Our research also implies that side channel attacks should be taken into account while designing the ballot sheets and planning physical protection measures in the polling stations.

This paper presented an attack on a rather specific form of paper voting. However, there are also many other designs of ballot sheets that deserve attention from the viewpoint of advanced technological attacks as well. This remains the subject for future research.

Acknowledgements

The research leading to these results has received funding from the Estonian Research Council under Institutional Research Grant IUT27-1 and the European Regional Development Fund through the Estonian Centre of Excellence in ICT Research (EXCITE) and the grant number EU48684. We would also like to thank all the volunteers contributing the writing samples used in this research, anonymous reviewers for their comments, and our shepherd Dr. Marco Prandini for helpful and thought-provoking discussions.

References

1. Kohaliku omavalitsuse volikogu valimiste käsiraamat 2017. https://www.valimised.ee/sites/default/files/uploads/kov2017/KOV2017_kasiraamat_web.pdf.
2. Ben Adida and C Andrew Neff. Ballot Casting Assurance. In *USENIX Electronic Voting Technology Workshop*, 2006.
3. Joseph A Calandrino, William Clarkson, and Edward W Felten. Some Consequences of Paper Fingerprinting for Elections. In *EVT/WOTE*, 2009.
4. Thomas M. Cover and Philip J. Hart. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1):21–27, September 1976.
5. Rop Gonggrijp and Willem-Jan Hengeveld. Studying the Nedap/Groenendaal ES3B Voting Computer: A Computer Security Perspective. In *Proceedings of the USENIX Workshop on Accurate Electronic Voting Technology*, Berkeley, CA, USA, 2007. USENIX Association.
6. Bart Jacobs and Wolter Pieters. Electronic Voting in the Netherlands: from early Adoption to early Abolishment. In *Foundations of security analysis and design V*, pages 121–144. Springer, 2009.
7. G. N. Lance and W. T. Williams. Computer programs for hierarchical polythetic classification ("similarity analyses"). *The Computer Journal*, 9(1):60–64, 1966.
8. Leontine Loeber. E-voting in the Netherlands; from general acceptance to general doubt in two years. In *3rd international Conference on Electronic Voting*, volume 131 of *GI-Edition Lecture Notes in Informatics*, pages 21–30. 2008.
9. Tal Moran and Moni Naor. Receipt-Free Universally-Verifiable Voting with Everlasting Privacy. In Cynthia Dwork, editor, *Advances in Cryptology – CRYPTO 2006*, volume 4117 of *LNCS*, pages 373–392, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.
10. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.
11. Ronald L Rivest. The ThreeBallot Voting System, 2006. http://people.csail.mit.edu/rivest/Rivest-TheThreeBallotVotingSystem.pdf.
12. W. Verhelst and M. Roelands. An overlap-add technique based on waveform similarity (WSOLA) for high quality time-scale modification of speech. In *1993 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 2, pages 554–557 vol.2, April 1993.
13. Tuo Yu, Haiming Jin, and Klara Nahrstedt. Writinghacker: Audio based eavesdropping of handwriting via mobile devices. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 463–473. ACM, 2016.