Cutting through the Emissions: Feature Selection from Electromagnetic Side-Channel Data for Activity Detection

Asanka Sayakkara, Luis Miralles-Pechuán, Nhien-An Le-Khac, and Mark Scanlon

UCD DUBLIN

UCD Forensics and Security Research Group

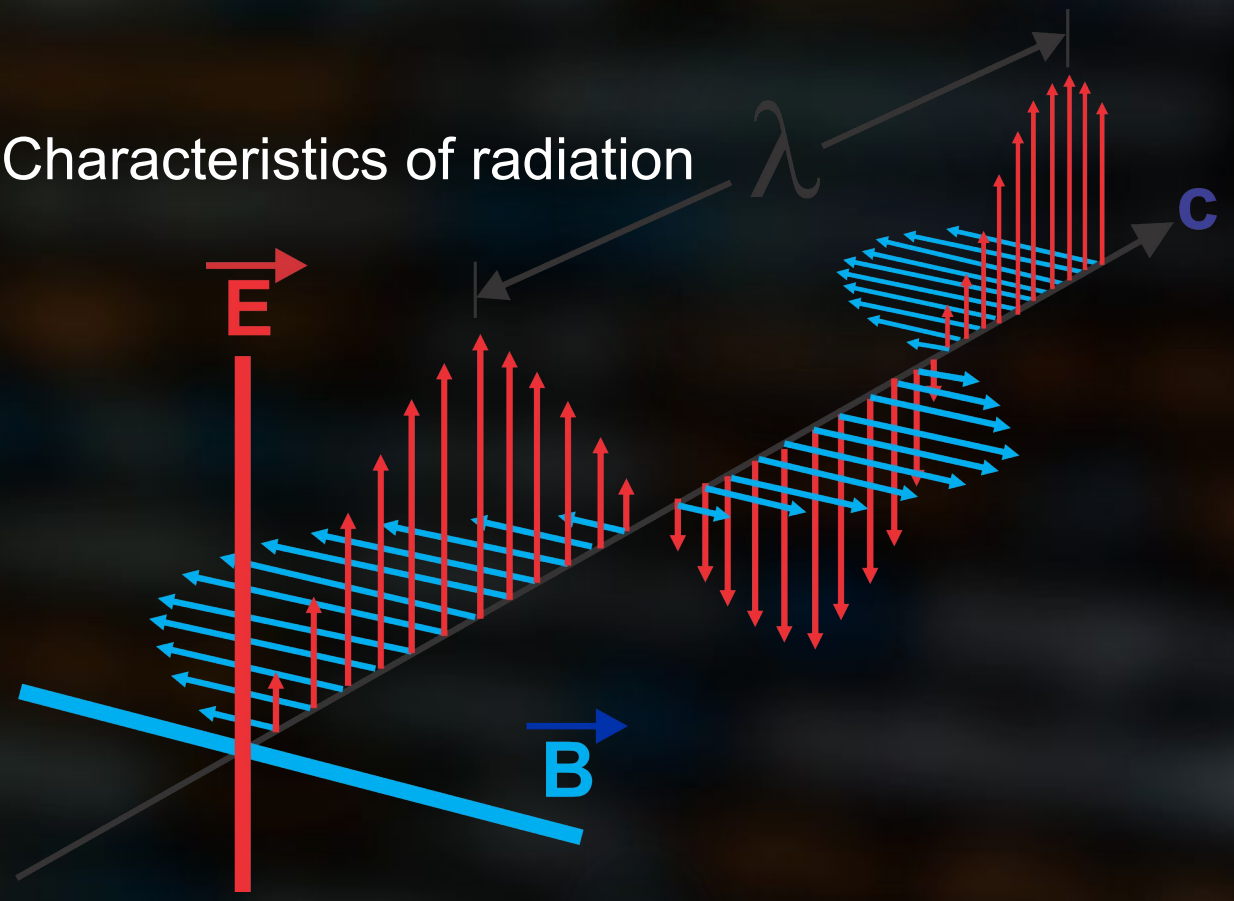
Electromagnetic Side-Channel Analysis

Time-varying electrical currents → Electromagnetic radiation

Nature of the time-varying current ← Characteristics of radiation

EM radiation from computer processors leak information

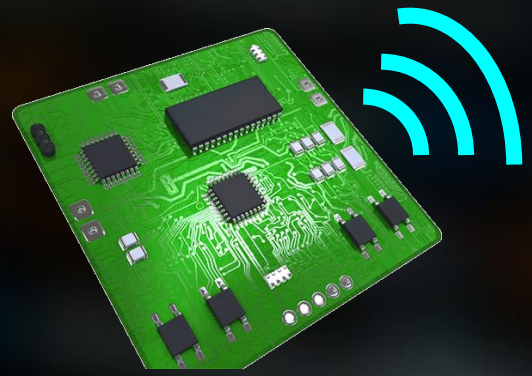
EM side-channel analysis (EM-SCA)



Electromagnetic Side-Channel Analysis

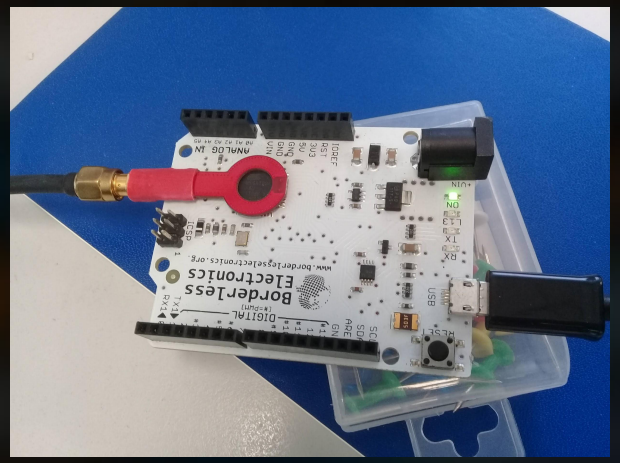
- EM-SCA is used for various information security purposes
  a. malicious modification detection to HW and SW
  b. device internal state detection
  c. cryptographic key retrieval

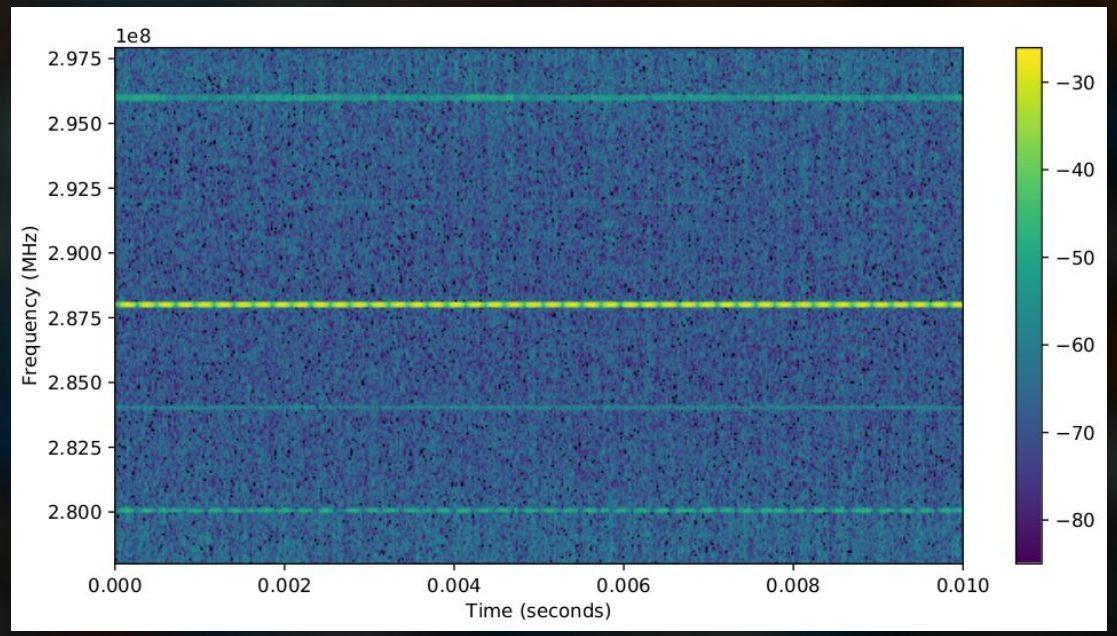
- Digital forensics benefits from EM-SCA.
  a. some devices don’t work with typical forensic methods, e.g., IoT
  b. trying to access data invasively can tamper the evidence
  c. EM-SCA can help gain forensically-useful insights

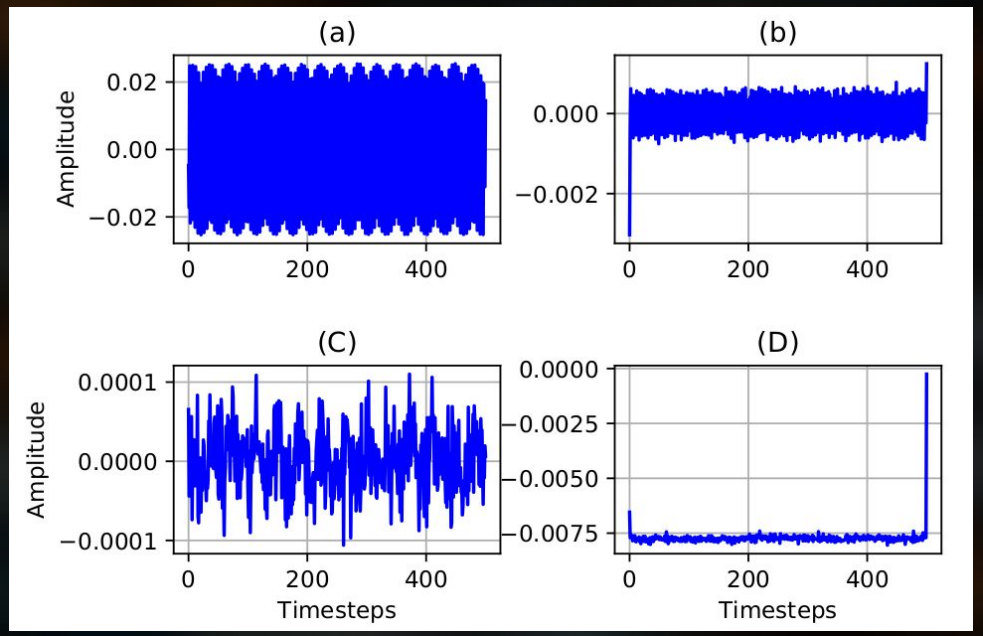


Troubled with dimensionality…

- We don’t know the exact information-leaking frequencies.
- So, we observe across very wide bandwidths.
- Resulting data is highly dimensional and not possible to use for real-time EM-SCA purposes.
- **How do we recognize the useful frequencies?**







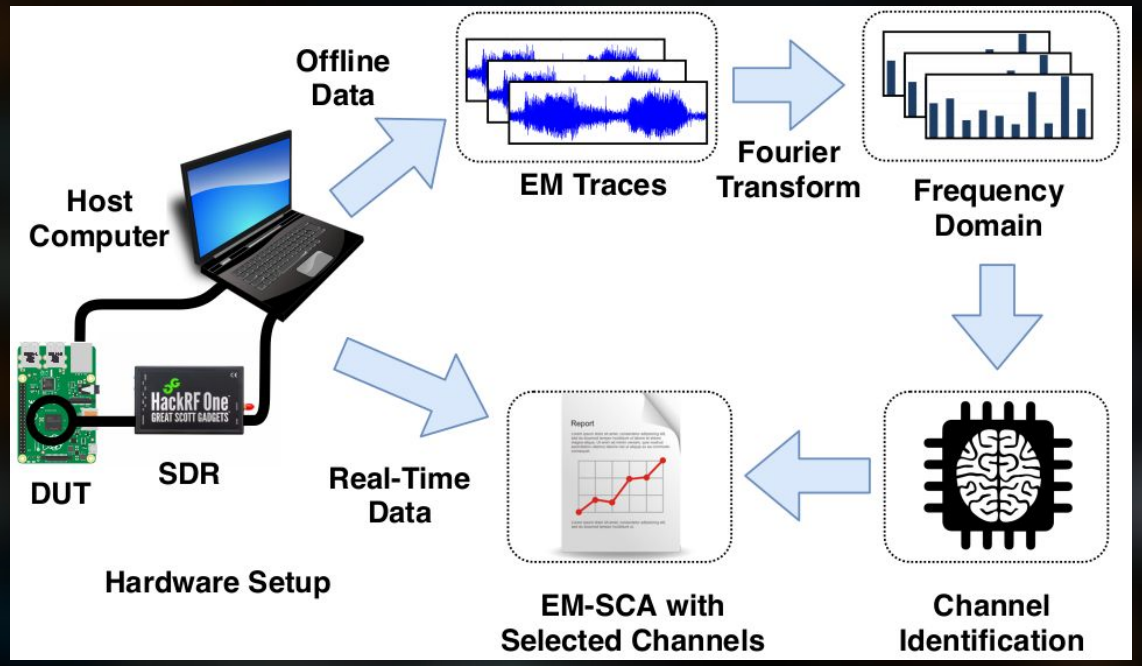
Contributions of this work:

- Experimental evaluation of multiple filtering methods to select a manageable number of frequency channels from a high dimensional EM data set.
- Introduction of a methodology using a Random Forest classifier to identify information-leaking frequency channels from high dimensional EM side-channel data.
- Demonstration of the effectiveness of the channel selection methodology by classifying software activities performed on a representative IoT device by observing its EM emissions.

Experimental Plan:

- Use Arduino Leonardo as the target device.
- Sampling rate = 20 MHz
- Fourier transform window = 1 millisecond
- Frequency domain signal has 20,000 frequency channels
- 10 different software activities as target classes
- Each activity is a program with *O(n)* time complexity
- Use various channel reduction methods before they are applied to a Random Forest (RF) classifier.

```
/* Arduino test program */
void setup(){
}
void loop(){
    for(int i=0, i<20, i++) { delay(10); }
    for(int i=0, i<20, i++) { delay(10); }
    /* further loops */
}
```

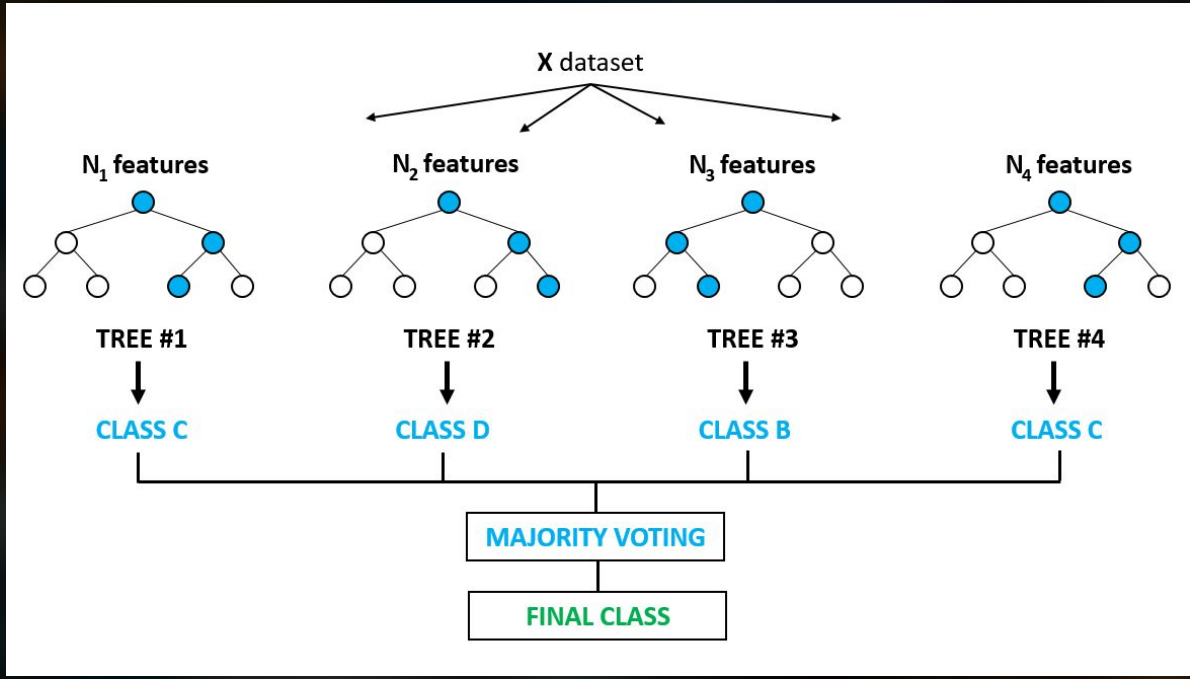


The objective is to reduce the number of frequency channels to 100 without compromising classification accuracy.

Random Forest Classification

- Random forest (RF) is a classification algorithm that uses a collection of decision trees.
- The two main parameters of are the number of trees (estimators) and the depth of those trees.
- The final classification prediction is the majority vote among all the created trees.
- In our experiments, we use 500 estimators and a maximum depth of 50 levels.
- A cross-validation of 5 partitions and 10 repetitions was used to decide the final classification accuracy.

Random Forest classification using decision trees



RF Reference: Breiman, L., 2001. Random forests. Machine learning, 45(1), pp.5-32.

Image source: https://www.globalsoftwaresupport.com/random-forest-classifier

Experiment 1: Using 20,000 channels

- All the channels are used as a baseline for other results
- 5000 RF trees.
- Average accuracy of 0.9315
- Time to predict 2,004 samples was 7.7435 seconds
- The accuracy for classes 0, 1, 2, 3, 4, 7, and 9 were 100% correct
- The accuracy of classes 5 and 8 are acceptable
- But the accuracy for class 6 is very low
- The algorithm confuses class 6 with classes 5 and 8 quite often (25% for each class).
- If class 6 was not considered the overall accuracy would be 0.9804.

Class	Accuracy
0	1
1	1
2	1
3	1
4	1
5	0.9043
6	0.5129
7	1
8	0.9199
9	1

Actual activities \ Predicted activities	0	1	2	3	4	5	6	7	8	9
0	2009	0	0	0	0	0	0	0	0	0
1	0	1985	0	0	0	0	0	0	0	0
2	0	0	1999	0	0	0	0	0	0	0
3	0	0	0	2024	0	0	0	0	0	0
4	0	0	0	0	2014	0	0	0	0	0
5	0	0	0	0	0	1821	0	0	195	0
6	0	0	0	0	0	854	1070	0	163	0
7	0	0	0	0	0	0	0	2011	0	0
8	0	0	0	0	0	160	0	0	1808	0
9	0	0	0	0	0	0	0	0	0	1927

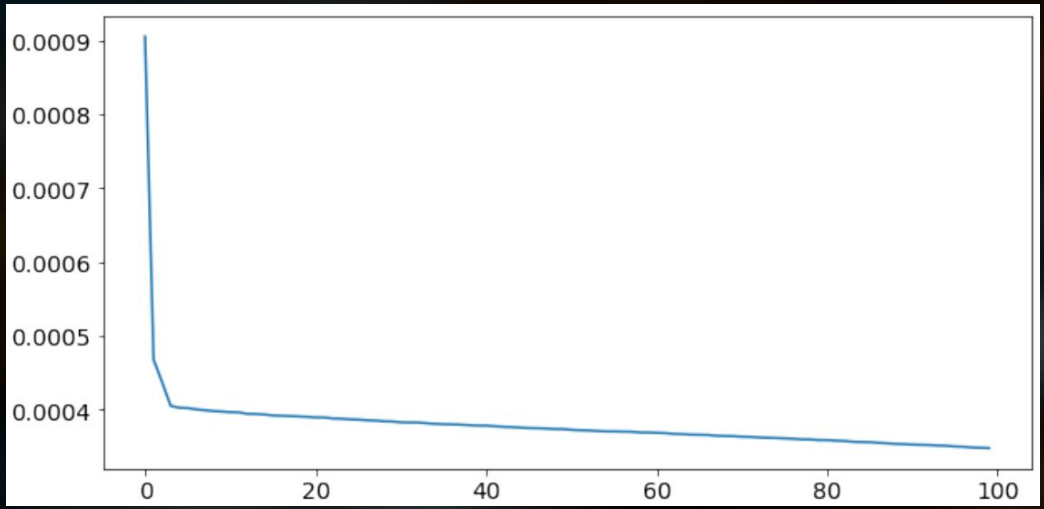
Experiment 2: Principal Component Analysis

- Principal component analysis (PCA) applies a linear combination of weighted variables to drastically reduce features.
- The new features are called eigenvectors (principal components).
- eigenvalues are ordered according to the amount of information they contain.
- Average accuracy was 0.1870, which is completely unfavourable.
- This is likely due to the high number of features, and most features being constant with low values.

Class	Accuracy
0	0.3438
1	0.2823
2	0.3882
3	0.1457
4	0.1373
5	0.0935
6	0.1768
7	0.2459
8	0.1375
9	0.1555

Conclusion: PCA is not suitable for feature selection on this type of EM side-channel data

Variance of top 100 eigenvalues from PCA



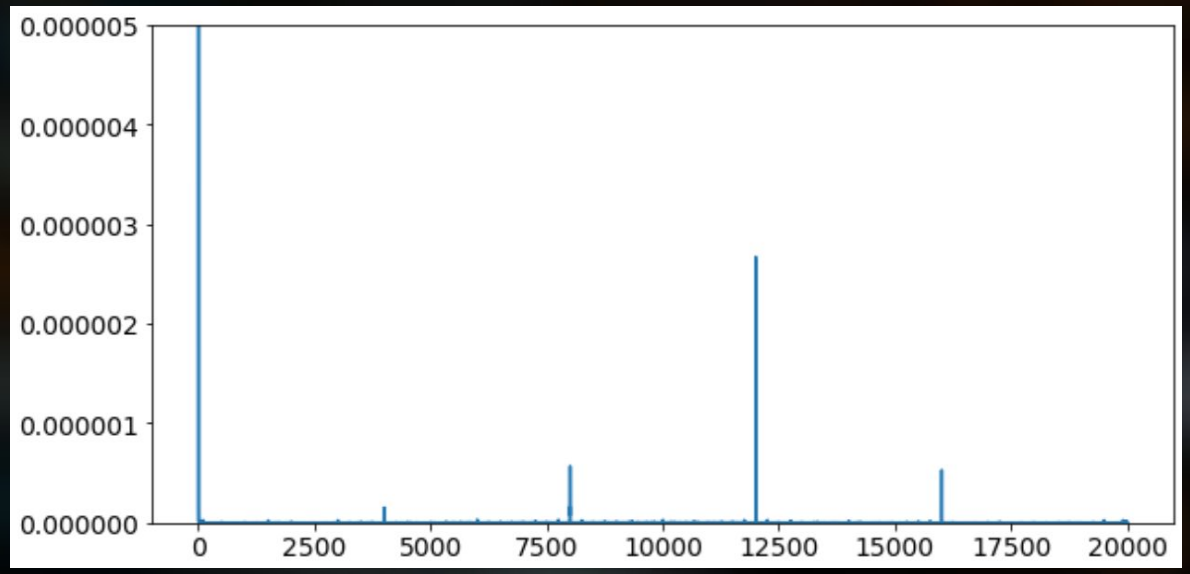
Experiment 3: Channel Selection Based on Variance

- The variance for each channel was calculated and subsequently, the highest 100 were selected.
- Outliers were removed before calculating variance.
- In order to select at least 100 channels with top variance, a threshold was set to 1.0632×10^{-8} that resulted 103 channels.
- Selected channels were used to test a RF classifier.
- Training and testing RF classifier took 1 minute and 37 seconds
- The average accuracy was 0.5431, which is still unsatisfactory.

Variance of the 20,000 channels.

Conclusion: Thresholding with variance is better than PCA but still not a sufficient classification accuracy.

Class	Accuracy
0	0.6473
1	0.5965
2	0.5391
3	0.5882
4	0.8027
5	0.2684
6	0.3845
7	0.4830
8	0.6272
9	0.5058

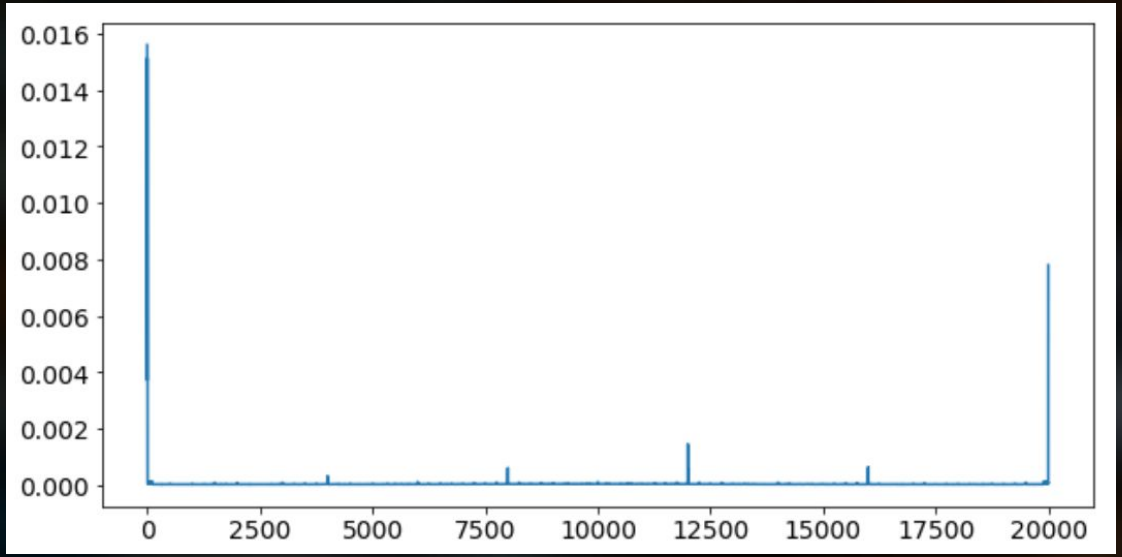


Experiment 4: Channels selection based on average

- The average for each channel was calculated and subsequently, the highest 100 were selected.
- Outliers were removed before calculating average.
- In order to select at least 100 channels with top average, a threshold was set to 6.9936×10^{-5}
- Selected channels were used to test a RF classifier.
- The average accuracy was 0.5423, which is still unsatisfactory.

Average of the 20,000 channels.

Class	Accuracy
0	0.6555
1	0.6067
2	0.5197
3	0.5770
4	0.8025
5	0.2694
6	0.3696
7	0.4939
8	0.6288
9	0.5196



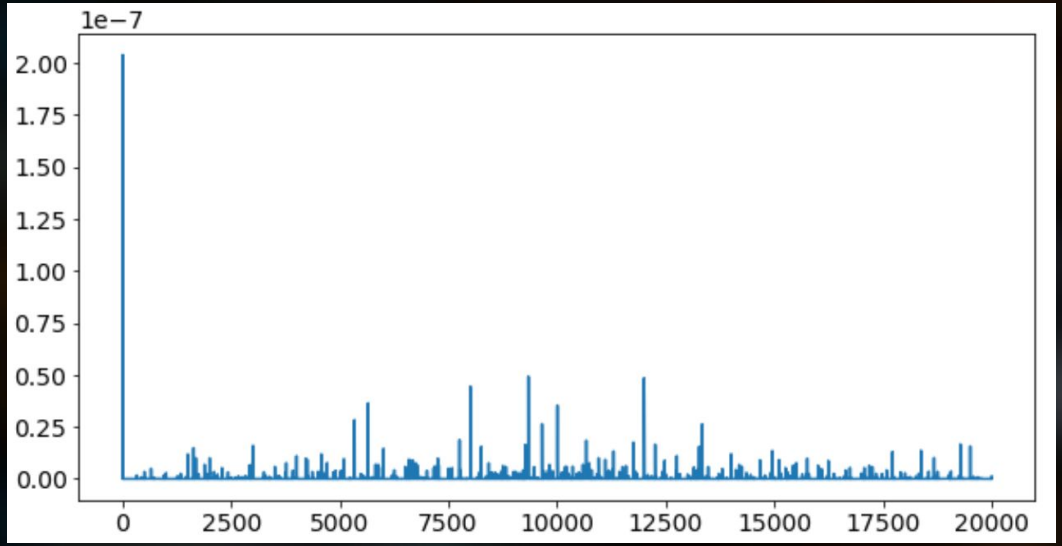
Conclusion: Thresholding with average almost same as variance and not a sufficient classification accuracy.

Experiment 5: Average per class and variance between the classes

- Firstly, calculate the average of the sample values of each class.
- This generates a [20,000 x 10] matrix containing average values for each channel for each class.
- Secondly, calculate the variance of these average values for each channel - this results in a [20,000 x 1] vector (see figure).
- Finally, a variance threshold of 3.3×10^{-5} was used to select the highest 100 channels that were used with RF classification.
- The average accuracy is 0.9047 for 100 channels. For 500 channels, this becomes 0.9395

Variance between the average of each class for the 20,000 channels.

Class	10 ch.	100 ch.	500 ch.
0	0.8220	0.9995	1
1	0.7364	1	1
2	0.8019	1	1
3	0.9090	1	1
4	0.5613	0.9990	1
5	0.3531	0.7749	0.9351
6	0.3592	0.5135	0.5179
7	0.4028	1	1
8	0.3828	0.7603	0.9422
9	0.4247	1	1
Avg	**0.5753**	**0.9047**	**0.9395**



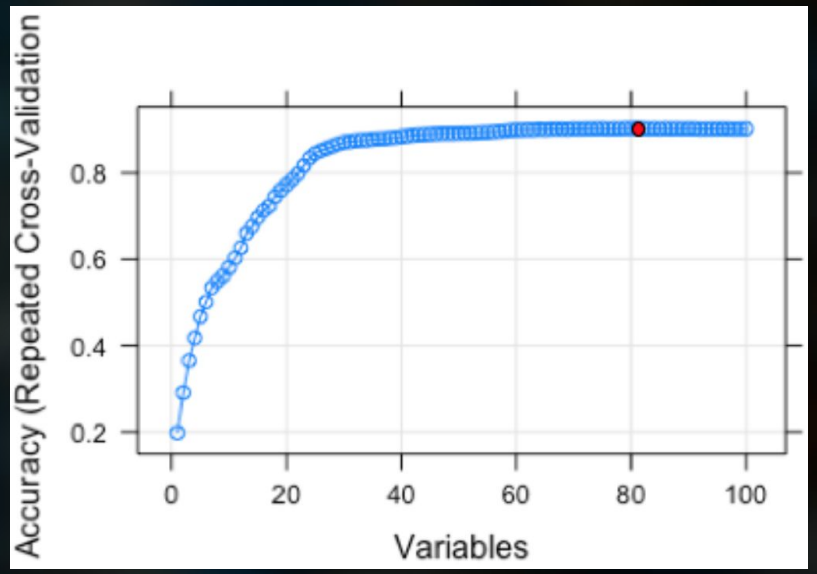
Conclusion: This approach generates better results than all the previous approaches.

Experiment 6: Applying Recursive Feature Selection

- Recursive feature elimination (RFE) is a method that creates supervised models starting with all possible attributes of data and rejecting weak attributes in each subsequent step. (*Guyon et al. 2002*)
- When attempting to select 100 features, the best result was achieved at 81 features using RFE.
- Average accuracy of the RF model created with those 81 channels is 0.9047.

Class	Accuracy
0	0.9986
1	1
2	1
3	1
4	0.9994
5	0.768
6	0.5129
7	1
8	0.7868
9	1

Conclusion: This approach generates similar results to experiment-5 for 100 channels.



Predicted activities

Actual activities	0	1	2	3	4	5	6	7	8	9
0	2006	0	0	0	3	0	0	0	0	0
1	0	1985	0	0	0	0	0	0	0	0
2	0	0	1999	0	0	0	0	0	0	0
3	0	0	0	2024	0	0	0	0	0	0
4	0	0	0	0	2013	0	1	0	0	0
5	0	0	0	1	0	1547	0	0	468	0
6	0	0	0	0	0	585	1070	0	432	0
7	0	0	0	0	0	0	0	2011	0	0
8	0	0	0	0	2	415	2	0	1549	0
9	0	0	0	0	0	0	0	0	0	1927

Experiment 7: Using a Time Window of 50 Timestamps

- We use a time window of 50 sample points.
- For each window, 18 statistical properties were calculated in time and frequency domains to consider as features.
- The average accuracy was 0.8000

- **Time domain:** mean, standard deviation, root mean square, maximal amplitude, minimal amplitude, median, number of zero-crossing, skewness, kurtosis, first-quartile, third-quartile, autocorrelation.
- **Frequency domain:** mean frequency, median frequency, entropy, energy, principal frequency, spectral centroid.

- The low accuracy was probably due to the fact that the model needs more samples to be trained. Applying a time window reduces the number of samples.

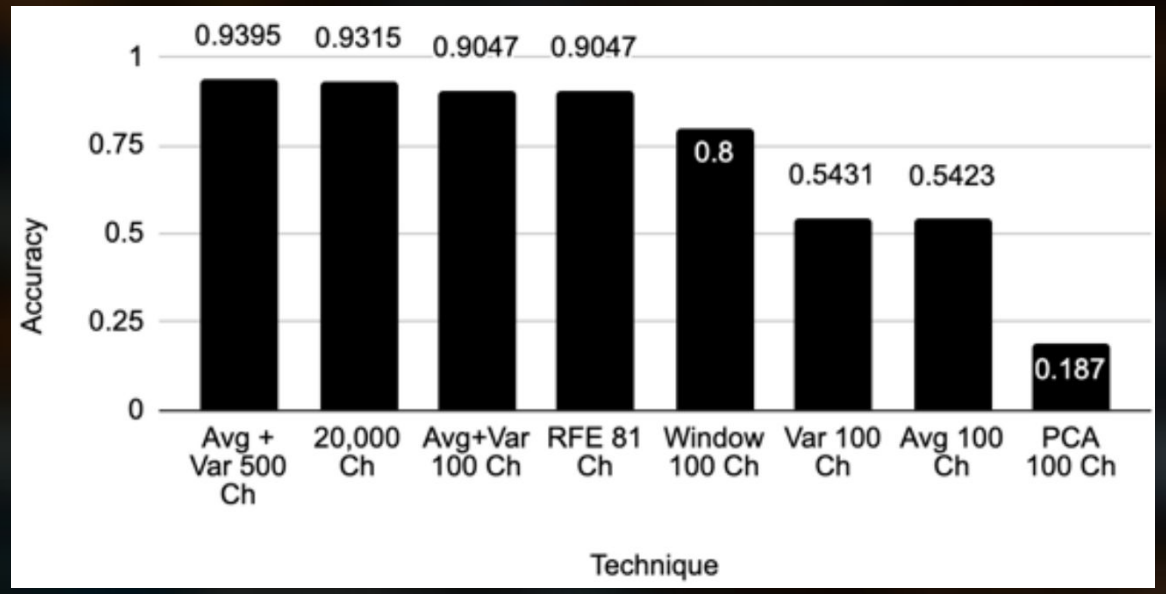
Conclusion: Compared to experiment-5 this accuracy is lower. Further research with more data is necessary.

Class	Accuracy
0	0.8857
1	0.9800
2	0.9500
3	0.9175
4	1.0000
5	0.5867
6	0.4255
7	0.8583
8	0.5467
9	0.9750

Summary & Conclusion

- We tested 7 methods to check which works best to reduce channels while maintaining a good classification accuracy.
- Among those 7 methods, [Avg+Var+500] was the best approach.
- Reduction of 20,000 channels to 500 is possible while achieving a much higher classification accuracy.
- Lower number of channels means faster processing with less processing power.
- And also, it permits us to store only those channels for future use saving storage space.

Message to take home: EM side-channel emissions can be incorporated into triage examination phase of digital investigations by using an appropriate channel selection methodology.



ASANKA.SAYAKKARA@UCDCONNECT.IE

WWW.FORENSICSANDSECURITY.COM

@FORSECRESEARCH

UCD Forensics and
Security Research Group