

Buried metal object recognition using deep neural networks

Laurynas Tamulevicius (2087510)

May 5, 2018

ABSTRACT

Nowadays metal detection has been employed in a vast range of applications such as landmine detection, archaeological excavation and airport security. However, the operation of a metal detector is non-trivial due to the complexity of the device and need of experience to interpret the feedback. In this paper, we propose a system to automate and, thus, ease metal detection task by exploiting latest advancements in deep learning. Our solution records metal detector response signal and extracts features that describe topology and composition characteristics of the target. We utilise convolutional neural networks to distinguish object shape and deep neural network classifier to determine the object type. We show that it is possible to infer symmetry of the buried metal item with high accuracy. Also, the results suggest that features describing the metal composition combined with spatial information produce unique target signatures, which can be successfully used to classify the objects by type.

1. INTRODUCTION

Metal detector (MD) is a device used to detect the presence of a metal object. There are two types of MDs, hand-held and stationary. The latter is usually used in the airports and banks to detect concealed weapons. In contrary, hand-held metal detectors (HHMD) are portable hence mostly used in archaeology and treasure hunting. Additionally, HHMD is a widely utilised approach to detect unexploded ordnance and landmines. It has been estimated that mines kill more than 28,000 people every year and over the last 50 years there were more victims affected by landmines than by nuclear and chemical weapons combined [27]. According to the United Nations, there are over 100 million mines scattered all over the world and pose a significant threat in at least 68 countries [12]. Therefore, a solution to increase the accuracy and efficiency of metal detecting is of great importance.

As depicted in Figure 1 a hand-held metal detector usually comprises a coil to transmit the signal and detect the response. A Digital Signal Processor (DSP) is used to generate a transmitting signal, process and analyse the response to provide comprehensible information to the operator. However, accurate interpretation of the feedback signal requires a significant amount of prior experience. While a misdetection in archaeology and treasure hunting does not affect the operator, in demining missions faulty identification of a landmine may pose a threat to human life. Therefore, accurate, portable and easy-to-use buried metal object identification system is crucial to prevent errors due to the lack of knowledge and experience. Additionally, such system would ease metal detection procedure, thus enabling clearance of larger hazardous areas.

Experience in metal detecting is the ability to determine the buried object type based on metrics provided by the detector.

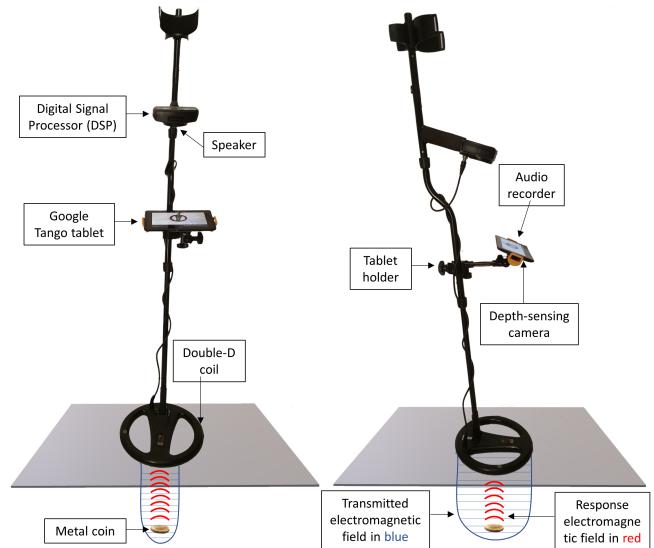


Figure 1: The basic setup used in the study.

This suggests that the target type identification procedure can be performed more accurately and efficiently by a machine learning model pre-trained on a broad set of known objects. Besides, such model would be better in inferring hidden correlations between target type and the response signal than a human operator. Hence, this paper will describe a multiple-neural-network system capable of identifying the type of the buried object and a mobile application that is used to collect target response signal.

Our main contribution is an accurate, portable, easy-to-use and low-cost buried metal object classification framework. To the best of our knowledge, this is the first system that tracks precise movement of the coil, records HHMD audio response and utilises deep learning techniques to infer collected auditory and visual data. Also, we show that spatial symmetry of the target can be determined by visualising and analysing response signal. We attained a transformation-invariant shape recogniser by introducing artificially augmented object representations in the training dataset. Our results reveal that extracted inductive, magnetic and conductive properties provide target-unique signatures. The latter features, along with the information about symmetry, could be used to distinguish targets, hence making the proposed system feasible.

The remainder of this paper is structured as follows: Section 2 discusses the basic setup and introduces metal detecting concepts. In Section 3 we overview relevant research that our solution builds upon. Section 4 describes every element of the pro-

posed system architecture in detail. In section 5 we introduce the experimental setup and discuss results obtained from simulated and real-world data. Section 6 concludes our work and proposes ideas for the future.

2. BASIC SETUP

This section will describe the basic setup and briefly discuss the operation of its components shown in Figure 1.

2.1 Metal detector

For this study, we exploited Minelab X-Terra 705 continuous-wave hand-held metal detector (see Figure 1). The MD is equipped with 7.5 kHz double-D coil [22], where the main compartment comprises two, transmitting and receiving, coils. The former generates an electromagnetic field (EMF) which is characterised and controlled by the DSP. The transmitted field induces eddy currents in the metal target which in turn produces a response EMF. The latter is obtained by the receiver coil and transmitted back to the DSP that analyses the differences between transmitted and received electromagnetic fields. Next, the digital signal processor emits a sound depending on the mode it operates in. The “response sound” directly represents the “response signal” hence we will use these terms interchangeably.

2.2 Modes of operation

The metal detector can operate in two modes, pinpoint and target identification.

The pinpoint mode produces a response sound whose intensity varies as bell-shaped function. The peak occurs when EMF generated by metal detector overlaps with the largest volume of the target (see component #1 in Figure 2). This appears due to the strongest response EMF which is triggered by the induced eddy currents. As coil and thus EMF moves away, the response signal becomes weaker due to decreased current induction. The metal detector generates a continuous sound proportional to response signal intensity.

The target identification mode produces a unique-frequency sound (i.e. target ID) for every object by extracting and processing various features of the perceived response signal. Therefore, the frequency of the generated sound may change for every sweep, especially if a coil is rotated or different cross-section of the object is scanned.

2.3 Target ID

The target ID is an integer value which varies in a range between 0 and 28 inclusively. It is determined based on a phase angle of the received signal with respect to the transmitted signal [15]. The phase angle is controlled by three metallic properties:

- **Conductance** - the ability for electric charge to flow in a certain path.
- **Inductance** - indicates how well a change in current through the conductor induces an electromotive force.
- **Magnetic response** - the magnetic field created by item without induced current flowing.

Our metal detector is capable of distinguishing the permeability of the object where target IDs ranging from 0 to 4 inclusively indicate ferrous and from 5 to 28 identify non-ferrous items. The greater the conductive and inductive properties the higher target ID is assigned to the buried object [15]. Additionally, the characteristics of the response EMF varies significantly with the following permeability features.

- **Non-ferrous** - these metals do not exhibit the magnetic properties; hence the response signal is determined purely by the inductive and conductive features. Targets of high symmetry (e.g. rings and coins) have enhanced inductance due to well defined current paths. Also, the response signal is stronger for more conductive (e.g. made of copper, gold) and thicker objects [15].

- **Ferrous** - these metals usually have low conductance, are inductive and, most importantly, they have a magnetic response due to the aligned magnetic moments of the atoms. The latter property overcomes the combined effect of inductance and conductance, therefore, ferrous materials can be easily distinguished from non-ferrous by the DSP.

2.4 Pinpoint signal and target ID extraction

Our system exploits pinpoint signal and target ID to determine the topology, conductive, inductive and magnetic properties of the target, hence the metal detector had to be augmented to track the coil movement and collect the response sound.

To associate the target response signals with the location of occurrence, we use Google Tango Project [23] tablet which records the sound and tracks a position of the coil. The latter is achieved with a depth-sensing camera and Tango API which enables smooth communication between the sensor and our mobile application. The recorded response signal is post-processed and sounds emitted by MD are identified.

3. BACKGROUND

Unfortunately, metal detector responds to any metal target, thus to discriminate undesired items, we must analyse the response signal. It appears that the task is non-trivial and target discrimination system with low false-alarm rate has been a long-standing goal in metal detecting, especially in land-mining, where every false signal decreases the deminer’s focus and raises the threat to life. Therefore, in this section, we will review several approaches to distinguish between the different type of objects. However, as most of the research of metal target classification is conducted in the demining field, the discussed trial objects will be either unexploded ordnance or landmines. In addition, we will overview the background literature on key components of our proposed system including convolutional neural networks which we use to infer the shape of the target. Also, deep fully-connected neural network classifiers employed to group targets by type and tools to combat overfitting.

3.1 Metal target discrimination

The study by [32] proposes a metal-detector array system which identifies areas of interest in the sensory input stream, extracts features and classifies the alarm signals which were triggered by the metal object underground. The feature vector consists of a range of morphological and wavelet-based properties of the target response signal (e.g. amplitude, width, kurtosis etc.). The objects are classified using supervised fuzzy Adaptive Resonance Theory Mapping (FAM) neural network and show 80% accuracy in varying-depth and target-type classification of 161 samples. One of the main issues was the misclassification of shallow-small targets as the large-deep objects due to characteristically similar response signals. While this technique achieves state-of-the-art accuracy, it only works on Minelab MD array and has not been tested on hand-held metal detectors, which contain only a single sensor and, hence, receive less descriptive signature of the target.

Furthermore, the paper [5] considers a continuous-wave hand-held metal detector. The authors propose a method to visualise

the eddy currents induced by the transmitting EMF so that the operator is aware of phase shift between transmitted and received signals. This provides not only the information about the permeability of the metal (i.e. ferrous or non-ferrous) but also contains details about the symmetry of the object. However, to extract this feature, multiple scans must be performed by rotating around the target. The researchers conclude that to fully exploit the proposed method a positioning mechanism must be installed to track the movement of a coil. Also, they suggest creating a knowledge database where target signatures could be stored and used along with pattern or image recognition algorithms to identify the object.

Moreover, [4] claims that it is possible to classify and identify objects based on their phase response. Similarly to the previous study, this work concludes that phase shift carries information about the topology of the target. It was shown that large metallic objects of at least 10cm in radius could be discriminated from metallic clutter. The thesis describes a technique to render near real-time 2-dimensional surface images of the shallow ferromagnetic targets (up to 10cm in depth). Unfortunately, the study points out the potential issues with composite objects, where response function might be complex and hence accurate object identification arduous. Also, the work states that smaller clutter can be distinguished from larger items, however, question whether small targets could be recognised in all-size clutter remains unanswered.

3.2 Convolutional Neural Networks

One of the critical components of our proposed system is a shape recognition mechanism which determines the target topology based on contour plots of response signal intensity and position. We utilise the convolutional neural network to accomplish that mainly because it can be easily enhanced to cope with a broader range and more complex target topologies by increasing number of parameters.

The convolutional neural networks introduced to computer vision by [20] were shown to be state-of-the-art in the image classification and detection task by [18]. The neural network architecture is especially applicable to highly locally correlated data. The latter pattern occurs in images, where nearby areas of pixels are associated.

A typical CNN model for computer vision task contains a set of convolutional layers that convolve adjacent input elements, and form features that are stored in feature maps [19]. The architecture differs from other neural networks in a sense that layers share weights and bias, thus reducing the total number of parameters. Every neuron in the layer is also followed by the activation function. Currently the most popular approach is ReLU non-linear activations [26], which were shown by [18] to speed up training several times compared to a CNN with saturating functions (e.g. tanh, sigmoid) [18].

The output of convolutional layer is usually fed to a pooling layer that further reduces the dimensionality of the input. It is the standard practice to attach fully connected layers at the end of the pipeline. They allow a model to learn non-linear combinations of high-level features and hence aggregate the ones that are common to a specific class.

3.3 Tools to prevent overfitting

Based on initial trials we expected to collect data of less than 100 targets, however, deep learning models tend to have a significant number of parameters and hence require a substantial amount of training data to be generalised. Here, we will overview

common methods to combat overfitting by means of regularisation. Also, we will discuss a technique to attain a transformation-invariant model.

3.3.1 Data Augmentation

Real-world objects are invariant (i.e. the identity of the item does not change with rotation, translation, scaling or illumination). This is critical when training a convolutional neural network as we want the model to detect and classify the same object regardless of the transformation it has undergone. Therefore, the simplest way to enhance the accuracy and reduce overfitting of the neural net is to artificially expand the original set of images by performing various label-preserving transformations (e.g. vertical/horizontal reflections, cropping, colour enhancement) [7, 6, 18, 2].

3.3.2 Dropout

Dropout first proposed by [13] is a regularisation mechanism used to alleviate overfitting. The latter problem is especially prevalent when training dataset is small, hence without any regularisation, network adjusts its parameters to perform well on training data only, thus failing to infer testing input accurately. The dropout mechanism forces the network to be redundant, learn same features as different combinations of neurons and therefore become more general. This is achieved by disabling a predefined fraction of units during the training stage. However, the approach causes additional complexity and, for instance, [18] claims that even though the dropout introduction reduced overfitting the training cost doubled.

3.4 Deep Neural Network Classifier

The last component of our system is a classifier which, when provided with information about target symmetry and composition, determines the object type. To tackle this task, we employ multilayer perceptron architecture, and the following subsection will overview relevant background literature.

A multilayer perceptron is a feed-forward fully-connected deep neural network (DNN) that comprises of at least a single hidden layer and continuous non-linearities. The feed-forward neural networks were proved [8] to be universal approximators for a numerous hidden-unit activation functions except for polynomials [21, 2]. Moreover, in many problem domains, DNN and Support Vector Machines (SVM) are the most-suitable solutions achieving comparable generalisation performance. However, in most cases, the DNN model classifies data more rapidly than SVM [2]. On the other hand, DNN usually requires much more computational resources to be trained due to a large number of parameters.

As aforementioned, fully-connected feed-forward layers are also utilised in other deep learning architectures to increase the separability and thus ease the classification of the final output space [2]. For instance, convolutional neural networks [30, 18, 16] or CNN-LSTM-DNN based speech recognition models [28] use fully-connected layers to learn non-linear combinations of the extracted features.

4 DESIGN

In this section we will discuss the elements of the proposed system in detail, hence to aid the explanation we will refer to Figure 2 that contains labelled pipeline components.

4.1 Mobile application for data collection

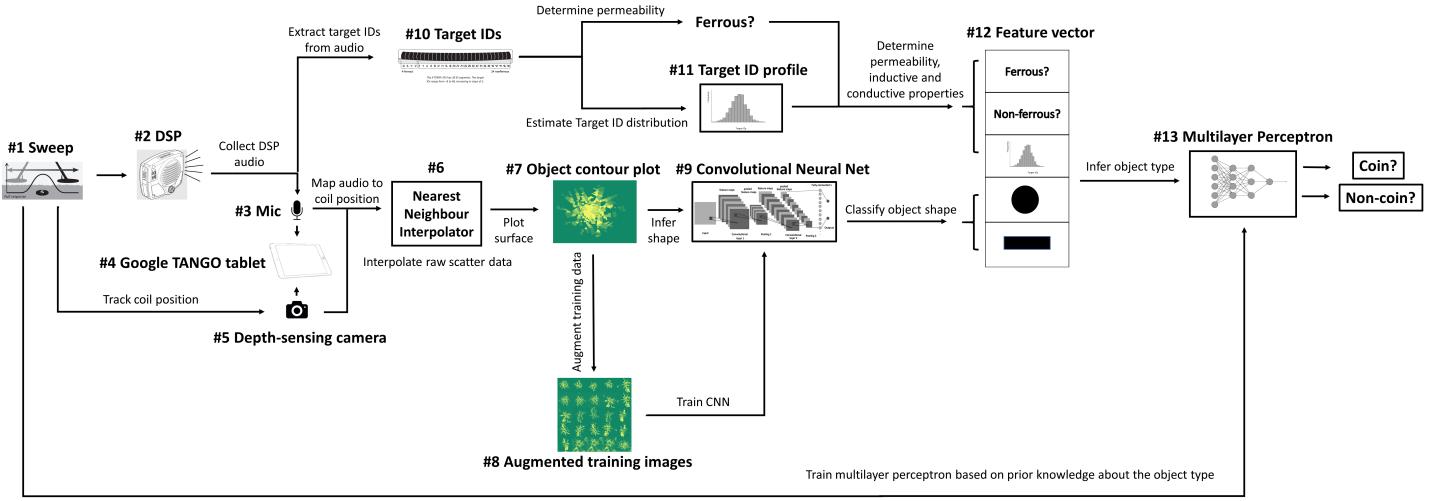


Figure 2: A proposed system pipeline discussed in Section 4 used to infer the target type. Starts at the left, component #1.

To track the metal detector coil movement, we use Google TANGO device (see #4). It provides built-in Tango augmented reality computing platform [23] which allows smooth interaction with the depth sensor (#5) through the API. The platform is queried for a depth-point cloud at the frequency of 100Hz. However, only a small fraction of this data is used as we are only interested in the depth-points residing on the coil. The latter region is displayed to the user with a green marker on the screen so that the camera could be aligned before taking a measurement.

In parallel to motion tracking, we also record audio (#3) to capture sounds generated by the DSP speaker (#2). Once the depth point is acquired the timestamp of the audio recording is captured. Later, audio recorded in pinpoint mode is used to map the recorded sound amplitude (signal intensity) to a spatial position of occurrence.

The application screen (#4) also displays the start and stop buttons that are used to launch or terminate depth-sensing and audio recording.

The depth-sensing camera was found to be sensitive to brightness and fail to operate in extreme outdoor lighting conditions due to ambient light which contributes to the noise [10]. Hence, to inform the operator, we provide a live count of detected points on the screen.

4.2 Object shape detection

In this section, we only consider the audio response signal acquired in pinpoint mode. To start with, once the detection process is over, we are ready to map the audio signal to the location of occurrence, which provides us with a set of 3-dimensional data points (i.e. X-Y coordinates and sound amplitude at that position). When plotted this data provides us with a noisy spatial representation of the object (#7). This was observed and clarified during initial trials.

4.2.1 Sound extraction and signal purification

We perform Fourier analysis of the audio to find the harmonic series of the signal. We calculate the autocorrelation of the Fourier components to quantify the periodicity and only consider audio segments with the sinusoidal signal generated by the DSP [24]. The latter procedure is based on the fact that noise and silence do not exhibit harmonics and hence can be identified and dis-

missed.

4.2.2 Response signal plotting

Next, we extract the amplitude of the audio at the recorded timestamp locations. It serves the Z axis in initial scatter diagram of the object response signal. We also gather X and Y components from the recorded depth-points. However, the Z coordinate in depth-point which indicates the distance from the camera to the surface is disregarded as our experimental procedure sets this distance to constant. Nevertheless, the depth could be used in future work to estimate a more precise position of the coil.

Furthermore, to reduce the parameter number in the following component (i.e. convolutional neural network), we decided to project the 3D scatter diagram to 2D contour plot. To obtain smoother surface, we perform Nearest Neighbor interpolation (see #6) of the scattered points. However, this step is optional and could be replaced by any other pre-processing algorithm. The value of the amplitude determines the colour gradient of a 2D point at the X-Y position. The final plot embeds the essential topological features of the object (i.e. size, length, width). Sample plots could be seen in Figure 6.

4.3 Object shape determination using CNN

4.3.1 Architecture

With the breakthrough in deep learning architectures used in image analysis [29] we decided to employ convolutional neural network to classify the contour plots by the shape (component #9). Also, this deep learning architecture can be easily scaled in the future to identify more complex shapes by simply increasing the number of parameters.

The CNN architecture we use bears resemblance to “AlexNet” [18]. However, the depth and parameter number are reduced as the complexity of a problem is expected to be lower than classifying the “ImageNet” [9].

The input of the CNN is a $96 \times 96 \times 3$ -dimensional image of the target contour plot.

There are four convolutional layers, where 32, 64, 128 and 256 feature maps are collected with 5×5 filter size. Every neuron utilises ReLU activation function.

Each convolutional layer is followed by a pooling layer where 2×2 max pooling with a stride of 2 is applied. This implies that none of the extracted regions overlaps and the input dimensionality is reduced twice.

The final stage comprises two fully-connected layers. The first one is responsible for aggregation of the extracted features. It has 1,024 neurons and exploits ReLU activation to introduce additional non-linearity to the model and, thus, increase mapping space of the classifier. Also, to avoid overfitting and generalise our model the first dense layer is regularised using dropout technique. The units are only dropped during the training of the neural network.

The final fully-connected layer applies logit function to the output of the predecessor to scale and prepare it for the softmax function, which squashes the input to a range of $[0, 1]$ and also normalises the output so that the sum is 1.

The output of the CNN is a distribution over 2 class labels (i.e. circular and rectangular shapes) produced by the 2-way softmax. The values indicate how confident the model is with the classification.

In total, the proposed convolutional neural network has 10,518,338 parameters if we assume that sample input contains $96 \times 96 \times 3$ -dimensional pixel array. Although this may seem an excessive amount of parameters, we found this a necessity to generalise the model on a substantial number of simulated images.

4.3.2 Data augmentation

The rendered contour plots may exhibit targets of varying scale, rotation and position. Hence to generalise the model and make it transformation-invariant we must either collect a significantly larger amount of transformed object representations or artificially expand our training dataset by augmenting original images. Our initial investigation showed that the latter approach would be suitable for our model. Hence, we used open-source library “Augmentor” [3] to augment the training images (see #8). The software provides a pipeline-based mechanism, where the user must define a set of desired transformations. These operations are applied stochastically according to predefined probabilities. We utilise the following transformations:

1. Image is rotated by 90° to the right 80% of times.
2. Image is rotated by 270° to the right 50% of times.
3. Image is flipped horizontally 80% of times.
4. Image is flipped vertically 80% of times.
5. Image is flipped vertically 80% of times.
6. 90% of the image is cropped 50% of times.
7. Image is resized back to 96×96 pixels as cropping does not preserve the dimensions.

4.4 Target ID profile extraction

At this stage (see #10) we only consider the audio signal recorded in the target identification mode to extract information about the target composition. However, as most of the objects are not pure, we expect the target ID to deviate. Therefore every object will have a distribution of recorded target IDs. We refer to it as the “target ID profile” (#11).

To obtain the profile, we start by computing Fast Fourier Transform (FFT) of 20-50ms audio segments. The interval was selected with respect to the mean duration of the DSP-generated sound

which is 200ms, hence by considering segments of 20-50ms we are confident to identify the majority of the target IDs.

Next, as aforementioned, we compute the autocorrelation of the segment to measure periodicity of the audio signal. The segments of low periodicity are identified as noise and disregarded. Moreover, the further consideration of the power of the harmonics allows us to determine the fundamental frequency of the signal and, thus, find the corresponding target ID.

The variance of the target ID distribution implicitly carries information about the composition uniformity of the object. For instance, electronic parts of the landmine might be made of copper (non-ferrous) but the cover of steel (ferrous). In this case, we would expect the variance to be broad as target IDs for copper reside on the right side (in a range [22..26]) of the scale and those for steel on the left side (in a range [0..4]). On the other hand, the items of high symmetry and homogeneous structure (i.e. a bronze 1p coin) would result in a distribution of low variance.

In addition, if the variance is low, the mean of the distribution indicates the most likely type of metal the object is made of. However, if the variance is high, the mean does not necessarily represent the metal type of the item. For instance, previously discussed cooper-steel landmine would result in a mean target ID in-between range [0..26] as it would contain both, copper and steel.

4.5 Deep neural network object type classifier

The last element in our architecture is the deep neural network classifier (#13). It accepts the extracted feature vector (#12) with information about object shape, magnetic, conductive, inductive properties and deduces the type of the target.

The decision to utilise a deep neural network was made because it could be easily downgraded to linear regression if the training dataset is not sufficient to optimise and generalise complex multi-hidden-layer DNN.

Our architecture exploits two fully-connected layers with 256 and 32 hidden units. Similarly to the CNN, we apply dropout regularisation to enhance the classifier’s accuracy. Every layer is followed by the ReLU activation. Finally, the cost of the model is optimised using Adaptive Moment Estimation (ADAM) algorithm [17].

5. EVALUATION

In previous sections, we introduced the system components that aim to detect, record and extract information about the topology, inductive, conductive and magnetic properties of the target. This section will discuss performance and results of the system on the simulated and real-world data. Besides, we will provide a methodology for the conducted experiments.

We claim that the proposed solution is capable of distinguishing any metal object given that:

1. Its symmetry can be deduced from the output data when metal detector is operating in pinpoint mode.
2. The item is shallow enough, and its size is sufficient for the detector to determine the target ID.

Furthermore, for our study, we try to distinguish coins from metal debris. The main reasons for choosing coins as testing objects were:

- High symmetry - we expect coins to exhibit distinct highly symmetric shape which could be easily distinguished from non-symmetric metal rubbish.

	Circular	Rectangular
Ferrous	Coins: 10 Non-coins: 7	Coins: 0 Non-coins: 12
Non-ferrous	Coins: 19 Non-coins: 13	Coins: 0 Non-coins: 15

Table 1: The number of permeability-shape-varying objects by type (i.e. coin or non-coin). Note, the figure does not represent the actual target distribution in real world.

- Known metal composition - conductive, inductive and magnetic properties can be tracked and analysed.

To simplify the model we selected trial items of two different shapes, rectangular and circular, and permeabilities, ferrous and non-ferrous. The proportion of coin and non-coin objects in each class can be seen in Table 1.

5.1 Target ID profile extraction

5.1.1 Experimental setup

To minimise the detector interference with external sources emitting electromagnetic radiation, we identified a clean-from-metal area far from transmission lines and established a soil bed with dimensions of $0.9 \times 0.9 \times 0.4$ meters. The volume was filled with 300 litres of all-purpose compost and verified to be clean from metallic clutter.

Moreover, we buried a test object under a 10cm layer of soil and swept the detector coil 2cm above the surface in a star pattern as depicted in Figure 3. The depth-sensing camera was set to point to the normal of the soil surface at all times as shown in Figure 1.

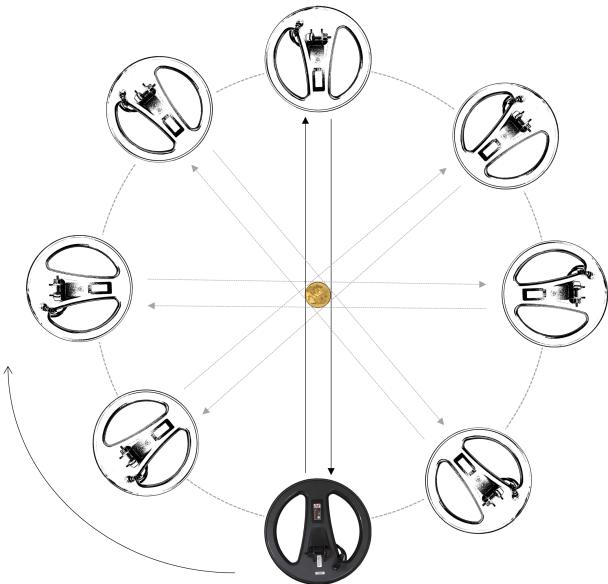


Figure 3: Object scanning procedure. The star pattern was followed by rotating clockwise around the object and moving metal detector coil back and forth.

Object	Composition	Volume (mm^3)
1p (≤ 1992)	Copper: 97% Zinc: 2.5% Tin: 0.5%	492.0
1p (> 1992)	Copper-plated steel	534.0
2p (> 1992)	Copper-plated steel	1,069.5
5p (≤ 2012)	Copper: 75% Nickel: 25%	432.6
5p (> 2012)	Nickel-plated steel	432.6
10p (≤ 2012)	Copper: 75% Nickel: 25%	872.2
10p (> 2012)	Nickel-plated steel	872.2
20p	Copper: 84% Nickel: 16%	611.5
50p	Copper: 75% Nickel: 25%	1,041.9
1£	Copper: 70% Zinc: 24.5% Nickel: 5.5%	1,252.5
rusty bottle cap	Aluminum-Steel	597.3
kitchen knife	Aluminum-Steel	11,040.0
L-shape hex key	Steel	10,995.6

Table 2: Specifications of the items used in the target ID profile analysis. Number in the bracket indicates the year of manufacture. Coin characteristics taken from [25].

To extract and compare target ID profiles we selected a set of coins and several non-coin objects of known composition [25]. The specifications of the trial items are shown in Table 2

5.1.2 Results

The target ID profiles in Figure 4 were plotted for test objects in Table 2. To compare characteristics of the profiles, we selected only a subset of all 76 test targets. The composition of the chosen objects was known therefore the correlation between object topology, conductive, inductive and magnetic properties could be studied in detail. The results suggest a number of insights and support our previous claims:

- Variance of the target ID distribution in non-ferrous coins is much smaller than the variance in non-coins, which are made of ferrous and non-ferrous metal mixtures.
- The greater the conductive and inductive properties the higher target ID is assigned to the buried object [15]. This implies that items of higher symmetry and made of better conductor should have greater target ID. The latter claim is visible in 1 penny coin (minted before 1992) made of bronze (97% copper) with target ID peaking around 24. Whereas, the 5 pence coin (created before 2012) has a mean

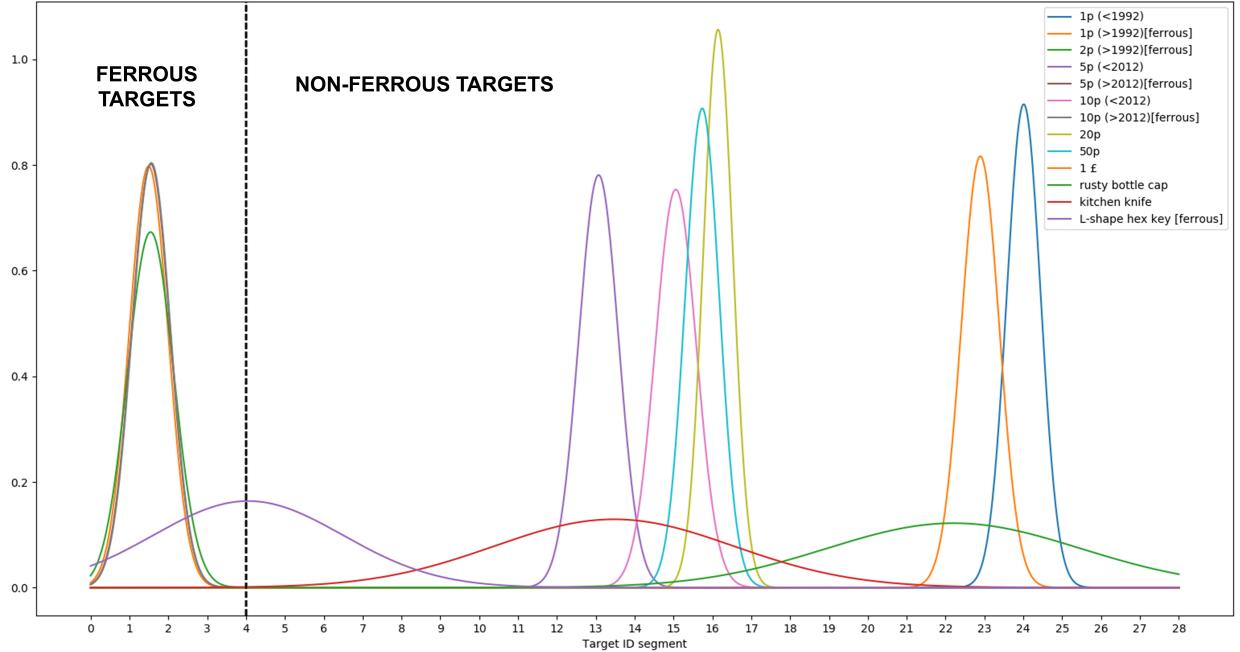


Figure 4: Comparison of target ID distributions of ferrous and non-ferrous targets. The original frequency diagrams were fitted with normal distribution.

at approximately 13 as it contains 75% copper and 25% nickel, which is four times less conductive material than copper.

- The correlation between the volume of the material and its target ID. This is visible in target IDs of 5p, 10p and 50p minted before 2012 and made of same material copper-nickel (75%-25% respectively). Here, the higher the volume of the coin the greater target ID gets assigned.
- Diverse-composition objects have a more significant variance of the target ID. While this trend is not visible in our trials with small items (i.e. coins), the large variance can be seen in bigger targets, primarily, the ones made of a mixture of a ferrous and non-ferrous element (e.g. rusty bottle cap, kitchen knife; see Figure 4).
- The inductance and conductance are dominated by a magnetic response in ferrous materials. This can be noticed in different-size ferrous coins (i.e. 1p, 2p, 5p or 10p which are made of steel). We see that all of them have similar variance and approximately same mean. However, larger items such as steel L-shape hex key was found to be peaking around higher ferrous values. The latter could be due to around ten times larger volume than that of a coin.

5.2 Object shape inference

In this subsection, we will overview the results of the CNN shape classifier trained on NVIDIA Tesla K80 GPU provided by Google Colab project [11]. We experimented with simulated and real-world data; thus we will divide this section accordingly.

5.2.1 Experiments on simulated data

During initial trials, we observed that target response signal in pinpoint mode was varying with frequency and amplitude. Further investigation showed that frequency of the generated sound changes only if the target is close to the coil hence we decided to keep frequency constant by burying object 10cm underground and only observe the change in signal amplitude.

Furthermore, we noticed that the volume of the sound was increasing as the coil was approaching the object and decreasing as metal detector moved away forming bell-shaped response function (see #1 in Figure 2). For symmetric items, this shape was preserved regardless of the scanning position as we rotated around the target. Whereas, rectangular objects produced an elongated signal response as coil was moving along the body and generated smaller width response signal as coil trajectory was perpendicular to the metal item.

To test our hypothesis that the classification of the response signal by rectangular and round objects is feasible, we sampled a set of images from the multivariate Gaussian distribution with distinct mean and covariance parameters to simulate response signal images.

However, to preserve the labels, control the shape, generate images of different scale and rotation we had to regulate ellipticity of the normal distribution. The latter was achieved by building customised covariance matrix as follows.

To begin with, the ellipticity of the distribution (i.e. $\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$) is defined by the major and minor axes. Thus having a covariance matrix ($\boldsymbol{\Sigma}$) the magnitude (M) of axes are [1]:

$$M_{min} = \sqrt{\lambda_{min}}, \quad M_{major} = \sqrt{\lambda_{max}} \quad \text{where}$$

$$\lambda_{min} = \min\{|\lambda| : \lambda \text{ is an eigenvalue of } \Sigma\}$$

$$\lambda_{max} = \max\{|\lambda| : \lambda \text{ is an eigenvalue of } \Sigma\}$$

Now, we can randomly choose eigenvalues that define the shape of the distribution and obey the predefined axes ratio. For example if we choose ratio $\frac{\lambda_{max}}{\lambda_{min}} = 1$ we would expect to get a circle-shaped Gaussian distribution. Hence, to build the desired Σ we impose several constraints:

- Σ must be not diagonal as we want the distribution to rotate and stretch in space. This implies that non-diagonal entries must not be zero.
- Σ must preserve same eigenvalues.
- Σ must be positive semi-definite to be valid covariance matrix.

We start by forming a diagonal eigenvalue matrix where eigenvalues are randomly chosen from uniform distribution and obey predefined ratio:

$$\Sigma_1 = \begin{pmatrix} \lambda_{min} & 0 \\ 0 & \lambda_{max} \end{pmatrix} \quad (1)$$

then we want to perform similarity transformation to get non-diagonal covariance matrix Σ_2 as follows:

$$P^{-1} \Sigma_1 P = \Sigma_2 \quad (2)$$

where $\Sigma_2 \sim \Sigma_1$ and P is invertible 2×2 matrix. As we can pick any matrix P given that it satisfies (2) we choose P to be orthonormal and hence (2) becomes

$$P^T \Sigma_1 P = \Sigma_2 \quad (3)$$

Next, we generate a random uniform real 2×2 matrix and use QR decomposition [14] to obtain orthonormal matrix P (we disregard upper triangular matrix R) which is used in transformation (3) to get non-diagonal covariance matrix (Σ_2) with same eigenvalues.

The ratio $\frac{\lambda_{max}}{\lambda_{min}}$ was chosen randomly in the following ranges to define different-shape objects:

- **Circular targets:** $\frac{\lambda_{max}}{\lambda_{min}} = [0.60, 1.40]$
- **Rectangular targets:** $\frac{\lambda_{max}}{\lambda_{min}} = [0.05, 0.40] \cup [1.60, 1.95]$

$$\begin{array}{c} \text{Predicted} \\ \begin{array}{cc} C & R \end{array} \\ \hline \text{Actual} \end{array} \begin{array}{c} C \\ R \end{array} \left[\begin{array}{cc} 0.46 & 0.04 \\ 0.22 & 0.28 \end{array} \right]$$

Figure 5: Confusion matrix of simulated dataset with 60 original images and 100 augmentations per original.

Originals	Augmentations per original	Accuracy (%)	F1 score
100	500	89.5	0.90
70	500	81.3	0.83
60	500	75.0	0.71
50	500	80.0	0.82
40	500	76.1	0.80
100	100	81.8	0.77
70	100	80.1	0.79
60	100	74.1	0.68
50	100	77.5	0.82
40	100	75.1	0.76

Table 3: Object shape classifier accuracy and F1 score with varying size simulated training dataset. The evaluation was performed using 5-fold cross-validation. Note, the evaluation image set was not augmented. Results were collected from successfully converged models only.

Using multivariate Gaussian distribution with these parameters we sampled 3D data points to represent simulated circular and rectangular objects. We used 3D points to generate contour plots as shown in Figure 6 that were fed to the CNN to test the classifier and estimate its performance with a low number of data points. Also, we augmented the input images to prevent the CNN from overfitting. For evaluation, we exploited 5-fold cross-validation, where the augmentations extended training dataset. Table 3 shows the results.

One can see the expected trend that with larger original dataset both, the accuracy and F1 score, increase, except for the trial with the 60-image set. To investigate the issue, we provide a confusion matrix in Figure 5, where we can see that model mostly misclassified rectangular objects as circular. This suggests that the rectangular shapes looked very similar to circular and thus we must adjust the boundaries of Gaussian distribution parameters so that the shapes seem more distinct.

Besides, the mean of the F1 score which is a harmonic average of the precision and recall increased with more Augmentations per Original (A/O). As $F1_{av} = 0.764$ for 100 A/O and $F1_{av} = 0.812$ when 500 A/O were used.

5.2.2 Experiments on real-world data

We collected 56 data samples of different objects, extracted the amplitude of recorded audio, mapped it to the position of occurrence and plotted contours as shown in Figure 6 (the four images at the bottom). As one may notice, the plot has many points with Z-axis value of zero while nearby points exhibit a high-valued gradient. We treat this as noise that may occur due to the MD tar-

Originals	Augmentations per original	Accuracy (%) \pm SD	F1 score \pm SD
56	100	63.4 ± 2.39	0.72 ± 0.02
56	1000	74.5 ± 2.63	0.81 ± 0.02

Table 4: Object shape classifier accuracy and F1 score with real-world training dataset and varying number of augmentations. The evaluation was performed using 5-fold cross-validation. Note, evaluation image set was not augmented. The standard deviation (SD) estimated from 4 attempts.

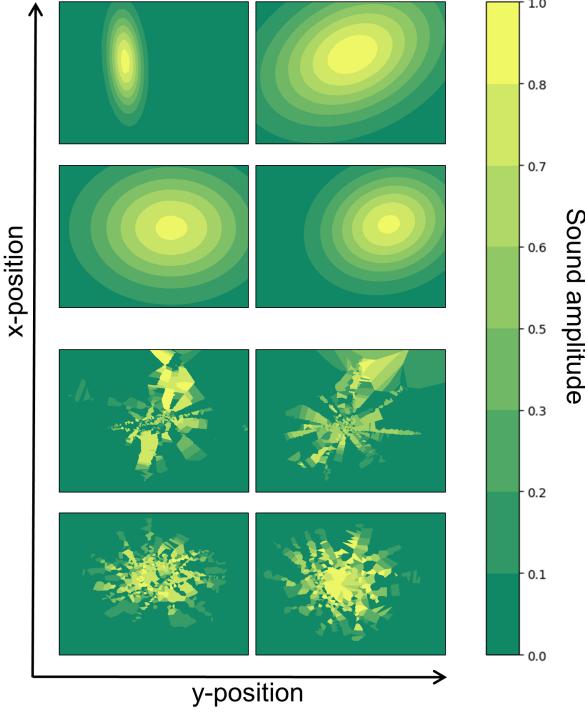


Figure 6: Sample contour plots for target response signal. Z-axis (colour gradient) indicates sound amplitude and X-Y axes define the position of occurrence. The top four images represent simulated data. The first 2 of them depict rectangular objects, while two below show circular targets. The four images at the bottom display real-world response pinpoint signals. Similarly, first 2 demonstrate the shape of rectangular targets and the bottom ones represent circular coins.

get stabilisation system which attempts to estimate the location of the target and generate the most intense signal around it. The estimation mechanism may introduce a relative error in position when measured at different locations. However, this information is proprietary and thus not available to us.

Nevertheless, the plots suggest us that symmetry can be inferred. We trained the same CNN that we used for simulated data to classify noisy real-world images of the response signal. In Table 4, one can see the results.

As suspected, by increasing the number of transformed images per point ten-fold we achieved 10% more accurate model. Similarly to simulated data, the mean F1 score increased with the higher number of augmentations.

Furthermore, confusion matrices in Figures 7 and 8 indicate that, similarly to simulated data, model misclassifies rectangular (R) shapes as circular (C) more often than C as R . This could be due to the uneven proportion of circular and rectangular objects (0.61 : 0.39 respectively) in the training dataset, which implies that model may be biased. Also, the latter issue remained with increasing number of augmented images as ratio (4) stayed approximately the same.

$$\left[\frac{C \text{ as } R}{R \text{ as } C} \right]_{100 \text{ aug}} \approx \left[\frac{C \text{ as } R}{R \text{ as } C} \right]_{1000 \text{ aug}} \approx 0.6 \quad (4)$$

	Predicted	
	C	R
Actual	C	$0.46 \pm 0.01 \quad 0.15 \pm 0.02$
	R	$0.22 \pm 0.02 \quad 0.17 \pm 0.02$

Figure 7: Confusion matrix of real-world dataset with 56 original images and 100 augmentations per original. C stands for circular and R - rectangular shape. The standard deviation estimated from 4 attempts.

	Predicted	
	C	R
Actual	C	$0.51 \pm 0.03 \quad 0.10 \pm 0.02$
	R	$0.16 \pm 0.02 \quad 0.23 \pm 0.02$

Figure 8: Confusion matrix of real-world dataset with 56 original images and 1000 augmentations per original. C stands for circular and R - rectangular shape. The standard deviation estimated from 4 attempts.

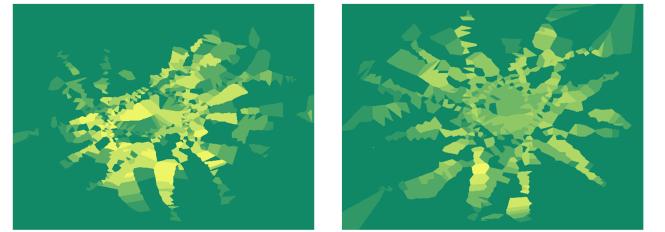


Figure 9: The typical misclassification example of real-world dataset with 56 original images and 100 augmentations per original. The left image shows rectangular object being identified as circular and the right image displays opposite misclassification.

To show a typical misclassification example, we provide Figure 9. The left diagram shows that characteristics of rectangular objects are not expressed well enough, therefore, the shape appears circular. In addition, right image shows that some circular-target graphs are noise-corrupted hence form a structure of a rectangular object. To overcome this issue, the experimental procedure must be revisited and background noise reduced.

5.3 Object type determination

In this section, we will overview the results of the DNN (multilayer perceptron) classifier used to determine the target type. Similarly to the previous component, the model was trained on NVIDIA Tesla K80 GPU provided by Google Colab project [11]. We tested the classifier's performance on simulated and real-world data; thus we will divide this section accordingly.

5.3.1 Experiments on simulated data

The final element of the system (multilayer perceptron) is accepting a feature vector made of the following information:

	Circular	Rectangular
Ferrous	$P(C) = 0.34$	$P(C) = 0.0$
	$P(\neg C) = 0.14$	$P(\neg C) = 0.23$
Non-ferrous	$P(C) = 0.66$	$P(C) = 0.0$
	$P(\neg C) = 0.35$	$P(\neg C) = 0.28$

Table 5: Probability distribution of finding different composition and shape objects in our real-world dataset. C indicates coins and $\neg C$ non-coins.

- Target ID profile.** It is a 29-dimensional array where every index of the entry represents target ID in a range $[0..28]$ and entry value shows the number of occurrences of a particular target ID.
- Magnetic properties.** It is a 1-dimensional array which indicates a ferrous metal with a value of 1 and non-ferrous with value 0.
- Shape of the object.** The feature is represented by the 2-dimensional array, where every entry represents a probability (i.e. softmax output) of the object belonging to circular or rectangular shape class.

Before starting data collection procedure, we selected 76 items which would span groups of circular and non-circular, ferrous and non-ferrous targets. Also, we calculated the distribution of the objects as shown in Table 5.

In order to determine the basic properties of Target ID distribution and encode them in simulation the trial session was conducted. We observed that target IDs for non-ferrous coins tend to deviate much less than for ferrous heterogeneous items such as rusty steel wires and bottle caps. Therefore, to prepare a simulated target ID profile we sampled from normal distribution with varying parameters as follows:

- Ferrous coins:** $\sigma = [0.1, 1.0]$ and $\mu = [0..4]$
- Non-ferrous coins:** $\sigma = [0.1, 1.0]$ and $\mu = [5..28]$
- Ferrous non-coins:** $\sigma = [1.0, 2.5]$ and $\mu = [0..4]$
- Non-ferrous non-coins:** $\sigma = [1.0, 2.5]$ and $\mu = [5..28]$

As in Figure 4, the mean (μ) directly corresponds to the target ID of the object and variance (σ) embeds our observed knowledge that non-coins tend to have higher variance than coins. The latter parameters were chosen from the given range randomly for every sample.

Furthermore, we sampled a set of simulated objects from this distribution to determine how the number of data points affects the model's performance. The samples were used to extract features and train the classifier.

During multiple training attempts, we observed significant overfitting. To address it, the fraction of dropout was gradually increased from 10% to 40% where accuracy and Area Under the ROC Curve (AUC) were maximised. Additionally, smaller batch size showed slower learning and required more steps to converge; hence we increased samples per batch from 5 to 20. The higher number of hidden units increased the complexity of the model. Thus it became more prone to overfitting. We found the 2-hidden layer architecture with 256 and 32 hidden units to be the most optimal solution for our problem.

Coins	Non-coins	Accuracy (%) \pm SD	AUC \pm SD
1000	1000	87.38 ± 0.40	0.92 ± 0.01
100	100	79.20 ± 2.87	0.82 ± 0.04
75	75	77.06 ± 3.15	0.79 ± 0.04
60	60	76.33 ± 7.33	0.79 ± 0.06
50	50	73.40 ± 6.10	0.74 ± 0.04
40	40	73.00 ± 4.97	0.70 ± 0.07

Table 6: The accuracy of object-type classifier with varying size simulated training dataset. The evaluation was performed using 10-fold cross-validation. The standard deviation (SD) estimated from 10 attempts.

The best performance was achieved using 2-hidden layer architecture with 256 and 32 hidden units per layer respectively, ADAM optimiser [17] with default Tensorflow parameters [31], 40% dropout and batch size of 20. The results can be seen in Table 6.

One may observe an expected trend that accuracy and AUC increased with more data samples. Also, the accuracy and AUC results were deviating less as more data was used in training.

However, we expect the evaluation measures to be boosted when training on real-world data, as the model could learn more correlations between the input features. For instance, [15] states that target ID is also influenced by the size of the object, which we do not consider in our simulation.

5.3.2 Experiments on real-world data

We collected 76 distinct data samples, however, after further processing only 62 (26 coins and 36 non-coins) were applicable for the DNN classifier. Mostly the data was corrupt due to the failure by our mobile application to record sound, the noise in the audio or noisy detector signals which misrepresented the target response.

We used the same set of multilayer perceptron hyperparameters as in the simulation. The training and evaluation procedure was performed using 10-fold cross-validation, and our DNN classifier managed to achieve 100% mean accuracy over ten attempts.

However, the perfect accuracy raises concerns whether the dataset was large enough to make generalised conclusions. The deep neural network managed to accurately separate different data points to classes because we did not consider complex-composition, varying orientation, depth and multiple target cases were response signals overlap. Also, we relied on a critical assumption that all targets have either rectangular or circular shape.

To sum up, this model is not generalised and achieves 100% accuracy only because we classify very few well-defined data points which would be not the case if we consider complex real-world targets with varying orientation, topology and composition.

5.4 Discussion

It is important to note that we relied on several critical assumptions which simplified our model and allowed to achieve high performance.

To begin with, we classified objects only as circular or rectangular, which is not the case in real-world. We identified these issues at the beginning and therefore employed convolutional neural network to determine the target shape. The CNN can be simply enhanced to work on a broader range of more complex topologies by increasing number of parameters. However, the

problem arises in the resolution of the pinpoint signal as sharp edges of the small targets (less than 10cm in length) are smoothed by the DSP. Also, the noise clutters the target representation and introduces misleading features as shown in Figure 9. This can be overcome by fusing target ID profile with the shape information to produce target-unique signatures.

Moreover, it has been observed by [32] that target response signal varies with depth and position of the object. However, in our experiments, the orientation of items was constant with target's flat surface facing the normal of a coil. We tried burying items at different positions, depth and response signal was indeed altered; thus it is important to collect more data points of objects with transformed target signatures and use them to generalise the neural networks.

While non-ferrous objects exhibit exclusive target ID profiles, we experienced difficulties in extracting unique signatures of ferrous items. In Figure 4 one can see that coins made of steel have almost identical distributions. In such cases, we had to rely on the spatial representation of the target. However, if the buried items were small, the topology characteristics were abstracted by the DSP, and our system failed to infer a unique target signature. To determine intrinsic properties of target signatures of ferrous objects we could cooperate with industry, though this may jeopardise our aim to open-source the system.

Finally, we only observed one item at the time, although, it is not the case in real-world where metal detector might receive interfering responses from multiple objects especially in areas filled with metallic clutter. While the built-in DSP stabilisation algorithm partially combats this problem, our system must be able to detect distinct objects and treat the responses individually.

6. CONCLUSIONS AND FUTURE WORK

This paper proposes a system to identify the type of a buried metal item based on hand-held metal detector output. We suggest methods to extract characteristic features describing the shape and composition of the object. To create a scalable solution which could be easily enhanced to work with complex real-world cases, we employ deep learning architectures used to identify target topology and classify objects by type.

Our final results indicate that with position tracking mechanism it is possible to determine the trivial symmetry of underground objects based on the audio signal generated by the DSP of the metal detector.

Furthermore, the results of the object-type classifier suggest that fusion of the information about shape and composition can be used to produce unique signatures corresponding to specific object types. We showed that target signature in non-ferrous metals is proportional to the volume of the object and conductivity. Besides, we observed that target ID variance aids to identify heterogenous-composition items and targets made of pure metals.

We also claim that our system is fast and light-weight as we use pre-trained deep learning models to infer the input data. Finally, our experiments hint that the mobile application used to collect the coil position coordinates, and the audio response signal could be unified with the processing pipeline to create a real-time object-type recognition framework. Although the system calls for a set of adjustments to work outdoors, such solution would be in high demand in post-conflict underdeveloped or developing countries where hand-held metal detectors are the only feasible solution for demining and clearance procedures.

6.1 Future Work

We investigated a simplified model of a metal target identification problem, and even though the conducted experiments show promising results, we must generalise the system to work accurately in outdoor conditions. Therefore further enhancements offer a great deal of potential future work.

Most importantly, we must collect a significantly larger amount of data points to learn existing topologies, unique target signatures and capture properties of desired object types. Crowdsourcing solution can be considered to obtain voluntary data from archaeologists or individuals practising hobby metal detecting. The collected target descriptions could be aggregated to a knowledge database and used to train our system. However, this approach would require every participant to have access to Google Tango Project tablet. The issue can be solved by updating our mobile application to use conventional camera and marker surface (above buried object) to track the movement of the coil on a ground. Such adjustment would make our system compatible with any Android tablet and thus increase system's accessibility.

The shape recogniser suffers from noisy input (i.e. contour plots of the pinpoint signal). Currently, to alleviate the problem, we employ Nearest Neighbour smoothing, however, the data acquisition procedure must be revisited and investigated to find less noisy data collection methods. The issue would be partially solved if our motion tracking mechanism could incorporate the behaviour of a MD stabilisation system, though it would require the cooperation with industry to obtain the proprietary information.

The proposed architecture does not utilise decision theory which could integrate prior information about probabilistic object distributions. While our study considers trivial-target dataset, where false negative responses are of no harm, in demining such mistake may be tragic, hence a mechanism to incorporate prior target distribution information is crucial for system usage in real-world.

Acknowledgments:

I would like to take this opportunity to thank my supervisor Dr Bjørn Sand Jensen for sincere support, guidance and insight throughout the year. Without his enormous dedication and invaluable feedback, this study would not have been possible.

7. REFERENCES

- [1] C. M. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, Inc., New York, NY, USA, 1995. Pages 140-148.
- [2] C. M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006. Pages 225-284.
- [3] M. D. Bloice, C. Stocker, and A. Holzinger. Augmentor: An image augmentation library for machine learning. *CoRR*, abs/1708.04680, 2017.
- [4] C. Bruschini. *A Multidisciplinary Analysis of Frequency Domain Metal Detectors for Humanitarian Demining*. PhD thesis, Vrije Universiteit Brussel Faculty Of Applied Sciences Department Of Electronics And Information Processing, 2002.
- [5] P. S. Z. F C. Bruschini. 3D EMI trajectories for the visualisation of metal object properties. Published and free to access on Academia.com.
- [6] D. Ciregan, U. Meier, and J. Schmidhuber. Multi-column deep neural networks for image classification. In *2012 IEEE*

- Conference on Computer Vision and Pattern Recognition*, pages 3642–3649, June 2012.
- [7] D. C. Ciresan, U. Meier, J. Masci, L. M. Gambardella, and J. Schmidhuber. High-performance neural networks for visual object classification. *CoRR*, abs/1102.0183, 2011.
 - [8] G. Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, 2(4):303–314, Dec 1989.
 - [9] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, June 2009.
 - [10] S. B. Gokturk, H. Yalcin, and C. Bamji. A time-of-flight depth sensor - system description, issues and solutions. In *2004 Conference on Computer Vision and Pattern Recognition Workshop*, pages 35–35, June 2004.
 - [11] Google. Google Colab project, Apr 2018. Available at <https://colab.research.google.com/> [visited on 5th April 2018].
 - [12] M. K. Habib. Humanitarian demining, mine detection and sensors. In *2011 IEEE International Symposium on Industrial Electronics*, pages 2237–2242, June 2011.
 - [13] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*, 2012.
 - [14] R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge University Press, New York, NY, USA, 2nd edition, 2012.
 - [15] R. Horton. Understanding your X-Terra. Minelab Electronics Pty Ltd, Minelab Electronics Pty Ltd, 118 Hayward Avenue, Torrensville, SA 5031, Australia, 2011.
 - [16] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei. Large-scale video classification with convolutional neural networks. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1725–1732, June 2014.
 - [17] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.
 - [18] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.
 - [19] Y. LeCun, Y. Bengio, et al. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 3361(10):1995, 1995.
 - [20] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten Zip code recognition. *Neural Computation*, 1(4):541–551, 1989.
 - [21] M. Leshno, V. Y. Lin, A. Pinkus, and S. Schocken. Multilayer feedforward networks with a nonpolynomial activation function can approximate any function. *Neural Networks*, 6(6):861 – 867, 1993.
 - [22] M. E. P. Ltd. Minelab X-Terra 705 instruction manual. Technical report, Minelab Electronics Pty Ltd, Minelab Electronics Pty Ltd, 2 Second Ave, Mawson Lakes, SA 5095, Australia, 2016.
 - [23] E. Marder-Eppstein. Project Tango. In *ACM SIGGRAPH 2016 Real-Time Live!, SIGGRAPH ’16*, pages 40:25–40:25, New York, NY, USA, 2016. ACM.
 - [24] R. N. McDonough and A. D. Whalen. *Detection of Signals in Noise*. Academic Press, Inc., Orlando, FL, USA, 2nd edition, 1995.
 - [25] T. R. Mint. Coin designs and specifications. Available at <https://www.royalmint.com/discover/uk-coins/coin-design-and-specifications/> [visited on 31st March 2018].
 - [26] V. Nair and G. E. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 807–814, 2010.
 - [27] I. C. of Red Cross. *Antipersonnel Mines- Friends or Foe?* ICRC Publication, 1996.
 - [28] T. N. Sainath, O. Vinyals, A. Senior, and H. Sak. Convolutional, long short-term memory, fully connected deep neural networks. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4580–4584, April 2015.
 - [29] J. Schmidhuber. Deep learning in neural networks: An overview. *Neural Networks*, 61:85 – 117, 2015.
 - [30] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. *CoRR*, abs/1312.6229, 2013.
 - [31] Tensorflow. Tensorflow documentation: AdamOptimizer, Apr 2018. Available at https://www.tensorflow.org/api_docs/python/tf/train/AdamOptimizer/ [visited on 20th April 2018].
 - [32] M. D. J. Tran, C. Abeynayake, and L. C. Jain. A target discrimination methodology utilizing wavelet-based and morphological feature extraction with metal detector array data. *IEEE Transactions on Geoscience and Remote Sensing*, 50(1):119–129, Jan 2012.