FN Herstal
Research & Development
FN® e-novation

University of Liège
Faculty of Applied Sciences
Department of Electrical Engineering & Computer Science
Doctoral College in Electricity, Electronics and Computer Science.

PhD Thesis

Embedded Detection of Gunshots by AI in Real-time (EDGAR)

Nathan Morsa

Advised by Hugues Libotte and Louis Wehenkel

Thesis submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy (PhD) in Engineering Science

July 2025

Abstract

Proper management and maintenance of infantry firearms are critical to operational readiness, safety, and cost control within modern military contexts. Weapon fleets degrade differently depending on their usage. Traditional methods of tracking weapon usage through manual shot counting are inaccurate and inefficient. Electronic shot counters enable preventive and predictive maintenance by providing armourers with precise, quantitative measures of weapon usage.

Traditional electronic shot-counting solutions require expert knowledge and extensive study for each firearm. In response to market demands for increased customisation and shorter lead times, recent machine-learning-based solutions have been proposed. However, these solutions are limited by the difficulty of acquiring sufficiently large, fully-labelled datasets, restricting their generalisation capabilities.

To address this limitation, we propose EDGAR (Embedded Detection of Gunshots by AI in Real-time), a novel technique capable of working directly with data labelled only by the total number of events in a time series. This approach significantly reduces labelling efforts, enabling the creation and use of datasets several orders of magnitude larger than those typically available. Furthermore, we show how the technique can be extended to effectively support discrimination between live and blank ammunition, as well as detection of suppressor usage, with minimal additional computational overhead. We demonstrate that these classification tasks can be executed in under 100 ms on highly constrained embedded microcontrollers, thus enabling real-time shot detection.

Extensive experiments conducted across a range of firearms, including FN Minimi, FN MAG, FN M2HB-QCB, and M134 Minigun, demonstrate that EDGAR significantly outperforms unsupervised algorithms and achieves comparable or superior performance to human-generated state-of-the-art algorithms, particularly for discrimination tasks. Practical field deployments validate the robustness and real-time capabilities of the proposed method.

Finally, leveraging the large datasets made accessible through our approach, we empirically investigate the impact of dataset size on model performance. We identify critical thresholds required for effective model generalisation and provide practical guidelines for efficient dataset acquisition and model training.

These results enable a new generation of electronic shot counters and targeted maintenance strategies, thereby reducing maintenance costs, preventing incidents, and increasing the operational availability of weapon fleets.

Résumé

La gestion et la maintenance appropriées des armes légères d'infanterie sont essentielles pour assurer la disponibilité opérationnelle, la sécurité et le contrôle des coûts dans les contextes militaires modernes. La dégradation des parcs d'armes varie selon leur profil d'utilisation. Les méthodes traditionnelles de suivi de l'usage des armes par comptage manuel des tirs sont inexactes et inefficaces. Les compteurs de coups électroniques permettent une maintenance préventive et prédictive en fournissant aux armuriers des mesures précises et quantitatives de l'utilisation des armes.

Les méthodes classiques de comptage électronique nécessitent des connaissances spécialisées et une étude approfondie pour chaque arme. En réponse aux exigences du marché en matière de personnalisation accrue et de réduction des délais de déploiement, des solutions récentes basées sur l'apprentissage automatique ont été proposées. Cependant, ces solutions sont limitées par la difficulté d'acquérir des ensembles de données suffisamment grands et entièrement annotés, ce qui restreint leurs capacités de généralisation.

Afin de surmonter cette limitation, nous proposons EDGAR (Embedded Detection of Gunshots by AI in Real-time), une technique novatrice capable de fonctionner directement à partir de données étiquetées uniquement par le nombre total d'événements dans une série temporelle. Cette approche réduit considérablement les efforts d'annotation, permettant la création et l'utilisation d'ensembles de données de plusieurs ordres de grandeur supérieurs à ceux généralement disponibles. De plus, nous montrons comment la technique peut être étendue pour supporter efficacement la discrimination entre munitions réelles des munitions à blanc, ainsi que la détection de l'utilisation d'atténuateur, avec un surcoût computationnel minimal. Nous démontrons que ces tâches de classification peuvent être exécutées en moins de 100 ms sur des microcontrôleurs embarqués à ressources très limitées, rendant ainsi possible la détection des tirs en temps réel.

Des campagnes d'essais menées sur un éventail d'armes, notamment la FN Minimi, la FN MAG, la FN M2HB-QCB et la M134 Minigun, montrent qu'EDGAR surpasse largement les algorithmes non supervisés et atteint des performances comparables ou supérieures aux algorithmes de référence générés par des experts humains, en particulier pour les tâches de discrimination. Des déploiements pratiques sur le terrain valident la robustesse et les capacités temps réel de la méthode proposée.

Enfin, en exploitant les ensembles de données étendus rendus accessibles grâce à notre approche, nous examinons empiriquement l'effet de la taille de l'ensemble de données sur les performances du modèle. Nous identifions les seuils critiques nécessaires à la généralisation du modèle et fournissons des recommandations pratiques concernant l'acquisition de données et l'entraînement des modèles.

Ces résultats ouvrent la voie à une nouvelle génération de compteurs de coups électroniques et de stratégies de maintenance ciblées, réduisant ainsi les coûts de maintenance, le risque d'incidents et augmentant la disponibilité opérationnelle des parcs d'armes.

Acknowledgements

This thesis represents the culmination of a challenging endeavour undertaken alongside professional responsibilities. Its successful completion owes much to the support and contributions of numerous individuals, to whom I extend my sincere appreciation.

First, I wish to thank Hugues Libotte for offering me the opportunity to start this project and for accompanying me through to its conclusion. Hugues, I certainly would not have embarked on this path had you not proposed it. Your recognition of the potential in this idea and your trust allowed the work the time it needed to mature.

I am also grateful to Louis Wehenkel for supervising this thesis and for consistently providing valuable feedback. His insistence on rigour significantly enhanced the quality of the work, and his guidance was essential in navigating the academic world and ensuring that ideas and results were clearly explained.

I wish to thank Pierre Geurts for his helpful feedback during the early stages of this project. Thanks are also due to Gilles Louppe, whose initial suggestions were instrumental in shaping my approach to this problem.

A particular acknowledgement goes to Louis Huggenberger, whose dedication to the practical components of this thesis was indispensable. His meticulous approach to experimental testing was critical in ensuring the reliability of the results. I greatly appreciate his responsiveness, attention to detail, and perseverance in overcoming inevitable obstacles encountered during testing. I also thank Fabien Coenegrachts for facilitating seamless data collection and providing timely assistance whenever difficulties arose.

I sincerely thank Benjamin Istaz and François Legras, who recognised the importance and necessity of this work, enabling me to pursue it despite competing priorities. Additionally, I am grateful to Jérôme Bronckart, whose insights into operational aspects informed several sections of this research.

Lastly, to everyone who provided their time, feedback, or support during this endeavour, I extend my heartfelt appreciation. Your collective contributions have enriched this thesis, and I am truly thankful for each one.

Contents

Chapter 1

Introduction

1.1 Motivation for electronic shot counters

The acquisition and ongoing maintenance of small arms, along with the procurement of spare parts and accessories, represent substantial investments for modern armies. The management and upkeep of this weapon fleet demand considerable time and directly impact the reliability of weapons, a factor critical to mission success and the safety of the user. In recent years, the defence industry has demonstrated a growing focus on preventive and predictive maintenance strategies, which aim to both improve operational availability and reduce costs in the long run.

For infantry firearms, the primary factor influencing deterioration is the number of rounds fired. Secondary factors, such as the firearm's age or periods of non-firing usage, also contribute to wear but are generally of lesser significance. Consequently, tracking the number of rounds fired is a crucial aspect of weapon maintenance, as it provides a quantitative measure of wear and tear. This can be likened to a car's odometer. Who would feel confident purchasing a modern vehicle without one?

Paradoxically, small arms management remains one of the few areas of military logistics that has yet to transition fully into the digital age. Weapon usage, and particularly the number and type of rounds fired, is still frequently tracked through manual paper records by either the end-user or the armourer. These notes are prone to human error and can prove unreliable. Entries might be omitted or subject to inaccurate estimations. In particular, it is not uncommon for these to be based on the total number of rounds fired by a group while the real distribution of shots between the weapons can be highly skewed. For example, weapons carried by support-role operators may not be fired at all during a mission, whereas others might be used extensively, even for tasks such as expending leftover ammunition, resulting in significant additional wear.

This reliance on paper logs makes the automated exploitation of data impossible, despite the increasing importance of big data approaches in logistical optimisation. Most attempts at digitalisation rely on manual transcription of these paper trails into spreadsheets, which is prone to typing errors. Decisions based on inaccurate usage tracking

may lead to inappropriate maintenance that either decreases operational readiness through under-maintenance, or increases costs through over-maintenance.

One solution to this problem is the systematic use of electronic shot counters. These devices not only continuously and accurately track weapon usage and maintenance status, but their systematic deployment also enables armouries to perform live inventory monitoring, a task that is still often performed manually. These devices are either clipped on or embedded in the firearm and usually rely on MEMS accelerometers, measuring in particular the acceleration along the firing axis. This enables them to detect and count shots, evaluate burst rate, and in some cases discriminate between types of ammunition. Armourers can thus prioritise maintenance for heavily used weapons and potentially defer servicing for those that have not been operated since their last check.

Additionally, maintenance can be performed more reliably through data-driven, predictive strategies. Modern firearms maintenance guidelines often specify estimated lifespans for individual components in terms of the number of shots fired. By leveraging this information, parts can be replaced preemptively, thereby preventing in-service failures. This new maintenance paradigm is estimated to save up to 50% on armourer labour time, increase fleet availability up to 90%, and lower operating costs by up to 20% [1].

Beyond simple shot counting, electronic shot counters can provide deeper insights into weapon usage via additional metrics derived from individual shot detection. For instance, the temporal distribution of shots allows the calculation of metrics such as:

1. **Moving parts speed:** Time between the ignition and the moving parts of the weapon reaching their maximum depth. This relates to backpressure, which is the force exerted by the combustion gases that flow back into the operating system of an automatic gas-operated firearm. Excessive speed can be caused by misconfiguration or the use of maladapted barrel accessories and may result in accelerated deterioration due to high impact forces. Modern firearms with hydraulic dampers tend to regulate the overall behaviour of the weapon, masking these issues over longer timespans, yet repeated high-force collisions still damage the mechanism.

2. **Cycle rate:** Time required for the weapon to complete a full operating cycle. Erratic cycle rates can be a symptom of different types of mechanical issues or of poor ammunition quality.

3. **Burst rate (or *cyclic rate of fire*):** Average number of rounds fired per unit of time, typically in rounds per minute (rpm), during continuous automatic fire. Higher-than-specification burst rates can indicate a misconfiguration and a risk of parts damage due to excessive forces during the automatic reloading operation. Lower than specification burst rates can indicate mechanical issues, such as a lack of lubrication, and frequently lead to ammunition feeding incidents.

4. **Schedule of firing (or *effective rate of fire*):** Average number of rounds fired over a longer timeframe, indicative of the intensity of usage. High schedules of firing lead to high weapon temperatures, particularly inside the barrel. This can result

in a permanent weakening of its structure. Schedules of firing above manufacturer-recommended specifications will reduce average lifespan and may lead to catastrophic failures during operation. Counters equipped with thermal models tailored to the firearm can estimate barrel temperature over time, enhancing preventive maintenance decisions.

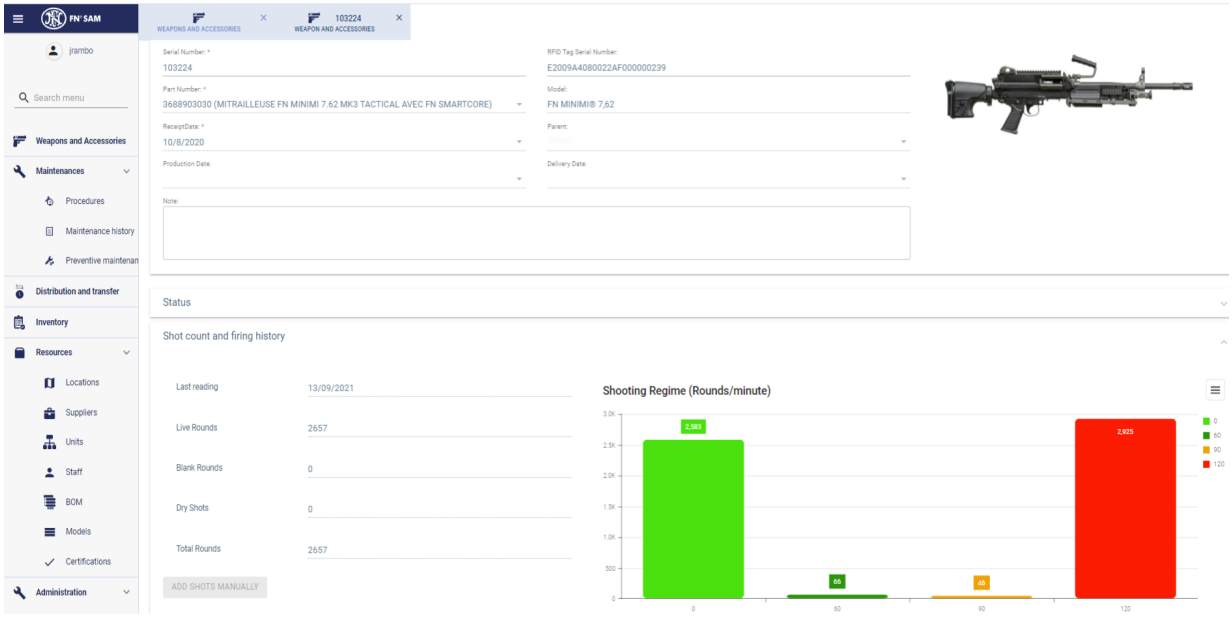


Figure 1.1: Example data retrieved from an electronic shot counter [2].

Once a reliable record of the number of rounds fired is obtained, more advanced shot counters can leverage the more subtle variations in their input signal to perform shot discrimination, which is the identification of the type of ammunition fired, or the weapon configuration in which it was fired. This capability enables armourers to carry out more targeted maintenance interventions based on detailed usage data.

In particular, blank rounds (training ammunition containing only powder and no projectile) present several maintenance challenges. One of the most significant is incomplete powder combustion, which results in increased residue deposits within the mechanism compared to standard live rounds. In contrast, live ammunition deposits copper residue inside the barrel due to projectile friction, progressively degrading accuracy. This also heats up the barrel significantly, leading to thermal fatigue. While the mobile parts experience similar stresses with both types of ammunition, live ammunition causes increased wear on the locking mechanism compared to blank ammunition. In summary, blank rounds result in less long-term mechanical damage but lead to increased residue accumulation, necessitating more frequent cleaning.

Similarly, the presence of accessories that influence weapon behaviour, such as suppressors, is also consequential. Suppressors function by trapping propellant gases, which raises backpressure within the system. This elevated backpressure accelerates wear on both the

mobile parts and the locking mechanism. Additionally, thermal conduction from the suppressor back into the barrel further exacerbates thermal fatigue. For example, a typical assault rifle rated for over 35,000 shots under nominal conditions may experience a major component failure after only 16,000 rounds if just 25% of the ammunition is fired using a suppressor. In the case where all rounds are fired with a suppressor, failure can occur before reaching 8000 rounds.

Another important type of event is the practice of "dry firing" a weapon, which involves of releasing the moving parts of a weapon in the absence of a round. While seemingly innocuous, repeated dry firings can cause substantial damage. The mechanism is designed to dissipate most of the closing energy through the resistance encountered while moving a round from its magazine (or equivalent) into the chamber, and subsequently lock it in place. Without a round, this energy is instead transferred directly through a forceful collision between the mobile parts and the locking mechanism, accelerating mechanical wear.

By detecting and individually tracking these different usages, electronic shot counters enable more accurate estimations of wear location and severity. This, in turn, supports more effective maintenance recommendations. Moreover, the automation of this process allows spare parts to be pre-ordered based on aggregate fleet-level degradation, thereby improving overall operational availability.

1.1.1 Smart armoury

As discussed, one of the principal applications of electronic shot counters lies in small arms management. These devices serve as both the digital identity card of a weapon and its comprehensive electronic logbook, containing complete records of its usage and maintenance history, accessible even in the field. The current trend in the defence market is to broaden this to a complete digitalisation of weapon fleet management, also known as a "smart armoury". This system integrates electronic shot counters for weapons or weapon parts (such as an individual barrel) and RFID tags on other equipment.

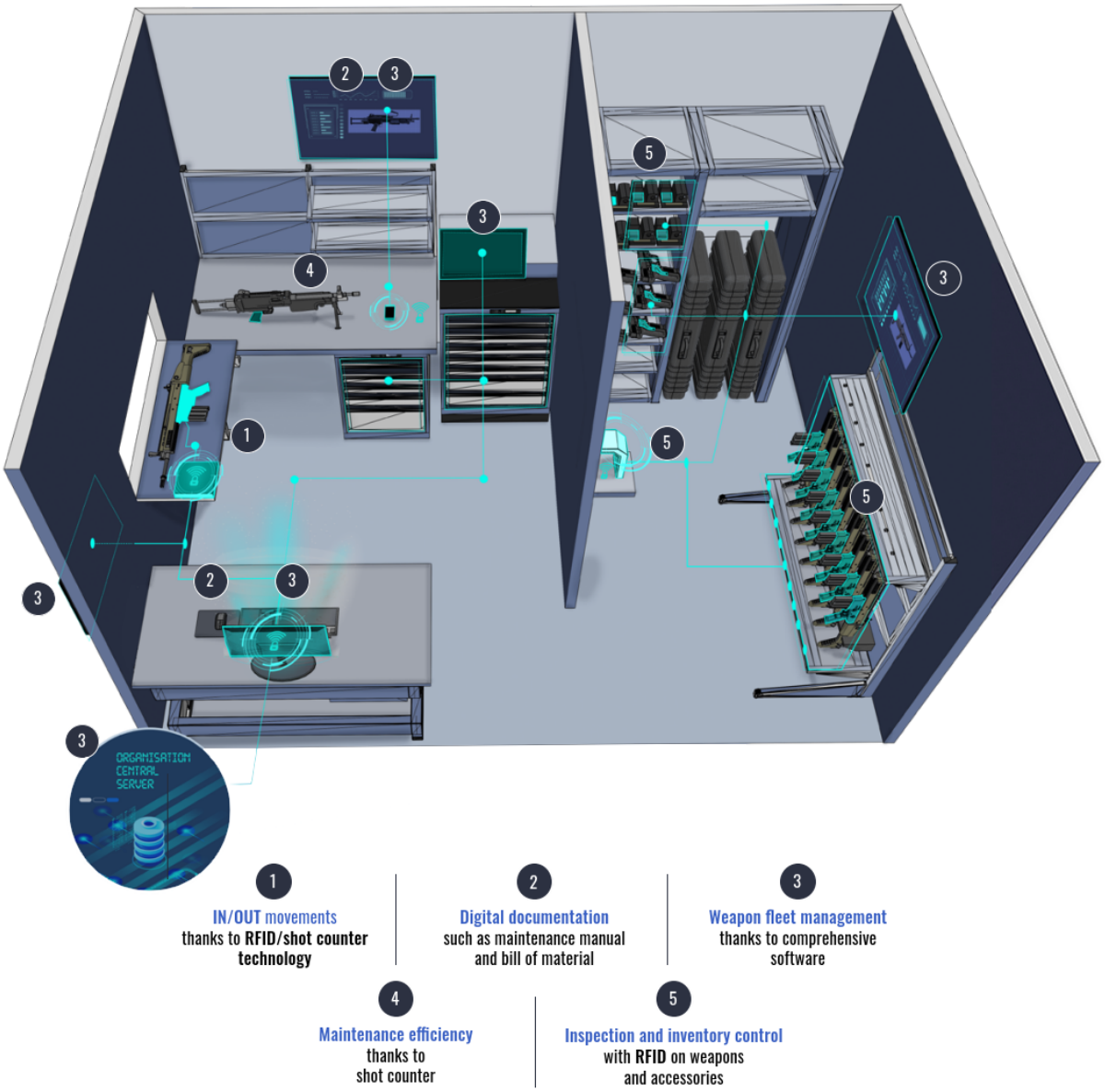


Figure 1.2: FN ® Smart Armoury concept [1].

Figure 1.3: CANiK Smart® Armory concept [3].

A smart armoury can be configured such that electronic shot counters are automatically detected and read when weapons enter or exit the armoury, consolidating this information into a centralised, army-wide database. This provides armourers with a consolidated overview of their fleet at any time. Aggregated wear-and-tear metrics allow for accurate forecasting of upcoming maintenance needs and spare parts requirements, analysis of part consumption rates across units, and optimisation of fleet utilisation through wear levelling. For each individual weapon, a complete chain of custody and its current location can be maintained within the centralised system. Users can also track all operations and suspected defects logged both in the centralised database and in the field via the electronic shot counter. Relevant technical documentation from the manufacturer is also immediately made available, eliminating the need for bulky paper manuals and catalogues.

To assess the potential cost savings offered by electronic shot counters, 150 machine guns (a mix of FN Minimi 5.56 and 7.62) from a modern army unit were equipped with electronic shot counters for a year. The resulting shot data are presented in Figure 1.4.

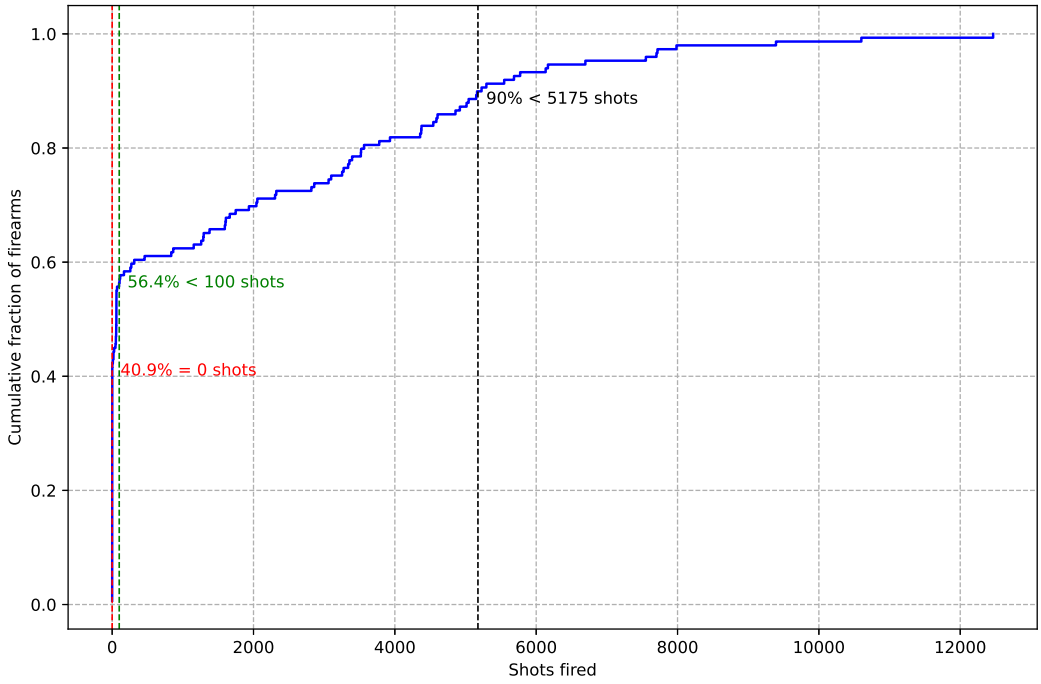


Figure 1.4: Distribution of the number of shots fired in real-world usage. Usage of 150 Minimi machine guns was tracked over a full year.

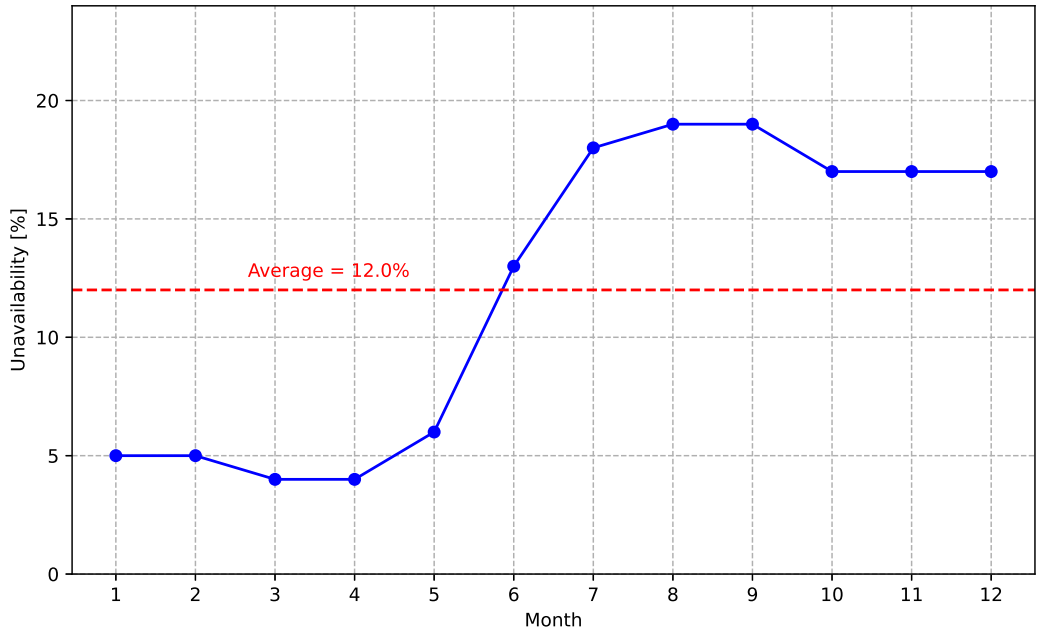


Figure 1.5: Fleet unavailability. Usage of 150 Minimi machine guns was tracked over a full year.

The data reveal that over 40% of the weapons were not fired at all during the observation period, and over 56% saw only minimal use. Among the remaining weapons, usage increased roughly linearly until the top decile, at which point a sharp rise was observed. The most heavily used weapons recorded more than twice the shot count of those at the 90th percentile. These findings confirm expectations of a highly uneven usage distribution across the fleet.

Figure 1.5 presents the percentage of the fleet deemed "unavailable" during that period. These are firearms which have incurred a major failure, and are in need of repairs before they can be used again. On average, these weapons remained out of service for four months prior to repair.

In a smart armoury, electronic shot counters show their value in multiple ways:

Time savings

Following the findings of Figure 1.4, more than half of the weapons could be cleared without inspection. An army unit is estimated to spend 96 h per year on weekly inspections, and 128 h on more in-depth yearly inspections. Halving those requirements frees 112 h from this workload.

Additionally, armourers are estimated to spend up to 144h per year on inventorying operations, tracking the distribution to end users, and tracking transfers between armouries. These operations are all performed using paper records, which are then manually encoded into spreadsheets at a later date. The RFID capabilities of the electronic shot counters allow the automation of these operations, and make the consultation of this data much faster. It is assumed that this would lower the associated workload by at least two-thirds.

Moreover, armourers can spend significant time retrieving (often outdated) paper documentation for the considered firearms. Since the deployment of electronic shot counters also comprises the associated software, armourers can be provided with an integrated software solution. When scanning a counter-equipped weapon as it enters the armoury, access to an interactive, digital version of the appropriate maintenance documentation can be automatically provided.

Overall, an army unit is estimated to save at least 208 man-hours of skilled armourer labour per year.

Availability and operational costs

The effect of electronic shot counters on operational availability is twofold. Firstly, by tracking the wear of each firearm part, the required spare parts can be pre-ordered, allowing for a much quicker maintenance cycle upon failure. Secondly, when intensive usage is detected on a specific firearm, preventive maintenance operations can be performed, replacing the required parts before failure. This allows the maintenance operation to be pre-planned, thus requiring minimal downtime and preventing much costlier firing incidents.

With such an improved maintenance procedure, it is estimated that half of the unavailable weapons in this study could have remained in use. By increasing the operational

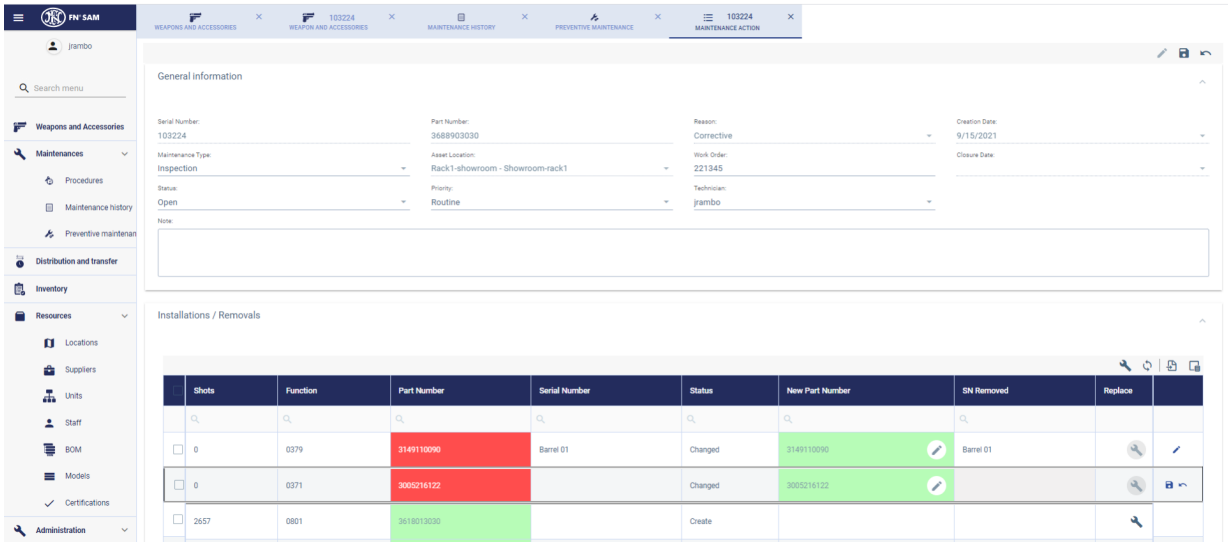


Figure 1.6: Example interface allowing armourers to perform spare parts tracking [2].

availability of a firearm, the overall cost of fleet acquisition is proportionally reduced. Additionally, firing incidents not only increase material loss but can also lead to significant operator injuries, associated with substantial human and financial costs.

A significant part of a weapon's transport costs arises from the need to ship it for maintenance and then back to the field. Even worse is the case where a weapon is shipped to the field, only to break down shortly after and require return shipment. Shot counters help avoid returning weapons that do not require in-depth maintenance, while ensuring that all necessary maintenance is performed on weapons destined for long-distance operations. Moreover, when considering an army's firearm fleet, electronic shot counters allow for wear levelling through rotation of the weapons between low and high-usage operators. This can significantly increase the expected lifespan of a firearm. Figure 1.7 compares the result after 15 years of different strategies:

- A "natural" levelling (partial random swap), in which 30% of the fleet is randomly redistributed each year. This simulates a fraction of the fleet being returned for maintenance and then reassigned at random.
- An intensive but uninformed strategy (full random swap), where the entire fleet is randomly redistributed yearly.
- A strategy informed by shot counters (top-bottom swap), where each year the top 15% of the fleet in terms of usage is swapped with the bottom 15%.

If we assume that a firearm would fail after 35,000 rounds, 15 years of usage with partial-random redistribution results in 27.4% of the fleet exceeding that threshold, and a full random redistribution 16.7%. By contrast, leveraging shot counters to perform a top-bottom swap reduces this figure to only 0.4%. This represents a difference of 27% in

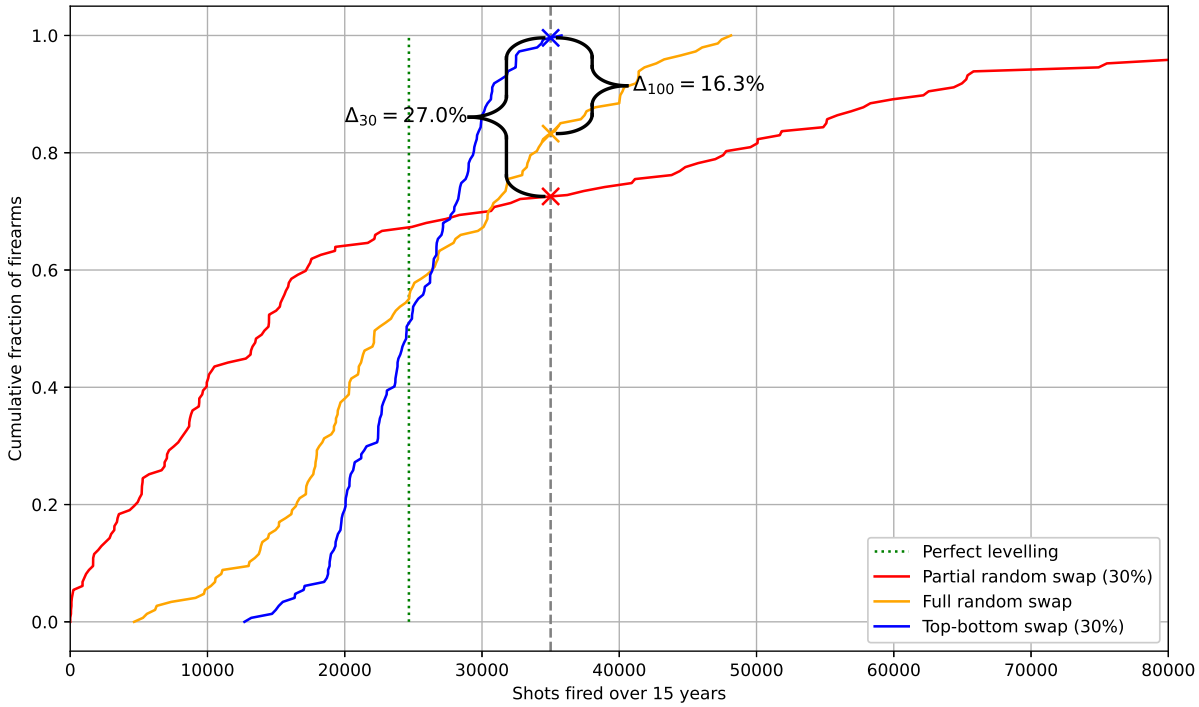


Figure 1.7: Wear-levelling strategies comparison. Data from Figure 1.4 is extrapolated over 15 years, and end of life is studied at 35,000 shots.

attrition after 15 years with the same number of swaps. Equivalently, the top-bottom swap strategy postpones reaching the same attrition level by 3 years.

Figure 1.8 illustrates the levelling effect of partial swap strategies. The informed top-bottom swap strategy yields significant improvements even for lower swap fractions, as the most and least-used weapons are correctly identified and exchanged.

Consequently, a smaller stock of reserve weapons and replacement parts can be maintained, leading to further savings in storage. In this pilot study, storage and transport costs are estimated to be reduced by 10%, and annual fleet attrition by 1–2%. Combined, these savings are expected to offset the acquisition cost of an electronic shot counter, even for lower-cost firearms. For a rifle (e.g. FN SCAR), savings on the order of 420 € are expected during the life cycle of the weapon (approximately 25 years). This increases to 900 € for a light machine gun (e.g. FN Minimi). For a heavy machine gun (e.g. FN M2), this represents a saving of approximately 2500 €.

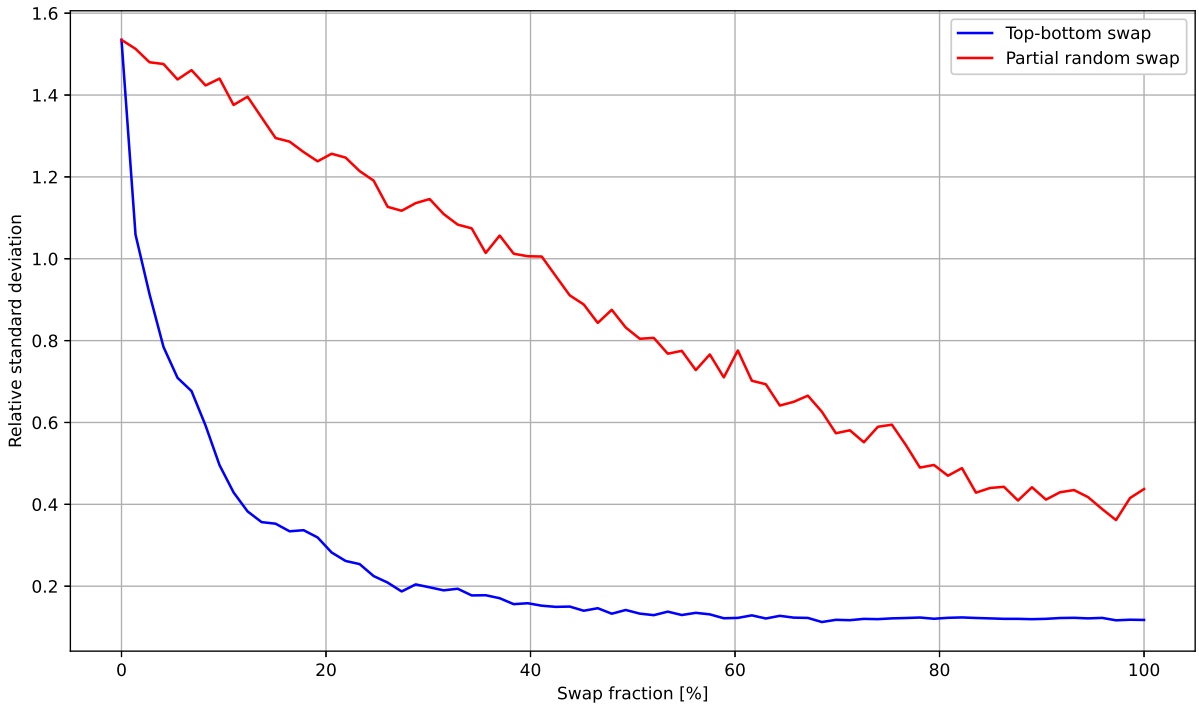


Figure 1.8: Levelling effect of partial swap strategies. Perfect wear levelling would result in a relative standard deviation of 0.

Fleet management

Once the data from electronic shot counters is consolidated into an army-wide small arms management system, a unified, real-time picture of fleet status becomes available. Fleet managers are able to inspect the current operational condition of weapons across and within units through a user-friendly interface. These data can be accessed independently without going through time-consuming requests between the fleet manager and individual armouries.

Currently, most armies size their weapons fleet using a push approach based on the available budget. Access to consolidated statistics allows fleet managers to incorporate lessons learned from the field and to identify the actual profile of parts consumption based on the type of training or the field operations performed. When deployed at scale, electronic shot counters also enable live tracking of the ammunition stock. Spare parts and ammunition procurement can then be carried out in a provisional manner, based on anticipated operations.

Furthermore, usage data and in particular the schedule of firing allow managers to assess whether the mandated doctrine is being followed in the field and whether it remains appropriate. Our field experiments have shown that changing a unit's equipment, such as substituting a MAG with a Minimi 7.62, without adjusting the associated doctrine, can increase maintenance needs by 30%.

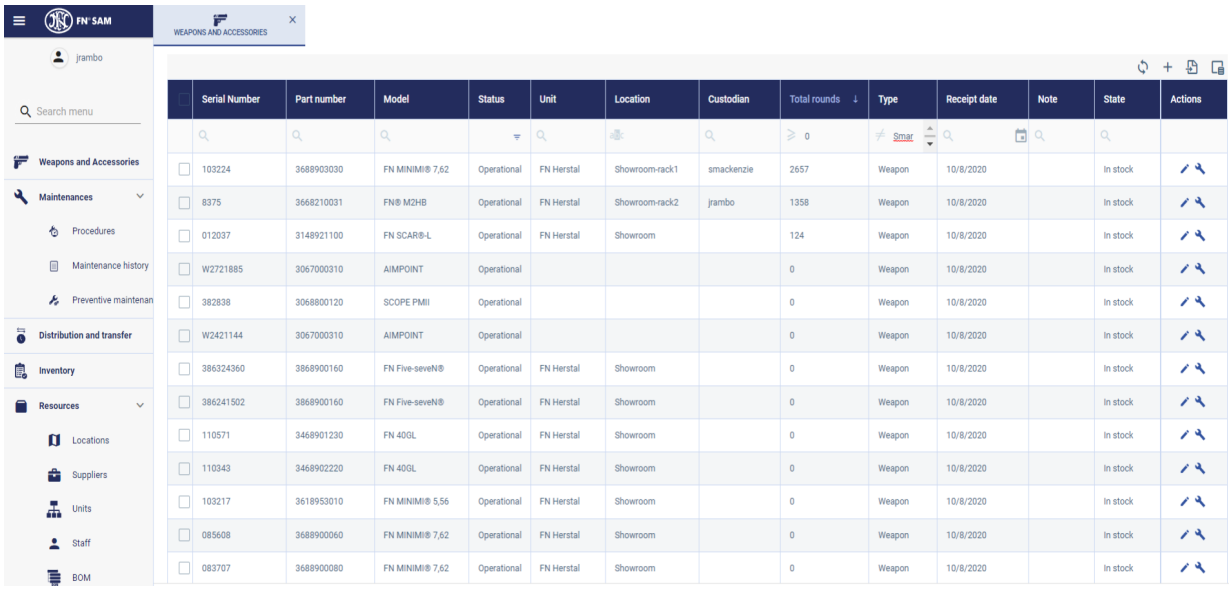


Figure 1.9: Example interface used by a fleet manager to consult fleet status [2].

1.1.2 Battlefield management systems

The second main application of electronic shot counters is through Battlefield Management Systems (BMS). BMS are suites of electronic devices and the associated software that provide military commanders and personnel with real-time information about the state of the battlefield. They typically allow a mix of both free and formatted communication within a squad, as well as between squad leaders and unit commanders. Formatted information can be plotted on a digital map, which is accessible in real time by the command station.

In this context, electronic shot counters bridge the gap between the soldier-level BMS device (often a smartphone) and the firearm itself. By detecting shots and communicating them in real time, the information that a soldier is engaged in active combat can be reported automatically. When combined with the orientation data from multiple soldiers, enemy positions can be estimated and plotted on the battlefield map. This significantly enhances real-time situational awareness.

Additionally, by accurately counting the number of shots fired, BMS can monitor the ammunition usage of each soldier. This enables commanders and logistics sections to plan resupply missions proactively, reducing the risk of catastrophic shortages.

These engagements can also be recorded and replayed for after-action analysis and training purposes. This is particularly valuable in law enforcement applications, providing precise records of who fired, where, and when. In the longer term, shot counters can also be used to verify compliance with training requirements and support training certification.

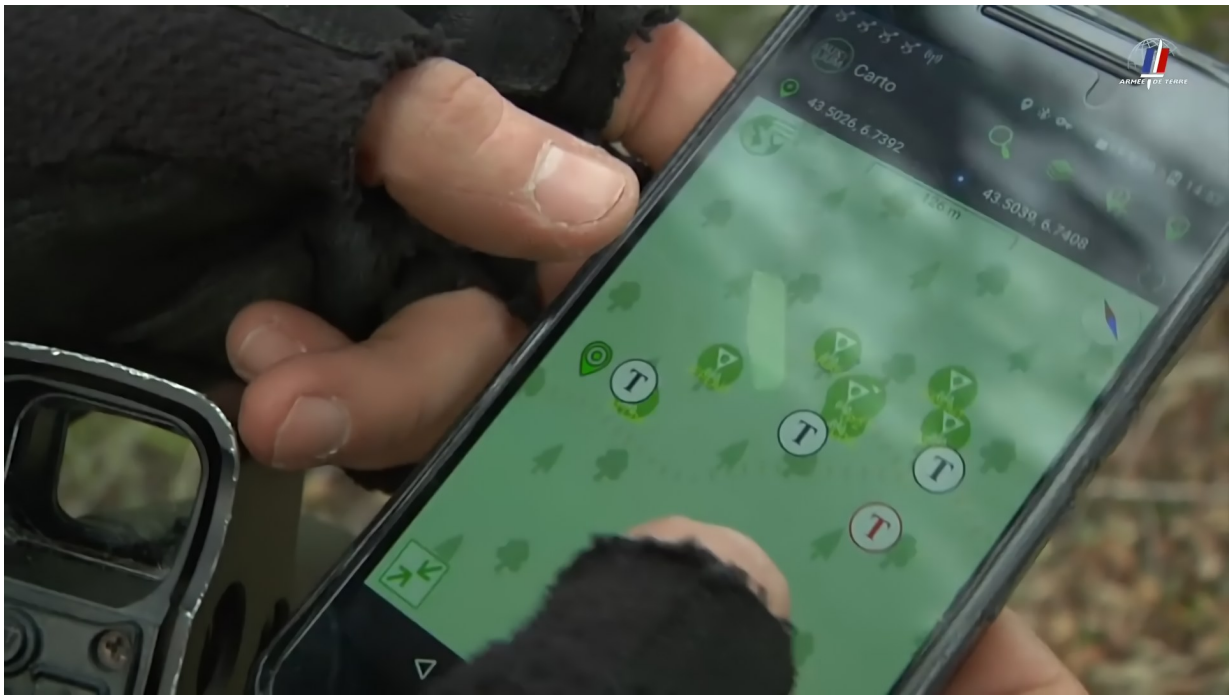


Figure 1.10: Soldier using the French BMS Auxylium, connected to FN SmartCores [4].

1.2 Types of shot counter and constraints

1.2.1 Consumer-grade shot counters

Shot counters in general encompass a wide range of accessories. For usage in controlled conditions such as firing ranges, the most widespread devices are acoustic shot counters and light barriers.

Acoustic shot sensors rely on a microphone to detect sound spikes, which are generally assumed to be shots. These devices are relatively inexpensive and can be used with any type of weapon. Their main drawback is their frequent need for manual calibration, especially in enclosed environments where echoes may occur. In noisy surroundings, they are frequently subject to false positives caused by other sources of highly impulsive sounds. They also become unreliable when multiple weapons are fired in close proximity. While suitable for consumer usage, the exposed microphone design limits their suitability for harsh military conditions.

Light barriers use a combination of light emitters and receivers (usually infrared), in which the receiver detects a beam interruption caused by a passing projectile. It is common to use a combination of two barriers, which drastically reduces false positives and allows measurement of the projectile velocity. Their main advantage is their high accuracy for both projectile detection, and shot timing. They are able to act as a reference standard for the burst rate measurements of other devices. Most common implementations require a static setup, which is only compatible with firing range conditions. Some

Figure 1.11: Example of an acoustic shot counting device [5].

Figure 1.12: Example of a shot counting device based on light barriers [5].

implementations attach the light sensors to the weapon, or even embed them inside the firing mechanism. While effective in the short run, they require frequent maintenance as the sensors quickly get obstructed by firing residue. The light emitters also require more power than alternatives relying on passive sensors.

Mechanical counters also exist and provide a battery-free implementation. However, their limited accuracy and lack of more advanced features make their use relatively unpopular.

Another type of device that might be considered as a shot counter uses a strain gauge to measure the slight expansion of the barrel when fired. These are also able to provide the full pressure curve experienced by the barrel, providing a deeper insight that is useful for characterising ammunition or tuning a weapon's backpressure system. This type of sensor is generally only used as lab equipment, as its placement around the barrel subjects it to intense heat during sustained firing in the field. Similarly, barrel oscillations can be considered [6].

1.2.2 Military-grade shot counters

For military-compatible use, which is the focus of our work, shot counter hardware has to be designed with a number of constraints in mind:

- **Full integration:** Requirements for high mobility in the field require the device to be fully integrated with its associated weapon. This rules out any accessory that is not fully attached to the weapon. While external attachments can be considered, they add undesirable bulk, take up space which might be required by high-value

accessories such as targeting systems, and are more easily broken off or removed. Therefore, fully embedded shot counters are usually preferred. The device should ideally be embedded close to core components of the firearm to prevent it from being removed or swapped out. This typically places it near the firing mechanism.

- **Retrofitting:** Firearms usually have a long (10+ years) development cycle, especially if they break away from existing designs. As a result, integration of electronics must usually be done as part of a retrofitting operation. Shot counters must therefore be designed in a way that allows their integration with minimal modification.

- **Limited available space:** As a consequence of the two previous requirements, the physical space available for shot counter integration is usually very limited. The integration must also work around pre-existing fixation points that cannot be altered in order to preserve compatibility with existing parts. This further constrains the available space and results in oddly shaped volumes for the electronics, batteries, and mechanical connections of the shot counter. In practice, this often limits the number of electronic components or the battery size. An example of constrained integration is illustrated in Figure 1.13.

- **Cost:** One of the main goals of electronic shot counters is to reduce costs through preventive maintenance. This typically limits the sales price of shot counters to the maximum potential savings for the user.[1] This amount is usually proportional to the value of the firearm, imposing tighter cost constraints on shot counters for less expensive weapons.

- **Maintenance-free:** For the shot counter to act as a monitoring device for the weapon's health even when maintenance is temporarily neglected, the shot counter itself should require no specific maintenance and in particular no battery replacement for a prolonged period of time. In practice, firearms are often placed in long-term storage possibly for more than a decade in non-ideal conditions. Conversely, they can also be suddenly deployed in the field and see intense and prolonged usage before a return to the armoury is possible. Therefore, shot counter batteries should ideally withstand at least 15 years in worst-case conditions, and be ready to operate immediately without any manual activation. This puts heavy constraints on the allowed power consumption, in particular on the current consumption while idle or sleeping. Considering the requirements on space and cost, current battery technology generally limits the maximum idle current of the system to 3 µA at 2.5 V, and peak current under 20 mA. When radio communication is used for data retrieval, supercapacitors or equivalent are often necessary to avoid exceeding this limit on peak current and guarantee the long-term integrity of the battery, further reducing the available space and budget.

[1] When shot counters are sold conjointly with a parent weapon, they can increase the overall value proposition of the ensemble, resulting in an increase in sales. In some specific cases where demand for shot-counting functionality is particularly high, cost requirements may thus be loosened.

Figure 1.13: FN SmartCore® for FN MAG® embedded inside its traditional pistol grip. The fixed positioning of the side plate screw holes limits the available space.

- **Environmental conditions:** During firing, shot counters can undergo peak accelerations exceeding 3000 g, and temperatures up to 85 °C. Electronic design and component selection must account for these constraints. For example, BGA packages have been known to detach from the PCB under such stresses, and QFN packages are therefore preferred. The sensor must also withstand the same environmental testing as its associated weapon, including full submersion in water, and direct exposure to dust, grease, saline environments, and various types of chemicals. For NATO-associated countries, the required testing is described in the STANAG 4370 standard.

Considering these constraints together, only two types of sensors are currently considered compatible with military applications: magnetic-based solutions and accelerometer-based solutions.

Magnetic shot counters

Shot counters using magnetic sensors generally add magnets to the moving parts of the weapon and monitor their displacement. One of the main advantages of this type of counter is the possibility of providing a battery-free solution by harvesting energy from the generated magnetic field to power the sensor during operation. In addition, the signal is directly generated by the actual motion of parts within the weapon, which simplifies analysis and results in excellent shot detection accuracy. However, the addition of magnetic components to the firing mechanism often means modifying critical components, and integration in a significantly space-constrained area of the weapon. This requires the solution to be heavily customised for each weapon, with high development and retrofitting costs. While the promise of a battery-free device is appealing, the lack of energy generation outside of shots makes the computation of the burst rate challenging and the computation of the schedule of firing is currently considered impossible. In addition, the device must be externally energised for data retrieval, therefore requiring a more complex reading solution.

The FN SmartCore® for FN SCAR® (Figure 1.14), described in a related patent [7], is an example of such a device. Through a slight modification of the hammer spring guide rod, a series of alternating magnets is attached. A fixed coil around the path of the magnets will pick up the excitation from the alternating magnetic field when the weapon is fired. An accelerometer is still used for live/blank ammunition discrimination. The device can then be externally powered through its NFC antenna for data retrieval.

Such an integration with relatively small modifications required to the existing parts is rarely possible. Another example of such a device is the Heckler & Koch Bolt Motion Sensor System (Figure 1.16), which uses a similar principle as described in promotional articles [8] and in greater detail in a related patent [9]. However in this case, important modifications were done to the mobile parts and upper receiver. These are among the most expensive parts of the weapon, as a result a retrofit to add a shot counter to an existing weapon would be prohibitively expensive with regard to the added value. They can thus only be offered for specifically designed new weapon models. Another example is

the Radetec RISCpro (Figure 1.15) which provides shot counting for Glock pistols through the addition and replacement of a significant number of parts.

Figure 1.14: FN SmartCore® for FN SCAR® being integrated at the back of a compatible lower receiver and powered by energy harvesting [10].

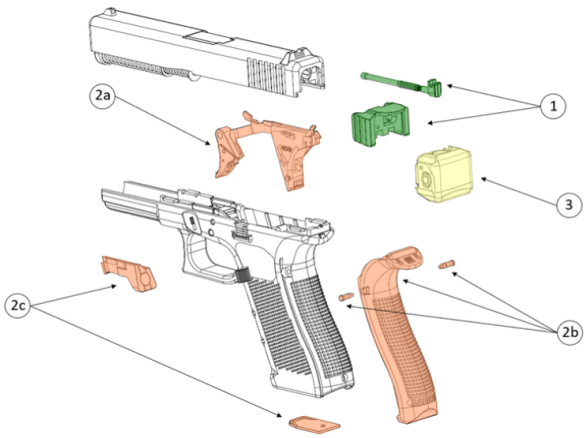


Figure 1.15: Assembly of the Radetec RISCpro in Glock pistols [11]. Parts in colour are added or replace original parts.

(a) Modified BMSS mobile parts.

(b) Modified BMSS upper receiver.

Figure 1.16: Heckler & Koch Bolt Motion Sensor System HK437RC [8]. The mobile parts of the weapon have been modified to include 4 magnets of alternating polarity. The upper receiver is also modified to include the fixed coils and the electronics.

Accelerometer shot counters

In contrast, accelerometer-based solutions offer wide hardware compatibility and easy integration. The electronics can be placed freely within or on the weapon,[2] and no mechanical parts are needed except for fixation resulting in better cost-effectiveness. In particular handles are often left hollow to reduce weight.[3] As a result, most shot counters for military applications are accelerometer-based. Their main drawback is the complexity of the signal analysis required for highly accurate shot detection. One of the main objectives of this work is reducing this complexity, lowering the development difficulty of accelerometer-based solutions. While our work is applicable to other types or a mix of input signals, accelerometer-only inputs will be the first application. Accelerometer-based shot counters and their challenges will therefore be discussed in more detail in the rest of this thesis.

Examples of commercially available devices include the FN SmartCore for FN Minimi and FN M3M by FN Herstal [10] (Figure 1.17), WeaponLogic devices by Senseer [12] (Figure 1.18), the PAL by LMT Advanced Technologies [13] (Figure 1.19), and the ShotPM device [14]. Except for ShotPM, on top of shot counting all of these provide capabilities to discriminate between live and blank rounds, with the PAL also detecting the use of a suppressor.

[2]Different placements of the sensor can result in significantly different acceleration signals. Some can provide a stronger and/or cleaner signal than others.

[3]Historically, a small field cleaning kit would sometimes be included inside the handle. This has, however, fallen out of fashion in recent years, weapon maintenance being reserved to specialised armoury personnel.

Figure 1.17: FN SmartCore® device embedded in a standard machine gun grip. The only mechanical modification needed is a lengthening of the screw that normally affixes the handle to the lower receiver to also affix the shot counter.

Figure 1.18: WeaponLogic devices by Senseer. These are accelerometer-based, and rely on an undisclosed machine learning to produce their shot counting algorithm [12].

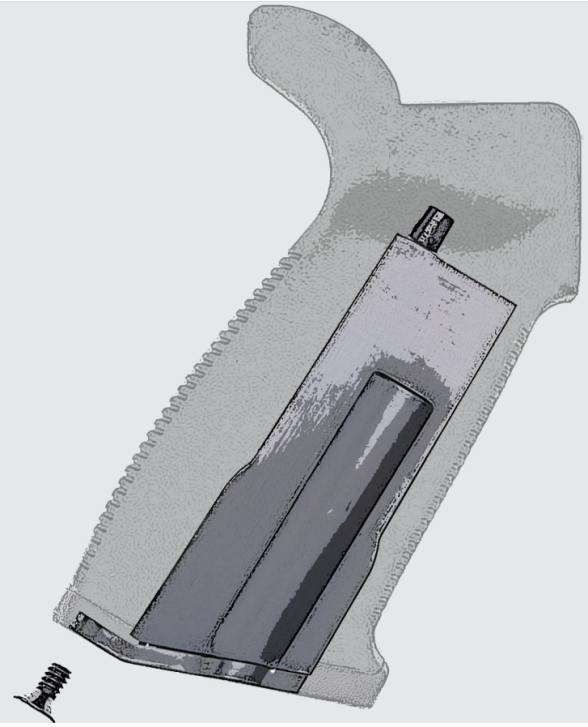


Figure 1.19: PAL by LMT Advanced Technologies embedded in a standard assault rifle grip [13].

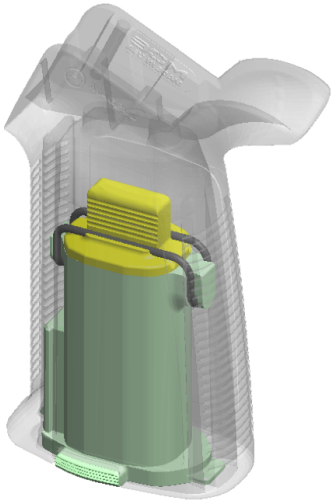


Figure 1.20: Example of an accelerometer-based ShotPM device embedded in a standard assault rifle grip [14].

1.3 The problems of shot counting

1.3.1 Dataset acquisition

Before one can look into the challenges of shot detection, it is important to understand which external factors can affect the behaviour of a weapon when firing. Doing so will allow us to construct a dataset that is representative of the full distribution of input signals that will be recorded by our sensor.

For a given model and calibre of firearm, the following external variables are known to have a significant impact on the weapon behaviour:

- Shooting sequence: The sequence of shots in time is influenced both by the operator through trigger pulls, and the weapon configuration such as semi-automatic, fully automatic, or automatic burst functionalities. This will lead to different overall signatures for the weapon as close shots might influence each other, and different mechanical effects can appear according to the different automatic firing mechanisms in use. In addition, specific mechanical effects can happen only at the end of a burst, and not between individual rounds.

- Ammunition type: We have described in Section 1.1 some of the behaviour differences appearing when using blank rounds. In addition, the lack of a projectile will lead to a significant difference in the forces and overall displacements of the firearm. The material and coating of the ammunition case also matter, different types leading to differences in the force needed to extract the case out of the chamber.

- Ammunition load: For a given type of ammunition, the quality and quantity of the powder can lead to a large variance in generated pressure and thus the force exerted during the automatic reload operation.

- Gas-operated reloading nozzle size: Modern firearms designs allow the manufacturer and armourers to fine-tune the backpressure by increasing or decreasing the size of a nozzle sitting between the barrel and the automatic reloading mechanism. This process is often performed iteratively according to the measured burst rate.

- Shooter position/mounting mechanism: The way an operator physically handles the weapon will influence its overall dynamic and displacement. Use of mounting mechanisms such as bipods, tripods, or complete firearm mounts (for example when used on a vehicle), will have an even larger influence.

- Mounted accessories weight: Modern firearms are often used with a large amount of accessories, adding several kilograms of weight to the platform. This increase in overall mass influences the displacement signature of the firearm.

- Ammunition loading type: Some firearms allow for multiple types of ammunition feeding such as magazines, linked belts fed from a pouch or free linked belts. The

type of loading, and the amount of ammunition left in the magazine/belt can lead to differences in the amount of force needed to feed the next round into the firing mechanism.

- Usage of a suppressor: Suppressors are a particularly impacting type of accessory. While an ideal suppressor will only modify the acoustic and visual signature of the weapon, in practice suppressors will have differing impacts on the backpressure. Internal testing has shown that some commercial suppressors can lead to significantly out-of-specification backpressures if the weapon is not tuned accordingly.

- Shooting angle: The angle of the firing axis from the horizontal changes the strength and direction of the influence of gravity on the reloading mechanism.

- Weapon and barrel temperature: Higher firearm temperature leads to increased generated propellant pressure. This can be observed by operators as the burst rate of their weapon progressively increases as it is operated. As a result, ambient temperature will also influence the longer-term behaviour of the weapon.

- Firearm dirtiness: Power residues and lack of lubrication increase internal friction in the weapon mechanisms, progressively increasing the required force of operation and ultimately leading to firing or feeding incidents if proper maintenance is not applied.

- Firearm wear and tear: Over the lifetime of the weapon, mechanical deformations and damage will influence the behaviour of the weapon in differing ways. In particular, the fixation linking the sensor to the platform might be loosened, leading to a difference in *perceived* signature of the weapon.

In order to acquire a dataset representative of the whole spectrum of possible real-life weapon behaviours, one would ideally have to record and control these external variables independently. However, the number of different variables and the large space of possible values for each of them can rapidly lead to an unmanageable number of combinations. As a result, a discrete and possibly reduced number of significantly differing settings for each variable will be chosen according to both expert armourer knowledge of the weapon and existing experimental results. A reduced number of combinations will then be chosen according to the development budget, attempting to capture both nominal and extreme behaviours of the weapon.

The sampling frequency of embedded accelerometers tends to shift by $\pm 10\%$ depending on sample and operating conditions. This causes our input time series to be unevenly sampled. The impact of this problem on model prediction is the subject of a future publication.

As we aim to maximise the quantity of information provided by each new recording, while minimising both the overall number of tests and the total time needed for it, increasingly complex shooting plans are produced. For example, switching the firearm from using live to blank ammunition requires removing all live ammunition from the firing range for safety reasons, mounting a specific accessory (blank-firing adaptor) which must be done on

a cooled-down barrel, and potentially re-tuning the weapon to ensure the correct amount of backpressure is applied. This transition thus requires a significant amount of time. A good shooting plan will thus avoid mixing live and blank firing in short succession to decrease the overall testing time, preferring for example transitions in shooting sequence or shooter position which can be done quickly.

On the other hand, data on the influence of weapon temperature or firearm dirtiness are often acquired indirectly through their progressive increase during testing, while taking into account the cooldown periods and maintenances in the shooting plan. A good shooting plan with regard to these variables will thus aim to mix weapon configuration as the temperature and deterioration increases, ensuring their influence is measured independently of the other variables.

These competing objectives must be blended together in such a way that the shooting plan is as optimised as possible while remaining comprehensible and easy to follow by the test operators. This is particularly a challenge when testing is done by the client, and the operators might not be informed of the full reasoning behind the test plan. Otherwise, this can lead to human error both involuntary (e.g. an overly complicated shooting plan has not been followed accurately) and voluntary (e.g. an operator might decide to reorder shooting operations to minimise their work time, not taking into account the effect on the acquired data).

The acquisition of negative examples (non-shot data) is relatively easier. Events occurring naturally before or after gunshots will be acquired at the same time. This also includes transport to and from the shooting range.

To further populate the dataset, specific acquisitions of specific repeated weapon manipulations can be performed. Experience with classical shot algorithms has shown that dry firing (i.e. releasing the mobile parts of the weapon as if to fire it, but without any ammunition present) is the most common source of false positives among non-shot events, and should thus receive particular attention. Most other weapon manipulations (such as reloading, handling, or aiming) do not result in large enough signals on the 200 g scale to be misclassified.

Falls on hard ground, particularly on the front or back of the weapon, produce complex and highly impulsive signals which can also result in false positives. However, testing has shown that falls of sufficient force to produce such signals also cause significant damage to the weapon. In the context of shot counting for preventive maintenance, firearms subjected to such falls will thus require emergency maintenance well before any false positive count becomes significant. Associated recordings should still be included in the dataset when possible, but specific destructive testing is judged unnecessary.

1.3.2 Counting problem

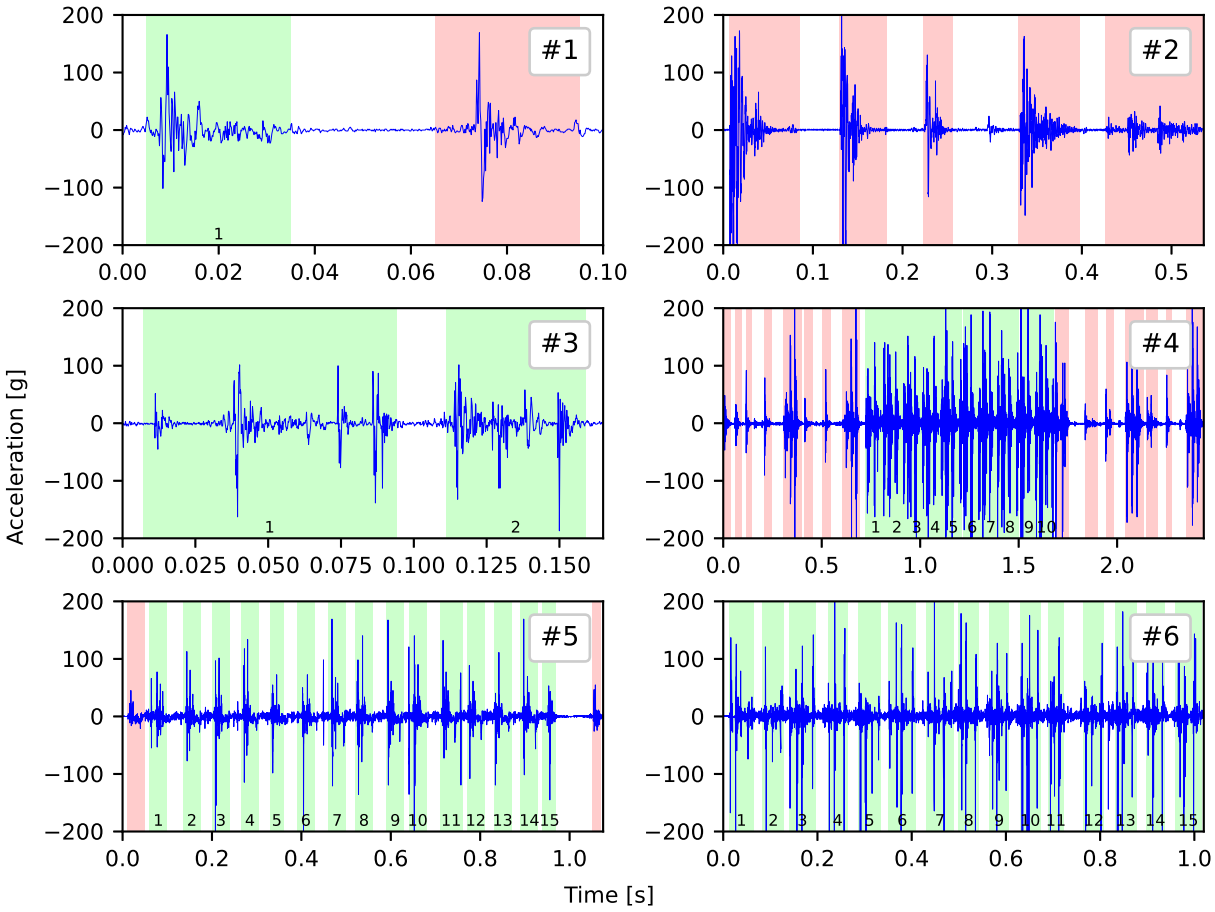


Figure 1.21: Example inputs (high resolution available in digital version). Events have been individually labelled by a human expert between shot in green (numbered at the bottom) and non-shot in red.

The problem of counting the number of shots in a time series lies firstly in detecting relatively rare individual events from unrelated ones, such as normal weapon manipulations or falls on hard ground interspersed between shots. We illustrate some example inputs in Figure 1.21. Example #1 shows a shot that is followed in close proximity by a purely mechanical event which should not be counted. Example #2 illustrates other non-shot events. In addition, shots need to be discerned from each other for proper counting and the start of each one properly identified for burst rate evaluation. However, shots can present a wide variety of signatures depending on external factors (see Section 1.3.1). Example #3 shows shots fired with the same weapon as #1 presenting differing signatures both from #1 and from each other even though they happen in close succession. Example #4 shows how shots can blend into each other rendering individual detection difficult, with the common occurrence of a mechanical event also blending in at the end of a burst. Mechanical events

visually very similar to shots also happen in close proximity. Examples #4, #5, and #6 are taken from different weapons and show how the signature can significantly differ between them.

We also observe that automatic firearms fire in concentrated bursts. In particular, machine guns can be used in prolonged (>1 min.) firing sequences. The frequent lack of a large working memory thus requires embedded algorithms to perform some form of real-time inference.

Reliable embedded data acquisition is also challenging. Figure 1.22 shows that even with a relatively low-calibre (5.56×45 mm NATO) rifle, significant accelerations above 1900 g can be measured in the firing axis. We can thus expect significant saturation when using low-power embedded sensors. Due to the highly impulsive nature of the signal, we also expect it to be undersampled. We also observe that acceleration in all axes is highly correlated in this case. Depending on the weapon and sensor placement, this is not always the case.

Figure 1.23 shows the acceleration for a high-calibre (12.7×99 mm NATO) heavy machine gun, which includes samples above 4000 g in the longitudinal axis. While the Pearson correlation coefficient between the longitudinal and vertical axis is 0.85 indicating a strong correlation, the correlation with the transverse signal is only -0.09. A possible explanation is the difference in weapon construction, the M2HB being firmly constrained from moving in the transverse direction. by the use of a tripod, while the reloading operation of a cycle is performed in that direction.

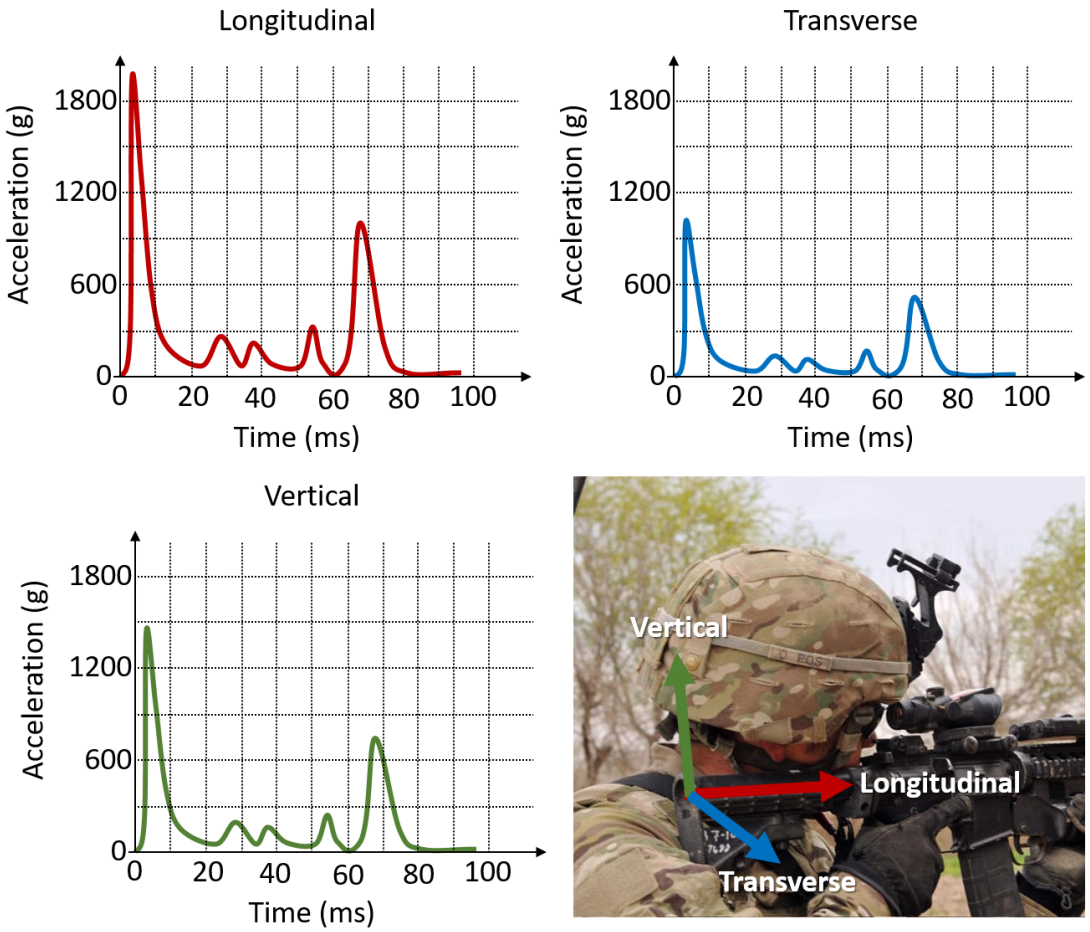


Figure 1.22: High-g acceleration data for an AR-15 rifle measured at the handle grip, filtered to low frequencies.

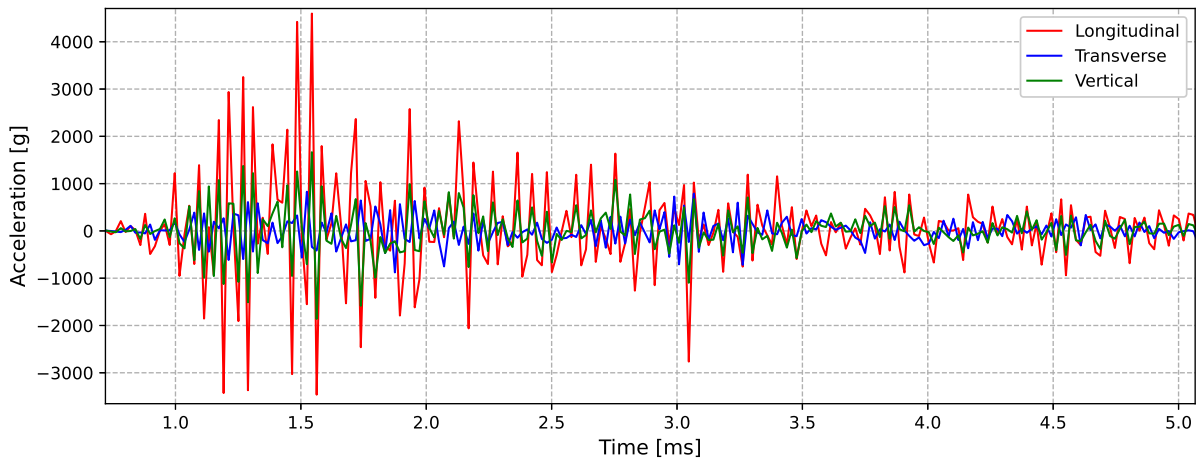


Figure 1.23: High-g, high-frequency acceleration data at the beginning of a shot on the FN® M2HB-QCB, measured at the base of the barrel.

1.3.3 Discrimination problem

Once shots are correctly identified, we can proceed to classify them according to their ammunition type. In particular, discriminating between live and blank rounds (illustrated in Figure 1.25) is often of significant interest. As we will discuss in greater detail in Chapter 3, the presence or absence of mass ejection causes a fundamental difference in the movement of the weapon. However, due to the large number of external variables, identifying a single metric which captures this difference is challenging, particularly because these differences may manifest in various ways under unknown external conditions. A few examples are given in Figure 1.24.

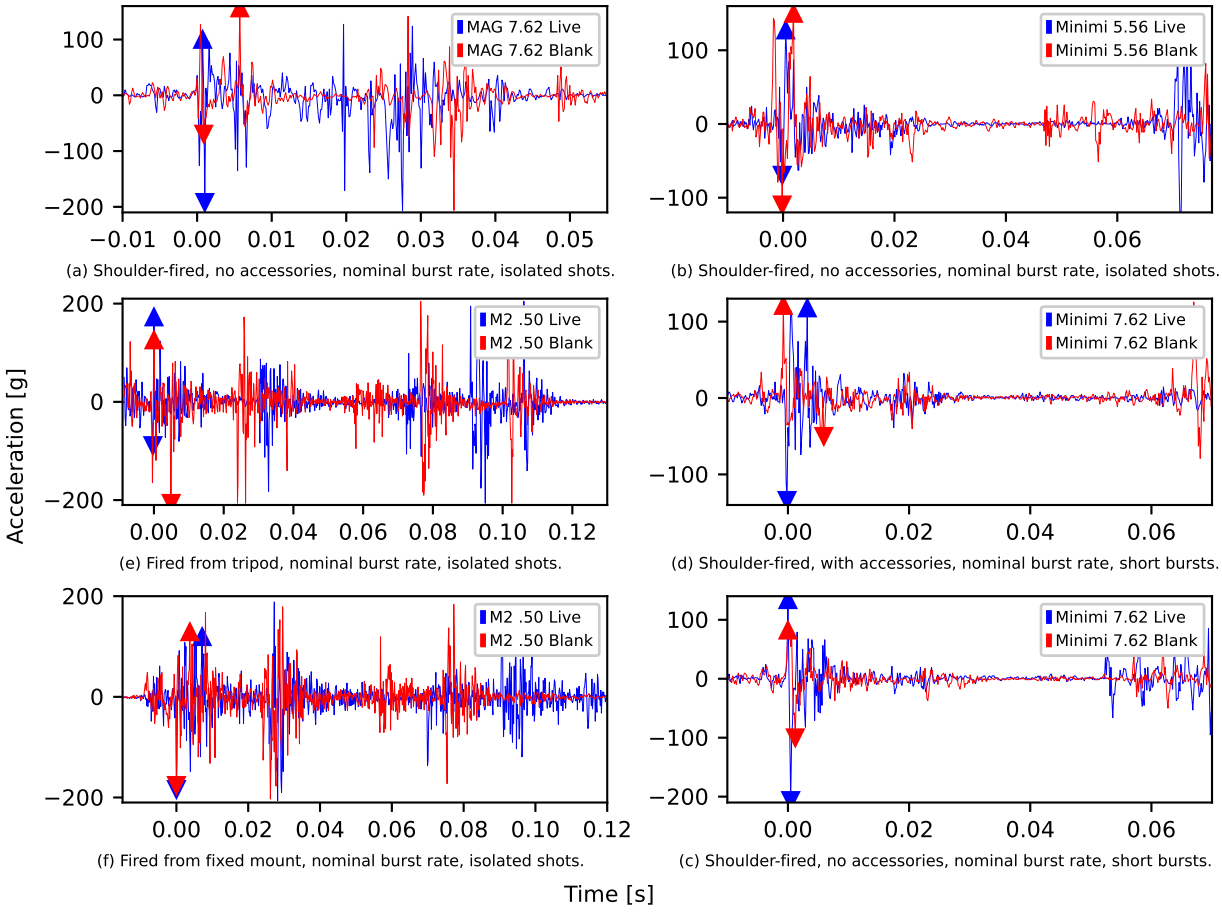


Figure 1.24: Example inputs (high-resolution images available in the digital version). Pairs of live/blank rounds were acquired under identical, controlled conditions. Each pair is aligned on an arbitrary significant peak in the signal (at $t = 0$). P_P and P_N are denoted by ▲ and ▼, respectively. The distinction between live and blank ammunition is obscured across different weapon systems due to substantial differences in firing cycles. Even within a single weapon, minor configuration adjustments or the intrinsic variability between individual rounds can lead to significant variations in behaviour.

(a) 5.56x45 mm NATO (b) 7.62x51 mm NATO (c) 12.7x99 mm NATO (.50 cal)

Figure 1.25: Types of ammunition used in this study. Each calibre features from left to right: a standard live cartridge ("ball"), a tracer round, and a training round ("blank").

1.3.4 Dataset labelling problem

In order to create a database of acceleration recordings with the aim of developing shot detection algorithms, raw data can be acquired by recording the sensor output while operating the weapon. However, the labelling of this data presents a significant challenge: Creating strong labels identifying the position and type of individual shots in the time series requires the intervention of a firearms expert to discern shots from unrelated acceleration events. This is a very labour-intensive task and is prone to human error. On the other hand, weak labels in the form of a total number of shots fired in a given recording can be produced cheaply and reliably by manually counting the number of rounds used, especially when these come in pre-numbered containers such as ammunition boxes and magazines as illustrated in Figure 1.26.

Firearms experts can readily leverage these weak labels to manually develop algorithms that achieve high performance by comparing their current predictions with the ground-

truth total number of shots in a given time series.

However, one of the main drawbacks is the large reliance on human work time. Although common techniques can be applied when adapting the system to support new types of firearms, their unique intrinsic signature must still be analysed by an expert in the specific weapon type to determine the quantitative characteristics of a shooting event and incorporate them into the detection algorithm.

As detailed in Section 1.3.1, this is further complicated by the fact that each single model of weapon can have a large number of signal-impacting variables. The combination of these can lead to a very wide range of different possible signatures that must be accounted for. Firearms can also accept different modes of behaviour. For example, incorporating a suppressor or being mounted on external fixation systems (vehicles, remote weapon stations, ...) results in multiple possible shot signatures which must be simultaneously accepted while still rejecting mechanical noise. As the range of possible behaviour increases, it becomes exponentially more difficult for the expert to formulate an algorithm highly accurate in all possible situations. This can result in significant development costs and lead time for every new weapon and weapon variant.

(a) 20-round magazine of 5.56mm ammunition [15].

(b) 100-round box of 12.7mm ammunition [16].

Figure 1.26: Illustration of typical ammunition containers, each holding a fixed number of rounds.

1.4 The case for machine learning

As the market for shot counters develops, requests for new types of shot counters come with increasingly high requirements and shorter acceptable lead times. This is part of a general trend in the firearms industry. Historically, firearms were produced in high volume and sold in configurations largely identical to the original design. Nowadays, clients of the defence industry expect a higher grade of personalisation, including modifications impacting the core design of the weapon and necessitating new R&D work. On the other hand, fast product delivery is also increasingly expected in a highly competitive market. This has led to a fundamental restructuring of manufacturing processes, from make-to-stock to assemble-to-order and even engineer-to-order strategies.

In recent public tenders in the firearms market, shot counters have progressively become an important factor in awarding contracts. It is thus increasingly important for shot counters to be available with minimal delay, ideally by the time the weapon prototype is delivered to the client for initial testing, so that the shot counter can be evaluated alongside its platform. However, since a shot counter can only be developed when at least one corresponding weapon prototype is available, the window of time left for development between the end of the weapon R&D phase and the first delivery can be slim, sometimes on the order of a few weeks.

Combined with the increasing complexity of the many existing weapon variants leading to a correspondingly high development time requirement as previously described, the current techniques of manually written algorithms can result in a failure to meet the market.

The automation of shot counting algorithm generation aims to alleviate this issue by trading labour time from a qualified expert for computation time, which can be scaled through cloud computing according to urgency. In addition, automatically generated algorithms have been shown to outperform and replace human-generated ones in some situations. This capability aligns with a broader trend towards using weakly supervised methods in predictive maintenance applications [17].

Several factors explain the advantage of automated over expert-designed algorithms:

1. Current state-of-the-art techniques employ intermediate representations, such as a decomposition of the signal in sub-events (shocks) [18]. The signal processing algorithm that produces this intermediate step is typically the most complex part of the overall pipeline, it is usually a generic algorithm that is only slightly tweaked when generating a new shot counter. Conversely, if a method is derived for a machine learning algorithm which would start from the raw signal and compute its own intermediate representations, it would be feasible to implement a new algorithm from scratch, including signal processing layers specifically tailored to the new weapon.

2. When manually implementing a shot detection algorithm, it is generally not realistically feasible for the programmer to analyse every data curve in a constantly growing database. While the counting might be correct on a small enough sample, closer inspection of the data might reveal that correct counting occurs only by coincidence.

Statistical techniques can then be employed to detect outliers and further improve the detection boundary, but this is again a time-consuming process as any slight modifications of a manually implemented algorithm must be evaluated across the entire database to determine whether it improves overall performance. Contrastingly, a machine learning algorithm is very suited to this process, establishing a confidence score for every prediction and being incentivised to continuously improve them through its loss function, and therefore improving the decision boundary leading to better generalisation.

3. The problem of ammunition discrimination has proven to be challenging to implement using classical programming techniques. Human experts can usually visually distinguish data curves according to the type of ammunition, and accurate metrics can often be identified for a specific weapon configuration. However, finding accurate metrics for all possible weapon configurations has proven difficult. Machine learning can provide more complex models, which more robustly capture the underlying physical differences.

1.5 Outline

Following this introduction, Chapter 2 summarises the primary contributions of the thesis and their significance to the field.

In Chapter 3 we first detail a state-of-the-art technique for classical accelerometer-based shot counting and discrimination, providing a baseline and context for the research problems addressed. Subsequently, we undertake a comprehensive review of scientific literature and patent documentation related to shot counting and discrimination, highlighting current state-of-the-art methods and evaluating their strengths and limitations. Through this, we also analyse key research objectives and recurring challenges in existing studies.

Chapter 4 offers an in-depth description of the hardware utilised for data collection, field experiments, and final deployment. Particular attention is given to the challenges surrounding sensor calibration and wake-up mechanisms.

Chapter 5 defines metrics for rigorously evaluating counting and discrimination performance using data labelled solely by the total number of events within a time series (weak labels). These metrics are used in the benchmark evaluations presented later in the thesis.

Chapter 6 introduces the main shot-counting technique, capable of learning directly from weakly-labelled data while providing accurate detection and localisation of individual shots. The chapter elaborates on the training loss function, model architecture, training strategies, and hyperparameter tuning necessary to achieve real-time, high-performance predictions on embedded hardware.

Chapter 7 expands this technique towards ammunition type discrimination, with a specific focus on distinguishing live rounds from blanks. We examine and compare different approaches, and leverage the similarities with the counting problem to significantly reduce the computational complexity of the discrimination problem.

Chapter 8 discusses how the training datasets were acquired to ensure that a wide range of possible firearm configurations is represented. Utilising significantly larger datasets made usable by our technique, we explore how dataset size affects model inference performance. We identify a critical minimum dataset size for representativeness and provide practical recommendations for future data collection. Additionally, we demonstrate how the model can inform alternative data acquisition scenarios and use interpretability techniques to gain deeper insights into model behaviour and the characteristics of the input signal.

In Chapter 9, we benchmark our approach against state-of-the-art unsupervised and human-generated algorithms using self-collected datasets from six weapon types. Our results show substantial improvements, confirming effective leverage of weak labels. We deploy our technique in practical field tests and confirm its real-time predictive performance under operational conditions.

Chapter 10 summarises the main findings and contributions of the thesis and outlines potential avenues for future research.

Finally, Chapter 11 provides a comprehensive list of acronyms used throughout the thesis for ease of reference.

The supplementary materials are organised into five appendices: Appendix A contains detailed analyses of hyperparameter tuning, studying the landscape of individual settings with respect to model performance and providing a reference for future fine-tuning efforts. Appendix B provides extended benchmarking results beyond those presented in Chapter 9, offering additional insights into the experimental validation. Appendix C evaluates our counting technique using a third-party dataset. Appendix D details the candidate density profiles discussed in Section 6.5 for all considered weapons. Appendix E presents comprehensive saliency maps discussed in Section 8.4 for all considered weapons.

1.6 Related publications

The core technique presented in this thesis was initially disclosed in patent application EP4269931A1 [19], which protects this invention. A portion of this technique, along with preliminary results, was subsequently published in a 2023 paper [20]. Content from both the patent and the paper has been adapted and significantly expanded for this introduction and Chapter 6.

Additionally, a 2025 article [21] describes the extension of this core technique to ammunition discrimination. This article has been adapted and incorporated into Chapter 7, this introduction, and Sections 3.1.2, 8.1, 8.2, as well as the results discussed in Section 9.3.1.

Chapter 2

Contributions

This thesis provides significant contributions to the domain of machine learning applied to electronic shot counting and discrimination:

- A comprehensive literature and patent review identifies current state-of-the-art methodologies for shot counting and ammunition discrimination, highlighting their strengths, limitations, and recurring research challenges. This review aims to bridge the gaps between isolated research efforts and provide an integrated perspective. (Chapter 3)
- Having identified dataset acquisition and diversity as a major challenge of the field, we introduce a novel machine learning technique able to work directly on data labelled only with the total numbers of events in a time series. This enables the creation and use of datasets several orders of magnitude larger than those typically available. We propose a neural network model optimized for real-time performance on heavily constrained embedded microcontrollers. We detail model architecture, hyperparameter tuning, and training strategies, resulting in a system capable of accurate, real-time shot counting and discrimination. (Chapter 6)
- We extend the technique to address the discrimination problem, with a specific focus on distinguishing live rounds from blanks. This approach leverages the inherent similarities between counting and discrimination tasks to streamline training and reduce computational complexity. (Chapter 7)
- Through extensive benchmarking against both unsupervised and expert-designed state-of-the-art algorithms across datasets from multiple weapon types, we demonstrate significant performance improvements, validating the effectiveness of our technique in leveraging weak labels. (Chapter 9)
- Enabled by the large datasets made accessible through our method, we empirically explore the influence of dataset size on model performance, identifying critical thresholds necessary for effective model generalisation and offering practical guidelines for efficient dataset acquisition and model training. (Chapter 8)
- Finally, we conduct practical field deployments, demonstrating robust real-time performance and investigating critical hardware challenges specific to embedded shot counting devices, such as battery robustness under extreme acceleration and accelerometer reliability in firearm-specific conditions. (Chapter 9 and Chapter 4)

Chapter 3

Related work and state of the art

3.1 State of the art

In this section, we will describe the state-of-the-art method for classical accelerometer-based shot counting and discrimination. Specifically, all references to "acceleration" correspond to the output of the MEMS accelerometer affixed to the weapon, measuring the net dynamic forces acting on the sensor in its local reference frame during the firing cycle. Unless otherwise specified, we focus on the acceleration component aligned with the firing axis of the firearm. In practice, the acceleration signal is sampled at several thousand hertz, with the precise sampling frequency determined by the specifications of the sensor employed. The basic technique was developed by Joannes et al. until 2015. While a partial description of the method is available in the related patent [18], many implementation details are lacking, and have not been made public otherwise before this thesis. From 2015 onwards, we iteratively improved upon the technique through our own work. As will be examined in Section 3.2.3, other public classical techniques can be categorised as subsets of this one, or have major drawbacks (hard hypothesis on the weapon configuration, need for manual calibration, ...) making them unreliable for our use case. This technique is thus suitable to serve as a basis of comparison for the machine learning application of this thesis.

3.1.1 Counting

Most modern firearms, in particular those of interest for electronic shot counters, include a form of auto-loading of ammunition. When fired, the weapon will present a firing cycle comprising multiple steps, the specifics of which depend on the type of firearm. The most basic example is that of a gas-operated rifle which cycles in 3 main steps:

1. Shot: The ammunition is struck, causing ignition and projectile ejection.

2. Back: Just before ejection, a portion of the propellant gas is used to push back the mobile parts of the weapon against a spring. During this motion, the spent cartridge

is removed from the chamber and ejected. Most often, the spring is fully compressed and the mobile parts "hit" the back of the weapon.

3. Rearm: Using the energy stored in the spring, the mobile parts are pushed back forward. During this movement, they pick up a new round from the magazine and chamber it.

The cycle immediately repeats when the rifle is fired in "automatic" mode. Each step will generally show as a distinct peak on the acceleration curve as this corresponds to events in which the mobile parts start, change or stop their movement.

The most limited techniques, which will be discussed later in Section 3.2.3, will only consider the ignition event which is very susceptible to false positives. While a lot of techniques consider the full firing cycle, they tend to do so by either looking at very specific, fixed parts of the cycle or comparing the whole waveform to a template. This assumes that the cycle of the firearm is relatively stable. This might be acceptable for some specific weapons, or in applications where the shooting conditions can be controlled. However, in field conditions, the cycle of firearms might show major differences. For example, due to a lack of gas return or lubrication, it is common for the second peak to be absent as the mobile parts are gradually slowed and pushed back forward without ever hitting the back of the weapon. Another very common case is the last round of a magazine in rifles equipped with a bolt catch. This will interrupt the firing cycle with the mobile parts kept at the back, leading to the absence of the third peak. Additionally, the burst rate of a weapon might vary by up to a factor of two according to the firing conditions, significantly stretching the nominal cycle the template is based on.

The most important idea behind the Joannes et al. patent [18] is to actively decompose the shot into its significant events. This allows one to leverage the full firing cycle, while accounting for non-nominal conditions. This decomposition allows missing or additional events to be detected and handled according to context. Furthermore, by explicitly detecting the events marking the beginning and end of a cycle, variations in burst rate are naturally accounted for.

In addition, identifying these events allows one to look into the intra-cycle behaviour. For example, the time between the shot event and the recoil event can be a good indicator of the strength of the gas return. From a single shot, one is able to compute what the natural burst rate of the weapon would be, even in the presence of dampers which will smooth out the overall burst rate. This allows us to measure over or under gassing, both of which are potentially damaging for the weapon.

First, the raw acceleration signal is processed by computing a moving sum of square values, which serves as a measure of local energy. While the following equations assume a one-dimensional acceleration, higher dimensional input can be easily reduced to this case by computing the magnitude of the acceleration vector.

The moving sum of square values is given by:[1]

$$s[t] = \sum_{i=-w/2}^{w/2} (a[t+i])^2, \tag{3.1}$$

This signal is then convolved with a kernel $\mathbf{k}$:

$$\mathbf{v} = \mathbf{s} * \mathbf{k}, \tag{3.2}$$

where $*$ is the convolution operator. More explicitly, for a kernel $\mathbf{k}$ of size K samples, then the convolution is computed as:

$$v[t] = \sum_{j=-K/2}^{K/2} s[t+j]\, k[j] = \sum_{j=-K/2}^{K/2} \left(\sum_{i=-w/2}^{w/2} (a[t+j+i])^2 \right) k[j]. \tag{3.3}$$

The kernel used is illustrated in Figure 3.1, and has the shape of the derivative of a Gaussian function. This is known as the variation transform, and is a measure of the local variation of energy around this point. This is simplified by taking:

$$q[t] = \begin{cases} 0 & |v[t]| < Q \\ 1 & v[t] \geq Q \\ -1 & v[t] \leq -Q \end{cases}, \tag{3.4}$$

where Q is a threshold, and $q[t]$ is called the qualitative variation. A value of $q[t] = 0$ represents the energy being relatively stable. A value of $q[t] = 1$ means that the energy is increasing in the surrounding time window, and a value of $q[t] = -1$ that it is decreasing.

Using those transformed signals, a first stage is tasked with identifying individual shocks. Two additional thresholds T_U and T_D are defined, with $T_U > T_D$. The first shock starts when $\mathbf{s}$ goes above T_U. A shock always ends when $\mathbf{s}$ goes under T_D. However, if $\mathbf{q}$ signals that the energy decreased ($q[i] = -1$) and subsequently increased ($q[j] = 1, j > i$), the point of sign inversion is considered the end of the first shock and the beginning of a second one. Optionally, one can look for the local minimum in $\mathbf{s}$ to further refine the separation. Older implementations would use only T_U and T_D to detect shocks. However, a problem arises when trying to separate shocks happening in close succession, such as when the end of a firing cycle is immediately followed by the beginning of the next one. To separate those two high intensity events one would have to use very high thresholds, thereby causing problems where low-intensity shocks would not be detected at all. The unusual technique would thus be to treat these together as a very high intensity super-event to be identified and treated separately by the next stage, and was generally unreliable. This problem is illustrated in Figure 3.2, which clearly presents at least 6 distinct events

[1] In practical implementation which uses 8-bit values for the acceleration, the lower 4 bits of the squared value are dropped to keep the sum inside the bounds of a 32-bit integer.

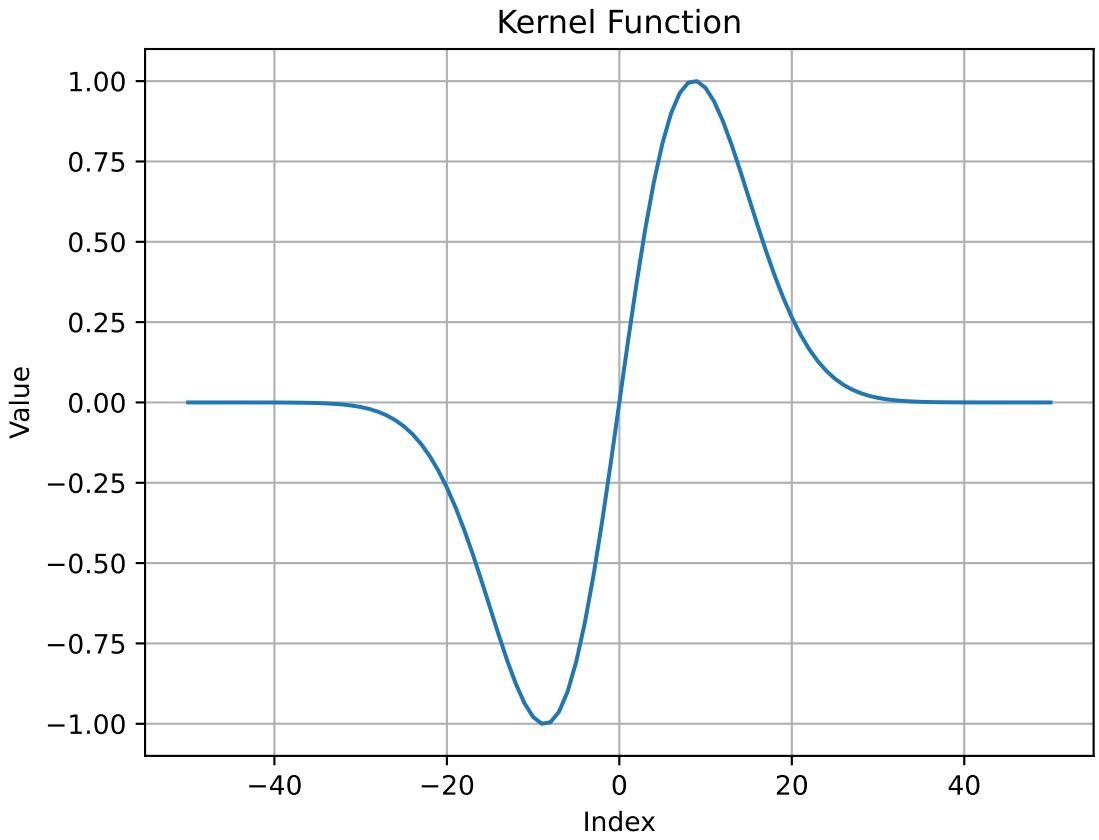


Figure 3.1: Kernel function **k** used in the variation transform.

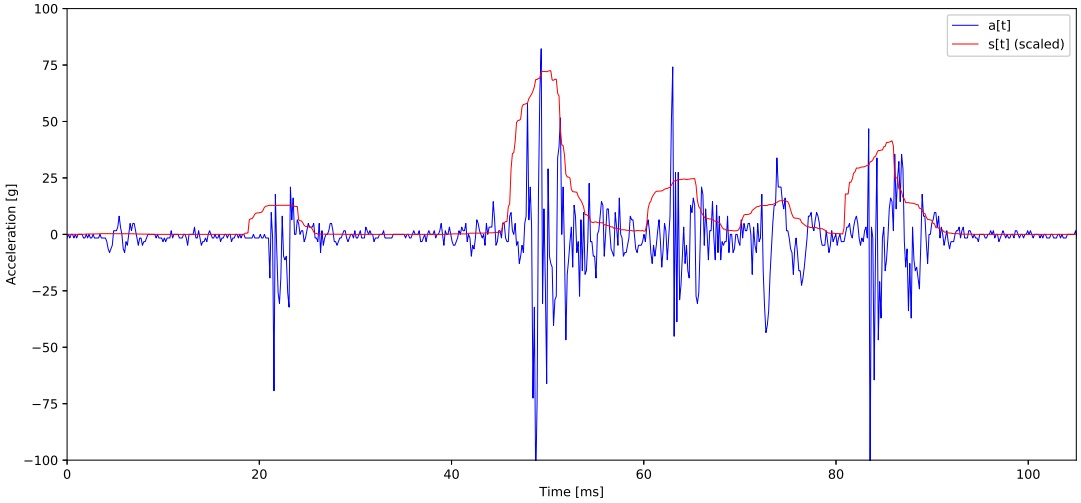
which would not be separable with fixed thresholds. Using the variation transform with low thresholds identifies many more individual shocks which can be passed to the next stage. Apart from the increase in computational complexity, the drawback is that meaningful physical events can get split up into multiple shocks without physical meaning. However, it is easier for the next stage to observe this tendency, and merge these shocks as needed afterwards.

We define a shock event as an ensemble of properties, represented by the set:

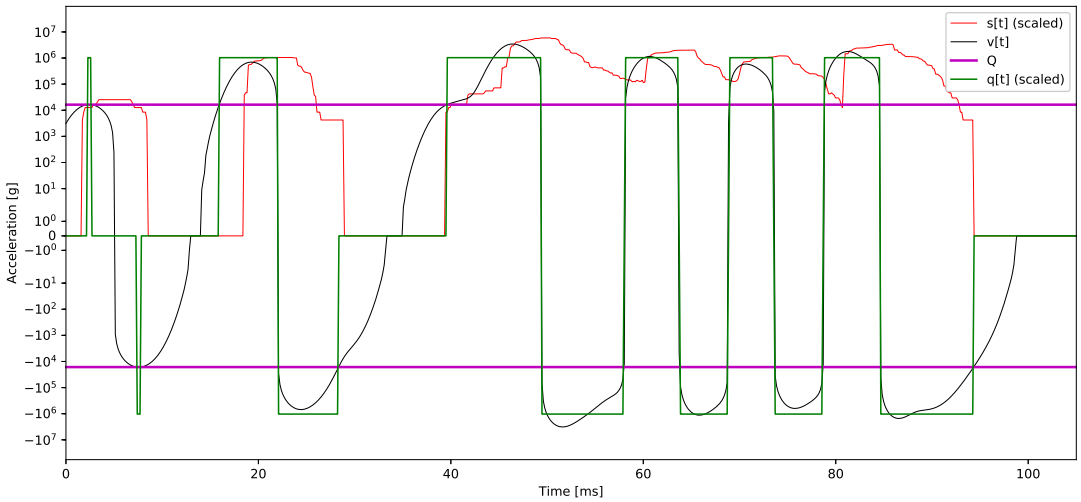
$$S = \{t_s, t_e, A\}\,,$$

where t_s and t_e are the start and end times of the shock, respectively and A is a measure of activity of the shock, defined as $A = \sum_{t=t_s}^{t_e} |a[t]|$.

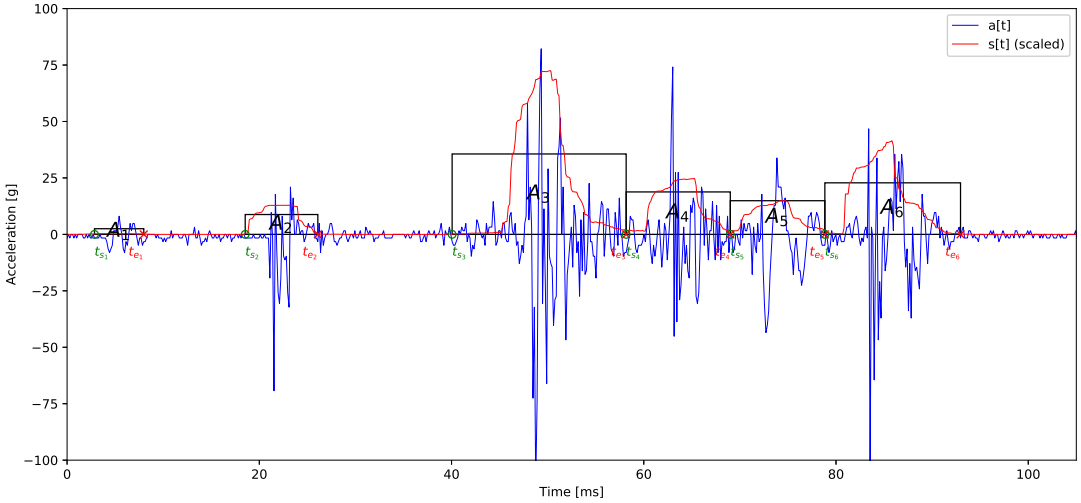
While adding a notion of direction to the shock properties is physically meaningful, due to the highly vibratory nature of the acceleration signal methods to determine the prevailing direction have shown highly unreliable. However, we will attempt to leverage a similar information in Section 3.1.2 when discussing ammunition discrimination.



(a) Step 1: Compute $s[t]$.



(b) Step 2: Compute $v[t]$ and $q[t]$, and infer shock bounds (symlog scale).



(c) Step 3: Shocks are defined by their start, stop, and activity.

Figure 3.2: Illustration of the shock detection process of classical shot counting.

The shock detection stage presented here is entirely generic, and only minimal adjustments to its parameters are required when developing an algorithm for a new weapon. Each shock S_i is then passed from this shock detection stage to the shot detection stage, which will be tasked with transforming a sequence of shocks into a sequence of shots. This is highly weapon-specific, and can be implemented in a number of different ways. Most often, a finite state machine (FSM) is used to interpret shocks as they come, and decide if they constitute the expected next step in that weapon's firing cycle. Note that for some more complex weapons, it might be necessary to either look ahead, or revert the decision that a shot has taken place as new shock information comes in.

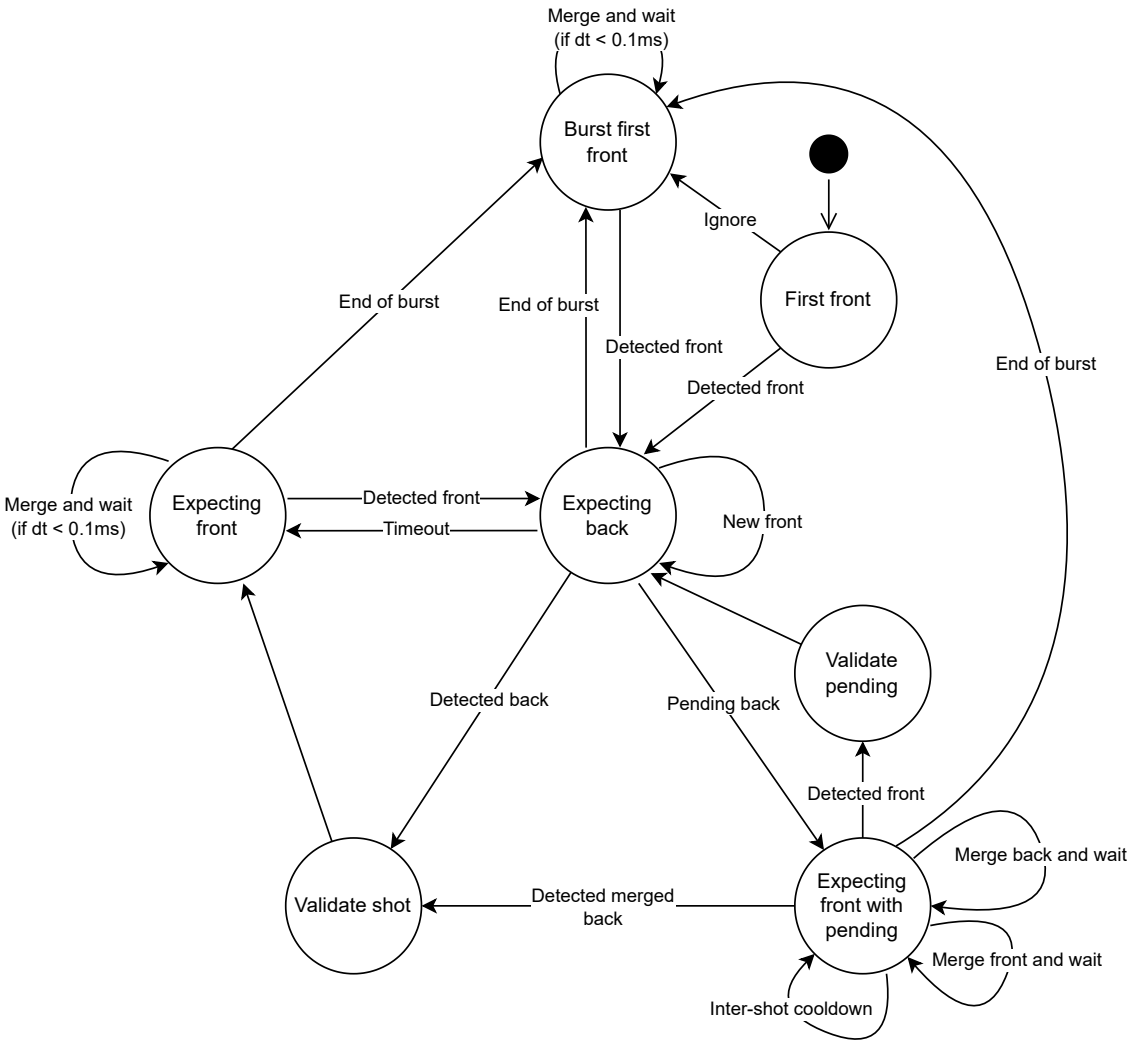


Figure 3.3: Finite state machine for the shot detection stage of a simple two-events (front/back) firing cycle.

Figure 3.3 shows an example of a relatively simple FSM for a machine gun which only looks for two meaningful events. Most FSMs in practice look for three events such as previously described. When applicable, they look for a fourth event such as a pre-firing shock (e.g. a multi-stepped chambering of the round) or an event related to bringing in the new ammunition.

Depending on the state, transitions are evaluated according to the (merged) shock duration, time since previous significant event, and activity. All can have minimum and maximum bounds. While also significant, bounds on activity should be relatively lax, as activity is much more susceptible to variations in the weapon configuration than time-based measurements. Although these will also vary, the physics of the firing cycle will eventually bound them with a minimum burst rate (under which the weapon fails to cycle) and a maximum burst rate (above which the weapon gets significantly damaged), although a significant margin must still be applied to take these abnormal cases into account. Additionally, the duration and activity bounds are not fixed, but instead relative to other confirmed shocks of the same or adjacent cycles (for example if a shot is strong/weak, the back shock is expected to be somewhat proportionally strong/weak). In the following description, "weaker" and "stronger" should be understood as relating to all of these properties and not simply activity.

First, it tries to identify the shock corresponding to the shot. A special provision is made for the first shot of a burst, which is expected to be weaker relative to the subsequent shots which benefit from the energy of the active cycle. A further special consideration is made for the common case when the acquisition after wakeup starts with the shock of the first shot. Since some of the data before the wakeup could not be acquired, this first shock is expected to appear relatively weaker. If a shock is deemed too weak to be considered a shot, it will be merged and accumulated with the following shocks if they occur within a short time frame, until the resulting shock can be considered strong enough.

Once the shot is identified, it looks for the previously described shock of the mobile parts hitting the back of the weapon. If a particularly strong shock occurs, it will be considered the actual shot and will replace the previously identified shock as the shot event. If it is reasonably strong, it is considered a "back" event and a shot is validated. Particularly weak shocks are ignored. In an intermediate case, the event is considered as a "pending" back event. Depending on the subsequent shocks, one possibility is to merge and accumulate until a full back event can be distinguished. If a new shot is instead detected, the pending back event will be validated. If neither case happens, the whole cycle is considered a non-shot event.

The decision to end a burst and move back to the state expecting a new burst can be made reliably using a simple time threshold, adjusted for each weapon. Since triggers are usually constructed with a hysteresis mechanism, there is a large gap between the minimum possible burst rate in automatic mode and the maximum speed at which an operator can physically release and press the trigger again to start a new burst.

When the firing conditions or weapon configuration induce significant changes to the shooting profile, this approach to solving the counting problem can easily take them into account by branching at specific states of the FSM. The presence or absence of some events

can be detected and interpreted depending on other characteristics of the shot, without having to resort to a large number of possible shot templates. Additionally, by deconstructing the shot into its physically meaningful steps, we are able to precisely compute metrics such as the speed of the mobile parts during the backwards movement, which is important information directly linked to backpressure strength.

Once proper shot detection is established, higher-level information such as the burst rate can be computed from the distribution in time of the individual shots. The duration of each shot can also be compared with the overall burst rate. Outliers can be indicative of either a weapon malfunction, or a counting error. A post-burst correction can then be applied, re-evaluating these outliers and their shocks. Depending on their number and their characteristics, outliers can then be split into multiple shots, merged with a neighbouring shot, or flagged as a weapon malfunction. These corrections further improve counting accuracy. From there on, the schedule of firing is computed by recording shots over a larger timeframe, typically an hour. The barrel temperature can be estimated from a mathematical model established from either simulations or empirical measurements.

3.1.2 Discrimination

The patent by Staiger et al. [9] states that "[ammunition discrimination] can be executed, for example, on the basis of the different acceleration pulses of moving parts such as the breech or the different recoil pulses on the overall system", although a method to do so is not described. In this technique, acceleration of the moving parts is measured through inductive sensors instead of an accelerometer, resulting in much lower resolution in acceleration measurements.

A patent by Joannes et al. [18] describes an approach for discrimination between live and blank rounds, in which the "direction of the first initial shock" is used. A related patent by Gering [7] goes into further details, describing a blank round as "having a phase of acceleration to the front for several milliseconds prior to the recoil." In this technique, an acceleration signal is considered as a blank round when "the recoil acceleration comprises two components of opposite directions." This difference is created by the way the ejection of the projectile mass (or lack thereof) affects the recoil of the weapon. However, due to the large vibrations created during the ejection and the difficulty of acquiring and processing an acceleration signal of sufficiently high resolution, detecting the direction of the initial shock is challenging. In particular, always-on sensors that rely on a low-power system (such as a piezoelectric element or a low-power accelerometer) to wake up from deep sleep are usually not able to capture the first milliseconds of a shooting event which, as just described, contain most of the information about the ammunition type used.

The aforementioned patents implement this by first detecting that a significant acceleration event is happening. Then, following the convention that accelerations are positive toward the front of the weapon, looking for the largest acceleration values in the positive (P_P) and negative (P_N) directions, as illustrated in Figure 3.4.

The direction of the initial shock according to this metric is then computed by looking at the sign of the difference between these peaks, optionally offset with a parameter P_0 as such: $\Delta P = P_0 + (P_P - P_N)$. Although intuitive, this method does not always yield the desired results. As an example, this approach fails on the Minimi 5.56 Live example in Figure 1.24 where the positive peak is significantly higher than the negative one, resulting in a "Blank" misclassification by this technique.

An alternative method for computing the initial shock direction which has shown strong discrimination properties is to look for the maximum recoil before the point of highest acceleration intensity (t_{max}). This can be computed with the following, where $a[t]$ is the input acceleration time series, and t_0 and t_{end} are the start and end indices of the current acceleration event as determined during the shock detection stage, with D_0, w_1, and w_2 as tunable parameters:

$$t_{max} = \underset{t \in [t_0, t_{end}]}{\arg\max} \sum_{i=-w_1/2}^{w_1/2} (a[t+i])^2, \tag{3.5}$$

$$D = D_0 + \min_{t \in [t_0, t_{max}]} \sum_{i=-w_2/2}^{w_2/2} a[t+i]. \tag{3.6}$$

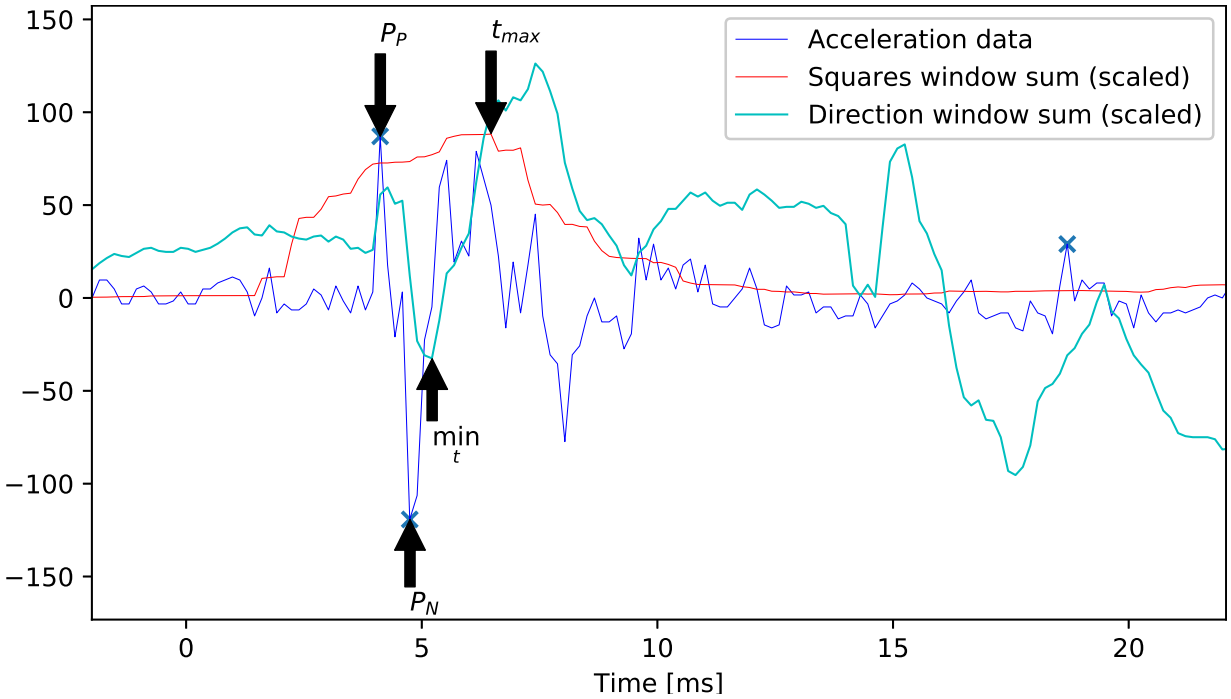


Figure 3.4: Example of classical ammunition discrimination. Metrics are computed from the raw acceleration data, and from moving windows on this signal.

Discrimination is then performed by examining the sign of D, with D_0 serving as an adjustable decision threshold.

By considering the absolute values of ΔP and D as confidence measures for each technique's discrimination capability, they can be combined into an ensemble metric, C, as follows, where α is a tunable parameter:

$$C = \begin{cases} \Delta P, & \text{if } \alpha(\Delta P)^2 > D^2, \\ D, & \text{otherwise.} \end{cases}$$

Finally, discrimination is performed by examining the sign of C. Decisions are first made individually, then the overall class is assigned based on the majority result across instances. While this approach provides very high accuracy for sufficiently long streaks of shots, its performance often declines over short, isolated bursts, typically dropping to around 75% accuracy (see Section 9.2.1). Whether this drop in accuracy is significant depends heavily on the weapon type. For example, this may be acceptable for heavy machine guns frequently subjected to sustained firing but inappropriate for assault rifles, which may primarily be fired in semi-automatic mode.

3.2 Related Work

Due to the specific nature of the problem of shot counting, as well as the customary level of trade secrecy in this industry, publicly available research on shot counting remains limited. It has been confined mostly to patent literature, with a notable surge in interest in the past three years, particularly regarding machine learning applications. Since ammunition discrimination relies on initial shot detection, studies in this area are even scarcer, and most publicly available information is available through patents.

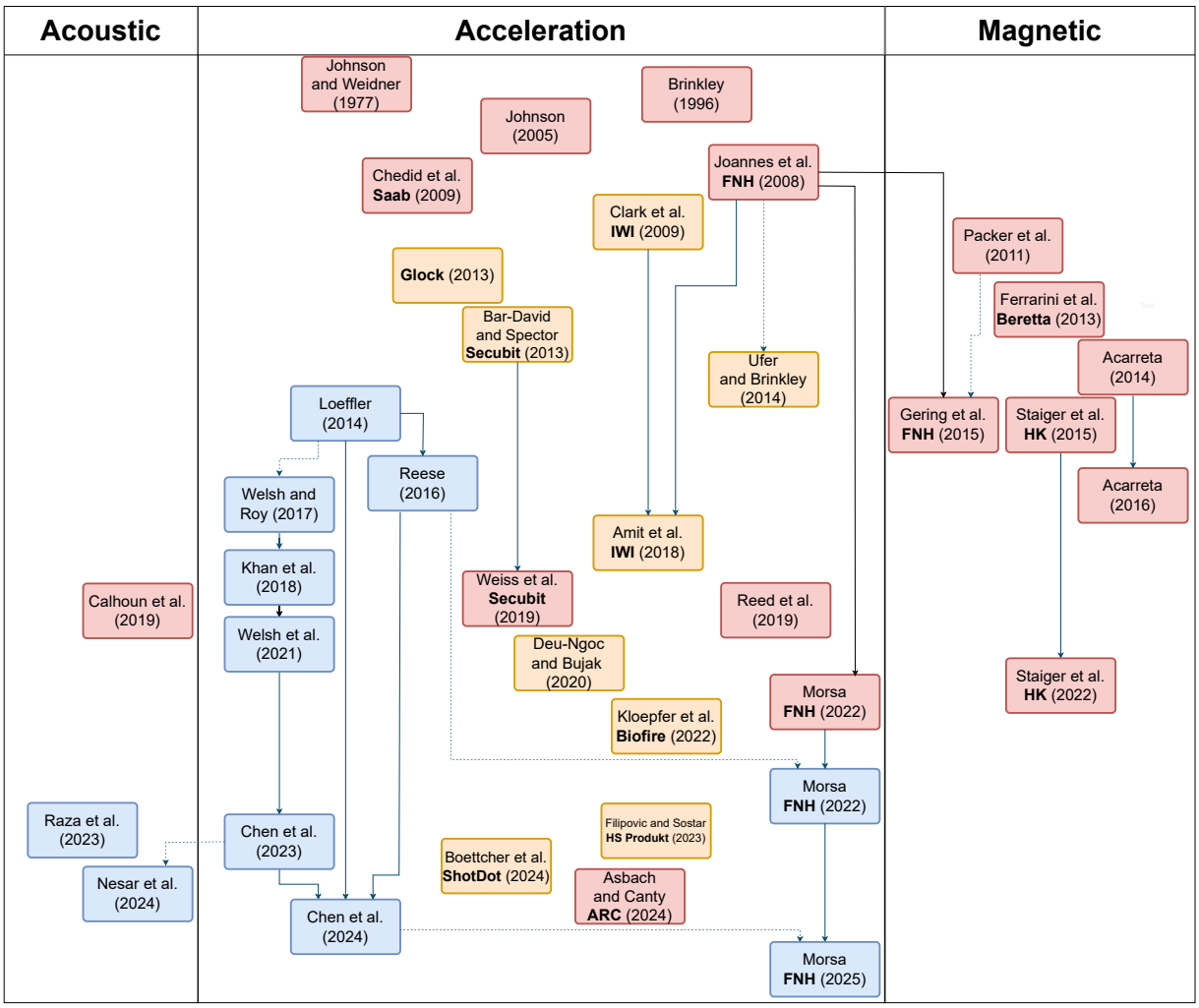


Figure 3.5: Map of the works cited in this section. Solid arrows indicate strong dependencies. Dotted arrows indicate noteworthy references. Academic publications are coloured in blue, patents in red/orange. Patents in orange can be considered subsets of the Joannes et al. [18] patent.

3.2.1 Acoustic gunfire detection

In Section 1.2, we discussed how the majority of gunshot detection devices rely on acoustic analysis, which is not suited to our use case. However, the associated literature is consequently much more developed. While we will not do an exhaustive analysis of it in this thesis, we wish to cite a few noteworthy publications:

The SoundThinking Inc. company (formerly ShotSpotter Inc.) details in a 2019 patent by Calhoun et al. [22] a commercial-grade method for gunshot detection involving deep learning in the context of a city-wide network of microphones. However, this technique is aimed at general gunfire detection and localisation, and does not enable counting of the individual shots. It also involves data processing in a central location and human confirmation.

A 2023 article by Raza et al. [23] presents a thorough analysis of the gunshot sound-related literature, comparing their results with 10 other techniques using supervised machine learning. They focus on general gunfire detection, considering audio samples of multiple seconds at a time. They further study the topic of firearm type recognition. The authors also mention the high computational complexity of their proposed model, which does not allow for real-time inference, even when using a GPU backend.

A 2024 conference paper by Nesar et al. [24] criticises the existing literature in this domain, including the Raza et al. article, for being performed on recordings acquired from YouTube or sound effect libraries. Since those typically consider relatively long recording with low sampling rates compared to the highly impulsive nature of gunshots, they assess that existing models are classifying the impulse response of the recording locations, rather than the characteristics required for generalisable firearm type recognition. Their proposed approach relies instead on synthetic generation of data from 30 samples of gunshots acquired in quasi-anechoic settings with a high sampling rate.[2] This illustrates the difficulty of acquiring high-quality, extensive datasets in this domain. A fact that is even more true for military-grade sensors which, as discussed previously, need to be mechanically fitted to different weapons, and will be a point of focus in this thesis. Interestingly, they also cite an article on accelerometer-based detection [25], showing that this domain is now sufficiently mature to influence work on acoustic gunshot detection.

3.2.2 Magnetic sensors

As discussed in Section 1.2.2, shot counters using magnetic sensors are much more difficult to integrate. However, they also greatly simplify the process of shot detection, as the sensor is only activated by a significant mechanical motion of the associated parts of the firearm. Measuring the intensity of this activation allows one to easily classify events as either low-intensity non-gunshot events (such as a manual movement of the slide) or high-intensity gunshot events with classical techniques. On the other hand, discrimination is more difficult as only the movement of parts inside the weapon can be observed, and not

[2] Unfortunately, they do not provide numbers on whether this approach generalises better in real-world applications.

the overall movement of the weapon itself. No publicly available magnetic shot counter capable of discrimination is known at the time of writing.

A 2011 patent[3] by Packer et al. [26] is the first known description of this technique. In this implementation, permanent magnets of alternating poles are mounted to a movable portion of the firearm, and a coil is mounted to an adjacent non-movable part. A technique to perform wireless energy transmission through the same receiver coil for data retrieval is also described.

A 2013 patent by Ferrarini et al. [27] (associated with Fabbrica d'Armi Pietro Beretta often known simply as "Beretta") describes an implementation of a shot-counting device that includes a fixed housing with a coil surrounding an internal magnet attached to a spring.[4] The device leverages the inertia of the magnet's mass and the firearm's recoil energy to generate an electrical signal. As such, the signal produced is more closely related to that produced by accelerometer-based techniques.

Patents by Raul Delgado Acarreta in 2014 [28] and 2016 [29], in addition to the aforementioned techniques, describe the use of a magnet attached to the follower of the magazine. As the cartridges are expended, the follower will rise inside the magazine. A fixed Hall-effect sensor can then measure the position of the magnet to estimate the remaining number of cartridges, as well as confirming a shot event through a corresponding change in the magnet's position.[5]

A 2020[6] patent by Gering et al. [7] (associated with FN Herstal) describe a similar technique to Packer et al., but with the addition of an accelerometer in order to provide ammunition discrimination (a topic discussed later).

Finally a 2023[7] patent by Staiger et al. [9] (associated with Heckler & Koch) compares the full-wave rectified signal of the magnets of alternating poles with the half-wave rectified signal. This allows individual magnets to be detected as they move along the coil. Given that the relative spacing of the magnets is known, the speed of the mobile parts can be calculated at different points during movement. From the difference between multiple discrete values of speed, one can estimate the acceleration of the mobile parts. This acceleration can then be used to perform ammunition discrimination in a similar way to Gering et al. In other words, by considering the specific configuration of magnets, the magnetic sensor is used as a limited accelerometer.

[3] This patent was acquired by FN Herstal in 2016.

[4] The patent also describes an alternative embodiment which instead uses piezoelectric elements to the same effect.

[5] An alternative embodiment using an optical sensor is also described.

[6] Anteriority of this patent can be traced back to March 2015 [30].

[7] Anteriority of this patent can be traced back to June 2015 [31]

3.2.3 Acceleration-based detection

Early works and FN Herstal patent

Early implementations can be traced back to 1975, where in a patent by Johnson and Weidner [32] a piezoelectric element or recoil-operated microswitch temporarily powers an electrochemical gauge, slowly shifting its indicator with each use. The intended use is already preventive maintenance, citing the replacement of a weapon part after a certain number of rounds have been fired. This simple method is easily subject to false detections, only provides an imprecise indication of firearm usage and must be read manually.

Another early example of acceleration-based shot counter is the 1996 patent by Kenneth L. Brinkley [33] (associated with Accu-Counter Technologies Inc.) which uses a ball and spring system to form an inertial switch, incrementing a counter with each actuation. Compared to later implementations, this type of device provides limited information on the shot and is prone to false positives. Being mechanical, it is also bulkier and prone to wear and tear.

A 2005 patent by Johnson and Kulesza [34] (associated with Advanced Design Consulting USA) provides the first description of modern accelerometer-based techniques. While the patent mentions both acoustic and acceleration inputs, its description focuses primarily on accelerometers. It is mainly intended to improve upon existing devices by recording the temperature at which rounds are fired, since rounds fired at higher temperatures have a greater impact on barrel health. This is implemented by attaching the counter to the barrel and using a temperature sensor. Due to the very high temperatures reached by machine gun barrels (easily in excess of 400°C), such implementations are a materials engineering challenge. Modern implementations usually instead precisely record each shot and the interval in-between, then approximate the barrel temperature from a mathematical model. While this early patent illustrates a typical acceleration signal for a rifle (see Figure 3.6) and identifies the complex nature of the signal; it fails to leverage this complexity to provide more accurate shot counting. Instead, it implements a pass-band filter through mechanical means and uses a simple threshold on the resulting signal to detect shots. A hold-off time is then applied to ignore the rest of the cycle. Such an implementation is subject to false positives, as other highly-impulsive non-shot signals might also trigger such a threshold. In addition, using a timer to fully ignore all other acceleration events of the firing cycle is also problematic in the context of sustained fire from machine guns. The burst rate of this kind of firearm can easily vary by as much as twofold. The associated timer would need to be precisely calibrated to the current burst rate in order to fully ignore all mechanical events right up to the beginning of the next shot. If the timer is too short or too long, this technique could easily get "desynchronised", particularly if it detects other events of the firing cycle as a shot, which will then start a period where the next actual shot will be ignored, leading to unpredictable under- or over-counting until the end of the burst.

An important 2008 patent by Joannes et al. [18][8] owned by FN Herstal. It covers the use

[8]While the cited US patent was issued in 2010, its anteriority can be traced to the associated 2008 Belgian patent [35]

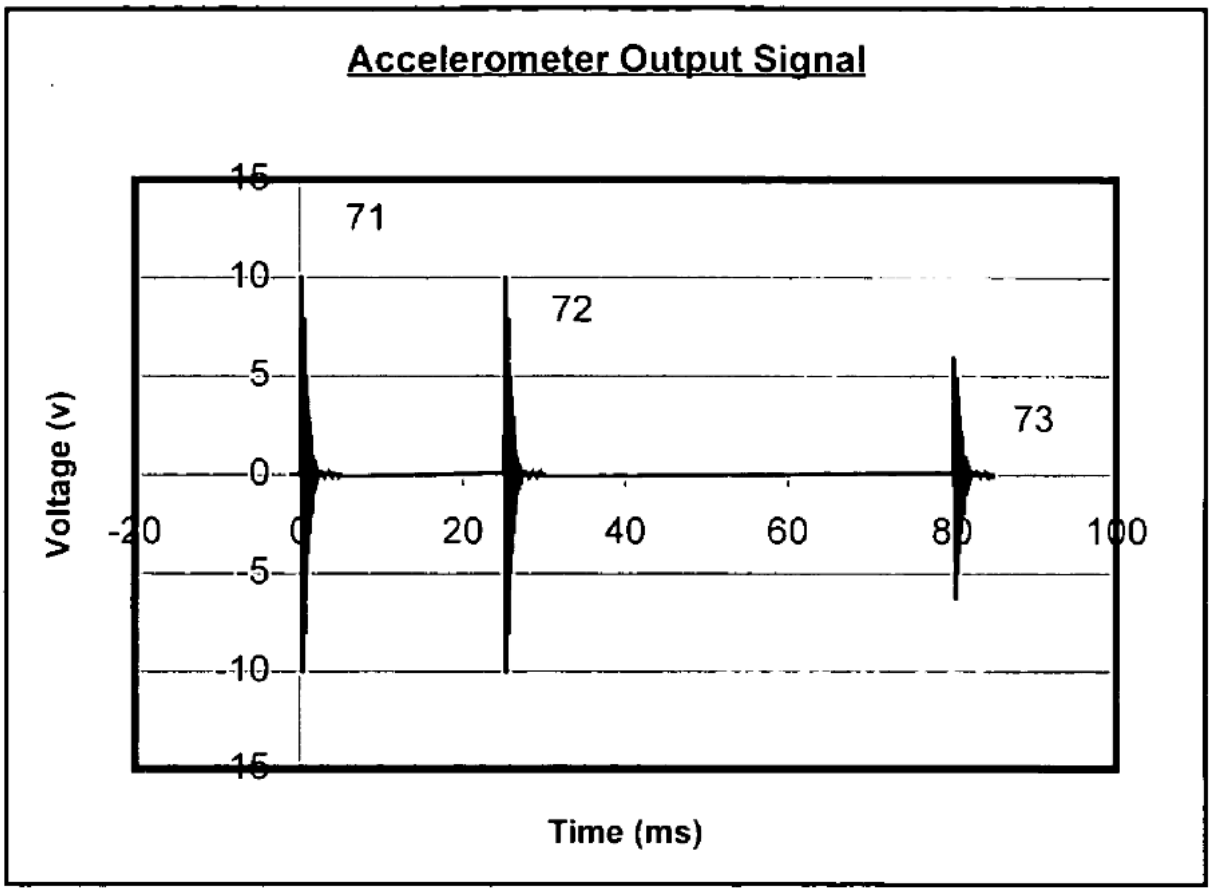


Figure 3.6: Acceleration signal of an unspecified firearm by Johnson and Kulesza [34], a typical 3-stepped cycle can be observed.

an accelerometer positioned in the firing axis of the weapon, comparing characteristics of the acceleration curve with typical characteristics of shots for that firearm, for the purpose of shot counting and ammunition discrimination. More specifically, an acceleration event is decomposed as a sequence of discrete shocks separated in time with values of amplitude, length, and direction. This allows for much better shot counting by taking into account the full complexity of the firing cycle, both eliminating false positives and dynamically observing each firing cycle to correctly separate individual shots in a burst without requiring calibration to a specific burst rate. However, the patent does not provide the shot counting algorithms used to identify the shocks and then process their sequence. The details of the latest implementation of this technique, which have not been published publicly before this work, are given in Section 3.1.1. We will see that subsequent descriptions of classical shot counting algorithms are mostly more limited versions of this implementation, justifying its choice as a baseline for those derived through machine learning. The strength of this patent has likely hindered further research into the development of accelerometer-based shot counters. The current surge in interest could be associated with its expiry in 2026.

A 2009 patent by Clark et al. [36] (associated with Israel Weapon Industries (IWI)) describes a similar early implementation relying on fixed minimum and maximum thresholds in two acceleration axes. After detecting an event in a given axis, significant acceleration is expected in another axis within a specific time window. A comparison of the signal with a stored "true shot" signature can also be performed. The process can be supplemented by acoustic or thermal sensors to further confirm gunshot detection.

A 2013 patent by Gaston Glock [37] (associated with Glock Ges.m.b.H.) describes an essentially identical device, albeit aimed at shot counting for handguns.

Another 2013 patent by Bar-David and Spector [38] (associated with Secubit Ltd. now known as Senseer) describes an invention similar to the Joannes et al. [18] work. To avoid the problem of losing the initial part of the shot event when waking up from sleep mode, the device focuses on a subsequent shock expected in a predetermined time window after wakeup. This is a simple subset of the work described by Joannes et al., in that only a recoil shock is considered in its energy instead of the full cycle signature. This is supported by the court decision in FN Herstal's favour in December 2020 [39], confirming the validity of the Joannes et al. patent (under EP1881292) and the counterfeiting of it by Secubit Ltd.

Ufer and Brinkley patent

A 2014 patent by Ufer and Brinkley [40] offers the first public full description of a shot counting algorithm. This technique relies on identifying four features in the part of the acceleration curve that corresponds to the "shot" event:

1. Windowed sum of absolute acceleration values with a high-pass filter: a measure of energy.

2. Windowed sum of signed acceleration values with a high-pass filter: a measure of direction.

3. Slew rate: They argue that shot events must have a high slew rate, as in a rapid speed of change in the acceleration signal. Additionally, they argue that while non-shot events can also have a high slew rate, these would take the form of a growing sinusoid instead, which can be used as a criterion for shot/non-shot detection. While we agree with the first point (and will build upon that idea later), the second is not necessarily true especially in the case of open-bolt weapons. This type of weapon, typically rapid-firing machine guns, keeps the mobile parts at the back of the firearm to allow for improved airflow and cooling of the mechanism. When firing, the mobile parts are released and move forward, chambering the next round and locking it in the chamber. These events produce significant acceleration prior to the shot ignition, as can be seen in Figure 3.7.

4. Zero crossings: They argue that counting the number of time the acceleration signal crosses zero during the shot event can be used as a measure of jerk, which would be low for shots and high for non-shot mechanical events, deemed more vibratory in nature. While this might be true in some specific situations, we do not believe that it is generally applicable. A counter-example can be seen in Figure 3.8. As the signal is known to be undersampled, and polarity inversions can be observed as single data points even at our sampling rate of 6400 Hz, this technique is also highly dependent on the bandwidth of the accelerometer, which can be unreliable, as discussed in Section 4.2.2.

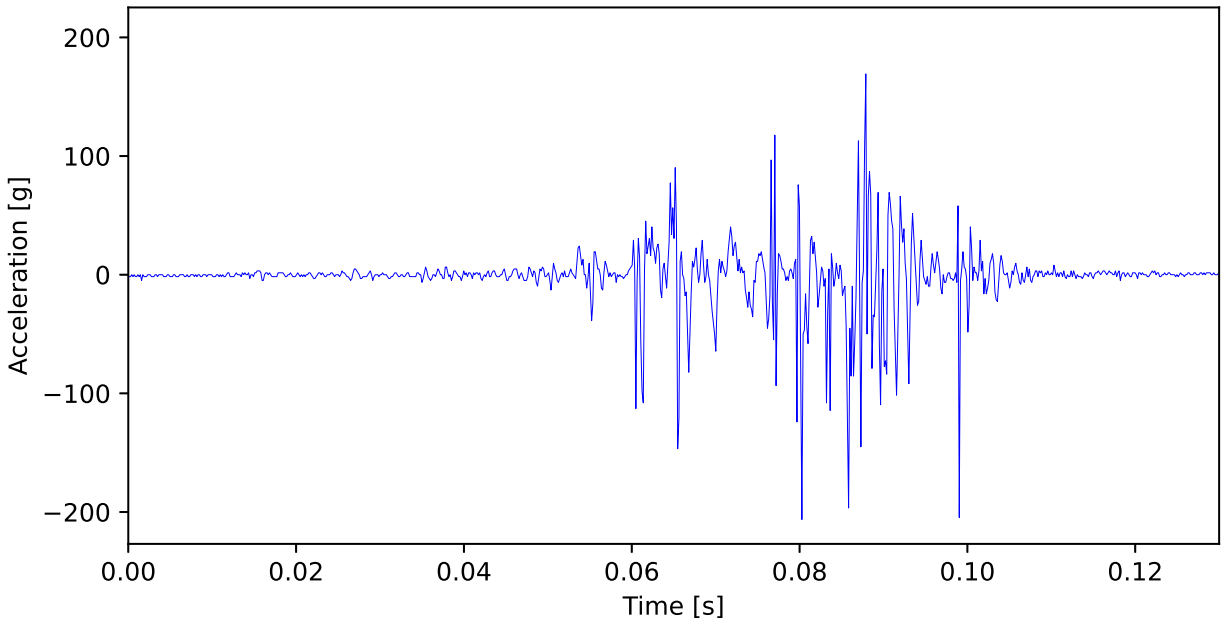


Figure 3.7: Example of a shot with the open-bolt MAG, showing a steadily increasing sinusoidal acceleration before the actual shot takes place.

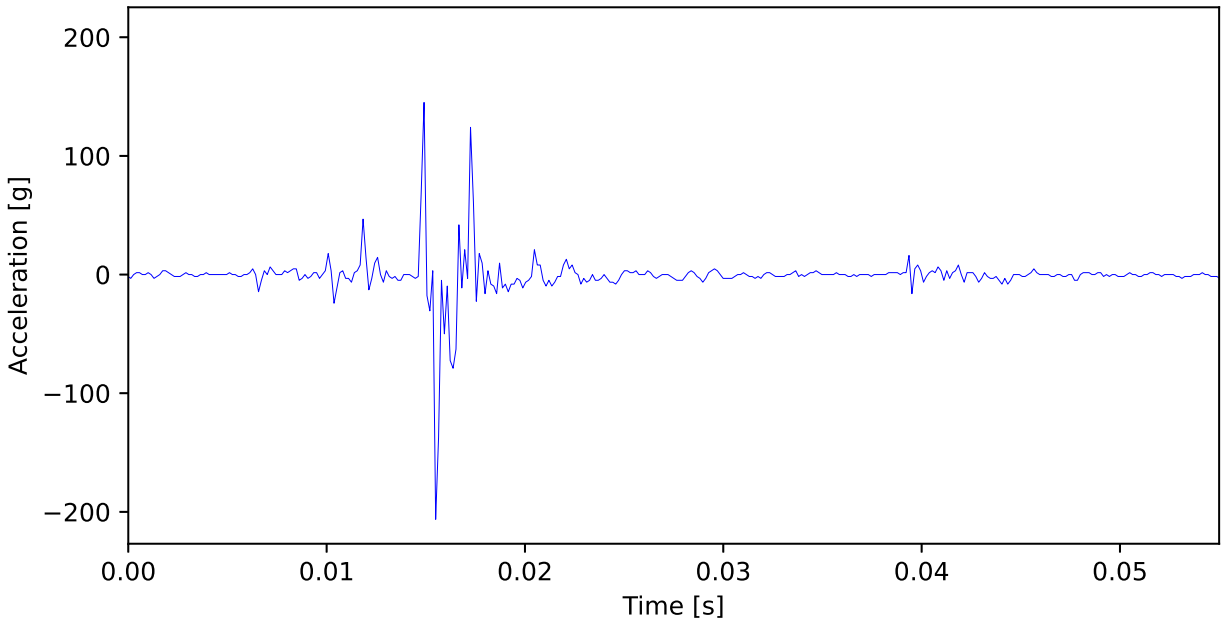


Figure 3.8: Example of a dry-fire with the MAG, showing an acceleration event with high intensity but relatively few zero crossings.

These measurements are taken only during the initial shock event of the shot, and a timer is then run to ignore all events occurring after a shot detection to prevent false positives. This can be seen as an implementation of shot detection which is similar to the FN Herstal technique described previously, but which only focuses on the shot event. In the context of low-power devices with acceleration-based wakeup systems, this is thus heavily reliant upon an early wakeup in order to acquire the full acceleration curve of the initial shot event. While the difficulty of this task has been alleviated thanks to new MEMS accelerometers capable of self-starting, this remains a challenge and also incurs a power consumption trade-off.

The use of a hold-off timer intended to fully cover the firing cycle is also problematic for identical reasons to the ones mentioned for the Johnson and Kulesza patent [34].

This patent however introduces the idea of measuring the rotation around the firing axis. As the rifling inside the barrel imparts a rotational movement to the bullet, they argue that by reaction the firearm should also observe a rotation in the opposite direction. However, no data is provided about the efficacy of this technique, and we are not aware of any successful practical implementation. While they do not make that conclusion, this would be a particularly interesting measurement for the task of live/blank ammunition discrimination, as this would be a physical quantity directly linked to the presence of a projectile.

In the patent, Ufer and Brinkley warn that their technique which relies heavily on pre-determined thresholds is highly sensitive to external factors such as temperature or the way the weapon is held or mounted. They suggest measuring these, with a temperature sensor for the former and through rotation for the latter, to apply different thresholds according to

the detected conditions. This would then be complemented with an individual calibration for each firearm, adjusting the thresholds for the specific configuration of usage. While this would indeed be effective in the short term, this calibration is highly impractical in the context of long-term usage, particularly at the level of weapon fleets. This is particularly true for military usage, as a change in the user's position (such as going prone or deploying a bipod), the mounting of an accessory, or a change of ammunition can all cause significant and rapid shifts in the acceleration signature, often occurring so quickly that individual calibration becomes ineffective.

Finally, they also introduce early "battlefield of things" concepts, proposing that their sensor be connected with other sensors and integrated into a wider police or army network for real-time battlefield management and resupply.

Academic publications by Loeffler and Reese

Academic research on the topic begins in 2014 with the work of Charles E. Loeffler [41] for the Department of Criminology of the University of Pennsylvania. The objective of this paper is to prove the feasibility of using wrist-worn devices for surveillance of individuals at high risk of gun crime, as an alternative to location-based monitoring devices. He constructed a publicly available dataset of around 1000 shots from 6 different handguns, as well as 68 hours of recording of construction work to serve as high-impact non-shooting activity. He hypothesises that, for the considered use case, firearm usage is always preceded by a period of very-low acceleration linked to the aiming task, which is not the case for other high impact activities which are preceded by significant pre-event accelerations. A threshold-based candidate selection method over the raw acceleration is applied. Candidates are then processed through a probabilistic classification method with hand-crafted features.

While successful, this work presents several limitations. While the paper contains the information that test dataset comprises 358 gunshots, no such information is provided for the learning dataset. The dataset does provide some metadata about the start and stop of each shooting sequence (along with the type of weapon, subject and handgrip type), however several mistakes are present in this labelling such as a start time after a stop time, or time sequences corresponding to obviously non-shot data. This illustrates the difficulty of proper firearm dataset construction, particularly when very low error rates are required, which we will aim to solve in this thesis.

It is also more restricted in scope, as the objective of the paper is related to activity detection in a civilian context rather than accurate gunshot counting. As such, it only considers semi-automatic fire from handguns at the level of the user's wrist. As illustrated in Figure 3.9, readings along the shooting axis (X-axis) are relatively low with peaks around only 10 g, whereas our use case is known to peak above 3000 g in some instances as discussed in Section 1.3.2. The readings on the Y and Z axes show very different acceleration profiles, with stronger readings in the Z-axis corresponding to a measured muzzle-lift. In contrast, we saw that the acceleration measured from embedded sensors show similar profiles along all axes, with the X-axis being the strongest. Thus, while related, the problem of shot

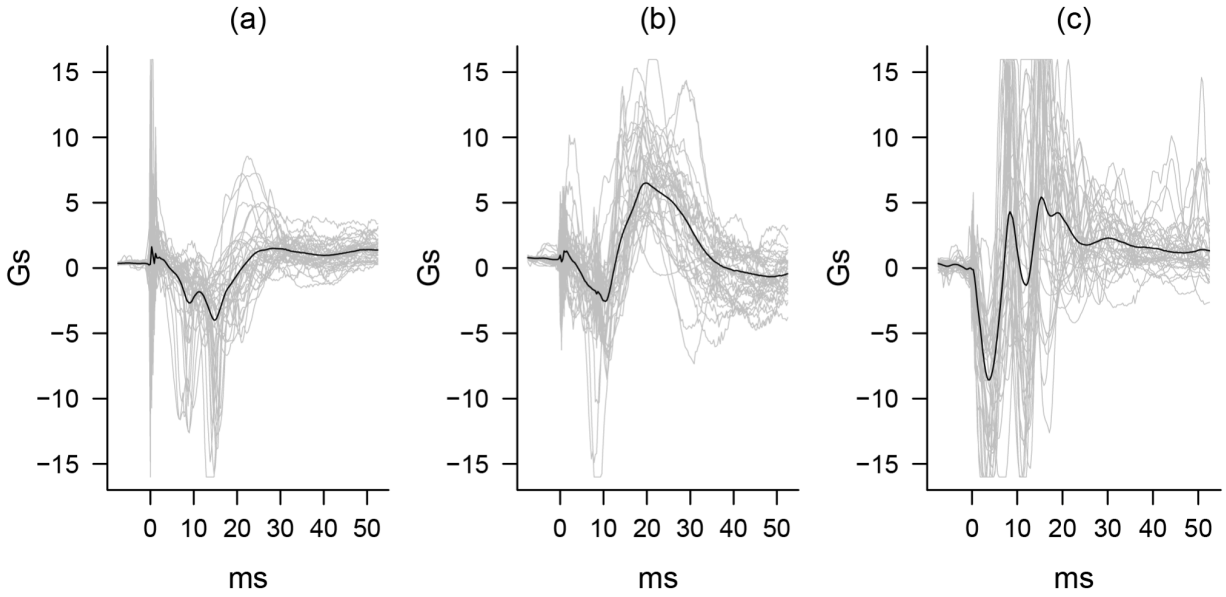


Figure 3.9: "Individual and averaged gunshot acceleration readings along the (a) X-axis, (b) Y-axis, and (c) Z-axis. Individual gunshot acceleration readings (in grey) are a 10 percent sample of the 359 gunshot acceleration readings (in black) in the sample average." [41]. Note even in this very controlled use-case the very high variance between individual samples.

detection at the level of the user's body differs fundamentally in nature and the optimal solution for it will differ from that used for embedded shot counting.

The method proposed by Loeffler also relies on features computed for the acceleration signal occurring before the shot takes place, which is not always possible in the case of embedded devices waking up from sleep mode. In addition, similarly to the limitations of the work by Ufer and Brinkley, the method developed by Loeffler does not consider firearms capable of automatic firing where the complex firing cycle will present multiple acceleration spikes per shot. While this is acceptable for activity detection, this method will likely miscount the number of shots in a burst.

In this work, Loeffler already identified parallels with the domain of fall detection in human activity. Indeed, falls are also high-intensity, impulsive events with internal complexity. Later work by Putra et al. [42] in 2017 studies accelerometer-based fall detection by decomposing the impact event into sub-events in the time domain, thereby showing similarities with our problem, and applying machine learning techniques. A 2019 paper by Santos et al. [43] applies deep-learning techniques to this same problem, and proposes a convolutional neural network with good results when using data augmentation techniques. These approaches, however, rely on strongly labelled data.

A master thesis from 2016[9] by Kiel A. Reese [44] for the Naval Postgraduate School

[9]It is predated by a 2016 conference paper by Yelkovan et al., however this publication has since been retracted.

attempts to bring the previous work of Loeffler to practical military application, using a sensor affixed to the weapon itself. In addition, an AR-15 rifle is studied as opposed to handguns.

However, the sensor used is an even more limited low-scale (±8 g), low-frequency (800 Hz) accelerometer. In Section 1.3.2 we provided high-scale values for the same platform measuring non-saturated accelerations up to 2000 g. Based on an estimation of the travel time of the projectile inside the barrel, Reese also computes a necessary minimal sampling rate of 2512 Hz for proper acquisition of the signal. This would however still be undersampled as the firing cycle comprises significantly shorter events. Due to these limitations on the input data, the signal processing technique relies on simpler thresholds. It suffers from the same limitations as Loeffler's by the measurement of acceleration before the initial shot takes place. While it similarly does not consider automatic firing, Reese extends the technique to consider semi-automatic shots fired in rapid succession,[10] although limiting it to a maximum of 3 consecutive shots.

Interestingly, Reese implements a measurement of "differential power" obtained by averaging the squares of the differences between successive data points. This is a more limited version of the variation transform (Equation 3.2), although derived independently as these details were not publicly available prior to the publication of this thesis. Reese also makes an attempt at real-time embedded processing on a Raspberry Pi Model B+, but was unable to meet the real-time constraints with the available computing power. Part of the work also integrates "battlefield of things" aspects, which fall outside the scope of this thesis.

Academic publications by Welsh et al. and introduction of machine learning

A 2017 paper by Welsh and Roy [45] uses the combined data from 10 different sensors of a user's smartphone to perform activity detection using a naive Bayes classifier, distinguishing between shooting and relatively low-intensity activities such as policing, running, and standing. This proof of concept uses a fully labelled dataset acquired in tightly controlled conditions. Only general activity can be detected, and the authors note that the technique is unable to distinguish individual gunshots. The intended use case is crime detection in the context of a smart city. This work marks the first publicly documented use of machine learning in the wider domain. In collaboration with Khan, they published a 2018 paper [46] for another proof-of-concept work in which they use wrist-worn tri-axial accelerometers to recognize the type of firearm being used. More specifically, they first classify between auto-loaded and manually-loaded firearms using a SVM, then between "handgun", "rifle", and "shotgun" using a decision tree, then further refine the category into an exact firearm model using another decision tree. Features are manually chosen from both the time and frequency domain, with an emphasis on the former.

[10] In this scenario, shots are fired at an approximately 2 seconds interval, whereas in an automatic firearm the next shot happens right at the end of the approximately 50–100 ms firing cycle. This blends the acceleration signal of the shots, and invalidates the "aiming task" criteria of Loeffler and Reese which requires a period of low acceleration before the shot takes place.

As already discussed with the work of Loeffler [41], data collected at the level of the wrist is fundamentally different from the data acquired with embedded sensors. Moreover, the dataset considered in this work is curated to only contain gunshot data, explicitly excluding even shot-related activities such as loading or securing the firearm or removing jams. Candidate windows are selected from a window of squared acceleration values as a measure of energy. Since the dataset is pre-filtered to only contain shots, this aims solely at selecting candidates out of the background noise between shots. The technique described is thus focused on the specific problem of firearm recognition, and is not applicable to shot counting. By using a low-pass filtered version of the signal with a cutoff frequency of 150 Hz, they observe the overall shape of the recoil and how it varies between weapons. This differs from ammunition discrimination, which focuses on subtle variations in high-frequency, high-g ranges.

However Welsh et al. follow up on their work in 2021 with the first publicly available work on ammunition discrimination [47]. They augment their 2017 technique [45] with a distance feature computed through Dynamic Time Warping (DTW). The focus of this work shifts from a civilian application to a Battlefield of Things one, and they use an accelerometer embedded in the grip of the weapon bringing it closer to our use case. It is noteworthy that their work seems unaware that this has long been the practice in military applications as described in the aforementioned patents, exposing the gap between the academic and industrial world that we aim to narrow in this thesis. They also identify that in a BoT context a wide number of firearms must be supported, each of which requiring specifically crafted detection algorithms which classically require expert knowledge intervention. Their use of DTW allows a generic comparison of samples with a template without the need of feature selection by a domain expert, which is one of the main motivations for our work (see Section 1.4). In this work, they perform in a single step both the relatively simple task of firearm recognition between a 5.56 mm AR-15 and a 7.62 mm AK-47 with perfect accuracy, and the much harder task of discriminating between 3 types of live ammunition for each weapon in which they achieve moderate success. By using a fully labelled dataset containing precise ground-truth timestamps for each shot, they are able to perform this study fully separated from the counting problem. However, their dataset is relatively small comprising only 100 single shots and 10 burst shots per weapon. While we will see in Section 8.2 that the discrimination problem can be solved with a smaller dataset than the counting problem, this is still too small to fully capture the complete spectrum of behaviours of the weapon in practical use. While Welsh et al. might be able to perform discrimination between live ammunition in strictly identical conditions, this might not be true when the weapon configuration is allowed to change. A focus of this thesis will be the comparison between live and training ammunition (or, to a lesser degree, the use of a suppressor), which produce a fundamental change in the physical response of the weapon. Discriminating between ammunition with different powder load or bullet weight is a much harder problem as changes in weapon weight, reloading nozzle size and shooter stance can have an effect more significant than the difference in ammunition type. As discussed for the Ufer et al. patent [40], which also used a comparison of the acceleration curve with a template, this type of technique will be particularly susceptible

to changes in the weapon configuration which might fundamentally alter the shape of the curve, thus further exacerbating the difficulty of generalising to a wide range of long-term usage conditions. Additionally, the technique is very computationally intensive, making it unsuitable for real-time usage in ultra-low-power devices.

Extension of the Welsh et al. work by Chen et al.

A 2023 article by Chen et al. [25] builds upon the previously mentioned works of Welsh et al. as well as Loeffler's [41]. In it, they argue that the previous works which rely on manually selected features may be overlooking better ones, and the difficulty of applying these earlier techniques in real-time wearable applications. They recommend the application of a deep learning method to address these issues. This is also the approach we suggested in our earlier 2023 paper [20] and its associated patent [19], as will be discussed later. While they aim to perform shot counting, they apply it to the wrist-device dataset of the 2018 Khan et al. paper [46] which went to significant lengths to exclude any non-shot data from it, and suffers from being filtered to a relatively low 150 Hz (which is problematic as already discussed in Reeses's work [44]). What Chen et al. identify as examples of non-shot events is restricted to cases of manual reloading in non-automatic weapons. Additionally, the dataset comprises only 117 shots without any burst firing. While they mention that weapon weight, propellant strength and shooter posture would all significantly affect the signal, this variation is also not present in the dataset of Khan et al. The added value of this article is their proposed neural network structure. They suggest a complex convolutional neural network with depthwise separable convolutions and an attention mechanism. They show increased prediction performance on the firearm recognition task when compared against support vector machines (SVM) and decision trees (DT), such as those employed by Welsh et al., although they do not implement the DTW feature, suggesting that the neural network leveraged the information better than hand-crafted features. Additionally, they compare their technique against a number of other machine learning techniques including recurrent neural networks (RNN), long short-term memory recurrent networks (LSTM), gated recurrent unit (GRU) neural networks and also show increased prediction performance, which they claim is particularly well suited to the spatial 3D nature of the signal from the wrist-worn device. As discussed previously, unfortunately this might not translate to sensors embedded in the firearm itself. Finally, they employ a knowledge distillation technique to reduce the size of the neural network without significantly losing prediction accuracy. Doing so, they provide a model that can be run on a smartphone or GPU-equipped smart holster. While this is an improvement over the previous techniques, this still does not allow the technique to run in real-time on the embedded device, thus requiring constant wireless connectivity and associated power consumption.

In a 2024 follow-up article [48], Chen et al. extend their method to proper shot detection by incorporating Reese's [44] candidate selection method, supplemented by an unspecified frequency-domain filtering technique for improved non-shot rejection. It is evaluated on Loeffler's [41] dataset, thus unfortunately inheriting both of their previously discussed caveats. Since no total shot information is available for Loeffler's learning dataset, they

split the initial test dataset, although erroneously counting 359 shots instead of the actual 358. They also evaluate this technique for shot detection on the Khan et al. dataset [46], and interestingly report a lower performance although this dataset only contains shot data. Although their stated goal is shot detection with a low-power gun-embedded device the method still only targets wrist-worn tri-axial datasets, uses pre-shot information, and is relatively computationally intensive, all of which they acknowledge as current limitations. They strongly underline the difficulty of accessing or cost of building appropriate gunshot datasets, one of the main motivations behind this thesis. They also study the large differences in signal between the two datasets even for similar weapons, a fact we would expect from our previous discussion about the importance of weapon configuration and shooting conditions. To alleviate these issues, their main contribution is the study of transfer learning between shot datasets. While they initially observe negative transfer due to the large differences, they then apply automated machine learning (AutoML) techniques to explore an optimal set of layers and parameters to add to the initial network which are trained to fine-tune the network on the new dataset. Their results with this technique are superior to those obtained by just training the larger network from scratch on the new dataset, showing that some of the original training was retained positively. However, no comparison is made with training from scratch on both datasets, which would be a simpler and faster process. A comparison with the use of the AutoML technique without prior training would also be interesting. Finally, since this technique further increases the complexity of their network, it would be worth exploring how their aforementioned knowledge distillation technique could be integrated with it.

Boettcher et al. patent

A 2024 patent by Boettcher et al. [49] (associated with ShotDot itself associated with KordTech Pty) provides a full description of their shot counting technique focusing on a family of Glock handguns in the context of law enforcement post-incident reviews. Similar to the previously described techniques, their implementation comprises an ensemble of tests performed on the acceleration data in multiple time windows of the candidate region.

More specifically, they first transform the signal to use the difference between successive acceleration data points, as already performed by Reese [44]. They justify this transformation by the need to prevent the effect of a zero-g offset in the accelerometer values, without resorting to a calibration of the sensor. They then perform moving averages of different sizes before testing the signal. These can be considered implementations of respectively a high-pass and low-pass filter, which taken together form a pass-band filter as suggested in the original Joannes et al. patent [18].

They then perform the following tests:

1. Between index 1 and 149, the sum of values must be within minimum and maximum thresholds.

2. Between index 1 and 194, the peak of a moving average of size 21 must be higher than a threshold.

3. Between index 172 and 326, the difference between the total sum of values and a moving average of size 35 must be higher than a threshold.

4. Between index 175 and 410, the peak of a moving average of size 21 must be higher than a threshold.

5. Between index 165 and 410, the peak of a moving average of size 21 divided by the mean value of that window over the interval must be within minimum and maximum thresholds.

6. Between index 1 and 299, the Pearson correlation coefficient of the signal with a precomputed representative one must be higher than a threshold.

Although very specific features are thus manually selected and computed from the signal, they do correspond to the identification of different shocks or groups of shocks resulting from meaningful acceleration events of the firing cycle. Compared to the technique described in Section 3.1.1, the lack of an intermediate decomposition of the signal in those meaningful events makes the technique more difficult to generalise to different platforms.

The time windows considered are also very rigid. This requires them to trigger the accelerometer from an external, more precise clock source to alleviate issues related to imprecise sampling frequencies as discussed in Section 4.2.2.[11] While this implementation might work for their considered use case, this would likely not generalise to automatic military firearms whose fire rate can vary by as much as twofold according to the firing conditions and ammunition. Additionally, the use of an external clock source will significantly increase the power consumption even in standby mode, making it only usable in rechargeable devices. Using this clock source only as needed is not a possibility as they typically require a significant startup time, which would lead to missing an important part of the first shot on wakeup. As already identified by Amit et al. [50], this reliance on only thresholds in the magnitude of the acceleration is highly unreliable as the intensity of the acceleration signal can vary according to factors such as the weapon mass, mounting system or shooter stance. The success of Boettcher et al. stems from their focus on handguns which are less susceptible to change in configurations compared to automatic weapons, although they still identify a need to account for different shooter grip strengths. They also likely only considered nominal cases of usage. However, in practical applications, use of lower-quality ammunition or lack of proper firearm maintenance will result in a diminution or intensification of the acceleration signal which might then fall outside the predetermined thresholds. As advised in our suggested implementation of a classical counting algorithm, the use of both the event intensities and a dynamically computed measurement of the time interval between events should both be taken into account, with a focus on the latter for maximum reliability.

While already present in other prior methods, the use of a whole-signal statistical comparison with a representative shot signal is the only technique which is not present in our recommended implementation. This also stems from the fact that, as we saw in

[11] The accelerometer used in the work of Boettcher et al. is identical to the one used in this thesis.

Section 1.3.2, in less controlled settings the acceleration signal of a firearm might present drastically different acceleration profiles, especially when using different configurations or mounting systems. This would at least require comparison with a significant number of typical signals, which tends to result in a weak test criterion.

Additionally, their implementation seems to allow little to no buffering of the signal, which results in a self-imposed restriction on the post-shot processing time, negatively impacting their choice of features.

Weiss et al. patent

A 2023 patent by Weiss et al. [51] (associated with Secubit Ltd. now known as Senseer) details concurrent work aimed at using deep-learning techniques for shot counting from at least accelerometer data, optionally augmented by other sensors. However, it does not provide a concrete description of its implementation. They suggest an encoder-decoder structure, with the encoder part being embedded in the firearm and used to reduce the input dimensionality. The encoder is suggested to be a Residual Attention Convolutional Neural Network (RA-CNN), a Recurrent Neural Network (RNN), a Transformer Neural Network (TNN) or Bidirectional Encoder Representations from Transformers (BERT) with a receptive field "larger than the longest prominent event."

The latent representation can then be exported to be exploited by the decoder network, possibly on a reader smartphone, to detect different mechanical events in the firearm, count and discriminate shots, and detect changes in user status. No concrete proposition is made with regard to the decoder implementation. They then suggest using statistical regression to infer other operational parameters such as projectile speed, barrel temperature, and weapon condition.

On top of this, a flow controller uses information from both networks to perform out-of-distribution detection and early stopping. It is unclear how the encoder input vectors are chosen and kept to a reasonable number for embedded usage, especially when the decoder is not part of the embedded device and thus cannot perform early stopping. The suggested implementations for the flow controller are several types of neural networks, including Dilated Convolutional Neural Networks (D-CNN), RNNs, Gated Recurrent Unit (GRU) or Long Short-Term Memory (LSTM) networks.

The intended workflow could be similar to the practical proposition of Chen et al. (see Section 3.2.3) who suggest processing on a smart holster. In their work, they also compare the performance of RNN, LSTM and GRU networks with their own implementation, which could guide future choices for this type of network structure.

Finally, Weiss et al. propose continuous unsupervised optimization of the network parameters in order to ensure that the predictions of the network respect a pre-established distribution. However, as described in Section 1.3.1 the signal distribution can be shifted consistently, for example producing more or fewer mechanical events with respect to the number of shots depending on the generated backpressure. While the technique might improve nominal performance on different launchers, we expect it to degrade performance out of nominal operation.

To solve the dataset labelling problem, they suggest to first train Generative Adversarial Networks (GAN) to synthesize both shot and non-shot data along with distorted variations. This allows the creation of fully labelled datasets, provided that the synthesized output is an accurate representation of the nature and distribution of real firearm data, a separate challenge in itself.

Asbach and Canty patent

A 2024 patent by Asbach and Canty [52] (associated with Armaments Research Company) is the most recent publicly available patent on the topic at the time of writing. It is posterior[12] to our own patent on the topic [19], which is not described in this section as it would be redundant with the rest of this thesis.

It first presents a more classical technique where a candidate is evaluated when the raw acceleration passes a threshold level. They then integrate the acceleration signal into a velocity signal, and impose a maximum and minimum profile along the length of the candidate vector. Although velocity is considered instead of acceleration, this is subject to the same caveats as the previous techniques using a predetermined whole-profile acceleration shot template such as the works of Chedid et al. [53], Ufer et al. [40] and Welsh et al. [47]. They also use the rotational information, as described previously by Ufer and Brinkley [40], for additional false-positive rejection.

More interestingly, they then present another technique which is a combination of a classical algorithm with machine learning. They first proceed to the identification of candidates in the acceleration signal through a rolling average and threshold. Candidates are optionally aligned to a template before being classified through a CNN into shot/non-shot. This alignment step is performed through a more classical algorithm as described in Section 3.1.1 which identifies the shock resulting from the actual shot, and shifts the candidate to it. When firing a burst with an automatic closed-bolt firearm, the end of a firing cycle (chambering and locking the round) is immediately followed with the beginning of the next one (percussion and ignition). Doing this alignment step avoids the case where the beginning of a candidate vector contains the end of the previous cycle. This step is potentially computationally extensive and requires the development of a reliable classical shot detection algorithm to be effective. As such, this method requires the intervention of expert knowledge for every new weapon and limits the added benefit of the machine learning model to improving prediction accuracy. Additionally, they suggest that the CNN might be multi-classed, in particular for the discrimination between live and training ammunition. A possibility that we will discuss extensively in Section 7.

The proposed solution to the dataset labelling problem is an iterative process. First, a small amount of data is labelled by a domain expert which is used to train a first model. This model is used to provide draft labels for the rest of the dataset, which can then be manually reviewed and corrected by the domain expert. While this is an improvement over a fully manual labelling of the dataset, this still requires the intervention of a domain expert,

[12]Patent applications are typically published approximately 18 months after filing. As such, the Asbach and Canty patent was filed without knowledge of our own prior patent.

and remains prone to input error. In contrast, the weakly-labelled technique suggested in this thesis removes this requirement.

Other notable patents

A 2009 patent application by Chedid et al. [53] (associated with Saab) attempted to claim the broad invention of detecting shots by comparing any physical temporal quantity with a template (although the suggested use is acceleration). Although withdrawn, it is noteworthy for its suggested technique. While the template is initially simply defined as an average of example shots, when computing the distance to the template of a candidate shot, each point is applied a weight inversely proportional to the standard deviation of example shots at that point. The intent is to give more weight to the most characteristic parts of the template. From this initial estimation, the template and weights are then refined with an iterative process to give optimal performance on the training data. This technique assumes that shots will have a consistent base fingerprint with added noise to be filtered. As discussed with other works, this will not generalise when using the weapon outside the intended nominal conditions and configuration. It is also likely to be susceptible to problems when candidates are not aligned temporally with the template. However, its intended application is shot detection from training ammunition (blanks) for the purpose of lighting up an infrared emitter in the context of laser-based shooting simulations. This application allows one to more tightly control the weapon configuration, be more relaxed with false positives, and not have to worry about individual shot detection/counting.

A 2018 patent by Amit et al. [50] (associated with Israel Weapon Industries) cites the Joannes et al. patent [18], and claims to improve upon it by being less susceptible to external variables such as the weapon mass, mounting system or shooter stance. The invention is described as using only a piezoelectric element instead of an accelerometer, but to the same effect, and is a possibility which already was present in previous works. In step 442, the round counter has to be initialized with calibration values including the firearm weight, barrel length, and ammunition caliber. It then describes a state machine algorithm similar to the prior techniques, except with an emphasis on the duration of shocks rather than on their energies, although both are still considered. The described technique is also focused on the implementation for the IWI Negev, instead of a generally applicable one. In summary, this technique is only a subset of the Joannes et al. work focusing on time intervals between peaks in a low-pass filtered version of the signal and not an actual improvement.

Around 2018, ShotPM offered shot counters for preventive maintenance based on accelerometer signal [14] (possibly in infringement of the Joannes et al. patent), integrated with the APB360 infrastructure. While their publicly available information makes claims of machine learning features [54], it is only a method of measuring the current burst rate of the weapon in order to apply a corrective factor on the number of shots detected with the aim of compensating undercounting by the default algorithm. It does not aim at deriving the shot counting algorithm itself.

A 2020 patent by Deu-Ngoc and Bujak [55] describes an invention for counting shots

and dry fires from accelerometer and gyroscope data. The technique is actually much simpler in nature than the previously described ones, only looking for the signal being above a certain threshold for a calibrated period of time. This patent is actually focused on applications in archery, where an interesting difference is that dry fires generate more energy than actual shots, which is the opposite of the case in firearms.

A 2023 patent by Reed et al. [56] (associated with the University of Central Florida) describes a system with similar intent to the Chedid et al. [53] work, performing trigger detection for force-on-force training. As such, it has similar drawbacks of only being concerned with shot activity detection (not counting), weak requirements on false-positives, and tightly controlled training conditions. They perform other kinds of activity detection, such as reloading, firing handle manipulation, etc. However, they also classify dry firing, which is sometimes considered part of the discrimination problem. In this implementation, instead of simply lighting up the IR emitter when a shot is detected, the information is transmitted wirelessly to a live training system such as an Augmented Reality (AR) device, along with information about the position and orientation of the weapon. While their filing is posterior to the Weiss et al. [51] patent, they suggest a practical implementation using a 3-layer multilayer perceptron.

A patent application filed in 2023 by Filipovic and Sostar [57] (associated with HS Produkt d.o.o.) does not seem to provide any significant improvement over the state of the art. It details the same two-stage system as prior techniques with a wakeup from a piezoelectric element and acceleration measurement from a digital accelerometer. The technique used for shot counting is only described as comparing the measured waveform with presets. Although not specific to shot counting, one significant difference is that they deliberately misalign their sensors relative to the firing axis to reduce the strength of the measured signal and avoid saturation. While this might be useful in their use case, we have seen in Section 1.3.2 that significant, correlated acceleration can also be present in the other axes.

In a similar vein, a 2024 patent by Kloepfer et al. [58] (associated with Biofire Technologies, also known simply as Biofire) describes a more limited version of prior techniques, but this time applying it to electrically-primed “smart guns” (i.e. firearms that can only be operated by a biometrically authorised user). The technique described uses simple acceleration thresholds in time windows, or acceptable ranges around average values of the raw acceleration. Optionally, different time windows can be applied in different axes, as already seen in prior techniques. A unique aspect of their technique is the use of trigger pull information, which is only available in the rare cases where firearms are specifically engineered to provide this data.

3.2.4 Conclusion on related works

In this section, we reviewed the existing literature on electronic shot counters, with a particular emphasis on acceleration-based techniques. Most approaches developed since 2006 are either similar to, or implement a subset of, the Joannes et al. technique [18] as described in Section 3.1.1. Truly novel methods have emerged only with the introduction of machine learning, pioneered by Loeffler [41], and have attracted increased industry interest as advances in artificial neural networks now enable more accurate real-time predictions than classical techniques. Consequently, there has been a marked surge in interest in shot counting from 2020 onwards. This justifies our choice of the previously described classical technique as a baseline for comparison with the machine learning approach presented in this thesis.

Furthermore, several works, including those of Gering et al.[7], Welsh et al.[47], Staiger et al.[9], Weiss et al.[51], Reed et al.[56], and Asbach and Canty[52], have identified ammunition discrimination, particularly between live and training ammunition, as an important topic. This motivates the extension of shot detection in this work to explicitly address ammunition discrimination, with a focus on this specific case.

Finally, almost all research involving machine learning techniques highlights the acquisition of suitable datasets as a major challenge. Academic studies [25], [41], [44], [46], [47], [48] have generally relied on relatively limited datasets, which are unlikely to capture the full range of real-world behaviours. Industrial solutions have proposed synthetic data generation [51], which introduces separate challenges, or iterative manual labelling [52], which remains labour-intensive and prone to error. These factors motivate our proposed approach based on weak labels, enabling the construction of large-scale datasets with minimal labelling effort. In addition, this motivates a dedicated study of dataset construction itself, in order to determine the minimal dataset sizes required for representativeness and the desired level of predictive performance.

Chapter 4

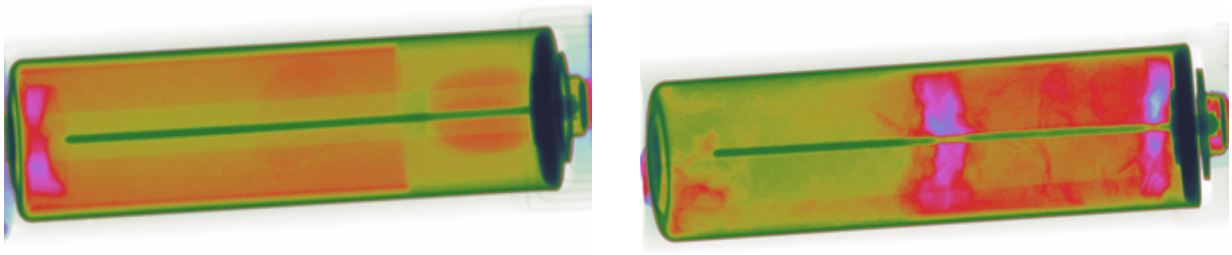
Hardware description

4.1 SmartCore hardware

The current FN SmartCore v3 hardware platform comprises:

- Microcontroller: Nordic Semiconductor nRF52840 [59], an ultra-low power 64 MHz ARM Cortex-M4F, 1 MB flash and 256 kB RAM, capable of Bluetooth Low Energy (BLE) communication.
- Accelerometer: Analog Devices ADXL372 [60], ±200 g, 6400 Hz 3-axis MEMS, capable of waking up from an ultra-low-power state upon detecting significant acceleration and acquiring data to its internal FIFO buffer.
- Power regulator: Texas Instruments TPS78225 [61], 2.5 V low-dropout linear regulator specialised in ultra-low quiescent current.
- Real-time clock: Abracon AB0805 [62], ultra-low-current RTC.

The circuit is powered by a non-replaceable battery, whose exact specification depends on the requirements of the particular application. An electronic shot counter battery should be able to provide the continuous sleep current (approximately 2–3 µA) and the intermittent working current (up to 20 mA for a few seconds). This is generally accomplished by the use of a low-power battery associated with a supercapacitor for peak-current support. In addition to its required temperature range, an important factor to consider is the construction of the battery with respect to the acceleration it will undergo along the firing axis. As seen in Figure 1.23, the firing (longitudinal) axis undergoes the highest acceleration. Figure 4.1 illustrates the effect of sustained firing on a battery. The cathode in this type of $LiSOCl_2$ battery is a porous carbon mass permeated by $SOCl_2$, with a current-collecting nail inserted into the carbon and connected to the top terminal. In this example, the internal carbon mass has completely migrated to the top of the battery and has lost contact with the internal nail, resulting in a complete loss of voltage at the battery terminals.



(a) Before (b) After

Figure 4.1: X-ray imagery of a battery before and after undergoing 1500 M2 shots in the longitudinal axis. Colours represent material density, with blue indicating low density and red indicating high density.

In the case of an accelerometer-based shot counter, the important choice of acceleration sensor deserves further discussion. As seen in Section 3.2.3, to reduce idle power consumption to its minimum it is common to use a piezoelectric element to passively monitor the acceleration and generate the wake-up interrupt. A major issue with this approach is that a significant delay is introduced before the acceleration signal can be processed. Indeed, once a significant acceleration occurs, we first incur the delay of the piezoelectric element reaction time, then the time required for the microcontroller to start up, initialise the accelerometer, and acquire the first data point. This leads to a loss of acceleration data during the initial firing event, which contains the most important information for shot detection and classification. In addition, a piezoelectric element large enough to generate the required wake-up voltage and sturdy enough to withstand the high accelerations presents both cost and encumbrance challenges.

This problem is alleviated with the use of a digital accelerometer capable of independent wake-up and acquisition to an internal data buffer. In this setup, the accelerometer operates in an ultra-low-power mode until a preset acceleration threshold is exceeded. Once this threshold is triggered, the accelerometer wakes and immediately begins acquiring data to its internal buffer. This enables delayed startup of the microcontroller without any loss of data.

This led us to choose the ADXL372, which provides this feature with the name “Instant On”, as illustrated in Figure 4.2. In this mode, the sensor uses a wakeup threshold around 10 g, while consuming only 1.4 µA at 2.5 V. In practice, the 512-sample buffer offered by the ADXL372 allows up to an 80 ms delay at the maximum 6400 Hz acquisition rate. In Section 3.1, we have seen that due to the highly impulsive nature of shots a high sampling frequency is desirable. Until recently, the high acceleration range of 200 g was also seen as a requirement. Although we will see in Section 8.3 that recent developments might enable the use of 16 g accelerometers in some cases, using a high-range accelerometer still tends to improve prediction accuracy. At the time of writing, the ADXL372 remains the best-known accelerometer for this use case, as few sensors on the market offer high-frequency, high-range signals together with self-wakeup and sampling features.

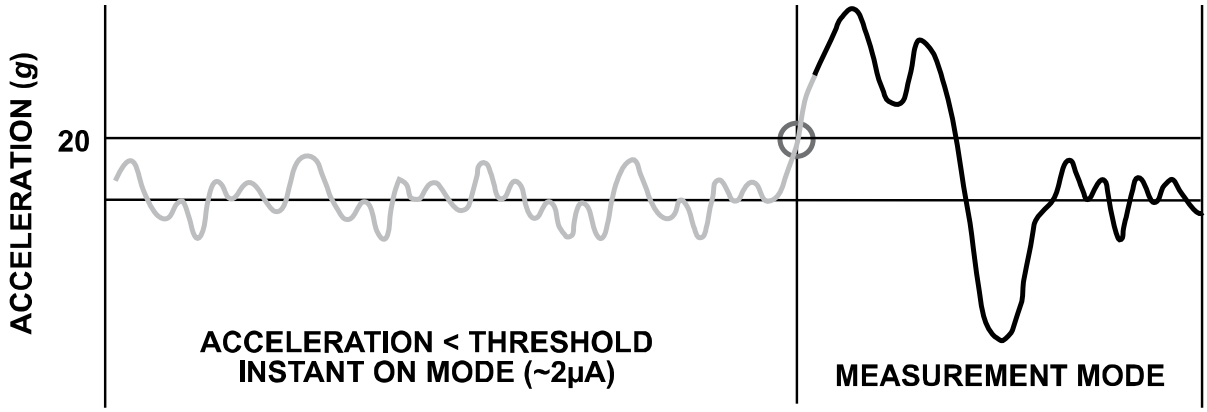


Figure 4.2: "Instant On" mode of the ADXL372 [60]. The grey trace shows acceleration monitored by the wake-up circuitry of the device, but no measurements are stored. When the signal exceeds a user-configured threshold, the device enters measurement mode after a delay of approximately 1 ms, and data acquisition begins.

4.2 ADXL372 study

4.2.1 Configuration

For reproducibility purposes, we include in Table 4.1 the device parameters used to acquire the shot database. Parameters not explicitly mentioned were kept at their default values.

Data was retrieved through SPI when the watermark interrupt was raised.[1] To save on both buffer memory and subsequent computing requirements, the raw 12-bit acceleration values were immediately rounded to the nearest 8-bit integer.

4.2.2 Calibration

The accelerometer is first validated through its self-testing functionality which, as described in the datasheet [60], applies an electrostatic force to simulate an acceleration and verifies that the correct response is obtained through the whole electromechanical acquisition chain.[2]

The device is documented to be sensitive to input voltage variations and temperature. In particular, sudden instabilities in the input voltage will skew instantaneous measurements and are sufficient to wake up the device from Instant On mode even when no acceleration is present. Care should be taken during hardware design to minimise ripples

[1] On configuration of the Instant On mode, the corresponding interrupt can be raised erroneously. As a workaround, these interrupts should be ignored between the start and the end of the configuration process.

[2] In addition to the procedure advised in the datasheet, we have found that disabling the FIFO buffer (i.e. setting FIFO_CTL to 0) is necessary. Otherwise, the self-test results in a systematic failure. The B-sample errata warns that false negatives are possible; testing over 15,000 cycles has shown a practical 0.19% rate of failed tests on correctly performing parts.

Register	Field	Value	Comment
Basic configuration			
FIFO_CTL	FIFO_SAMPLES	255	Internal buffer watermark level
	FIFO_MODE	1	Streaming
	FIFO_FORMAT	1	X-axis only
TIMING	ODR	4	6400 Hz
MEASURE	BANDWIDTH	4	3200 Hz
	LOW_NOISE	true	Power/noise tradeoff
INT1_MAP	MAP_FIFO_FULL	true	Watermark interrupt
INT2_MAP	MAP_FIFO_OVR	true	Data missed interrupt
Instant On mode			
POWER_CTL	MODE	2	Instant On
	INSTANT_ON_THRESH	0	Low threshold
	FILTER_SETTLE	1	16ms settling time
	HPF_DISABLE	true	No high-pass filter
	LPF_DISABLE	true	No low-pass filter
Full curve acquisition			
POWER_CTL	MODE	3	Full bandwidth measurement
	FILTER_SETTLE	true	16ms settling time
	HPF_DISABLE	true	No high-pass filter
	LPF_DISABLE	true	No low-pass filter

Table 4.1: ADXL372 device configuration.

in the input voltage, and in particular to avoid switching regulators. An example is given in Figure 4.3 where, in a faulty design, a 100nF decoupling capacitor was not sufficient to prevent a switching regulator from causing an 8.8 kHz, 81 mV peak-to-peak ripple and constantly waking up the accelerometer. Similarly, attempting to minimise sleep-mode power consumption by switching off the power regulator and enabling it upon wake-up risks impacting the data acquisition due to the sudden change in input voltage. Conversely, not regulating the input voltage results in drift in both acceleration values and sampling frequency as the battery depletes, requiring regular recalibration.

As a result, we opted to use a constantly-on LDO power regulator in the form of the Texas Instruments TPS7822. Its particularly low quiescent current enables this application with the given battery size. In practice, the resulting electronics uses overall between 2 and 3 µA in sleep mode at 3.3 V.

Even in identical conditions, the sampling frequency of the ADXL372 can vary by more than ±10% depending on the individual part, as shown in Figure 4.4. This variation can

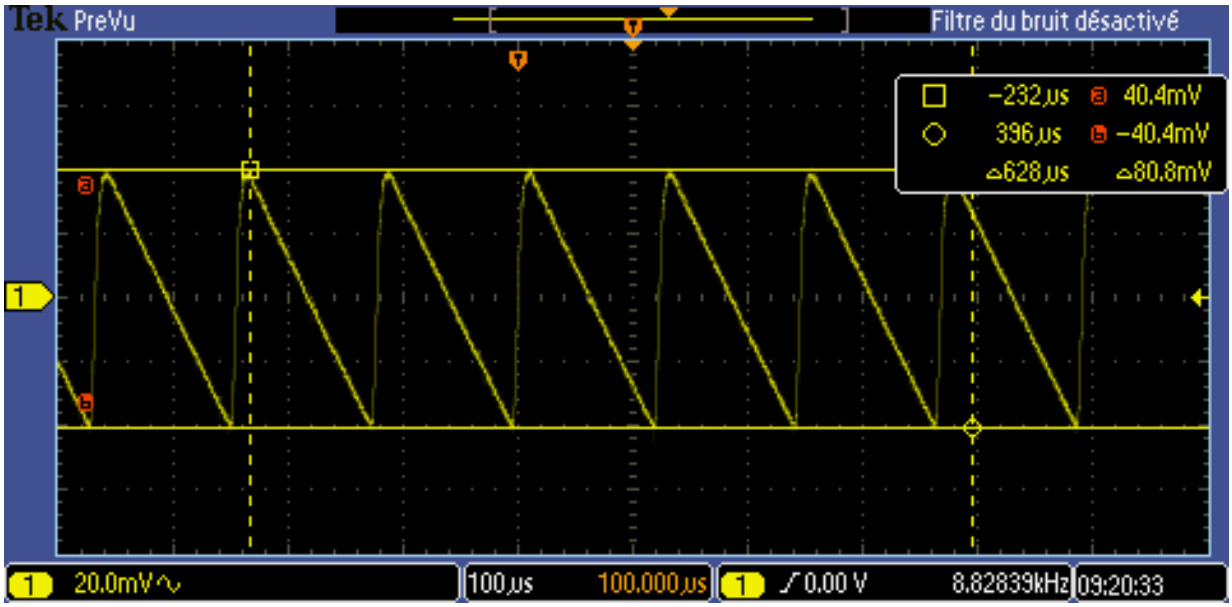


Figure 4.3: Example of ripples sufficient to cause spontaneous wake-up from Instant On mode.

be even greater depending on input voltage and temperature. One solution is to trigger the sampling using an external, more precise clock reference. However, in practice, this is often incompatible with the use of the Instant On feature, as a constantly powered reference clock would significantly increase the current draw in standby mode, which must remain low for non-rechargeable devices. While powering on the external clock after wakeup is possible, the relatively long startup time of typical components means that the internal clock of the accelerometer would be used for the first shot, before reconfiguration to the external clock becomes possible.

Sampling frequency self-measurement can be performed by acquiring data for a fixed amount of time as measured by the more accurate 16 MHz crystal oscillator of the microcontroller. We then compute the ratio of the actual sampling frequency to the expected sampling frequency, which we denote as TDF for Time-Dilation Factor:

$$TDF = \frac{T_{actual}}{T_{expected}}. \tag{4.1}$$

Empirical data illustrated in Figure 4.6 shows that after about 113 s of sampling, the measured TDF deviates from the 445 s long-term average by less than 0.06%, a negligible error. Accordingly, we used the 113 s measurement to calibrate all devices in this study.

Parts with a deviation greater than ±10% from the 6400 Hz target are rejected. For accepted parts, all recorded acceleration data is associated with the measured effective sampling frequency as metadata. Measurements of time intervals and signal transforms can be corrected proportionally with the measured TDF to correct this constant offset.

If the sampling frequency measurement is done while the device is at rest, with its

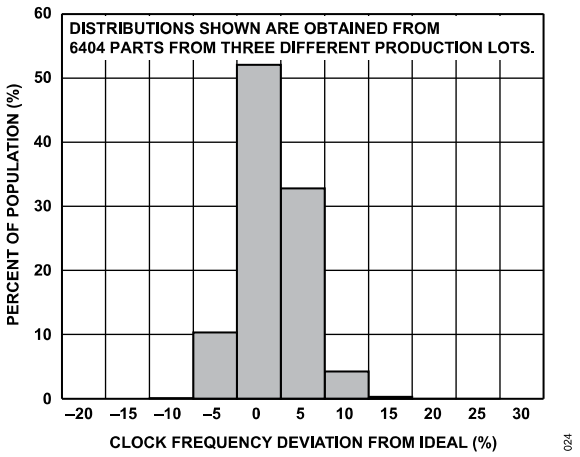


Figure 4.4: Clock Frequency Deviation from Ideal at 25 °C, 6400 Hz sampling frequency, 2.5 V input voltage [60].

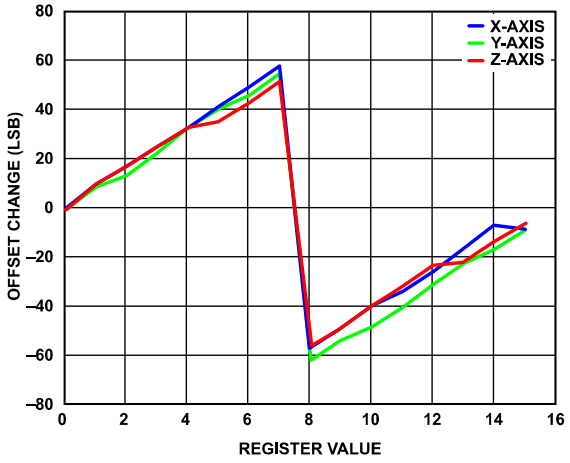


Figure 4.5: Official estimation of the hardware-level acceleration offsets for the ADXL372 [60].

firing axis aligned with the horizontal, we can also proceed to a zero-g calibration at the same time. The average measured acceleration can be compared to the expected value of 0 g, and compensated accordingly. The ADXL372 provides calibration of the device at the electromechanical level, which enables us to correct the larger part of the expected drift without inducing saturation effects. The exact offset applied is unfortunately also part-dependent, and would ideally need to be calibrated as well. In order to avoid unacceptably long calibration times during device production, we computed an estimation for the X axis over 3 parts, each calibrated 3 times for each possible offset value for a total of 144 data points. For each offset value, we measure the average difference between the value obtained after a full calibration, and the one obtained without any offset applied. The resulting estimations are shown in Table 4.2. On average, the measured maximum and minimum offsets fall within 3.9% of these estimations.

In practice, we first measure the zero-g offset with default parameters (no corrections applied) and apply the closest correction from Table 4.2. Any residual error will be corrected in firmware at the microcontroller level. A second pass of calibration lasting 10 s is then performed with this estimated correction applied to fine-tune the calibration for the considered ADXL372 part. The average remaining error is applied as an additional firmware-level offset.

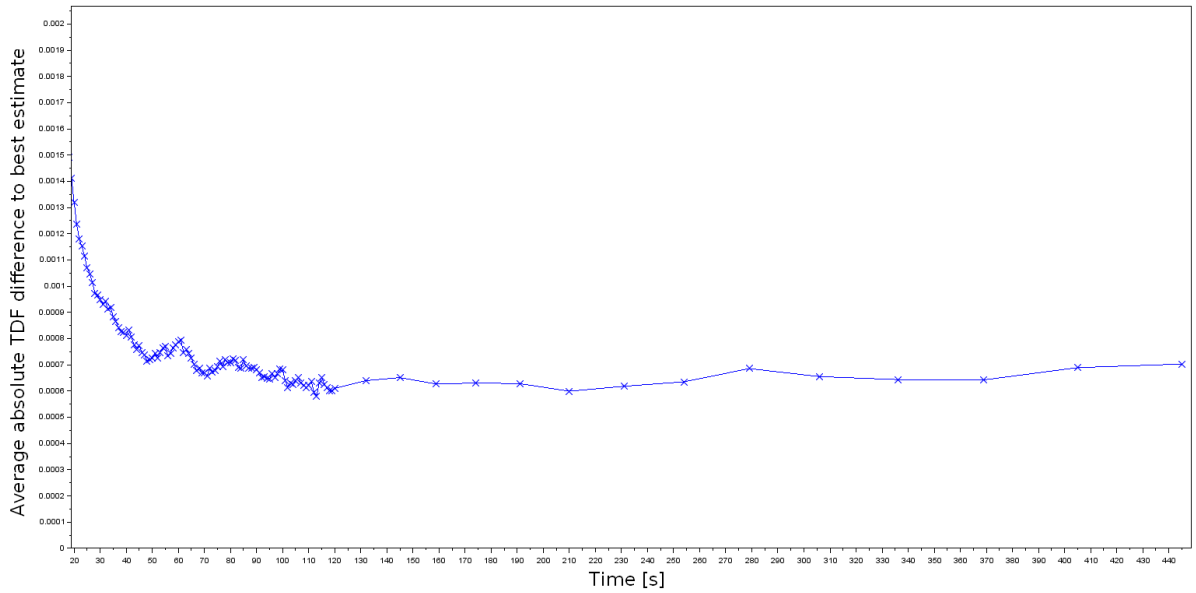


Figure 4.6: Average over 24 runs of the difference between the TDF obtained after a certain amount of time and the average TDF obtained after 445 s of sampling for all runs.

Setting value	Measured offset [LSB]	Setting value	Measured offset [LSB]
0	0	8	-55.8
1	8.1	9	-47.9
2	14.4	10	-45.1
3	21.4	11	-38.2
4	29.4	12	-30.2
5	37.3	13	-22.2
6	43.4	14	-16.0
7	50.4	15	-8.7

Table 4.2: Estimation of the effect of the "OFFSET_X" offset trim register.

4.2.3 Instant On wake-up

In Instant On mode, the ADXL372 is supposed to wake up when the acceleration signal rises above a certain threshold. The part specifications offer two levels of threshold with a 10–15 g range (low threshold) and a 30–40 g range (high threshold). However, experiments have shown that the practical range is heavily part-dependent and that the actual wakeup can occur significantly outside of this advertised range. Tests were run on 15 devices from 3 different lots. In low threshold mode, practical thresholds were observed from 6–22 g. In high threshold mode, a practical range of 20–40 g was observed. In addition, discussions with the parts manufacturer uncovered that the Instant On sampling rate is a fixed 2 kHz, independent of the configured normal measurement rate. We also discovered that the wake up circuit is independent of the configured measuring axis; for example a device configured to measure only the X axis will still be woken up by a significant acceleration event in the Y or Z axes.

Figure 4.7 shows the practical consequence of these differences. In this example, we observe that the device exits its Instant On mode well before (>40 ms) the actual shot occurs, and before a significant acceleration over 10 g is observed in the firing axis. Repeating this experiment 31 times and recording the difference between the wake-up from Instant On and the actual start of the shot (as decided by classical shot detection algorithms), we obtain the histogram shown in Figure 4.8. We observe two distinct windows in which the device is likely to undergo an early wake-up. Mapping these windows onto the continuously acquired data (without Instant On mode) in the firing axis yields Figure 4.9. We observe that while some activity is indeed present in these windows, it remains well below 10 g.

If we now observe the other acceleration axes as illustrated in Figure 4.10, we indeed observe events on the Y axis (up-down) and Z axis (sideways), sufficient to trigger at least the low threshold.[3] We suppose that this is due to the open-bolt cycle of the Minimi platform. Open-bolt weapons have their automatic cycle interrupted when the mobile parts of the weapon are at the rear, and the spring compressed. This allows the weapon to have its chamber kept open while idle, enabling air to enter it and passively cool down the system, avoiding overheating which could cause spontaneous cartridge ignitions (cook-offs). When the trigger is pulled, the mobile parts are released and travel to the chamber before locking and allowing percussion, and thus ignition to occur. Our results indicate that the release of the mobile parts causes a significant, detectable acceleration, but only along the axes perpendicular to the firing axis.

For our application, these kinds of early wake-ups are actually desirable, allowing us to acquire the entirety of the first shot, including the locking phase. In general, wake-ups on events in other axes can at worst be ignored, and at best allow for an earlier wake-up than if only the firing axis was considered.

[3] Surprisingly, this signal is also sufficient to trigger an early high threshold (>30 g), despite even the XYZ signal being lower than this threshold. This is currently unexplained. Discussions with the part manufacturer led to believe that some high-acceleration but very transitory events could still trigger the wake-up from Instant On mode. Further testing with a higher sampling frequency device is required to confirm this hypothesis.

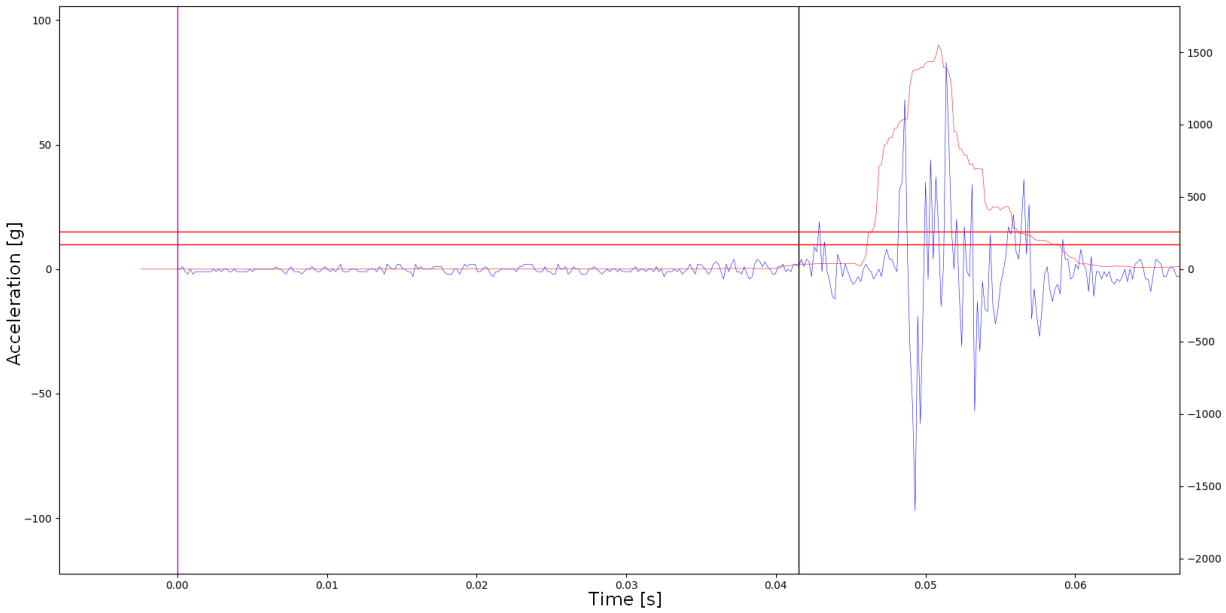


Figure 4.7: Example of early wake-up on Minimi 5.56 acceleration signal (blue). The black bar signals the start of the shot as detected by a classical algorithm. The magenta bar signals wake-up from Instant On mode.

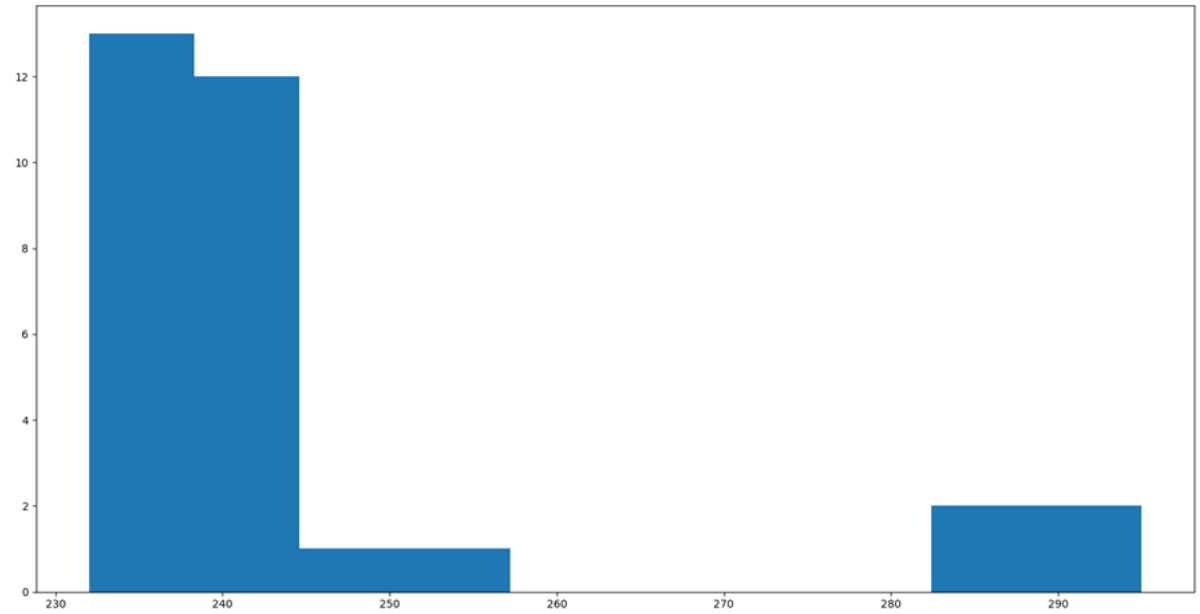


Figure 4.8: Distribution of the number of samples between Instant On wake-up and the actual start of the first shot.

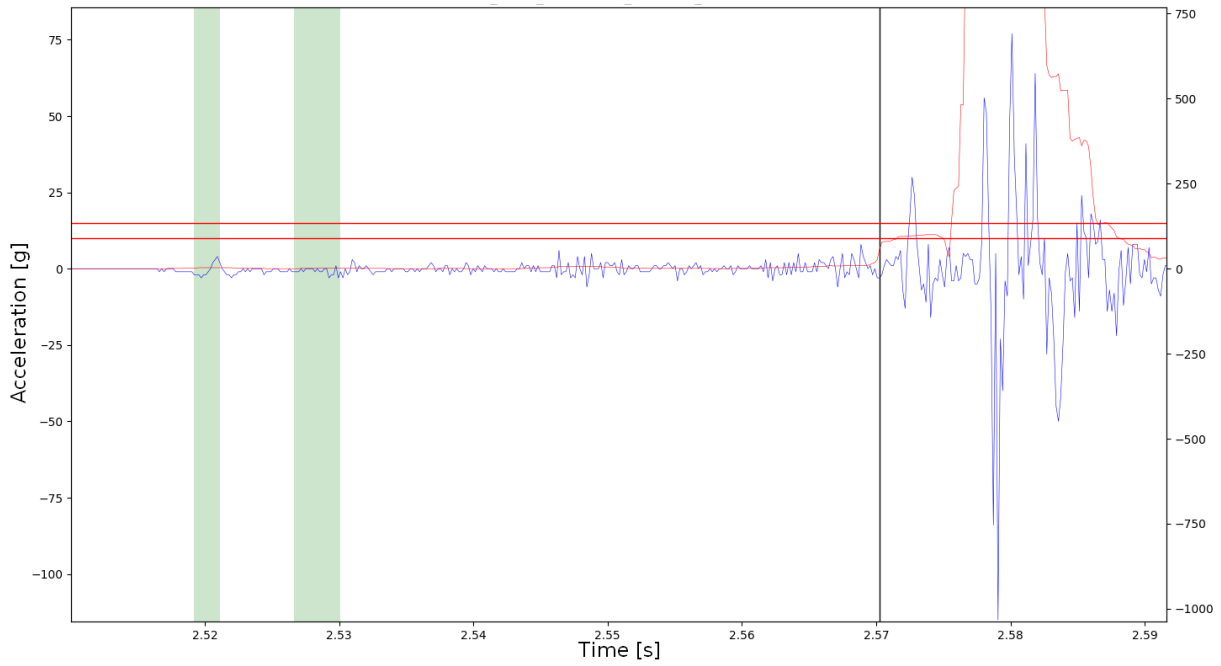


Figure 4.9: Early wake-up windows (in green) on Minimi 5.56. Horizontal red bars represent the datasheet-estimated wakeup region for low threshold, between 10 and 15 g.

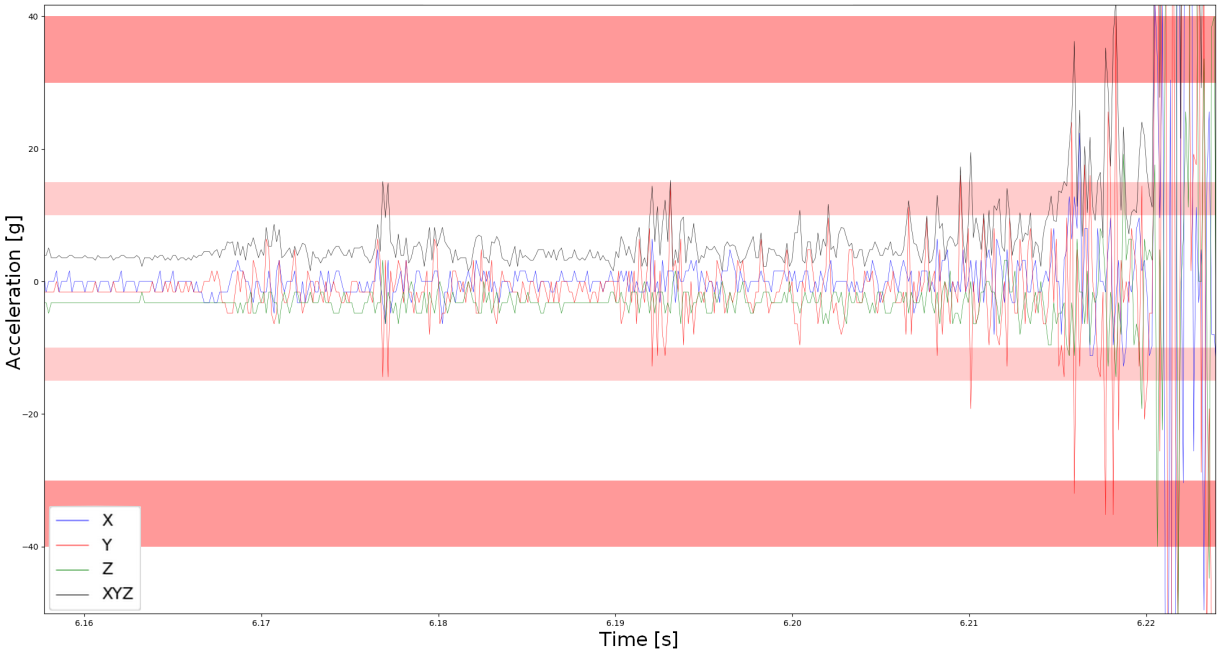


Figure 4.10: Data acquisition on Minimi 5.56 before the initial shot. The XYZ data is the magnitude of the total acceleration vector. Horizontal red sections represent the datasheet-estimated wakeup regions for the low and high thresholds.

On the other hand, this makes constructing a shot dataset from continuously acquired data inadequate. These experiments have shown that it is impossible to reliably predict in which part of the firearm cycle a given part will be woken up from Instant On mode and start the actual acquisition. However, accurately representing the initial part of the signal is crucial. In an experiment, introducing a simulated 1 ms wakeup delay can increase a model's error rate from around 0.5% to over 6%. Yet, when trained on the shifted inputs, the error rate falls once again to around 1%. This implies that most of the information has not been lost, but that proper characterisation of the wakeup signal has a large impact on model performance. In order to take this unpredictability into account, data acquisition thus must be performed with a representative sample of parts configured in Instant On mode, thereby reproducing the true distribution of wakeup signals.

Figure 4.11 shows data acquired after waking up from Instant On mode, on a weapon which does not produce early wake-ups. As such, the overall acceleration around the shot event shows a visibly distinct pattern between the first shot of a burst and subsequent shots. Interestingly, the associated counting network learned to produce valid predictions for both, even though the input vector will be significantly different.

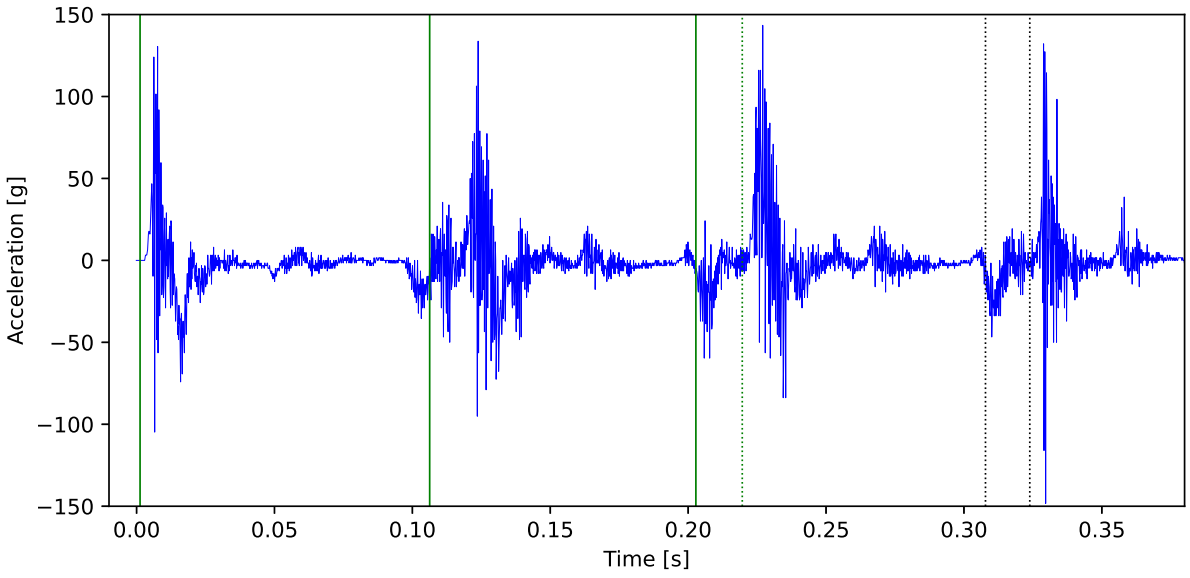


Figure 4.11: Burst of 3 shots with M2 barrel after band-stop filtering (see Section 6.8), acquired with Instant On mode. Shot candidates are indicated with vertical lines, with full lines being classified shots and dotted line non-shots.

4.3 Considered weapons

In this thesis we will evaluate our shot counters on five weapons, for which we will use the following abbreviations:

1. FN Minimi® 5.56 (also known as M249), referred to as Minimi 5.56 or M556.
2. FN Minimi® 7.62 (also known as Mk 48), referred to as Minimi 7.62 or M762.
3. FN MAG® (also known as M240 or GPMG), referred to as MAG.
4. FN® M2HB-QCB (also known as Browning .50 cal or M2A1), referred to as M2.
5. M134 Minigun (also known as GAU-2/A or GAU-17/A), referred to as M134.

4.3.1 FN Minimi®

The FN Minimi® 5.56 is illustrated in Figure 4.12. It fires from an open bolt, meaning that the firing cycle starts with releasing the mobile parts from the back of the weapon. It uses a long-stroke reloading system, in which a portion of the high-pressure gas released from the cartridge is used to push back on the mobile parts of the weapon, providing the energy for the reloading cycle. We integrated the electronic shot counter inside the handle. This part is originally hollow, and could be used to store an optional field-maintenance kit.

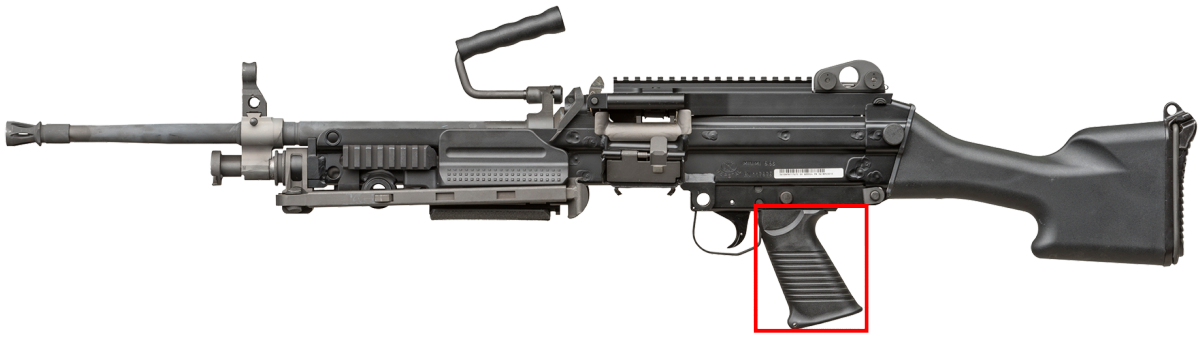


Figure 4.12: FN Minimi® 5.56. The shot counter is embedded inside the handle, highlighted in red.

The FN Minimi® 5.56 and FN Minimi® 7.62 differ in their ammunition calibre, but share an identical handle. The shot counter hardware can thus be fully shared between the two models. We, however, consider them different firearms in this thesis, as their differing ammunition produces a very different acceleration signal. Since shot counters are not frequently switched between weapons in an army setting, different counting models can be used with the shot counter being manually configured for the associated calibre.

The 5.56 model is notoriously difficult to develop counting algorithms for due to its very wide number of possible configurations. It presents short and long barrel variants,

can include mechanical or hydraulic damping systems. It can be fired from a bipod, tripod, prone, shouldered or from the hip. A large range of accessories can be mounted, including suppressors. This wide range of possible setups induces a large variance in the associated acceleration signals. In addition, the weapon tends to show low-information measurements in some configurations due to its heavy weight relative to the low-power 5.56 mm ammunition.

4.3.2 FN MAG®

The FN MAG®, illustrated in Figure 4.13, presents a different, heavier design but a firing cycle similar to the Minimi. It shares a common ammunition with the Minimi 7.62. The shot counter is also integrated in the handle. However, the original handle of this firearm is partly full and comprises two traversing screws, making practical integration more challenging.

Figure 4.13: FN MAG®. The shot counter is embedded inside the handle, highlighted in red.

Compared to the Minimi, it presents a lower number of variants. However, it is frequently used with fixed mounting systems which can have a significant influence on the measured signal.

4.3.3 FN® M2HB-QCB

The FN® M2HB-QCB is a heavy machine gun using a short recoil firing cycle. This mechanism leverages the recoil energy of a shot to power its firing cycle. Its barrel is stopped shortly after the shot, while the mobile parts continue, thus presenting a complex firing signature. It is one of the heaviest firearms still considered an infantry weapon, and its 12.7 mm serves as an upper bound for our study. We integrate a first shot counter in the handle of the backplate, as illustrated in Figure 4.14.

The QCB of its designation stands for Quick Change Barrel. In this variant, the barrel can be quickly unlocked and swapped for a fresh one, allowing for sustained fire while

Figure 4.14: FN® M2HB-QCB. A shot counter is embedded inside the backplate handle, highlighted in red.

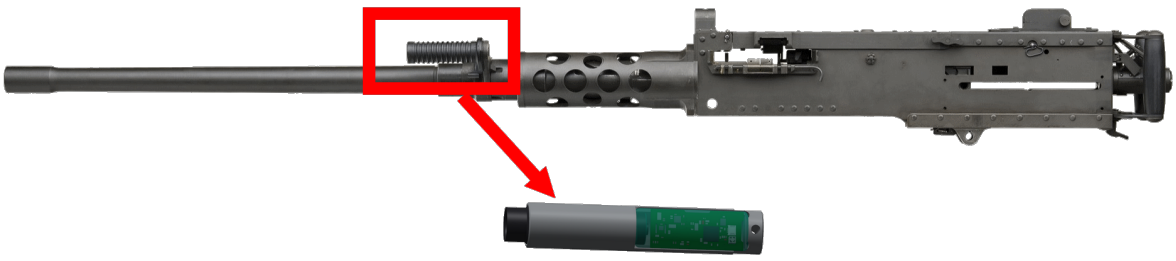


Figure 4.15: FN® M2HB-QCB. A shot counter is embedded inside the barrel handle, highlighted in red.

preventing overheating. These barrels are an important and expensive part of the weapon, and in some cases can be shared among multiple weapons. As such, it is useful to track the number of shots per barrel. A second shot counter can be included in the barrel handle, as illustrated in Figure 4.15. Due to its short-recoil reloading system, this firearm uses a different type of barrel to fire training ammunition. Barrel shot counters thus do not require ammunition discrimination capabilities, as they can be manually configured with the associated barrel type. However, the significant heat emanating from the barrel under fire (in excess of 400°C) presents unique challenges for electronics integration.

In order to differentiate the two systems in short-form, we will abbreviate the backplate handle counter as M2P (for backPlate) and the barrel handle counter as M2R (for baRrel).

4.3.4 M134 Minigun

The M134 Minigun is illustrated in Figure 4.16. It comprises six rotating barrels, powered by an electric motor along with the reloading system. This allows for very high rates of fire up to 6000 rounds per minute. This is an order of magnitude higher than infantry machine guns. Although our study of this weapon remains limited, it allows us to test our implementation at these high rates of fire requiring similarly fast inference times. Its electrically-driven firing cycle also presents a unique challenge.

Figure 4.16: M134 Minigun. The standard (Picatinny) accessory rail is highlighted in red.

Since this weapon was only included for theoretical analysis rather than practical deployment, the shot counter was attached to the standard accessory rail of the firearm.

Chapter 5

Error rate definition

In Section 1.3.4, we identified the potential of weak labels, meaning labels that provide only the total number of shots in a time series rather than marking the exact position of each shot, to simplify dataset construction and thus the leveraging of much larger datasets. However, when dealing with weak labels, we are unable to verify model predictions at the instance level and compute usual performance indicators such as accuracy. Instead, we have error measurements per weak label and thus per time series.

For our problem, two types of error are possible:

1. **Counting error**: The difference between the ground-truth number of shots and the estimated number in each time series.

2. **Discrimination error**: The classification error between all possible ammunition classes.

In this section, we define two corresponding error rates that serve as the primary metrics for evaluating model performance. Although machine learning models are trained using a loss function adapted to weakly labelled data, these error rates are used to select the best models after training, to compare benchmark performance on both simulated and real-world datasets, and to assess the impact of different loss functions or training strategies. In the absence of instance-level ground truth, these metrics provide a consistent and interpretable basis for quantitative comparison throughout this thesis.

5.1 Counting error

The final counting error on a given dataset will be:

$$\sum_{x \in \mathcal{A}} \left| \hat{c}^{(x)} - c^{(x)} \right|, \tag{5.1}$$

where $\mathcal{A}$ denotes the set of all acceleration time series, $c^{(x)}$ is the real number of shots in time series x, and $\hat{c}^{(x)}$ is the predicted number. Since this will naturally increase with the

size of the dataset, we can express this as a relative counting error by dividing by the total number of shots, yielding a counting error rate:

$$E_C = \frac{\sum_{x \in \mathcal{A}} \left| \hat{c}^{(x)} - c^{(x)} \right|}{\sum_{x \in \mathcal{A}} c^{(x)}}. \tag{5.2}$$

This is ultimately the metric that will be seen and evaluated by the typical end-user, who is interested in the estimated number of shots fired by a weapon over a long period.

However, in the general case, this metric does not fully reflect classification performance. This is because false positives (overcounting) and false negatives (undercounting) can offset each other when only the total count is considered. As $c^{(x)}$ increases, such compensation becomes more likely, so the relative counting error per time series may decrease even if the underlying classification performance remains unchanged.

In the general case, we can model counting as the sum of two binomial distributions as

$$T = S + N, S \sim \text{Binomial}\big(c^{(x)},\ \text{TPR}\big), N \sim \text{Binomial}\big(n^{(x)},\ \text{FPR}\big), \tag{5.3}$$

where $n^{(x)}$ is the number of non-shot candidates in x, TPR is the True Positive Rate and FPR is the False Positive Rate. The values of $n^{(x)}$, TPR, and FPR will vary depending on the model and cannot be computed exactly. However, when considering only acceleration curves of shot data, we can assume that $n^{(x)}$ will be proportional to $c^{(x)}$. If $n^{(x)} = r\ c^{(x)}$, then:

$$\text{Var}(T) = c^{(x)} TPR(1 - TPR) + n^{(x)} FPR(1 - FPR) =: c^{(x)} C, \tag{5.4}$$

where C is a constant that depends on the model. Thus, the standard deviation of the count is proportional to $\sqrt{c^{(x)}}$. The normalized counting error can then be defined as:

$$E_{C_{\text{norm}}} = \frac{1}{|\mathcal{A}|} \sum_{x \in \mathcal{A}} \frac{\left| \hat{c}^{(x)} - c^{(x)} \right|}{\sqrt{c^{(x)}}}. \tag{5.5}$$

Unfortunately, this metric is not defined for input containing only non-shot data. Additionally, it is difficult to interpret directly, as one must compute $e = \sqrt{c^{(x)}} E_{C_{\text{norm}}}$ in order to obtain the standard error on a time series with $c^{(x)}$ shots. However, datasets acquired in controlled conditions will often consist of a fixed number of shots per time series, as the ammunition container providing the weak label will be the same. In that case, $E_{C_{\text{norm}}} = E_C \sqrt{c^{(x)}}$ and thus $E_C = \frac{e}{c^{(x)}}$, the relative standard error, which is directly interpretable. For example, for an input with 100 shots and $E_C = 5\%$, the standard error will be ± 5 shots. Note that for small variations of $c^{(x)}$, $E_{C_{\text{norm}}} \approx E_C \sqrt{\bar{c}}$, where $\bar{c} = \frac{1}{|\mathcal{A}|} \sum_{x \in \mathcal{A}} c^{(x)}$.

Since the datasets used in this study indeed present fixed $c^{(x)}$ (see Section 8.1), we will prefer to report E_C when comparing relative performance on the same dataset for ease of interpretation, and only report $E_{C_{\text{norm}}}$ in the final benchmark.

5.2 Discrimination error

Since counting error is evaluated independently, it is important to ensure that discrimination error is not affected by overcounting or undercounting. In the general case of N ammunition types, let d_i denote the ground-truth number of shots (or other countable events, if applicable) of type i, and $\hat{d}_i$ the value estimated by the model. We have:

$$c^{(x)} = \sum_{i=1}^{N} d_i^{(x)}, \ \hat{c}^{(x)} = \sum_{i=1}^{N} \hat{d}_i^{(x)}. \tag{5.6}$$

When summing errors across categories, the total splits into the overall count error plus twice the number of misclassified shots, since each misclassification decreases the count in the true category and increases it in the predicted category:

$$\sum_{i=1}^{N} \left| \hat{d}_i^{(x)} - d_i^{(x)} \right| \ = \ \left| \hat{c}^{(x)} - c^{(x)} \right| \ + \ 2 \left(\# \text{ of misclassified shots} \right). \tag{5.7}$$

We can then define the normalized discrimination error similarly to $E_{C_{\text{norm}}}$:

$$E_{D_{\text{norm}}} \ = \ \frac{1}{|\mathcal{A}|} \sum_{x \in \mathcal{A}} \frac{\sum_{i=1}^{N} \left| \hat{d}_i^{(x)} - d_i^{(x)} \right| \ - \ \left| \hat{c}^{(x)} - c^{(x)} \right|}{2\sqrt{c^{(x)}}}. \tag{5.8}$$

When acquiring data under controlled conditions, it is sometimes possible to cheaply obtain time series in which each sequence features only a single type of ammunition. Although this is not always feasible, for instance in the case of linked blank and tracer rounds ("4B1T"), the live/blank discrimination task, which forms the focus of the experimental component of this thesis, does allow for such clean separation, since live and blank rounds cannot physically be fired together.

If we know the ground-truth type of each shot in our dataset, similar to E_C, we can instead define a more easily interpreted metric, the misclassification error rate E_D:

$$E_D \ = \ \frac{\sum_{x \in \mathcal{A}} \sum_i \sum_{j \neq i} \hat{d}_{ij}^{(x)}}{\sum_{x \in \mathcal{A}} \hat{c}^{(x)}}, \tag{5.9}$$

where $\hat{d}_{ij}$ represents the count of shots of type i that are classified as type j by the model[1].

Moreover, since we do not know the real-world distribution of the different types of ammunition, it is desirable to define balanced versions of these error metrics by averaging across types. This prevents artificially improving the error metric by adding data focused on well-performing cases.

$$\bar{E}_D = \frac{1}{N} \sum_{i=1}^{N} \frac{\sum_{j \neq i} \hat{d}_{ij}}{\sum_j \hat{d}_{ij}}, \tag{5.10}$$

[1] We divide by $\hat{c}$ instead of c to prevent the influence of counting errors on the discrimination error.

For the simplified case of discrimination between live and training ammunition, this becomes:

$$\bar{E}_D = \frac{E_B + E_K}{2}, E_B = \frac{\hat{d}_{BK}}{\hat{d}_{BB} + \hat{d}_{BK}}, E_K = \frac{\hat{d}_{KB}}{\hat{d}_{KK} + \hat{d}_{KB}}, \tag{5.11}$$

where B (for **B**all) denotes live rounds and K (for blan**K**) denotes blank rounds.

Similarly, we adapt the counting error from Equation 5.2 by balancing across ammunition types as follows:

$$\bar{E}_C = \frac{1}{N} \sum_{i=1}^{N} \frac{\sum_{x \in \mathcal{A}_i} \left| \hat{c}^{(x)} - c^{(x)} \right|}{\sum_{x \in \mathcal{A}_i} c^{(x)}}, \tag{5.12}$$

where $\mathcal{A}_i$ denotes the set of acceleration time series for ammunition type i. The normalized versions of the errors can also be balanced as such:

$$\bar{E}_{C_{\text{norm}}} = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{|\mathcal{A}_i|} \sum_{x \in \mathcal{A}_i} \frac{\left| \hat{c}^{(x)} - c^{(x)} \right|}{\sqrt{c^{(x)}}}, \tag{5.13}$$

$$\bar{E}_{D_{\text{norm}}} = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{|\mathcal{A}_i|} \sum_{x \in \mathcal{A}_i} \frac{\sum_{j=1}^{N} \left| \hat{d}_j^{(x)} - d_j^{(x)} \right| - \left| \hat{c}^{(x)} - c^{(x)} \right|}{2\sqrt{c^{(x)}}}. \tag{5.14}$$

5.3 Sample size

In the following chapters, we will study models that are trained from random initialization. We will see that a significant variance exists in the resulting counting or discrimination error depending on the initial values. For each case, we train a significant number of models with different random initializations in order to properly compare the prediction quality between models.

Counting models being slower to train, the sample size will usually be 20 networks unless specified otherwise. Discrimination models using label-proportion loss are also sampled in groups of 20. Discrimination models using cross-entropy loss are faster to train (see Section 7.2.3) and are sampled in groups of 200.

When graphically representing performance in figures, we use box plots along with a line plot marked with crosses to indicate the average. For the early versions of the technique, the plots omit non-converging networks from the sample group.

Chapter 6

EDGAR: Embedded Detection of Gunshots by AI in Real-time

6.1 Introducing the technique

The objective of this work is first to provide an innovative way to derive an instance-level projectile launch detector through deep learning from a database labelled only with counts of each type of projectile in a series. By instance-level, we mean that even though learning is occurring from aggregated labels (a total number of shots), the classifier will be able to identify each shot individually. The resulting detector will be able to run in real on a heavily constrained device and provide reliable detection of individual projectile events immediately after they occur, enabling interactive usage. The detector will avoid triggering on false positives (e.g. weapon dry firing, manipulation and transport, or falls onto hard ground).

We name this technique EDGAR for "Embedded Detection of Gunshots by AI in Real-time". This technique serves as the core component of a broader, fully automatable workflow for shot counter development, as illustrated in Figure 6.1 and detailed in the associated patent [19]. Upon receiving a client request for a new type of shot counter, the provider automatically generates a tailored shooting plan informed by the EDGAR technique (see Section 8). The client executes this plan and returns the collected sensor data to the provider, where it is consolidated into a larger shot database and used to generate the model for the desired firearm. Since the entire process, including shooting plan generation, does not require expert knowledge, it can be implemented as an end-to-end software solution deployable and executable by the client. This allows for a very fast development loop whose bottleneck lies solely in the data collection process performed by the client, whose length can be adjusted and optimised at shooting plan stage depending on the budget and time constraints given by the client themselves.

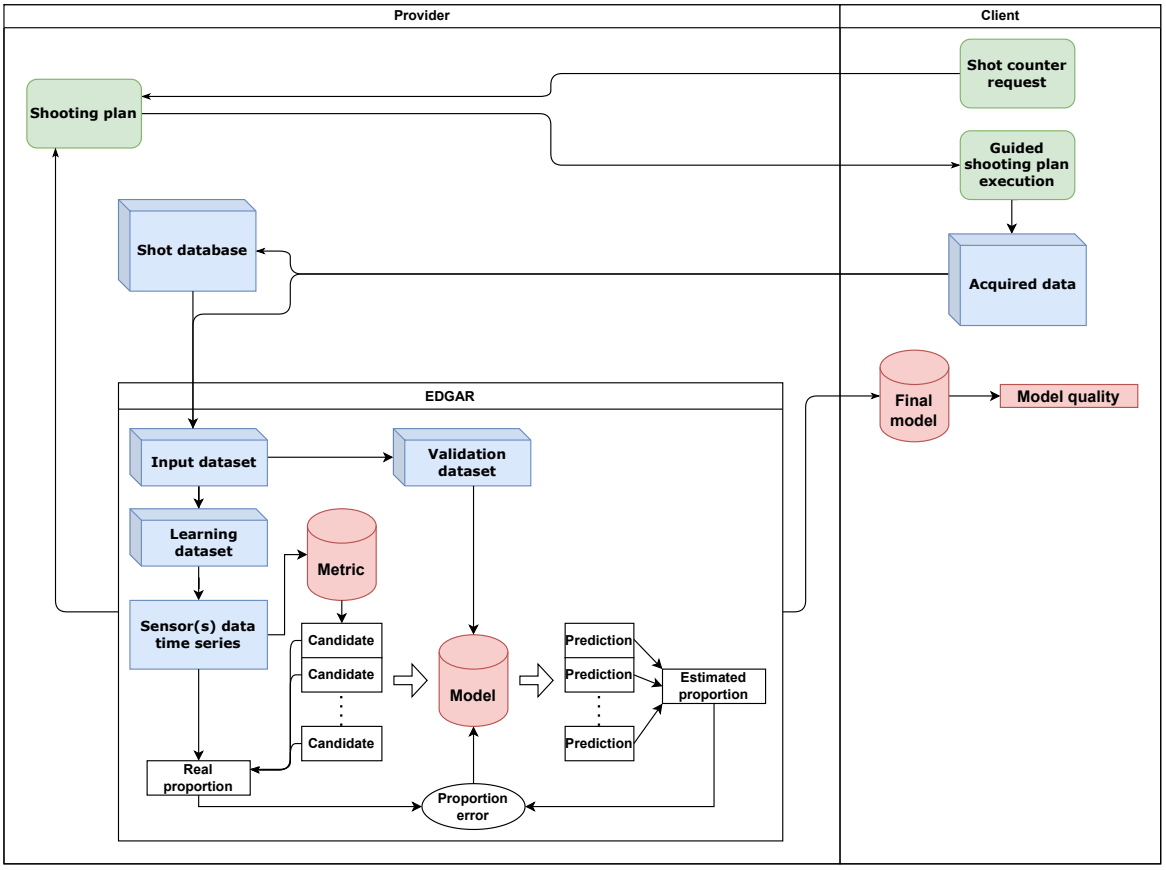


Figure 6.1: Flowchart of an EDGAR-driven shot counter development workflow. The process begins with a client request, after which a guided shooting plan specifying the required data is generated. The client executes the plan and returns the acquired sensor time series, which are integrated into a shot database. The data are then used to assemble input datasets, split into learning and validation subsets, which serve to train and validate a shot detection model using the EDGAR technique. The final model is delivered back to the client for immediate deployment and evaluation. The process is fully automatable and can operate without expert intervention.

6.2 EDGAR

In Section 1.3.4, we have seen that if we desire minimal human input, we can only obtain weak labels in the form of a total ammunition count per time series. The key to best leverage the information in our database is thus to derive a technique which will enable us to exploit these weak labels as part of our training. This can be accomplished by finding a way to reformulate the counting problem into a category proportion problem. To do so while keeping a detector with suitable time and space constraints for embedded use, two hypotheses are made:

1. A shot has a finite and known maximum duration.
2. At least one sub-event of a shooting event can always be reliably distinguished from the background noise.

Hypothesis 1 is related to the length of the candidate windows that will be considered. Since it does not impose a strict upper bound on duration, a sufficiently large window length can always be chosen to satisfy this hypothesis. However, larger numbers will negatively impact the performance of the resulting detector. An appropriate window length can be derived from the minimum theoretical burst rate of the weapon.

Hypothesis 2 is reasonable and has been used in related works as early as Loeffler et al. [41]. This allows us to focus our candidate windows around the actual shot, as opposed to the common use of sliding windows of constant density. It is done by producing candidate windows only when a certain metric is satisfied, preventing useless computations during rest periods where the only input is background noise, while producing multiple candidates in quick succession during high activity. Additionally, an effective candidate selection method will ensure that actual shots are reliably aligned within their candidate windows.

Unlike the detector, the metric is not required to minimize the number of false positives, although reducing false positives further decreases computation time. It is important that the metric avoids false negatives, which would result in a valid candidate not being presented to the detector.

The choice of a good candidate metric will depend on the nature of the input signal. For the most common input signal of accelerometer time series, we suggest the use of a rolling average on the instantaneous accelerations squared:

$$m[t] = \frac{1}{w} \sum_{i=-w/2}^{w/2} (a[t+i+o])^2, \tag{6.1}$$

where $a[t]$ is the input acceleration time series, w is a positive integer hyperparameter representing the size of the metric window, and o is an optional integer offset which can be applied to shift the position of the metric relative to the input signal. This is an adaptation of the shock detection metric used successfully by the baseline classical technique through Equation 3.1. Note that this can be computed efficiently in a rolling manner by considering that:

$$m[t] = m[t-1] + \frac{1}{w}(a[t + \frac{w}{2} + o])^2 - \frac{1}{w}(a[(t - \frac{w}{2} - 1) + o])^2. \tag{6.2}$$

The input signal and metric are illustrated in Figure 6.2. It can then be compared against the high (T_H) and low (T_L) thresholds, whose values are hyperparameters of the model. A candidate would, for example, only be generated when the signal dipped below T_L and increased above T_H.

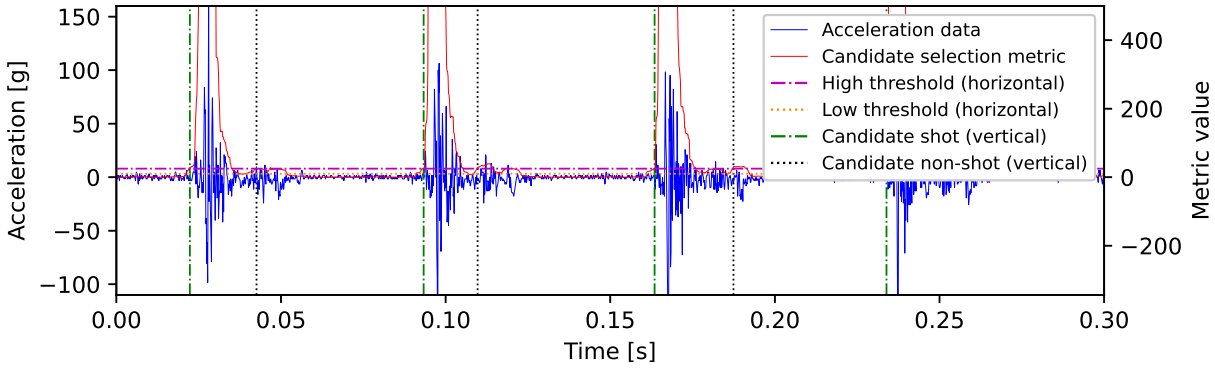


Figure 6.2: Example of candidate generation on an ideal input.

Shot candidates can then be generated accordingly as shown in Figure 6.3. Each one defines a slice of the input series of a fixed size determined according to Hypothesis 1 at most. In practice, experiments have shown that similar or better classification performance can be obtained with a significantly smaller input vector (see Section 6.4.1). A smaller input vector has a major benefit regarding inference-time performance.

Let N be the number of categories of inputs to classify. The classification categories can be chosen arbitrarily, as long as their number of occurrences can be known for each input series. In the simplest case, a database could be classified as *non-shot* or *shot*, leading to $N = 2$. A more complex example would be classifying between *non-shot*, *shot with live ammunition*, and *shot with training ammunition* leading to $N = 3$. Shot types could then be further subdivided according to whether or not they include a suppressor with $N = 5$.

Consider a time series containing a shooting sequence, with $\mathbf{c}_i$ ($i \in \{1, \ldots, N\}$) denoting the number of occurrences of each event category. Let its division into candidate slices be $\mathbb{X}$, where each element $\mathbf{x} \in \mathbb{X}$ is a fixed-length input vector extracted from the time series, corresponding to a candidate event identified by the metric. Each input vector represents a segment of the signal that may or may not contain a shot event. We can then define event proportions between the known number of events of a certain category and the total number of candidates as:

$$\mathbf{p}_i = \frac{\mathbf{c}_i}{|\mathbb{X}|}. \tag{6.3}$$

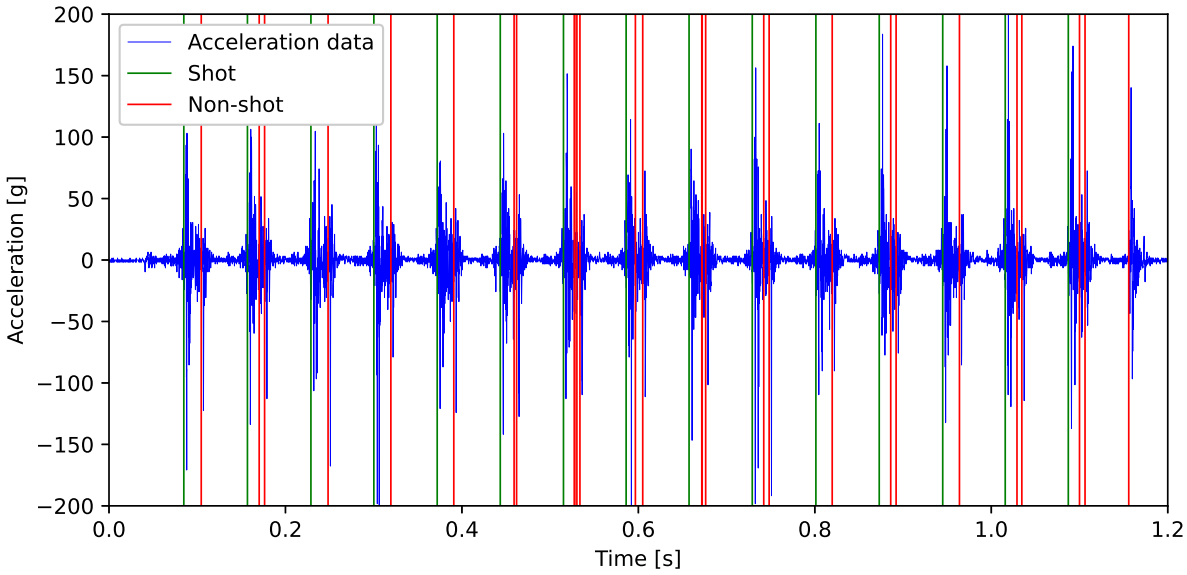


Figure 6.3: Candidate selection on a 15-shots burst with the Minimi 5.56. All vertical lines indicate a candidate generated from the metric. Candidates corresponding to a shot event were manually coloured in green, others in red.

A machine learning-based instance-level classifier $f_\theta(\mathbf{x})$ with trainable parameters θ can then be run on each input vector $\mathbf{x} \in \mathbb{X}$, attempting to classify it into its corresponding category $i \in \{1, \ldots, N\}$ (where N is the number of possible output categories). The output of $f_\theta(\mathbf{x})$ is a vector of N values, which are typically normalised within the model to represent class probabilities, for example by using a softmax function. The resulting class probabilities over a given time series can then be aggregated into proportions as follows:

$$\hat{\mathbf{p}} = \frac{1}{|\mathbb{X}|} \sum_{\mathbf{x} \in \mathbb{X}} f_\theta(\mathbf{x}). \tag{6.4}$$

The two proportions can then be compared using a label proportion technique such as that described by Tsai et al. [63] to tune the trainable parameters θ to optimal values. In particular, the relevant part of their loss function is:

$$\mathcal{L}_{\text{prop}} = -\sum_{i=1}^{N} \mathbf{p}_i log(\hat{\mathbf{p}}_i). \tag{6.5}$$

In our work, we add a second term, resulting in the Kullback-Leibler (KL) divergence:

$$\mathcal{L}_{\text{prop}} = -\sum_{i=1}^{N} \mathbf{p}_i log(\hat{\mathbf{p}}_i) + \sum_{i=1}^{N} \mathbf{p}_i log(\mathbf{p}_i) = D_{\text{KL}}(\mathbf{p} \,||\, \hat{\mathbf{p}}). \tag{6.6}$$

This formulation ensures that the loss is always zero for perfect predictions (i.e., when

$\hat{\mathbf{p}} = \mathbf{p}$). Without this adjustment, the minimal achievable loss has different values depending on the ratio of the different proportions. This makes the comparison of loss values across different inputs more meaningful, as well as their aggregation in an average loss over the dataset. Average loss can be viewed as a smooth metric of model quality, whereas the model accuracy can only take discrete values, corresponding to a discrete number of counting errors. This enables us to monitor the model learning evolution more precisely and employ early stopping techniques based on the average loss on the validation dataset. Comparative experiments have shown that its addition provides increased numerical stability and prediction accuracy.

In theory, this formulation does not guarantee that the model will learn to assign the correct labels to each class in the rare case where two or more classes always appear in exactly the same proportion across all training samples (that is, $\mathbf{p}_i = \mathbf{p}_j$ for all samples). In such a scenario, the model has no way to distinguish between these classes based solely on label proportions, since they are indistinguishable in the training data. However, this issue does not arise in practice, as the natural variability in class occurrences across samples ensures that class proportions typically differ. Furthermore, it is generally easy to obtain samples containing only non-shot data or samples with a single type of shot together with non-shot data. Such heavily skewed samples provide clear signals for the model and prevent this problem from occurring.

This technique allows us to train a model from our weak aggregated labels. Note that our proposed method only relies on a particular definition of the loss function, while not imposing any constraints on the type or structure of the model as long as it is able to train from such a loss function. A proposed model for embedded usage is described in Section 6.4, but can be replaced with other types of neural networks or machine learning techniques as long as it is able to train from such a loss function.

6.3 Minimum cycle time

An important factor limiting the performance of the technique, as previously described, appears when the metric generates two candidates very close in time. The model will receive two very similar input vectors, different slices of the same shot, which it is likely to both classify as being a shot, leading to false positives in the counting as a typical firearm is only able to shoot one projectile at a time in a cycle. However, we could leverage the known information of the minimum cycle time of the weapon to alleviate these false positives. For example, the modern Minimi 5.56 specifications allow a maximum firing rate of 950 rounds per minute (rpm), corresponding to a 63 ms average cycle time. Since this burst rate can be slightly exceeded in exceptional situations and individual cycle times might vary, it is necessary to incorporate a margin of error above that theoretical number. A value of 40 ms (corresponding to a theoretical cycle rate of 1500 rpm) has been used in our experiments.

While we could consider filtering out candidates at the metric level, at this step, we do not yet know which candidate the model would usually classify as a shot. Filtering at this

stage therefore risks removing the candidate closest to the actual shot ignition in favour of one before or after it. This could lead to non-detections and would diminish the model's ability to accurately localize the shot in the time series. A better approach is to apply this filtering after the network predictions. We first change successive *shot* predictions to *non-shot* if they fall within an exclusion window of the first prediction. Experimental data (see Section 6.10) show that this already leads to a significant reduction in error rate. In addition, we can avoid performing model inference on the ignored predictions, leading to an overall computational performance increase in deployment.

A drawback of implementing this as a post-processing step is that the model has no knowledge of it and remains penalized during training for the duplicate predictions. This might produce a model unnecessarily reluctant to predict shots. In other words, if the model were aware of this post-filtering, it would not hesitate to predict a shot for slightly offset candidates, knowing that it would not be penalized for duplicate predictions. We can accomplish this during training by implementing Algorithm 1, which corrects the predictions while ensuring the duplicates do not take part in the loss computation. This process is performed iteratively since a masked-out shot prediction must not itself create a mask. Redefining Equation 6.4 to substitute $f_\theta(\mathbf{x})$ by $f'_\theta(\mathbf{x})$ will then prevent backpropagation for the masked-out predictions. This leads to another significant reduction in error rate in experiments.

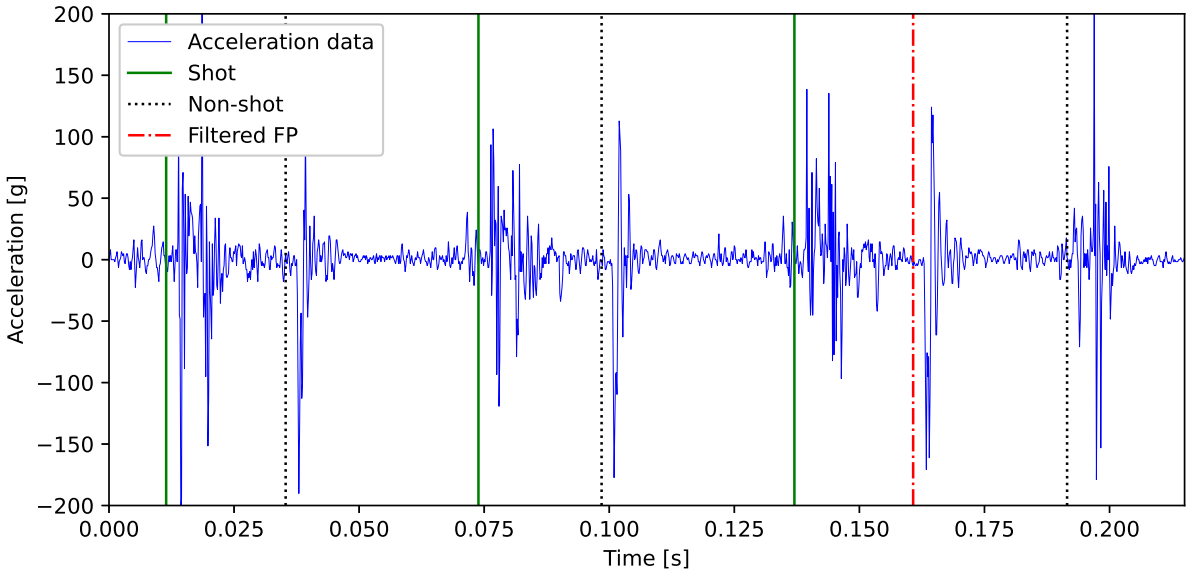


Figure 6.4: Burst of 3 shots with the Minimi 7.62. Each shot generates 2 or 3 candidates. For the third shot, a false positive (FP) of the second candidate is filtered out.

Algorithm 1 Remove duplicate predictions in training

Require: t_i: timestamp for each candidate $\mathbf{x}_i$.
Require: T_M: minimum event duration
Require: $f_\theta(\mathbf{x})$: instance-level classifier with trainable parameters θ,
Require: $\mathbf{e}$: vector corresponding to a non-shot prediction with maximum certainty

$\hat{\mathbf{y}}_i \leftarrow \max_{1 \leq k \leq N} f_\theta(\mathbf{x})_{i,k}$ ▷ Individual category predictions
$\mathbf{m}_i \leftarrow \top$ ▷ Mask to be computed
for $i = 1, ..., |\mathbf{x}|$ **do**
  $\hat{\mathbf{y}}'_j \leftarrow \begin{cases} \hat{\mathbf{y}}_j, & \text{if } \mathbf{m}_j \\ 0, & \text{if } \neg\mathbf{m}_j \end{cases}, j \in \{1, ..., |\mathbf{x}|\}$
  if $\hat{\mathbf{y}}'_i$ predicts a shot **then**
    $\mathbf{update}_j \leftarrow (t_j \leq t_i) \vee (t_j > t_i + T_M), j \in \{1, ..., |\mathbf{x}|\}$
    $\mathbf{m} \leftarrow \mathbf{m} \wedge \mathbf{update}$
  end if
end for
$f'_\theta(\mathbf{x})_j \leftarrow \begin{cases} f_\theta(\mathbf{x})_j, & \text{if } \mathbf{m}_j \\ \mathbf{e}, & \text{if } \neg\mathbf{m}_j \end{cases}, j \in \{1, ..., |\mathbf{x}|\}$
return $f'_\theta(\mathbf{x})$

6.4 Basic model structure

The basic technique and loss function improvements described in Sections 6.2 and 6.3 are independent of the specific trained model. Focusing on eventual implementation on heavily-constrained embedded devices, our initial network structure is a convolutional neural network (CNN) inspired by the one described by Santos et al. [43], who showed strong real-time performance on IoT devices in the related domain of fall detection. An example of our structure is shown in Figure 6.5 and totals only 33,242 parameters.

6.4.1 Input vector size

In Section 6.2, we saw that the input vectors of the network are candidates generated according to a metric. The size of these candidates is based on a hypothesis on the maximum possible duration of a shooting event. However, experimental results such as the one shown in Figure 6.6 have shown that the input vector size can be significantly reduced with similar or better model accuracy. While the potential accuracy improvement is marginal, reducing the input vector size leads to a proportional reduction in inference time. Naturally, the input size cannot be reduced below a certain threshold, which depends on the specific weapon and metric being used. Reducing the input size further causes critical information about the shooting event to be lost, leading to a catastrophic drop in prediction accuracy.

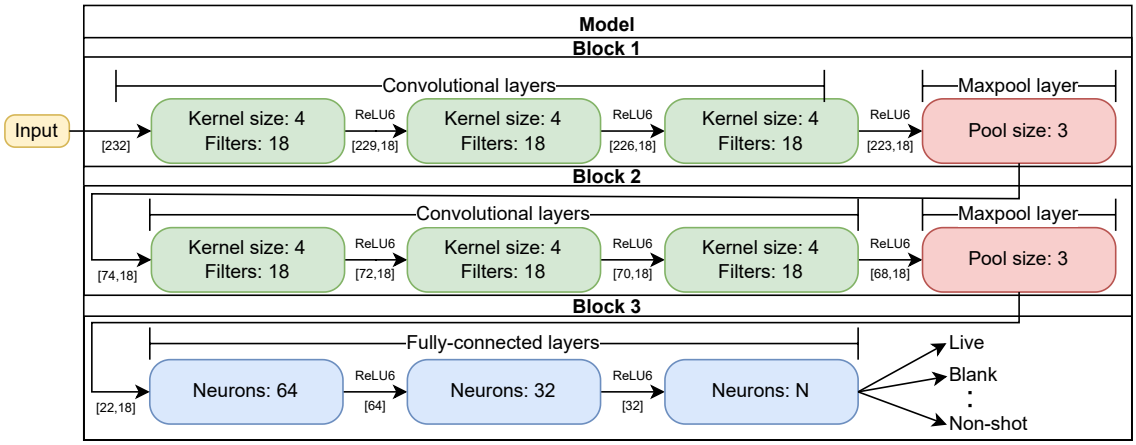


Figure 6.5: Example of the basic neural network structure for use with the EDGAR technique. The convolutions use neither strides nor padding. The input size and number of convolution channels depend on the application and the available computational budget.

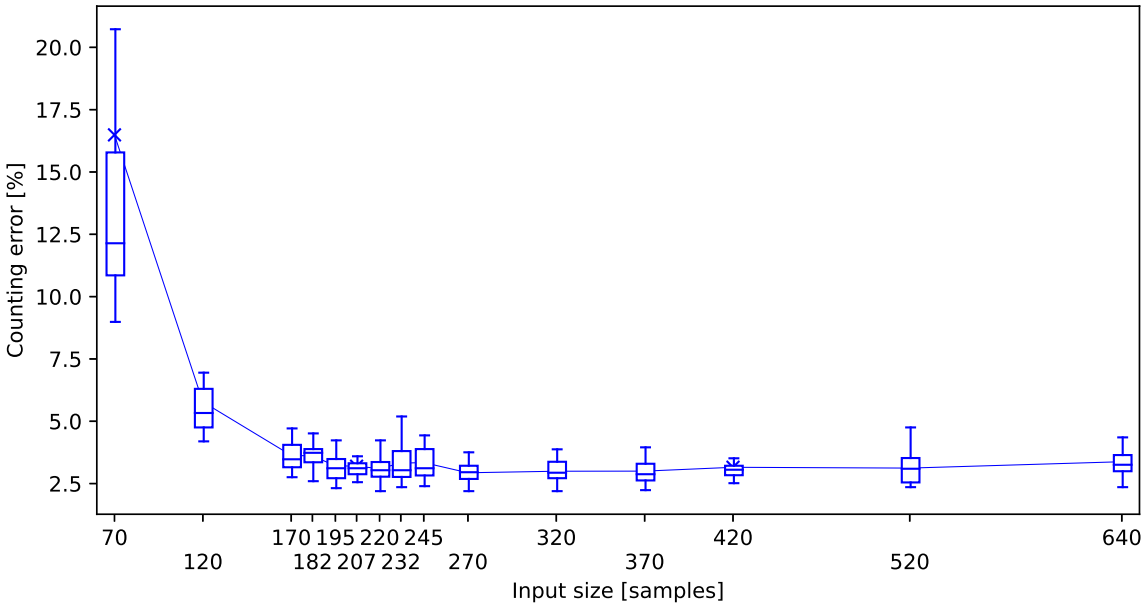


Figure 6.6: Effect of input size on prediction performance for Minimi 5.56. After a critical point, the shot is sufficiently captured, and a longer input size decreases performance while significantly increasing model size and inference time.

While the accuracy improvement with less information about the event might seem paradoxical, we speculate that this is due to the fact that the later parts of the firing cycle are less reliable due to variations in burst rate. Additionally, for larger input vectors the next shot and/or mechanical events may not always be present in the input vector. In this case, two shots that would otherwise be identical can result in different input vectors depending on subsequent events. This results in additional signal complexity. In principle, this contextual information could be leveraged in order to improve predictions (e.g. a candidate is more likely to be a shot if we can observe another shot occurring soon after). However, the networks suitable for use in constrained embedded use considered in this work are likely to be underparametrized. We speculate that the additional complexity renders the task too complex for the smaller network to handle, and that it benefits from focusing on the simpler task of predicting from the main part of the shooting event.

6.4.2 Convolution channels

The convolution layers account for the vast majority (over 95% in typical configurations) of the inference time. Once the input size has been minimised to capture the relevant part of the firing cycle, tuning the number of filters is the most important parameter to control the trade-off between inference time and counting error. Examples of this trade-off are given in Figure 6.7.

Supplementary figures which break down performance according to the number of filters are given in Section A.3.1, and show that around 16 filters are sufficient for most weapons to reach near-minimal counting error.

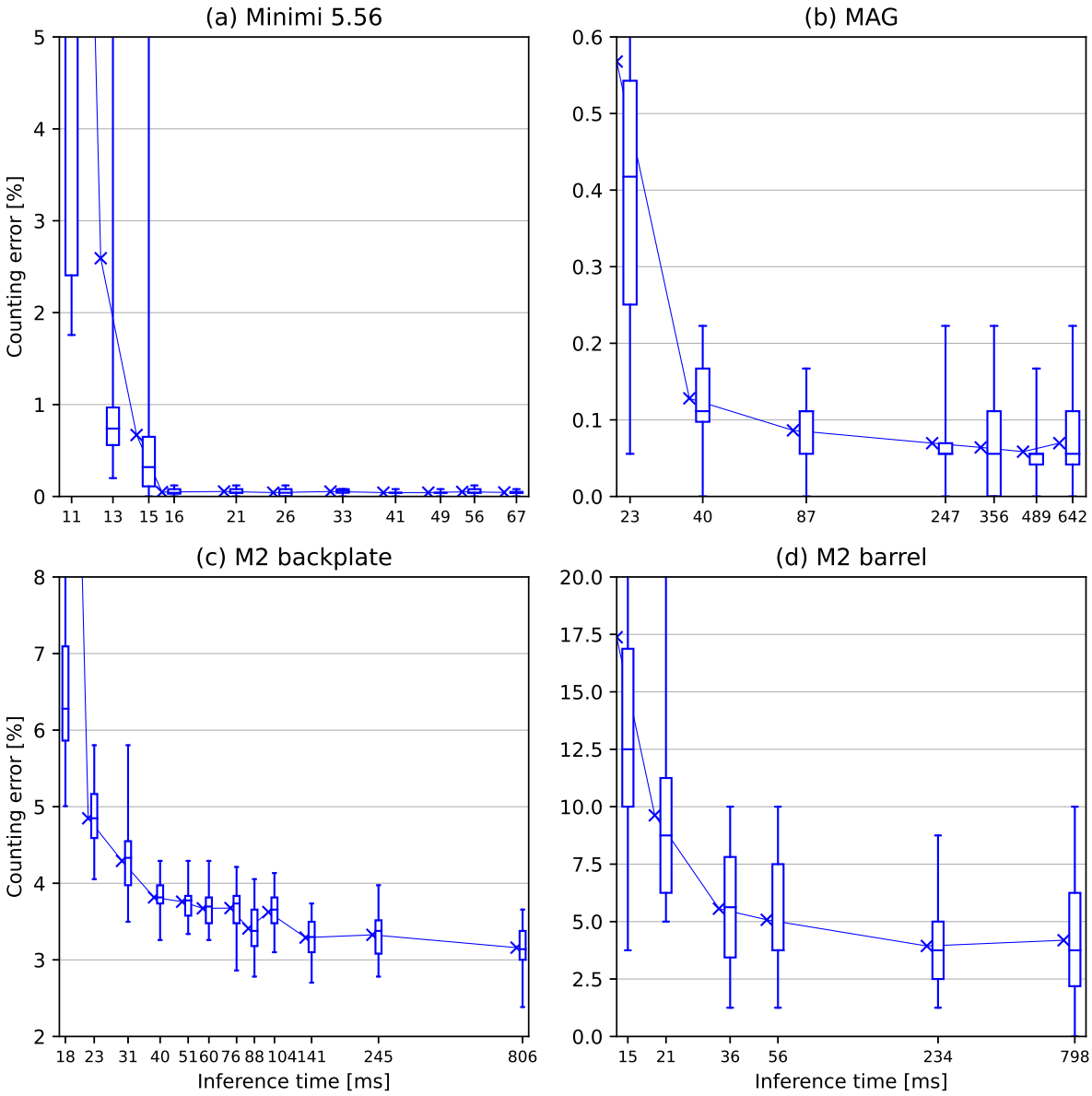


Figure 6.7: Counting error vs. inference time trade-off, obtained by varying the number of convolution channels.

6.5 Inference time constraints

In an embedded application dealing with a continuously incoming, unbounded time series, the metric described in Section 6.2 must be continuously computed on the device. When it detects a candidate, the associated input vector can be provided to the classifier which will output its corresponding class. To guarantee real-time interactive reporting, the total processing time per shot should be under the maximum burst rate of the associated weapon. Example burst rates are given in Table 6.1.

	Min. BR [rpm]	Max. BR [rpm]	Min. BR [ms]	Max. BR [ms]
Minimi	750	850	80	71
MAG	650	1000	92	60
M2	485	635	124	94
M134	2000	6000	30	10

Table 6.1: Minimum and maximum burst rate (BR) for the considered weapons, reported in rounds per minute, and the equivalent time per shot in milliseconds.

The total processing time comprises candidate selection, and the model inference for each candidate. The number of candidates generated per shot will depend on the weapon configuration. To gain insight into their distribution, we studied single bursts of over 10 shots in our learning sets and recorded the average time between candidates. Results are given in Table 6.2 and an example distribution in Figure 6.8 (supplementary figures for other weapons are given in Appendix D).

	Avg. T	Avg. $\bar{T}_{max}$	Avg. T_{buf}	P1 T	P1 $\bar{T}_{max}$	P1 T_{buf}
Minimi 5.56	73.9	73.2	123.6	67.3	66.7	112.6
Minimi 7.62	71.7	71.0	120.0	63.6	62.9	106.3
MAG	68.8	68.1	115.0	50.9	50.4	85.1
M2 backplate	56.3	55.7	94.2	33.4	33.1	55.9
M2 barrel	87.3	86.4	146.0	68.7	68.0	114.9
M134	18.2	18.0	30.4	17.2	17.1	28.8

Table 6.2: Average and first percentile candidate period, basic inference time budget ($\bar{T}_{max}$), and buffered inference time budget (T_{buf}) on our hardware target for the considered weapons. All values are in milliseconds.

From a given candidate period T, we obtain the average time available for model inference $\bar{T}_{max}$ by removing the preprocessing time as such:

$$\bar{T}_{max} = T(1 - F_s T_{CS}) \tag{6.7}$$

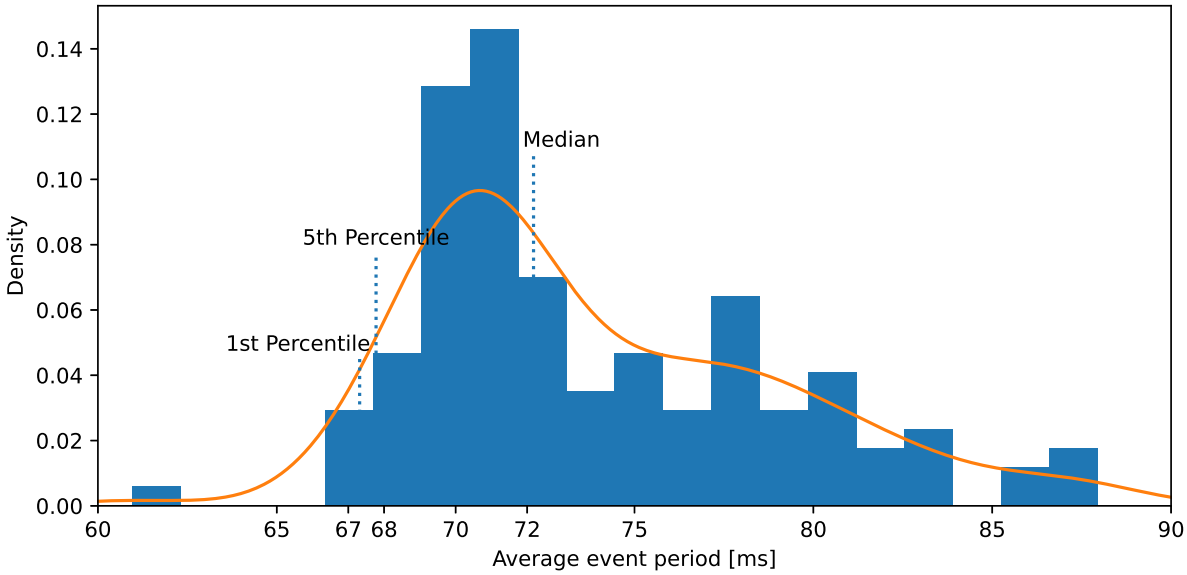


Figure 6.8: Filtered candidate density distribution on the learning set for Minimi 5.56.

where F_s is the sampling frequency in Hz and T_{CS} the time per sample required for candidate selection. For reference, our optimised implementation on a 64 MHz Cortex-M4 target requires 1.52 µs per sample, and data is acquired at 6400 Hz. This is approximately 1% of the available time. The computed budgets for this target are also reported in Table 6.2. Staying under the first percentile budget should ensure that input is processed, on average, faster than it arrives for all weapon configurations.

However, these values are obtained during sustained firing which cannot be maintained. After less than 30 seconds at this regime, infantry weapons risk overheating and require at least a barrel swap before firing can continue. If a sizeable buffer is available and near-real-time processing is acceptable, the allowed inference time can increase to T_{buf}:

$$T_{buf} = \bar{T}_{max} + \frac{QT}{T_F F_S B_S}, \tag{6.8}$$

where T_F is the maximum allowed sustained fire duration, B_S the number of bytes per sample, and Q the buffer size in bytes. Our reference embedded implementation uses 1 byte per sample and a buffer of 131,072 (0x20000) bytes. The values of T_{buf} for a maximum sustained burst of 30 seconds are also reported in Table 6.2.

6.6 Quantization

Embedded shot counter algorithms are most likely to be deployed on low-cost generic microcontrollers which have limited support for floating point operations. In order to make the most of the available computing power, one will usually need to proceed to a quantization of the network to operate on integer values. In addition, the use of smaller fixed-point integer representations (16 or 8 bits over the usual 32) will both reduce the size of the network and potentially improve inference time on platforms where more efficient low-level operations are available for smaller integer values.

Quantizing the network introduces rounding and saturation effects within the network which can lead to lower accuracy, making the choice of a proper quantization scheme important. We compared the implementations of the TensorFlow Lite and NNoM [64] libraries. While NNoM allowed for faster (up to 28%) inference time, it also led to an increase in error rate (up to 35%) and did not allow for quantization-aware training (QAT). In this technique, artificial quantization effects are simulated during the learning phase, enabling the network to adopt a structure that at the very least mitigates, and possibly leverages the effects of quantization. QAT further decreased the error rate with TensorFlow Lite by up to 19%. For the most time-constrained platforms, the number of convolution filters in the TensorFlow Lite version can be brought down to perform an accuracy/performance trade-off (see Figure 6.7). However, increasing the number of filters of the NNoM version did not significantly decrease the error rate. We thus proceeded with quantization by TensorFlow Lite for the rest of this study.

Additionally, we can try to adapt the basic structure of the network in order to make it more suitable to quantization. To do so, we replaced the usual ReLU activation layers with ReLU6. The difference between these functions is illustrated in Figure 6.9. Since ReLU activations are unbounded for positive values, their range can be arbitrarily wide. In contrast, ReLU6 limits activations to the $[0; 6]$ range. This allows us to use a fixed-point representation with more bits allocated to the fractional part, reducing quantization artefacts. This improves overall prediction accuracy, but most importantly improves the consistency of the results between the unquantized and quantized versions of the model (see Figures A.1 and A.2 for more details). Numerical instabilities during training are also reduced.

In testing, the maximum value of 6 has shown to produce the best median results, whereas ReLU2 has shown the highest best-case performance (see Figure A.3). Since 6 is not a power of 2, the full $[0; 6]$ range does not quantize efficiently, needing up to 3 bits for the integer part while not making full use of them. However, we speculate that even when the distribution of activation values uses the full range, only a low percentage of values actually fall near the bounds. Better performance can thus be achieved in those cases by only quantizing a 2-bit range around the average, saturating the outliers but leaving one additional bit to be used in the fractional part.

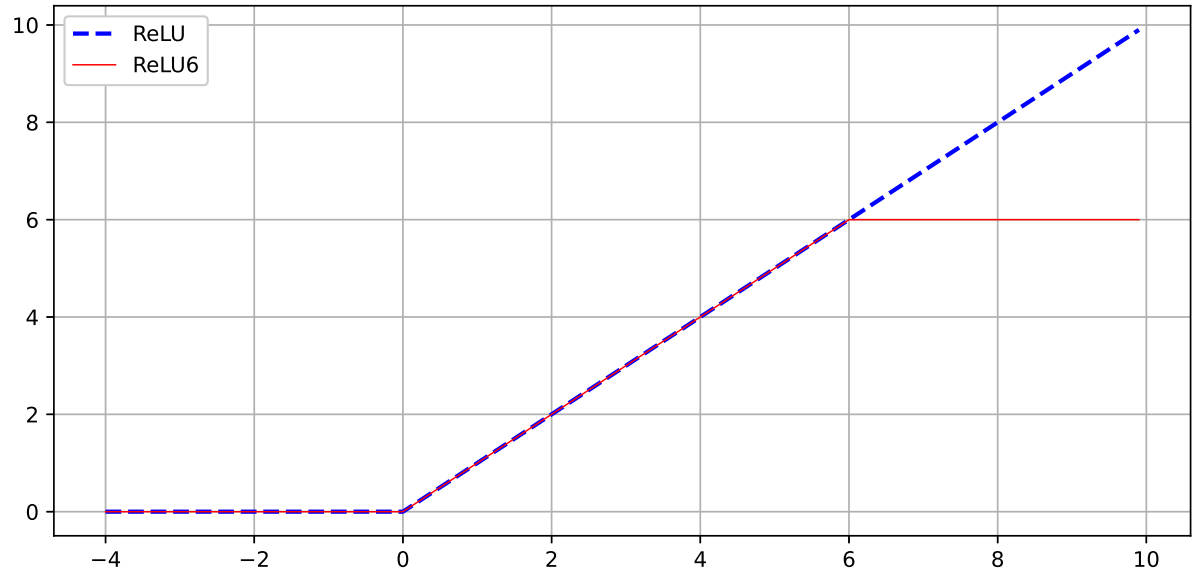


Figure 6.9: Comparison of ReLU and ReLU6 activations.

6.7 VAT

Following the good results of Tsai et al. [63], we apply Virtual Adversarial Training (VAT) as described by Miyato et al. [65]. As advised in the original paper, optimization is performed only through the perturbation size ϵ. The regularization coefficient α is fixed at 1, the finite difference factor ξ is fixed at 10^{-6}, and a single power iteration is performed ($K = 1$). Contrary to the findings of Laine et al. [66], our experiments have shown that the best results on our problem are obtained when the VAT loss is introduced as soon as possible (see Figure A.33 for more details). Thus, we do not include any ramping up to the VAT loss component.

6.8 Pre-filtering

The positioning of the sensor can have a major influence on the recorded signal. In particular, care should be taken to place it inside a part that moves in unison with the main body of the weapon during firing. The broad-spectrum impulsive nature of a shot will create significant wobbling in parts subject to resonance.

An interesting case study is that of the M2HB-QCB (illustrated in Figure 6.10), which features a removable barrel with a dedicated handle. When this weapon is used with multiple barrels, it can be interesting for the user to track shots fired by each barrel independently. The barrel handle is thus a natural place to house an electronic shot counter. However, this weapon operates by short-recoil, in which the barrel is a movable part which will slide back inside the weapon to perform the cycle. This creates violent accelerations at

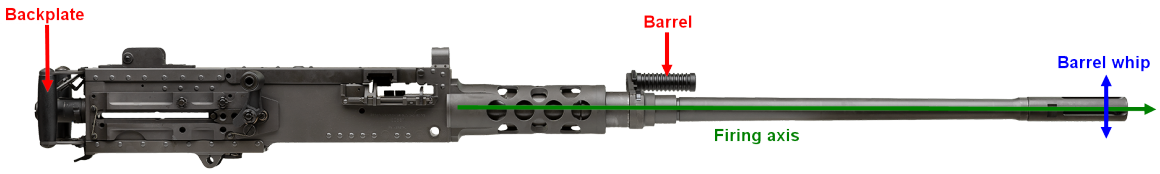


Figure 6.10: FN® M2HB-QCB Mk2.

the barrel. Figure 6.11 shows that the peak acceleration in the vertical axis is more than three times that of the fixing axis. This phenomenon is known as barrel whip. In addition, the barrel handle is linked to the barrel body through only one attachment point.

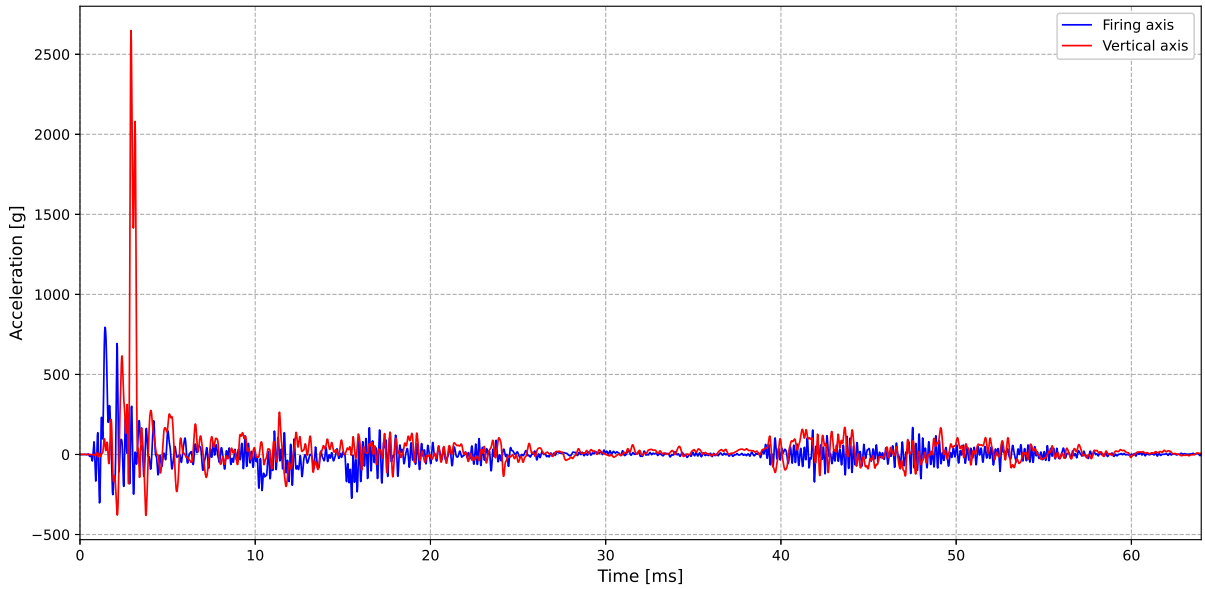
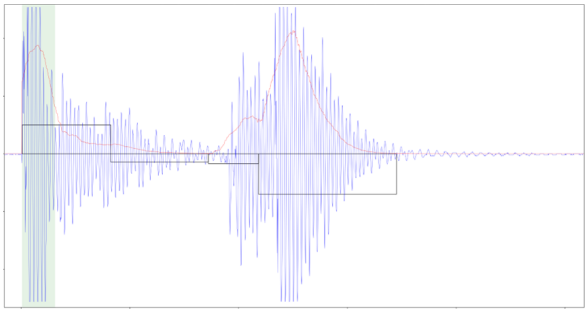


Figure 6.11: Acceleration data of a M2HB-QCB acquired with high-amplitude accelerometers. Data was acquired at 25 kHz. A low-pass filter with a 5 kHz cut-off frequency was applied.

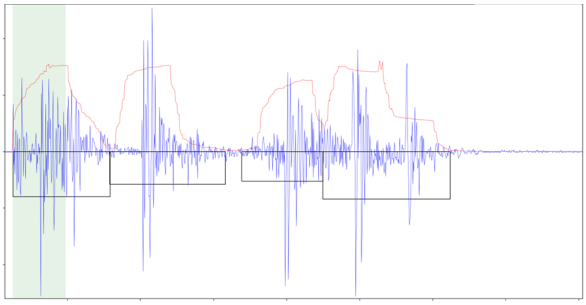
In combination, this creates a strong resonant wobbling at the level of the barrel, as illustrated in Figure 6.12. We observe that while classical shock detection is still able to distinguish the individual sub-events, they are drowned in a relatively low-frequency, high-amplitude wobble. A frequency analysis, shown in Figure 6.13, shows a significant spike around 535 Hz for the barrel-mounted sensor. While significant accelerations around this frequency are also measured at the backplate, they are more in line with the rest of the spectrum.

Through video analysis, we were able to confirm that this wobble is specific to the barrel handle and not the whole barrel. By tracking in a high-frequency video the movement of the barrel and the barrel handle against a reference backdrop (respectively features 4, 3 and 2 in Figure 6.14), we are able to measure the displacement shown in Figure 6.15, which

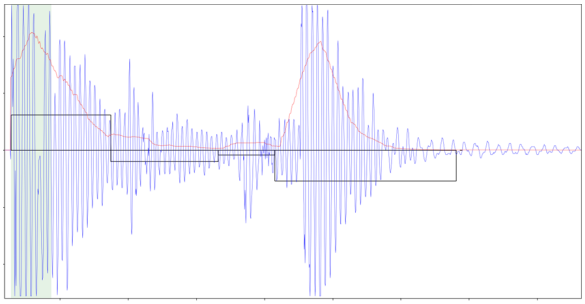
already indicates perturbations for the barrel handle. Making a frequency analysis of the second order derivative of this signal to obtain the acceleration, we observe a significant peak around the same 535 Hz frequency for the handle, but not for the barrel itself. This proves that the observed acceleration is not characteristic of the weapon itself, but rather results from the way the sensor is attached, and is therefore a parasitic effect.



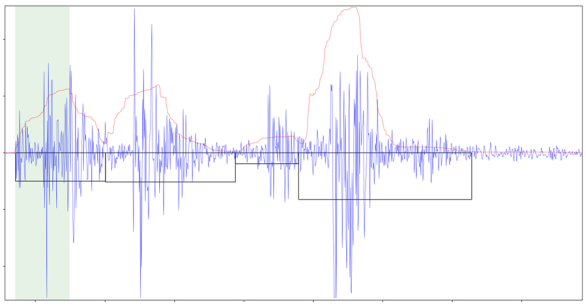
(a) Tripod - barrel



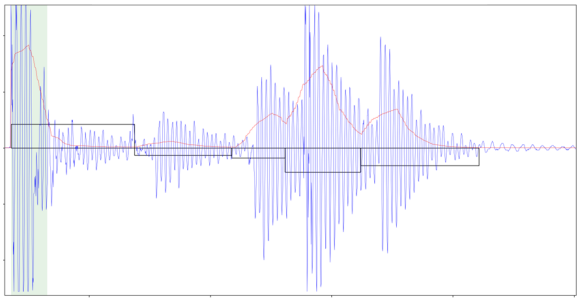
(b) Tripod - backplate



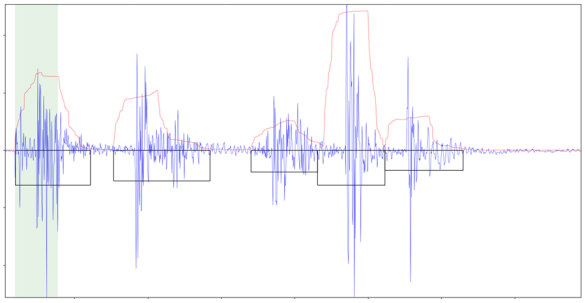
(c) Fixed mount - barrel



(d) Fixed mount - backplate

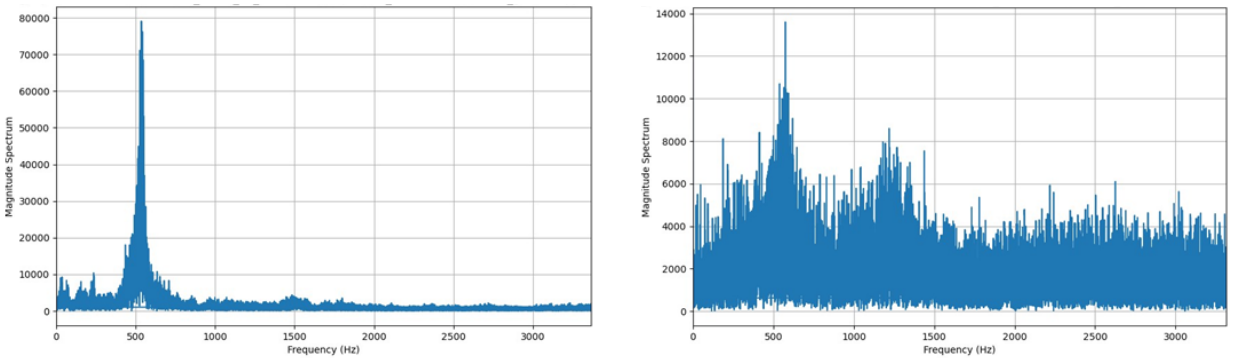


(e) Elastic mount - barrel



(f) Elastic mount - backplate

Figure 6.12: Qualitative comparison in M2 shot acceleration between a barrel-mounted and a backplate-mounted sensor across three different weapon setups. Data is interpreted through classical shock detection (as described in Section 3.1.1).



(a) Barrel (b) Backplate

Figure 6.13: Frequency analysis of a single shot of M2HB-QCB with a barrel-mounted sensor and a backplate-mounted sensor.

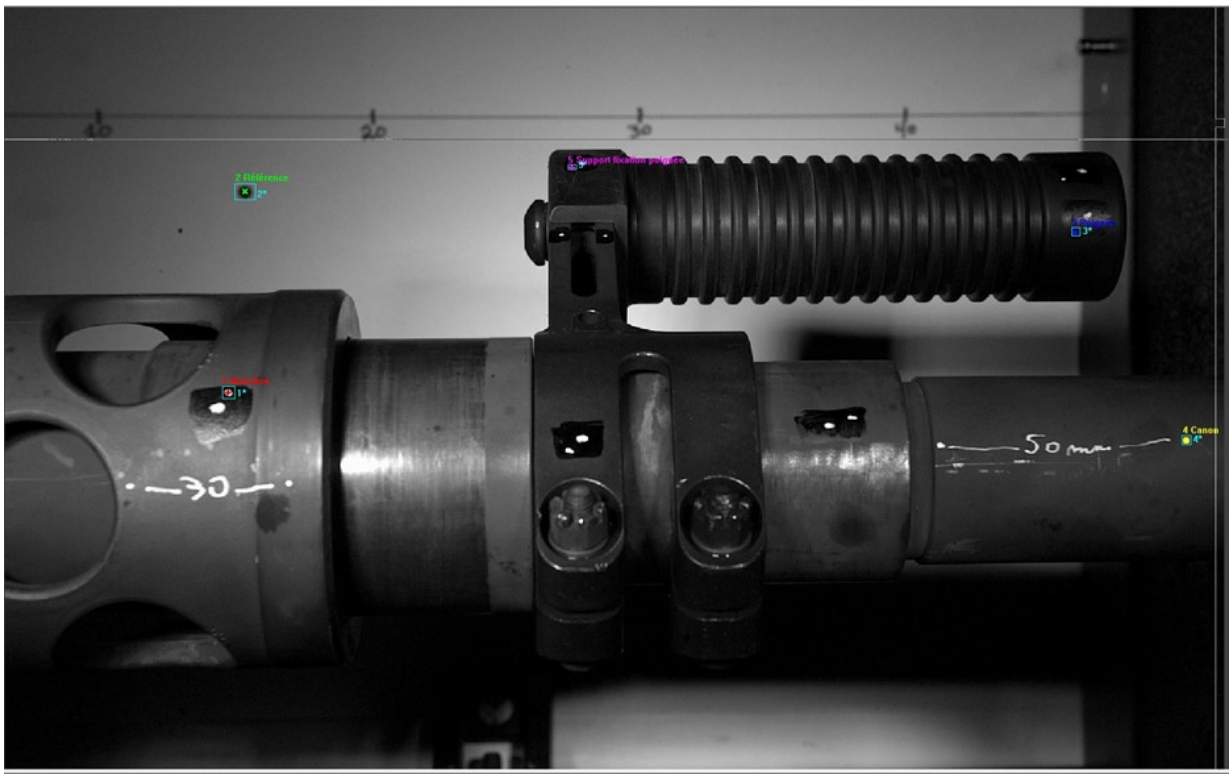


Figure 6.14: Single frame from a high-speed video of a M2HB-QCB shot, with 5 features being tracked visually.

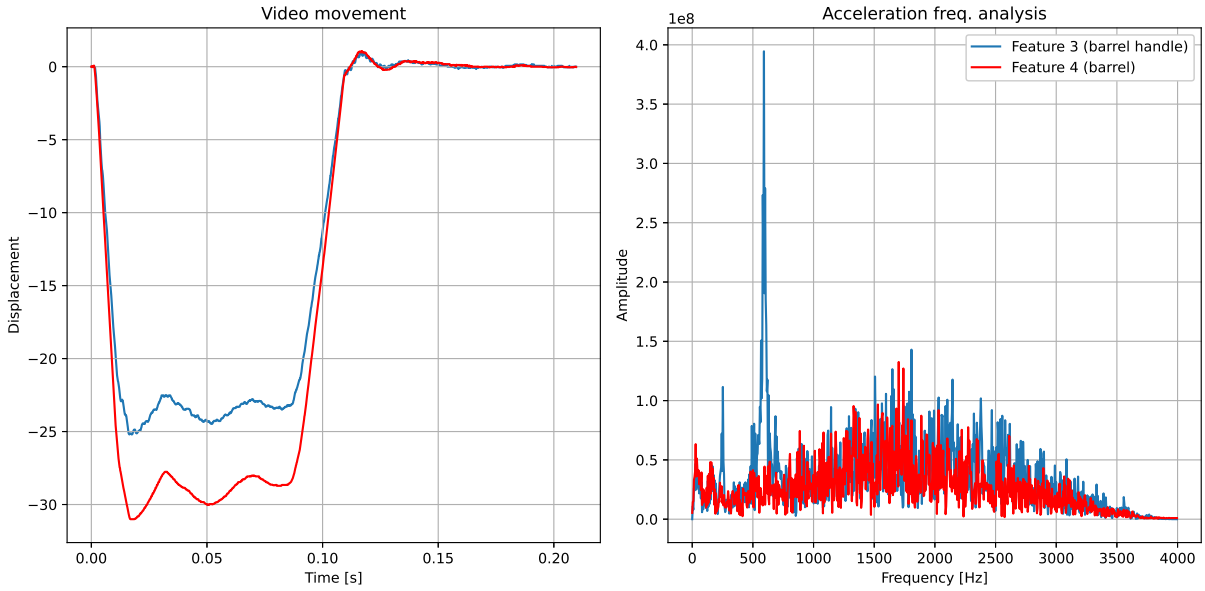


Figure 6.15: Tracking data of the barrel and the barrel handle from a high-speed video of a M2HB-CQB shot.

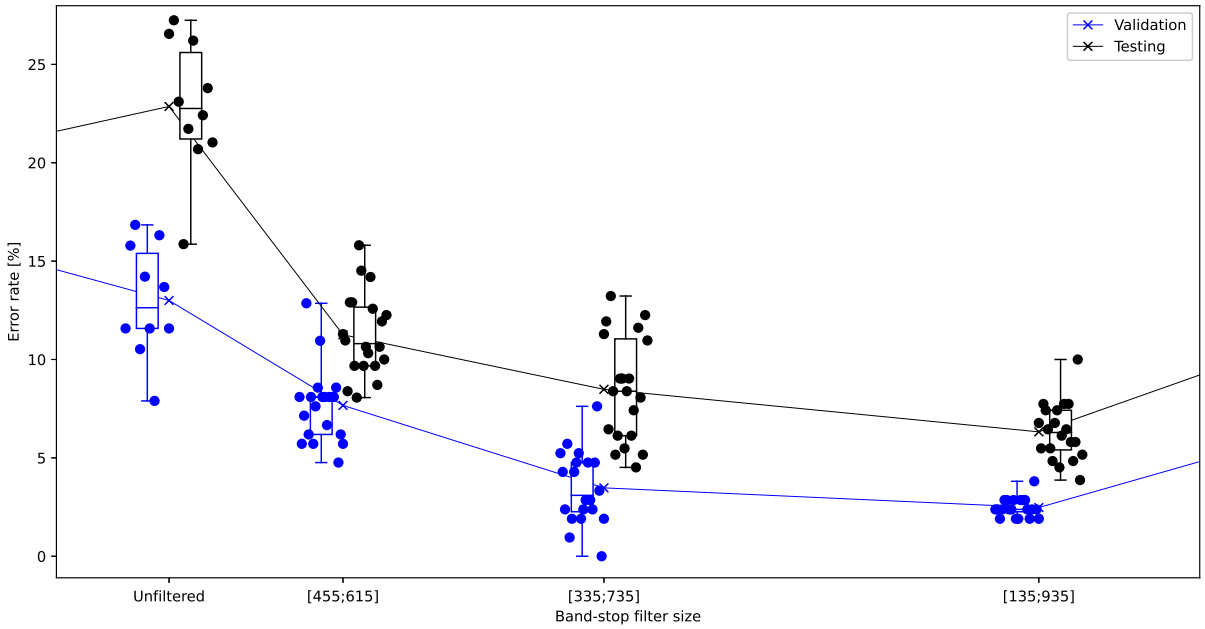


Figure 6.16: Performance of the EDGAR technique for a barrel-mounted M2HB-QCB sensor with either unfiltered data, or different width of band-stop filters around 535 Hz.

As shown in Figure 6.16, learning from the unfiltered data is ineffective, with a median counting error of 23% on the test dataset. Implementing a band-stop filter around the parasitic 535 Hz significantly improves performance. The best results are obtained with a relatively wide filter between 135 Hz and 935 Hz, reducing the error rate to 6% with the same input dataset. When considering real-time embedded usage with an unbounded time series, a time-domain implementation of the filter is preferred. We chose to implement an 81-tap Finite Impulse Response (FIR) filter with a Blackman window. The result is the signal illustrated in Figure 6.17.

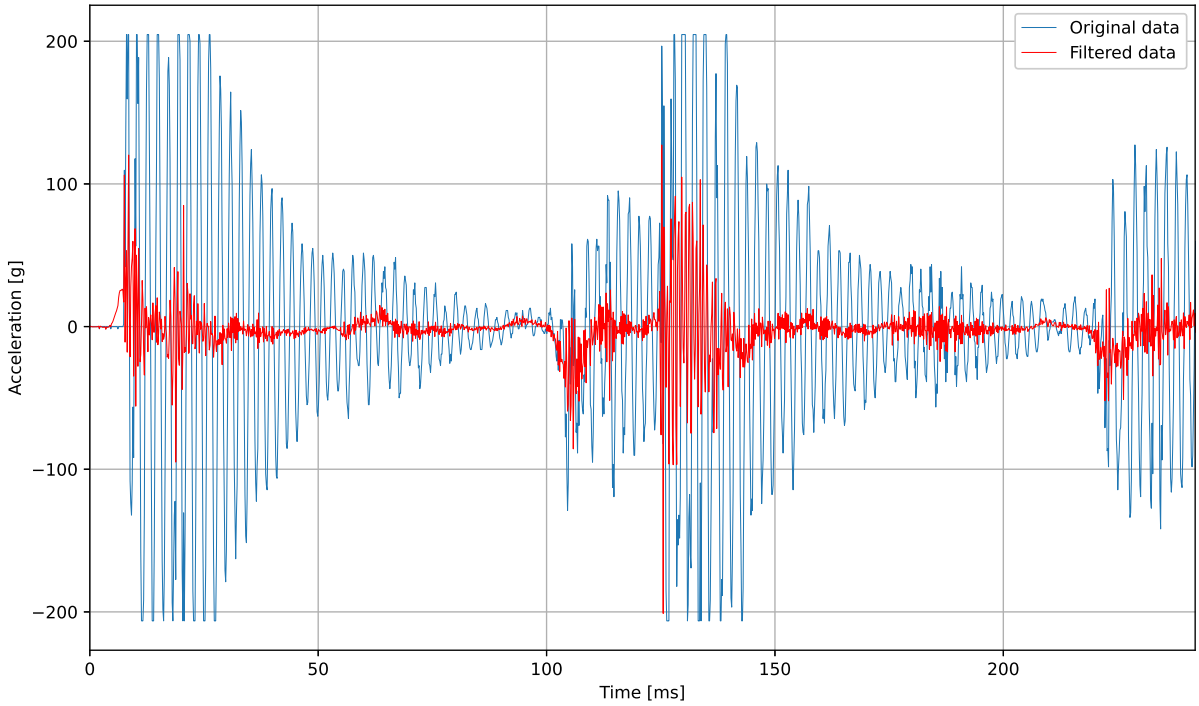


Figure 6.17: Recovered signal after filtering, for an ongoing burst of 2 shots.

The filter coefficients are adjusted to take into account the actual sampling frequency of the accelerometer, as measured during calibration in Section 4.2.2. This leads to a significant performance improvement as shown in Figure 6.18. During training, each series is processed using the measured actual sampling frequency stored as metadata. For inference on real devices, the FIR coefficients are recomputed on the target during device calibration.

Enabling this feature brings down the median counting error on the test set from over 15% to less than 1%, with some networks achieving a perfect score (see Figure A.13 and Figure A.15).

To minimise the impact on real-time performance, we used the CMSIS implementation of FIR filtering with 32-bits floats, which optimises the implementation through usage of SIMD (Single Instruction Multiple Data) instructions and block memory accesses.

In the same idea, this filtering should be integrated with candidate selection in a way

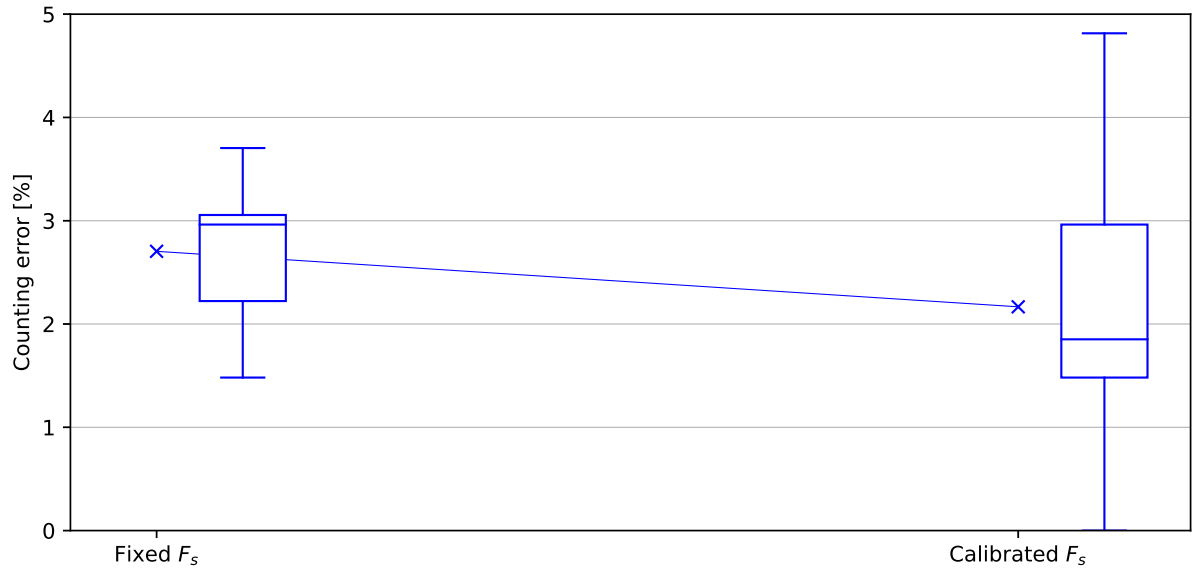


Figure 6.18: Difference between fixed and calibrated FIR filter coefficients for M2 barrel.

that favours batched processing and loop unrolling. Additionally, all circular buffer sizes are powers of two, allowing for efficient indexing by considering that

$$(x+i) \mod 2^n = (x+i)\&(2^n-1), \tag{6.9}$$

where & denotes the bitwise AND operation.

To favour batch processing and grouped memory accesses, data is first accumulated in a temporary buffer. It is then evaluated in fixed batch sizes in the following steps:

1. Compute the 32-bit float filtered version of the current batch.
2. Round all filtered values to 8-bit signed integers.
3. Compute the candidate metric for all points in the batch, using Equation 6.2.
4. Check all values for a satisfaction of candidate selection or end of signal criteria, and evaluate accordingly.

Increasing the batch size shows diminishing returns, although the decrease is non-linear as some sizes will not align well for memory operations. For our application, a batch size of 360 was chosen, identical to the input vector size. A similarly implemented but slower function that handles variable data is used only at the end of an acquisition, which might not provide a full batch.

6.9 Pretraining

Data from different firearms is expected to still present similarities. For example, the Minimi 5.56 and Minimi 7.62 feature similar mechanical designs and firing cycles, while using a different ammunition type. Meanwhile, although the overall design differs, the Minimi 7.62 and MAG share a similar firing cycle and ammunition. We can thus assume that this data may be helpful to the learning process.

To leverage it, we perform a first training phase on the combined datasets.[1] Validation during this phase is also performed on the combined validation datasets. The resulting weights are then used as initial weights for the specialised training phases (floating-point and quantization-aware trainings). For the discrimination problem (which will be discussed in Chapter 7), the pretraining datasets must first be processed using the best counting model derived for each individual dataset. This step requires additional groundwork on the implementation side.

Supplementary data presented in Section A.5 studies how the inclusion of different pretraining datasets affects inference performance. We find that in most cases, using the broadest pretraining dataset leads to the best performance, even when the considered firearms are very different in ammunition calibre and firing cycles. This suggests that a shared, learnable concept of "shot" exists across the firearms. However, as shown by Figure A.65, trying to learn without weapon-specific data leads to very poor performance. We also observe that the relative usefulness of pretraining remains constant for both small and large target-specific datasets.

Table 6.3 shows the effect of pretraining on the counting problem, and Table 6.4 on the discrimination problem. Pretraining proves to be very effective for the counting problem. Although the effect on the discrimination problem is smaller, it is still generally beneficial. In cases where pretraining is detrimental for either task, the performance decrease is minor. The main drawback of pretraining is the significant increase in training time, proportional to the amount of pretraining data. This is illustrated in Section 6.10.

	Validation dataset			Test dataset		
	No PT	PT	Diff.	No PT	PT	Diff.
MAG	1.12%	1.19%	-6.41%	1.49%	1.57%	-4.88%
Minimi 5.56	0.22%	0.11%	50.00%	3.29%	2.46%	25.26%
Minimi 7.62	0.62%	0.15%	75.00%	-	-	-
M2 backplate	4.72%	3.35%	29.10%	5.88%	3.43%	41.67%
M2 barrel	0.74%	0.19%	75.00%	3.7%	1.11%	70.00%
Avg.			**44.54%**			**33.01%**

Table 6.3: Counting error before and after pretraining (PT).

[1] No difference was found between using the target dataset candidate selection parameters for all pretraining datasets, and per-dataset parameters (see Figure A.64).

	Validation dataset			Test dataset		
	No PT	PT	Diff.	No PT	PT	Diff.
MAG	0.66%	0.63%	5.64%	1.31%	1.12%	14.51%
Minimi 5.56	0.77%	0.74%	4.58%	6.24%	5.22%	16.37%
Minimi 7.62	1.37%	1.27%	7.72%	-	-	-
M2 backplate	1.05%	0.87%	16.67%	3.57%	3.38%	5.39%
Avg.			**8.65%**			**12.09%**

Table 6.4: Discrimination error before and after pretraining (PT).

6.10 Breakdown of performance improvements

In Figure 6.19, we compare the performance of incremental improvements to the base method as discussed in Section 6.2. We start with the base technique. As we move to the right we successively add improvements, conserving all previously enabled ones. Hyperparameters are kept identical.

We first add our second term of Equation 6.6 in "Zero-Loss". While the resulting error rate is roughly unaffected, it reduces the number of non-converging networks, which are not included in the results for readability (as their counting error rate is 100%). We then replace ReLU activations with ReLU6 of Section 6.4. This has only a small positive effect on the counting error rate, however consistency between the unquantized and quantized predictions is improved (see Figure A.2). Post-filtering from Section 6.3 is then introduced, leading to a significant improvement in the Minimi 7.62, M2 backplate and MAG cases, and a neutral effect in the others. Unfortunately, training time increases because this step is computed during training in order to continuously evaluate the model's quality. Adding our implementation of Algorithm 1 in "Learned P-F" has a large positive effect in all cases. Enabling VAT can have either a positive or a negative impact on the counting error. However, as discussed in Section 6.7, it is expected to also enhance out-of-distribution performance which is not represented here. VAT significantly increases training time. Finally, pretraining is introduced with a substantial positive effect on Minimi 7.62 and M2 barrel, and a smaller but significant positive effect on Minimi 5.56 and M2 backplate. Pretraining for MAG shows a small negative effect, which can be explained by the already low counting error, and the relatively larger base dataset. Unfortunately, training time is largely increased with pretraining, proportionally to the size of the pretraining dataset.

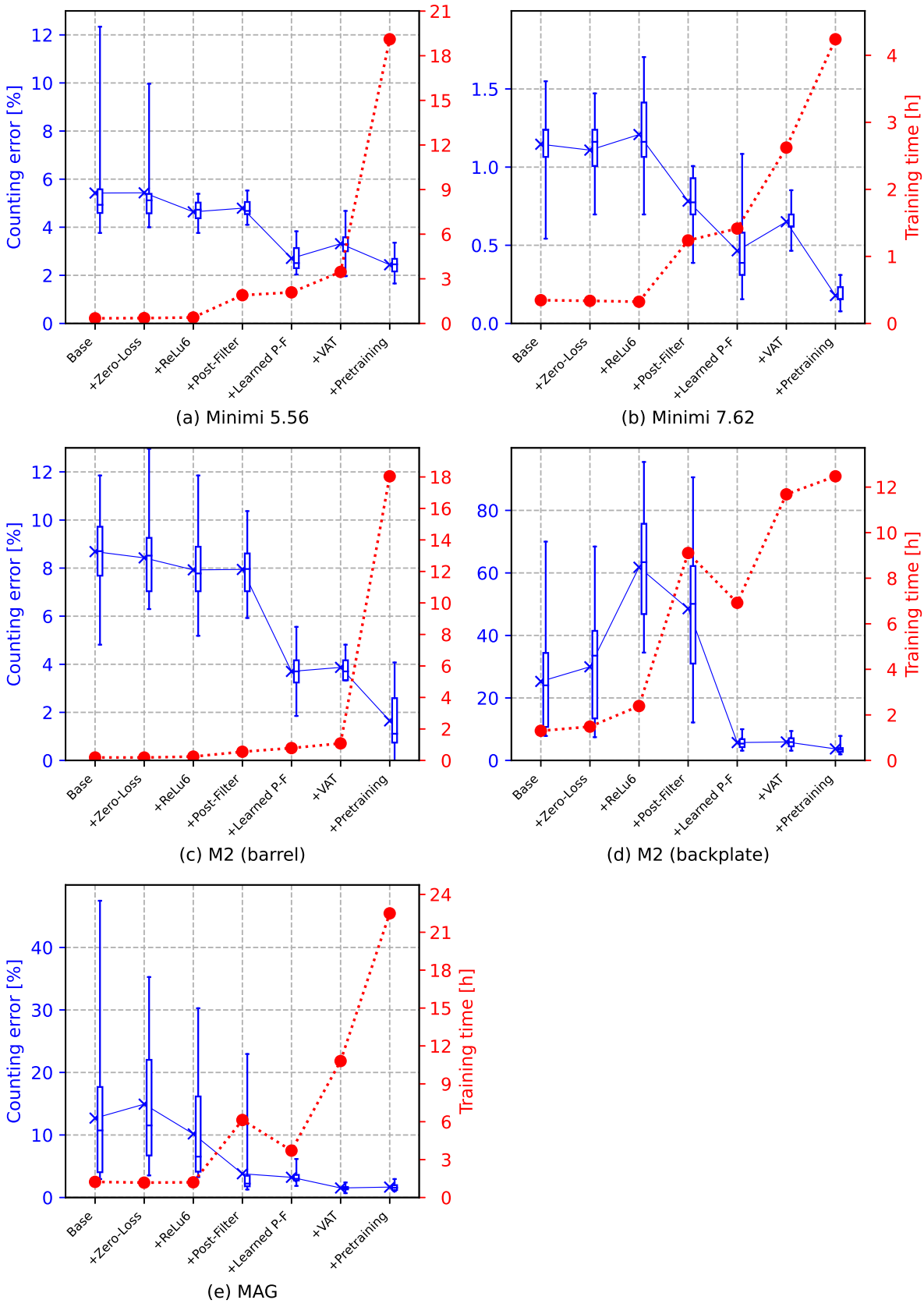


Figure 6.19: Effects of incremental improvements. Each box represents the counting error rate of 20 randomly initialized models on the test set (except Minimi 7.62, which uses the validation set). Non-converging models are excluded.

6.11 Hyperparameter tuning and orthogonal array testing

Our initial approach to hyperparameter tuning involved an extensive search in which we first varied each parameter individually, allowing us to identify those with significant effects, those that could present multiple local minima, and to determine good initial values and search areas. Details on this search are provided in Section A. We will see in Table 9.1 that optimal performance for very different firearms can be reached with similar hyperparameters. Using these as a starting point, tuning of future models can be performed through only a local search around these values. When narrowing our search, we can account for multivariate effects while maintaining a manageable number of experiments by using orthogonal array testing.

By design, the candidate metric window size (w) and selection thresholds (T_L and T_H) can influence each other. The M2 barrel case introduced an additional need for band-stop filtering (see Section 6.8), which could also influence the candidate selection process. This filtering introduces two supplementary hyperparameters: the number of taps (which should be kept as low as possible to minimise computation time) and the band-stop width. Adjusting the thresholds evenly leaves us with four different variables to optimize. For each variable, we defined three values to be tested as reported in the first part of Table 6.6. Performing a full factorial analysis would require $3^4 = 81$ experiments. However, we can reduce this to the 10 experiments detailed in Table 6.5. Experiment #0 serves as a baseline, with our best initial guess. Experiments #1 to #9 are designed as an orthogonal array. To score an experiment, we train 20 randomly initialised networks and report their median and best number of errors on the validation set. The score used is the sum of these two values.[2]

For each variable value in Table 6.6, we add the scores of all experiments where this variable has this particular value.[3] By design, this sum will have exactly one contribution from each possible level of all other variables, allowing their effects to compensate and isolate the effect of the considered variable. Looking at the lowest score for each variable points us towards the optimal value for each parameter. The combination of these best settings, reported as "T" in Table 6.5, indeed provides the best score. It can then serve as a new baseline if necessary. In this case, the obtained performance was already sufficient to stop further hyperparameter tuning after only this small amount of training experiments.

[2]The use of the best value can be subject to outliers and should be used with care. Its inclusion here is viewed as a tie-breaker, and one should consider giving it a lower weight compared to the median value. However, this would make no difference in these particular experimental results, and was thus kept out for the sake of simplicity.

[3]For example, the score for "High" (81) FIR taps is the sum of experiments #7, #8 and #9. Similarly, the score for "Low" w is the sum of experiments #1, #5 and #9.

#	FIR taps	Filter width	Thresholds	w	Median	Best	Score
0	Neutral	Neutral	Neutral	Neutral	9	4	13
1	Low	Very low	Low	Low	35	19	54
2	Low	Low	Neutral	Neutral	32	21	53
3	Low	Neutral	High	High	9	6	15
4	Neutral	Very low	Neutral	High	24	11	35
5	Neutral	Low	High	Low	19	1	20
6	Neutral	Neutral	Low	Neutral	11	5	16
7	High	Very low	High	Neutral	16	5	21
8	High	Low	Low	High	4	2	6
9	High	Neutral	Neutral	Low	5	2	7
T	High	Neutral	High	High	4	1	5

Table 6.5: Experimental plan and results of candidate selection parameters tuning for M2 barrel using an orthogonal array.

Parameter	Setting	Value	Score
FIR taps	Low	21	122
	Neutral	41	71
	High	81	**34**
Band-stop width [Hz]	Very low	[435;635]	110
	Low	[335;735]	79
	Neutral	[135;935]	**38**
Thresholds $\{T_L, T_H\}$	Low	{54,69}	76
	Neutral	{72,91}	95
	High	{90,114}	**56**
w [ms]	Low	3	81
	Neutral	5	90
	High	10	**56**

Table 6.6: Considered variables values and associated score of candidate selection parameters tuning for M2 barrel using an orthogonal array.

The M134 gave us another interesting test case. Due to the significantly higher rate of fire of this weapon, in addition to the candidate selection parameters we expected the input size ($|x|$) and the minimum cycle time (reported as Min. Δt for short) to require significant adjustment. An initial rough guess was made from looking at a few example inputs, and an experimental plan was designed accordingly. The results are provided in Table 6.7 and Table 6.8. Since experiment #6 yielded a perfect result on the validation set, these settings could be chosen without further adjustment of other hyperparameters.

#	$\|x\|$	Min. Δt	Thresholds	w	Median	Best	Sum
0	Neutral	Neutral	Neutral	Neutral	9	2	11
1	Low	Low	Low	Low	14	6	20
2	Low	Neutral	Neutral	Neutral	1	0	1
3	Low	High	High	High	4	3	7
4	Neutral	Low	Neutral	High	3	2	5
5	Neutral	Neutral	High	Low	4	1	5
6	**Neutral**	**High**	**Low**	**Neutral**	**0**	**0**	**0**
7	High	Low	High	Neutral	12	5	17
8	High	Neutral	Low	High	2	2	4
9	High	High	Neutral	Low	11	7	18

Table 6.7: Experimental plan and results of candidate selection parameters tuning for M134 using an orthogonal array.

Parameter	Setting	Value	Score
$\|x\|$ [samples]	Low	60	28
	Neutral	110	**10**
	High	200	39
Min. Δt [ms]	Low	7	42
	Neutral	10	**10**
	High	13	25
Thresholds	Low	440/360	**24**
	Neutral	550/450	**24**
	High	660/540	29
w [ms]	Low	1	43
	Neutral	2	18
	High	3	**16**

Table 6.8: Considered variables values and associated score of candidate selection parameters tuning for M134 using an orthogonal array.

Chapter 7

Extending towards ammunition discrimination

7.1 Learning from weak labels

In shot counters, ammunition type discrimination is generally the second most important feature after accurate counting capabilities. Once sufficient counting accuracy is reached, we can attempt to include ammunition type prediction into the network. The loss function as described in Section 6.2 and the corresponding network described in Section 6.4 can be trivially extended from counting to discrimination by increasing the number of output classes (N) considered.

Until now only two categories, *non-shot* and *shot* ($N = 2$), were considered. In this chapter, we study an example of ammunition discrimination that classifies between *non-shot*, *shot with live ammunition* (or *live*), and *shot with training ammunition* (or *blank*) leading to three categories ($N = 3$). While discrimination categories can be considered, we saw in Section 3.2 that the live/blank case is the most popular application. Other cases such as the use of a suppressor will still be considered in later sections. Although we did not get an opportunity to study it, the case of "4B1T" ammunition deserves mention. This type of linked ammunition comprises tracer rounds (T) after every 4 normal live rounds (B). As such, the two can hardly be studied separately, making this a case for which the weakly-labelled approach is particularly suited.

The previously introduced loss function and corresponding neural network can be easily extended from counting to discrimination by increasing the number of classes (N) considered. However, as the *non-shot* class becomes one of many, the loss function increasingly emphasises the discrimination accuracy over the counting accuracy, which is counterproductive for the general shot counting problem. Indeed, information on weapon usage and degradation from accurate shot detection is far more important than that provided by discrimination. To address this issue, we introduce a new hyperparameter α, to balance these competing objectives:

$$\mathcal{L}_{\text{prop}} = \mathcal{L}_{\text{counting}} + \alpha\mathcal{L}_{\text{discrimination}}, \alpha \in]0, \infty[. \tag{7.1}$$

Assuming that $\mathbf{p}_1$ represents the proportion of *non-shot* candidates, we define new sets of proportions to recover the original counting problem by summing the proportions of all *shot-type* classes.

$$\mathbf{p}' = (\mathbf{p}_1, \sum_{i=2}^{N} \mathbf{p}_i), \hat{\mathbf{p}}' = (\hat{\mathbf{p}}_1, \sum_{i=2}^{N} \hat{\mathbf{p}}_i). \tag{7.2}$$

The counting loss is then defined using these reduced proportions:

$$\mathcal{L}_{\text{counting}} = -\sum_{i=1}^{2} \mathbf{p}'_i \log(\hat{\mathbf{p}}'_i) + \sum_{i=1}^{2} \mathbf{p}'_i \log(\mathbf{p}'_i). \tag{7.3}$$

The discrimination loss follows the original formulation:

$$\mathcal{L}_{\text{discrimination}} = -\sum_{i=1}^{N} \mathbf{p}_i \log(\hat{\mathbf{p}}_i) + \sum_{i=1}^{N} \mathbf{p}_i \log(\mathbf{p}_i). \tag{7.4}$$

The value of α can then be adjusted, revealing a Pareto front between counting and discrimination accuracy, from which an optimal balance can be selected, as illustrated in Figure 7.1. An alternative formulation that excludes the *non-shot* proportion from the discrimination loss produces a similar result, though over a different range of α values.

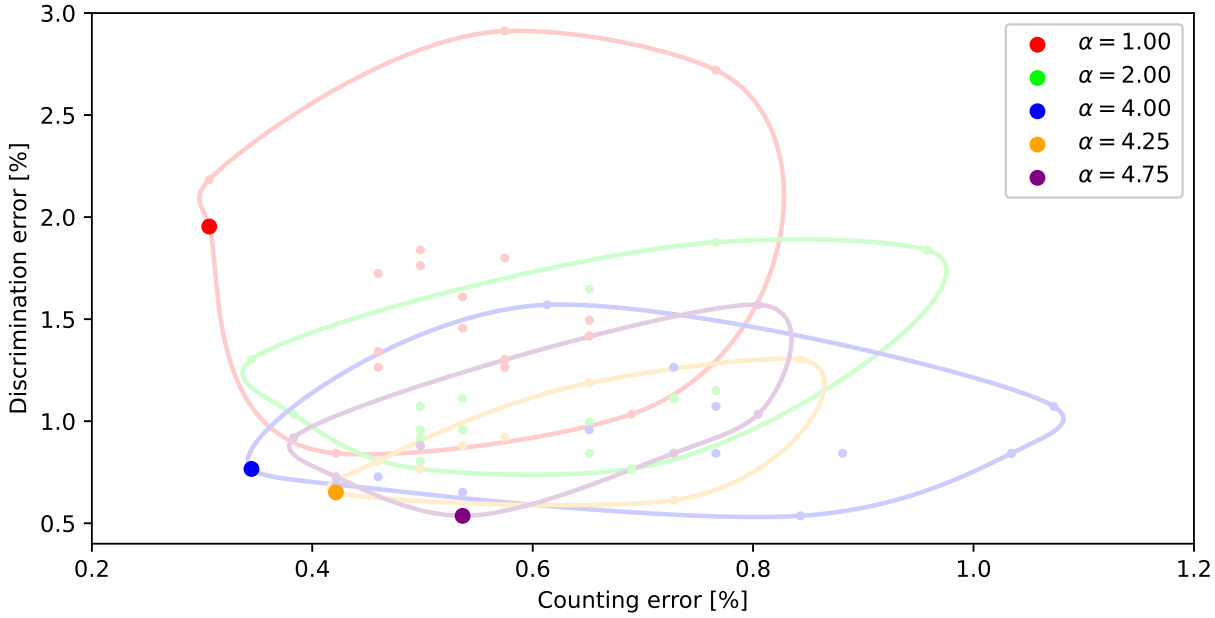


Figure 7.1: Error rates on an unseen test dataset for clouds of 20 randomly initialized runs with different values of α. The contour of each cloud is outlined and non-dominated solutions are highlighted.

7.2 Separated networks

The technique described in the previous section demonstrates acceptable practical performance while providing the fastest and most compact representation. However, experimental results indicate that a significant improvement can be made by dividing the network into a two-step process, as illustrated in Figure 7.2.

In this new structure, the first part of the network processes every candidate and is tasked solely with the counting sub-problem. If the candidate is confirmed as a shot, the input vector is passed to a second, similar network tasked with the discrimination sub-problem. This technique offers several advantages:

7.2.1 Computational budget

Separating the two sub-problems clarifies the distribution of computational budget between the counting and discrimination sub-problems. The budget for the second network is derived from the computational power remaining after allocation to the higher-priority counting network. Moreover, the discrimination network only needs to run on candidates confirmed as shots by the counting network. Assuming a larger network is required to handle the complexity of both counting and discrimination (a matter discussed in Section 7.2.2), the total inference time can be reduced by running the discrimination part only when necessary. Provided that some level of buffering is feasible, if *shot* candidates represent a proportion λ of the total number of candidates, then, for a fixed maximum average inference time $\bar{T}_{max}$ and a counting inference time $\bar{T}_C$, the maximum average discrimination inference time $\bar{T}_D$ can be estimated as follows:

$$\bar{T}_{max} \geq \lambda(\bar{T}_C + \bar{T}_D) + (1-\lambda)\bar{T}_C \iff \bar{T}_D \leq \frac{\bar{T}_{max} - \bar{T}_C}{\lambda}. \tag{7.5}$$

A good estimate of λ (and of the worst-case activity burst in practical use regarding the necessary buffering space) will therefore allow for a larger $\bar{T}_D$ budget. This optimisation is not feasible with a monolithic network. Estimations of λ for the considered firearms can be found in Table 7.1. These estimations were computed using only acceleration curves containing a shot sequence and close non-shot events, where discrimination is relevant.

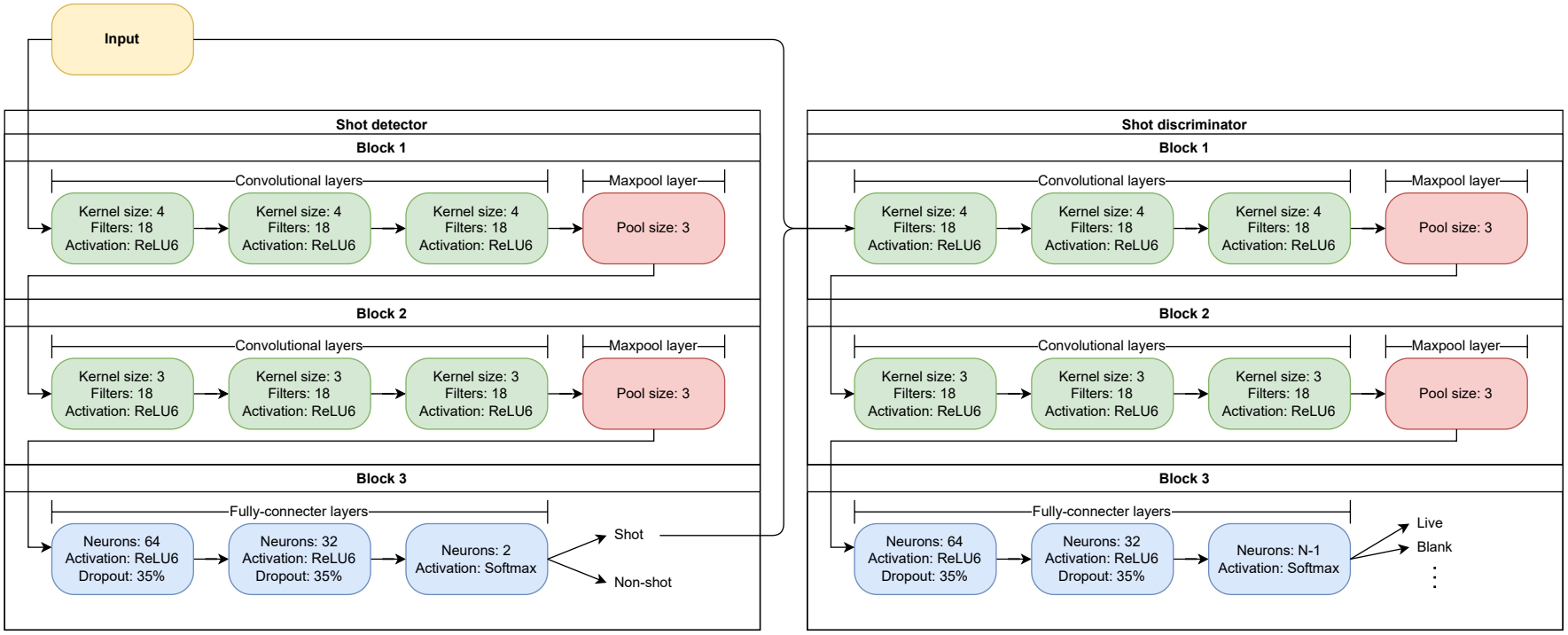


Figure 7.2: Two-step network structure for discrimination. The exact number of parameters depends on the available computational budget.

	Raw λ	Filtered λ
Minimi 5.56	47%	77%
Minimi 7.62	55%	82%
MAG	40%	67%
M2 backplate	18%	21%
M2 barrel	35%	41%
M134	55%	95%
Average	**42%**	**64%**

Table 7.1: Estimation of λ for the considered weapons using the preprocessing parameters of Table 9.1 on learning sets. Only input curves containing shots are considered. Filtered λ considers only candidates after post-filtering from Section 6.3 using the best available counting model.

7.2.2 Discrimination performance

Figure 7.3 compares the proposed two-step process with large single networks. Larger networks are obtained by proportionally increasing the number of convolution channels in the convolutional layers and the number of neurons in the dense layers. For example, the models denoted as "Single network 2x" have their number of convolution channels increased by $\sqrt{2}$, and the number of neurons per dense layer also increased by $\sqrt{2}$. This results in a total number of parameters approximately twice that of the original network. Note that, in the case of separated networks, the counting network is first selected as the best-performing model out of 80 runs (as describes in Section 7.2.3). Its positive shot predictions are then used to train the separated discrimination networks. As a result, the counting error is fixed for all associated discrimination networks. The most direct comparison is with a single network that has the same total number of parameters as the separated networks, which is twice the number of parameters in the original counting network. This is represented in red in Figure 7.3. Importantly, while these models are identical in size, as discussed in Section 7.2.1, the larger single model must be run in full and requires twice the inference time for *non-shot* candidates.

We observe that the separated discrimination network performs significantly better, achieving a reduction in the median discrimination error of more than 80%. However, we also notice that a very small proportion of networks show a slight decrease in counting error. These outliers have likely allocated the increased network size towards the counting sub-problem rather than the discrimination sub-problem. As expected, increasing the size of the single network improves performance in both counting and discrimination. However, even networks with 100 times more parameters still exhibit a median discrimination error 86% higher compared with the separated networks. These results suggest that isolating the two sub-problems enhances the effectiveness of the discrimination part significantly.

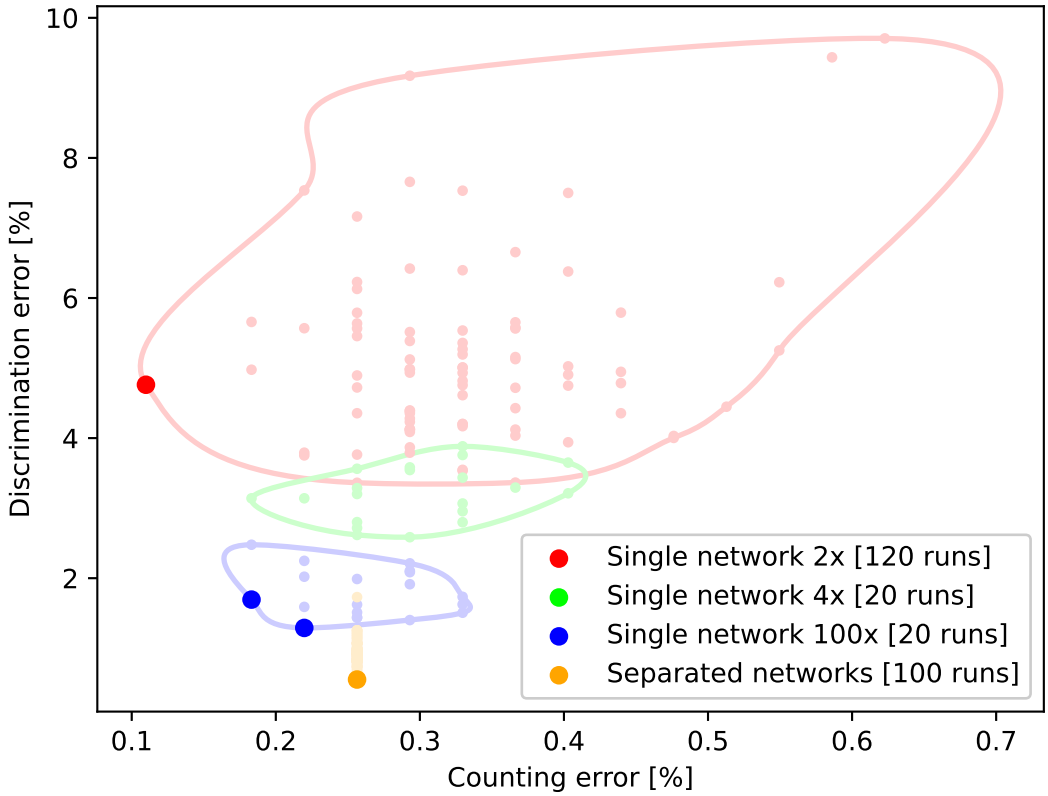


Figure 7.3: Error rates on an unseen test dataset for clouds of randomly initialized runs, comparing large single networks against a two-step process. The contour of each cloud is outlined and non-dominated solutions are highlighted.

7.2.3 Ease of training

Separating the two sub-problems allows their respective networks to be defined and trained independently. The higher-priority counting model can thus be derived first and locked to ensure optimal accuracy, independent of discrimination performance. To reduce development lead time, a key reason for the use of machine learning in shot counters, one could consider deploying a counting-only model before deriving the discrimination model. This additional feature could then be deployed later as a firmware update.

In addition, as illustrated in Figure 7.3, attempts to reduce the discrimination error by increasing the network capacity might instead result only in improved counting performance, necessitating further adjustment of α to ensure that the increased capacity is allocated to the discrimination sub-problem. Separating the networks ensures that any supplementary capacity given to the discrimination network will be used for that purpose.

Our original proposal has the primary advantage of being able to be trained using only

weak labels on time series that may contain mixed classes, thus reducing the need for extensive labelling efforts. However, when constructing a dataset under controlled conditions, it is possible to ensure that each time series contains only a single type of ammunition. Since the counting network separates the hard-to-label *non-shots*, the remaining *shot* vectors can be labelled inexpensively on an individual basis (though this will also include the false positives given by the counting network). The problem addressed by the discrimination network then becomes fully supervised, enabling us to leverage a more usual formulation of the loss function, such as cross-entropy. The main benefit is a reduction in training time by approximately 65%. The median discrimination error is also reduced by 20%, as shown in the rightmost boxes of Figure 7.4. Notably, this represents a difference of only 0.23% in absolute discrimination error, which remains stable across different discrimination network capacities as obtained by the technique described in Section 7.3 with higher error rates. This suggests that, generally, using weakly supervised learning with the EDGAR technique does not significantly penalise prediction performance.

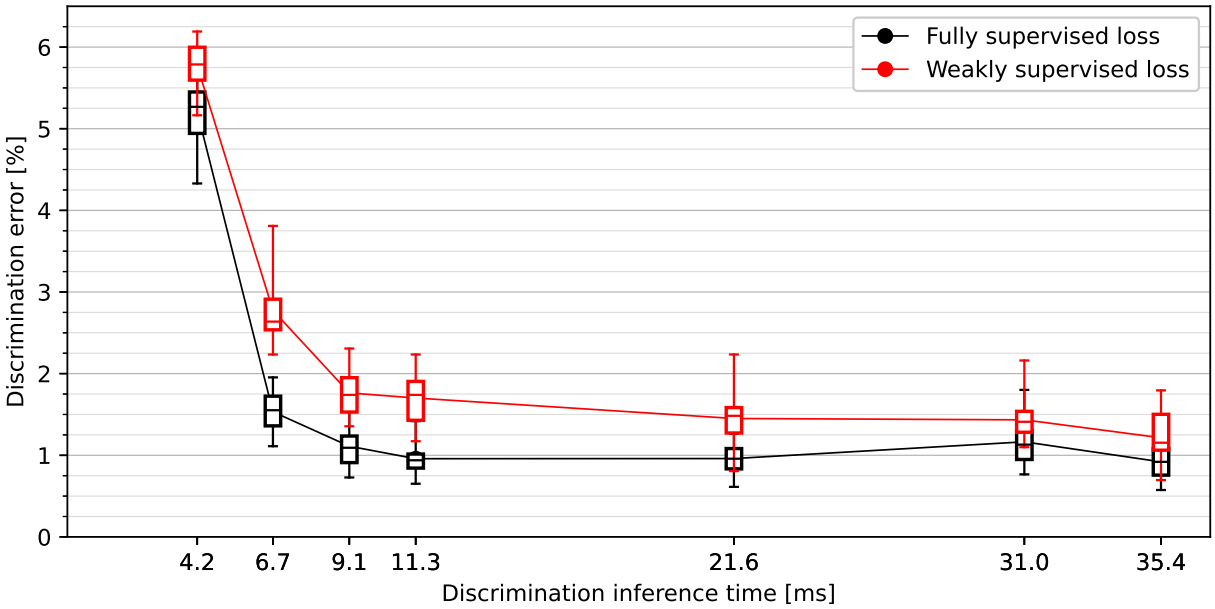


Figure 7.4: Discrimination error rates on an unseen test dataset for batches of 20 randomly initialized networks at various levels of D. The highest inference time corresponds to fully separated networks. Performance between fully and weakly supervised losses is also compared. Inference time is evaluated on a 64 MHz Cortex-M4F microcontroller.

7.3 Shared activations

While the previous section has shown that splitting the two sub-problems leads to several benefits, we can assume that some redundancy will exist between the two networks, as they both operate on the same input vector to solve a similar problem. Since parallelisation of the two sub-problems is often unavailable on resource-constrained embedded hardware, we can reduce the inference time and network size of the discrimination network by repurposing intermediate activations of the counting network.

The initial layers of the convolutional neural network, prior to the first pooling operation, are the most computationally expensive, accounting for about 60–80% of the inference time in typical use cases.

Depending on the specific problem and available resources, two hyperparameters are introduced:

- The depth at which the counting network is reused (D), measured in layers from the input. Deeper copies further reduce inference time and memory footprint, but also reduce prediction performance as discrimination-specific information is not recovered from the original signal.

- Optionally, layers before D can be replicated in a narrower fashion (fewer convolution channels/neurons) before being concatenated to the copied activations. These extra channels/neurons, whose number is denoted as E, allow the final network to recover some of the discrimination-specific information that may be missing from the activations of the counting network.

An example of the resulting structure is given in Figure 7.5.

Training is accomplished by building the whole network structure, then copying and freezing the weights of the layers shared with the counting network. At inference time, the counting sub-graph is traversed first up to the counting outputs. If a *shot* is predicted, the discrimination sub-graph, starting from the shared activations (and optionally the input of the extra layers), is traversed separately.

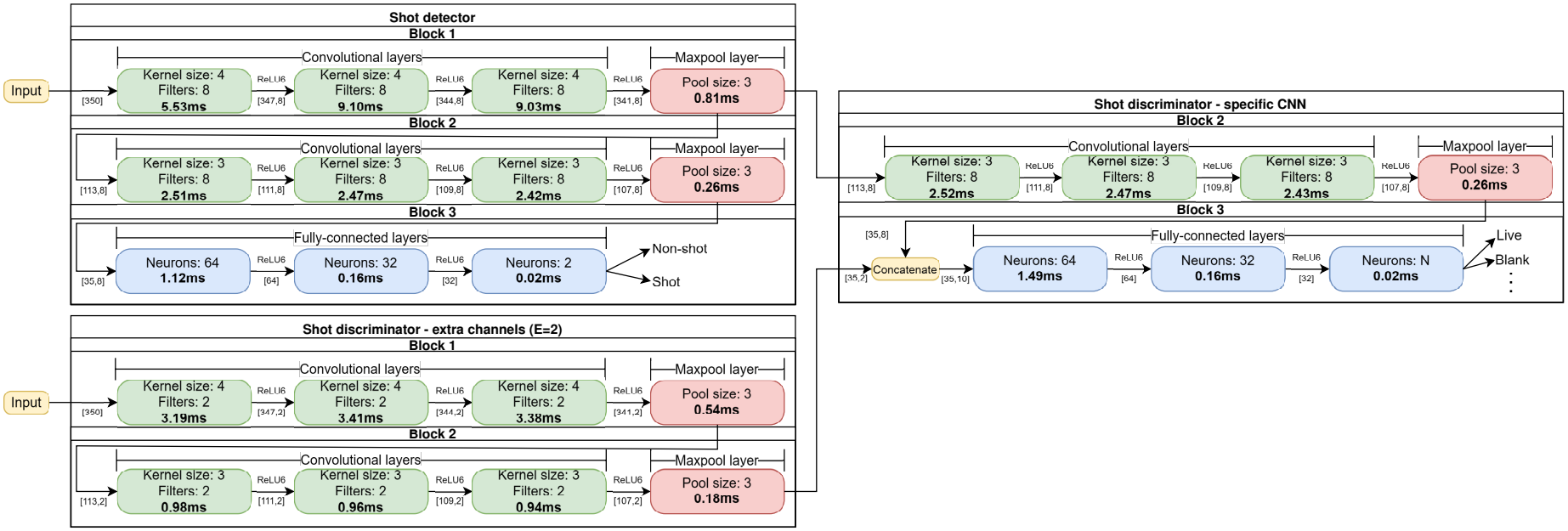


Figure 7.5: Example of a network with shared activations used for MAG discrimination with $D = 4$ and $E = 2$.

In Figure 7.6, we study how the discrimination error rate and inference time evolve when D and E are tweaked. As a basis, we compare this with how the error/time trade-off is otherwise performed by varying the number of convolution channels, denoted as F. For reference, the associated counting network runs in 33.9ms.

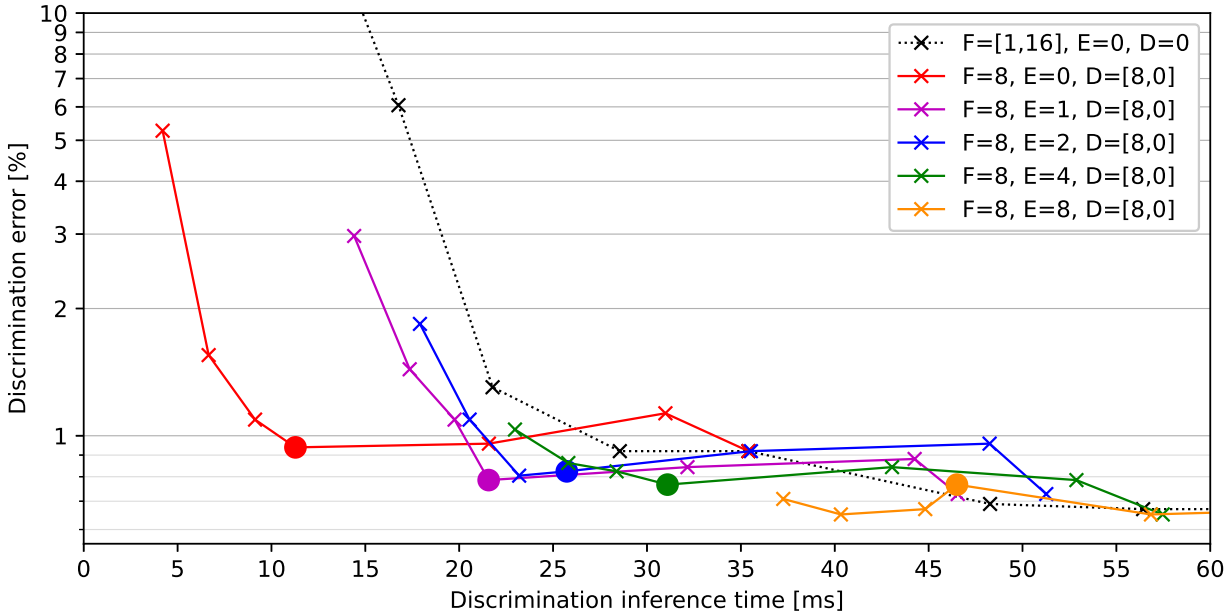


Figure 7.6: Logarithmic discrimination error rates and inference time on an unseen MAG test dataset at various levels of D, E, and F. Points at $D = 4$ are highlighted. Each point is the median of 20 runs. $D = 3$ and $D = 7$ are identical to $D = 4$ and $D = 8$ respectively. Inference time is evaluated on a 64 MHz Cortex-M4F microcontroller.

Looking first at variations of D for each fixed E, we observe that increasing D (from right to left) up to $D = 4$ has only a minor impact on discrimination performance, with an average increase of 12% and as low as 2% for $E = 0$. Meanwhile, inference time is reduced by 25ms, representing a 68% decrease in the $D = 0$ case. After that point, further increases of D generally increase the error rate exponentially while providing relatively lower inference time gains.

Comparing with the reference approach of managing the error/time trade-off by varying the number of convolution channels F (shown in dotted black), we notice that with $E = 0$, significantly faster networks are obtained for a similar error rate. Considering a maximum acceptable error rate of 5%, the shared activations technique allows us to run in 4.2ms instead of 16.8ms, or a 75% reduction. Considering that shots represent about 67% of the evaluated candidates in this dataset, the ammunition discrimination feature can be added with as little as 2.8ms per candidate, representing an 8% overhead on the counting inference time. At a desirable 1% error rate, discrimination runs in 11.3ms instead of 28.5ms, representing a 60% reduction.

Looking now at the different curves obtained when E is increased (from left to right), we observe an expected reduction in discrimination error at the cost of increased inference

time. Comparing with the reference, for a restricted computational budget of around 35ms similar to that of the counting network, we notice that the discrimination error can be reduced by up to 20%. For unrestricted inference time, the difference becomes progressively smaller as the "extra" network becomes sufficient to capture the full discrimination sub-problem independently.

Based on these results, our general recommendation is to use shared activations with $D = 4$ and $E = 0$ as a starting point. From there, either E can be increased until the maximum acceptable inference time is reached for best prediction performance, or D can be increased until the maximal acceptable error is reached for best inference time performance.

The main drawback of this approach is that the discrimination network is directly tied to a specific counting network. Any change to the latter will likely require retraining the former. We thus recommend keeping fully separated networks in the case of less constrained environments.

7.3.1 Practical considerations

Current implementation of TensorFlow Lite for Microcontrollers does not support partial execution of a neural network. This is necessary to avoid running the discrimination network when the counting network classifies the candidate as non-shot.

One possible workaround is to cut the network graph into subgraphs which can be evaluated separately, copying the output of one subgraph into the input of another as necessary. This is illustrated in Figure 7.7. First, the main CNN and the counting head are evaluated. If the main CNN classifies the candidate as a shot, the discrimination and extra CNNs are evaluated and their outputs concatenated. Finally, the discrimination head gives us ammunition type classification.

The problem with this implementation resides in the quantization parameters of the resulting subgraphs. When moving data between subgraphs, one must be careful to apply the scale factor and zero-point offset of both the output of the first subgraph and the input of the second one. These operations have a significant computational cost, and introduce rounding errors. Additionally, when using quantization-aware training the freezing of the layers does not correctly freeze the learning of quantization parameters. Although we did not confirm this directly, we suspect that it exacerbates the problem. Memory consumption is also an issue, as the allocator does not properly reuse tensor memory space between subgraphs. In Table 7.2, we observe that these rounding errors more than double when using split subgraphs. RAM consumption is also increased by a factor of 2.5.

	5 subgraphs	Single subgraph
Counting difference	0.08%	0.03%
Discrimination difference	0.51%	0.24%
Output difference ≥ 1	7946	4308
Output difference ≥ 2	311	62
Output difference ≥ 3	11	6
Output difference ≥ 4	1	0
Memory	30kB	12kB

Table 7.2: Comparison of the differences due to rounding errors between the PC and embedded implementations for a split-subgraphs solution against a single subgraph. Counting and discrimination differences indicate the percentage of cases where the model's decision differs between the PC and embedded implementations. Output difference represents the sum of absolute differences in the raw counting and discrimination outputs.

Considering these issues, we implemented a custom evaluation of the neural network that allows for partial execution. By reusing the tensors allocated for the counting problem, inference for the discrimination problem can be performed with no extra RAM requirement.

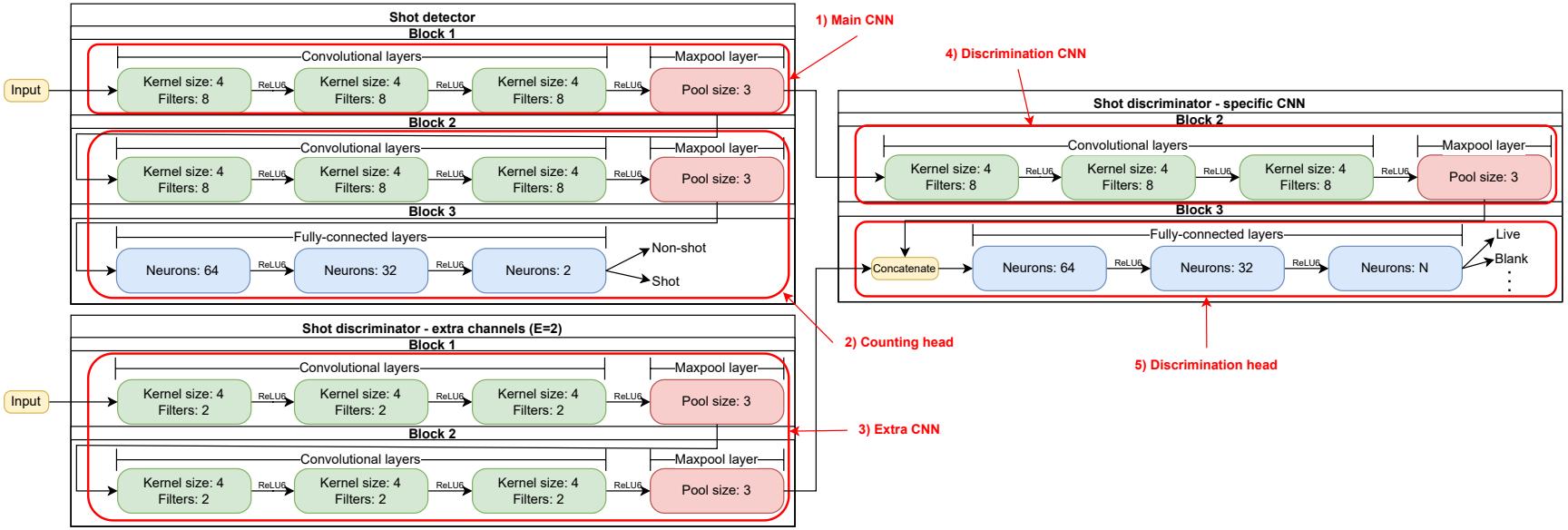


Figure 7.7: Network with shared activations split into 5 subgraphs.

7.4 Weapon configuration majority voting

There is a physical limitation on how quickly a weapon can be switched from one configuration to another, particularly when specific accessories need to be mounted on the platform. Notably, switching from firing live rounds to blank rounds typically requires attaching a blank-firing adaptor (illustrated in Figure 7.8a) to the end of the barrel for both functional and safety reasons. A similar operation is required when switching to and from the use of a suppressor (illustrated in Figure 7.8c). We can therefore assume that all shots detected in a burst (i.e. uninterrupted automatic firing) must belong to the same weapon configuration.

In addition, affixing the required accessory to the weapon requires a minimal amount of time, and these operations cannot be safely performed shortly after the firearm has been operated, requiring the operator to wait until the barrel cools down to safe handling temperatures. Therefore, provided that all intervals between detected shots are shorter than this minimum configuration switching time, the detected shots should all belong to the same weapon configuration.

Similarly to how the minimum cycle time was exploited to improve counting performance in Section 6.3, this physical information can be leveraged to improve the overall discrimination performance by performing a majority vote on the detected shot classes. Individual predictions are accumulated until there is a period without detected shots that exceeds the minimum time required to switch configurations. At that point, all pending predictions can be permanently recorded with the ammunition type that corresponds to the majority of the predictions. Note that if the device is to be powered off during periods of inactivity, individual predictions must be stored in persistent memory.

(a) Blank-firing adaptor.

(b) Blank-firing adaptor affixed to a FN Minimi.

(c) Suppressor.

(d) Suppressor affixed to a FN Minimi.

Figure 7.8: Illustration of barrel-mounted accessories.

Chapter 8

Dataset acquisition and model analysis

8.1 Acquired datasets and configurations

8.1.1 Acquisition methodology and weapon setups

Shot data collection and configurations

Training data was acquired on the FN Minimi® 5.56, FN Minimi® 7.62, FN MAG®, FN® M2HB-QCB, and the M134 Minigun.

For the Minimi 5.56, data was collected using both short and long barrels at various gas return settings, including extreme values. Most measurements were taken with mid-life barrels that were used until decommissioning, with additional data gathered from new barrels. The weapon was fired from the shoulder, hip, bipod, and tripod, both without accessories and with added accessories totalling 3 kg. The Minimi 7.62 was tested using a new barrel across multiple gas return settings, including extremes. It was fired from the shoulder, hip, and bipod, with and without 3 kg of accessories. For the FN MAG, data was gathered using three weapons, one new and two nearing decommissioning, alternated across trials. Each was set to different gas return levels, including extreme settings. Firing configurations included the shoulder, bipod, tripod, elastic mount, and two fixed mounts. Accessories totalling 3 kg were used in all configurations except the shoulder. Data for the backplate-mounted sensor for the M2HB-QCB was acquired in live fire using one new and one nearly-decommissioned barrel, employed alternately. For blank firing, a specialised blank-firing barrel was required due to the constraints of the short-recoil system. Data for the barrel-mounted version was collected separately, and included an additional weapon body and barrel. The weapon was fired from a tripod, fixed mount, and an elastic mount. A deFNder teleoperated station was also tested with the backplate version. Feeding was done using both a minimal 10-round belt and a standard 100-round belt to account for the additional mass. The M134 Minigun was fired from a single type of fixed mount, and should be considered an exploratory dataset. However, its electrical feeding system,

independent of any gas return, allows for a very consistent firing cycle.

The acceleration curves in our datasets typically consist of 15 shots for the Minimi and MAG variants and 10 shots for M2. This ensures that the counting error rate can be reliably reported. Only live or blank rounds are fired in a single acquisition, allowing for a full evaluation of the discrimination error. Each weapon was tested across four firing sequences: isolated shots, small random bursts of 3–6 rounds, bursts of four shots spaced by single shots, and long continuous bursts. Due to its very high rate of fire, an exception was made for the M134, which was fired in bursts between 15 and 70 rounds.

A roughly equal number of shots is fired in each setup in an attempt to sufficiently capture firearm behaviour at the edge of acceptable working specifications. This approach naturally over-represents extreme cases, leading to error rates that are higher than those obtained on a benchmark designed for nominal usage. While weapon accessories are typically evaluated only in nominal cases, we believe that focusing on edge cases is important for maintenance purposes, as shots performed at or slightly beyond the specifications of the firearm (for example, with a very high or very low burst rate) are likely to be most damaging to the hardware. However, data containing firing incidents (weapon malfunctions) was rejected.

A significant amount of non-shot data is naturally acquired before, during, and after firing. This includes both firing-related mechanical events and operator manipulations before and after firing sequences. Supplementary non-shot data is acquired to further populate the dataset and reproduce cases (such as falls) which are not included in normal testing. These also provide an opportunity to evaluate the network on inputs where all false positives can be detected.

Additional non-shot data acquisition for the Minimi and MAG variants includes dry firing, 1.5m falls onto concrete, opening and closing the top cover, full reloading manipulation, and the user randomly bumping the weapon. M2HB-QCB supplementary non-shot data comprises dry firing, rearming and releasing the mobile parts, opening and closing the top cover, and user randomly bumping the weapon. For the M134 Minigun, non-shot data was limited to rotating the barrels using the electric engine and dry-firing the weapon.

Note that the number of non-shot events in the datasets is not exactly known as there is no easy and reliable way to provide weak labels for this data, meaning we would have to rely on operator estimations. Thus, we only provide a lower bound based on the number of explicitly produced non-shot events. A more accurate estimate of non-shot events can be obtained from the total number of generated candidates.

The available data used for training and testing are reported in Table 8.1. Data using continuous acquisition are marked with “FC” (Full Curve). Data using the Instant On mode as described in Section 4.2.3 are marked with “IO”. The Minimi 5.56 mostly comprises full-curve data, but has been supplemented with some Instant On data and is thus marked with “M” for Mixed. The number of candidates is calculated with the preprocessing hyperparameters of Table 9.1.

	Minimi 5.56 (M)			Minimi 7.62 (FC)	
	Learning	Validation	Test	Learning	Validation
Live	4596	1920	1800	4707	943
Blank	2370	839	1150	1800	349
Non-shot	> 400	> 55	> 0	> 97	> 35
Candidates	15848	6289	6108	12033	2338
Curves	571	198	59	444	91

	MAG (IO)			M134 (IO)		
	Learning	Validation	Test	Learning	Validation	Test
Live	9400	2156	1755	1015	290	290
Blank	4609	1335	990	N/A	N/A	N/A
Non-shot	> 1067	> 206	> 192	> 3000	> 450	> 450
Candidates	38236	9361	7336	2123	585	625
Curves	994	245	196	55	11	11

	M2 backplate (IO)			M2 barrel (IO)		
	Learning	Validation	Test	Learning	Validation	Test
Live	4923	669	260	1550	270	270
Blank	3839	499	250	N/A	N/A	N/A
Non-shot	> 1261	> 174	> 30	> 375	> 100	> 120
Candidates	54416	7354	2962	4908	883	910
Curves	1033	137	54	184	34	36

Table 8.1: Summary of the number of shots per dataset.

8.1.2 Dataset splitting strategy

For the MAG, M134, and M2 barrels, validation and test datasets are created by splitting the original input datasets. To ensure that these datasets capture the behaviour of the firearm across a broad range of configurations, the input series are first grouped into bins based on all available combinations of external variables. Each bin is then randomly shuffled and divided into three parts, with the test and validation sets comprising at least 10% of the total data for that bin. If there is insufficient data for a given configuration to meet this ratio, priority is given to allocating data to the test set, followed by the validation set. Configurations tested less than 3 times are thus not present in the learning set.

For the Minimi variants and M2 backplate versions, the same procedure was followed to split the validation set but without a test set. The Minimi 5.56 and M2 backplate test sets were acquired independently of the training data. The Minimi 5.56 test set shows a distribution shift, as it includes some configurations which were not captured in the training data. Therefore, results on the validation set are also reported. The Minimi 7.62

does not have a test set, as its independent acquisition has not yet been completed; thus, results on the validation set are reported instead.

8.2 Minimal required dataset

8.2.1 Scaling analysis and error rates

Neural network performance relative to dataset size has been successfully modelled by a power law across a wide range of domains and scales [67], [68], [69]. In Figure 8.1, we can observe that this relationship holds true for both the counting and the discrimination sub-problems, in both the MAG and M2 datasets. For this study, the majority voting technique described in Section 7.4 is disabled for discrimination, as the voted error rate otherwise quickly falls to zero. The discrimination behaviour starts off with a worse error rate for very small datasets, but is more responsive to the increase in dataset size. Interestingly, the MAG dataset for the discrimination sub-problem shows a deviation from a power law at low levels of discrimination error. This deviation is expected from the known literature as we approach the saturation of our model (due to its fixed size) around $\bar{E}_D = 1\%$.

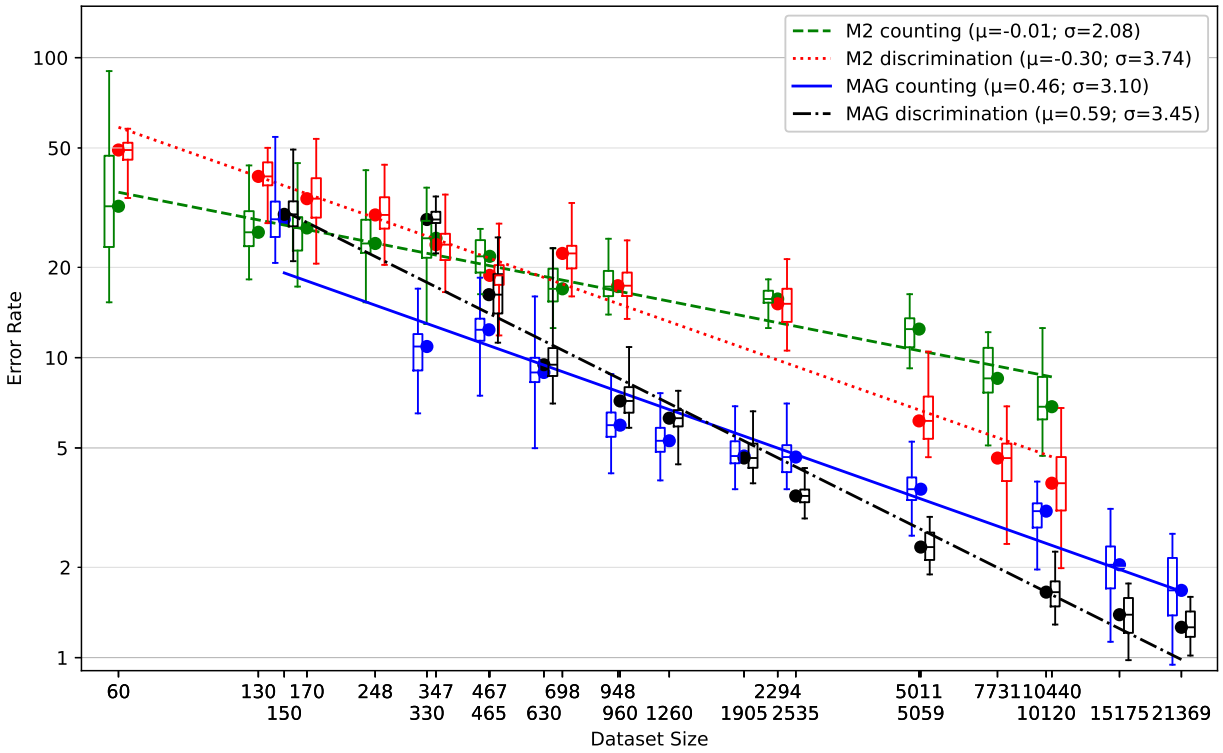


Figure 8.1: Error rate versus dataset size with power-law model fitting on the median values (log-log scale). Hyperparameters are fixed over the whole range. Reduced datasets are sampled from the full dataset across weapon configurations, preserving its diversity.

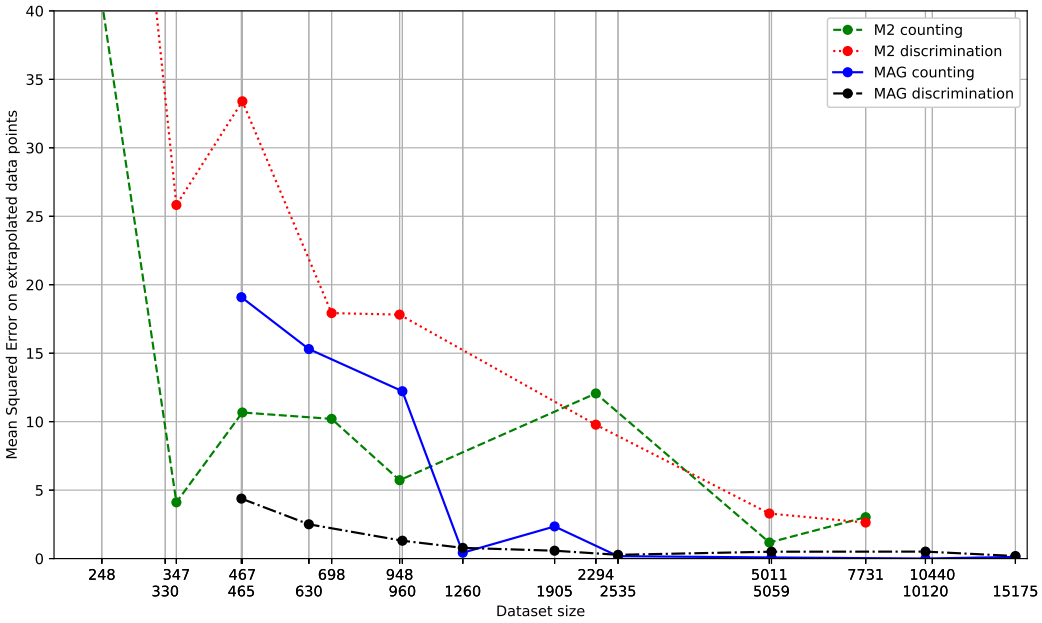


Figure 8.2: Mean-square error between extrapolated values and test values against dataset size. Extrapolated values are obtained by fitting a model on datasets $\leq N$ and then extrapolating from it. This assesses the quality of the extrapolation. Test values are the median error rates obtained when evaluating each dataset $>N$ on the test set of biggest available dataset.

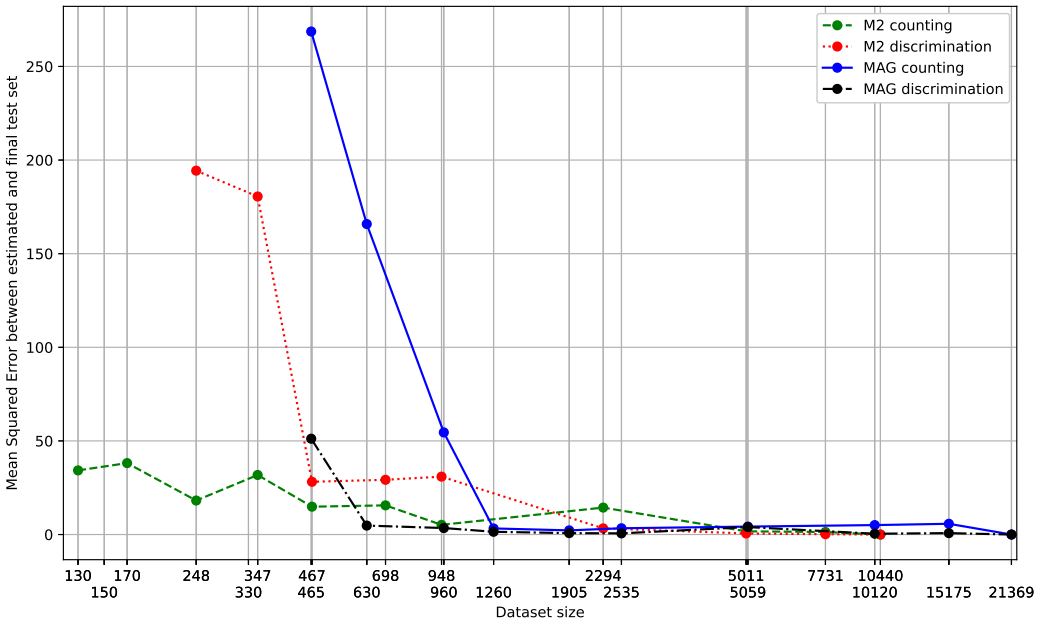


Figure 8.3: Mean-square error between error rates on the reduced test set and error rates on the full test set. This assesses the quality of the estimated errors rates, with a high quality being a necessary prerequisite for a good fit of the decay model.

This information has a very practical use for our application. Each new weapon requires a new dataset, and acquiring the samples is expensive, time-consuming, and often requires significant planning to ensure the availability of hardware and testing grounds. Quantifying in advance the necessary number of shots to achieve adequate performance is therefore of great interest. Rosenfeld et al. [70] apply extrapolation to the power law deduced from smaller datasets to deduce the performance given by a bigger dataset on a fixed benchmark.

In our case, an independent benchmark is generally unavailable during preliminary testing of a new firearm and must therefore be constructed alongside our learning datasets. In Figure 8.2, we observe the error rate between the extrapolated values and the actual values as estimated on the largest available dataset. We generally observe a sharp drop before a more gradual improvement. Figure 8.3 gives us further insight into this behaviour by comparing the error rates obtained on the test sets of the reduced datasets and those obtained on the test set of the biggest available dataset. This represents the error in the estimation of the error rates on which the decay model will be fitted. Before a sufficiently low error is reached, the reduced test dataset is not yet representative of the full distribution and significantly underestimates the actual error rate, rendering any fit, and thus extrapolation, inaccurate.

8.2.2 Practical guidelines for dataset sizing

From Figure 8.3, we can thus make a general recommendation for a dataset creation procedure that includes preliminary testing with 1000–2000 shots, allowing the test set to reach representativeness. We hypothesise that the exact value required is proportional to the number of possible configurations of the weapon. This is supported by the MAG counting dataset being the last to reach representativeness, as this firearm offers a very high number of possible configurations and, thus, high variance in possible behaviours under use.

Type	Problem	Shots	E	Extrapolated	Full fit	Recommendation	Achievement value
MAG	Count.	1260	2%	9767	14397	11720	10120
MAG	Discr.	1260	2%	3572	7705	4286	5059
M2	Count.	948	10%	3950	5094	4740	5011
M2	Discr.	948	5%	4123	9207	4947	5011

Table 8.2: Extrapolations from Minimal Representative Dataset

Using this exploratory dataset, one can fit the power law and estimate the number of shots required to achieve the desired counting and discrimination performance. Examples are provided in Table 8.2, where the smallest dataset that reaches representativeness is used to provide predictions for a desired error rate E (chosen arbitrarily with values low enough to ensure meaningful extrapolation while remaining verifiable from our data). We observe that, generally, the extrapolated values appear overly optimistic compared to the

fit on the largest dataset. However, this fit considers only the median error rate. When examining the best performance from each training batch of 20 networks, i.e. the network which would actually be selected for deployment, the required number of shots decreases. The "achievement value" column reports the number of shots in a dataset that produces the first network to reach the desired error rate. Although the extrapolated values remain slightly optimistic, they are generally close to the actual values for practical purposes. We therefore recommend using the extrapolated prediction increased by a 20% margin. Further research with additional data is needed to provide a generalisable adjustment for this underestimation. In addition, we recommend monitoring the error rate as data acquisition progresses to detect any saturation through a gradual deviation from the power law, as observed in the case of MAG discrimination. In such cases, data acquisition can be stopped early, as additional data would provide only marginal improvement with a fixed network size.

8.3 Leveraging the model for hypothesis testing

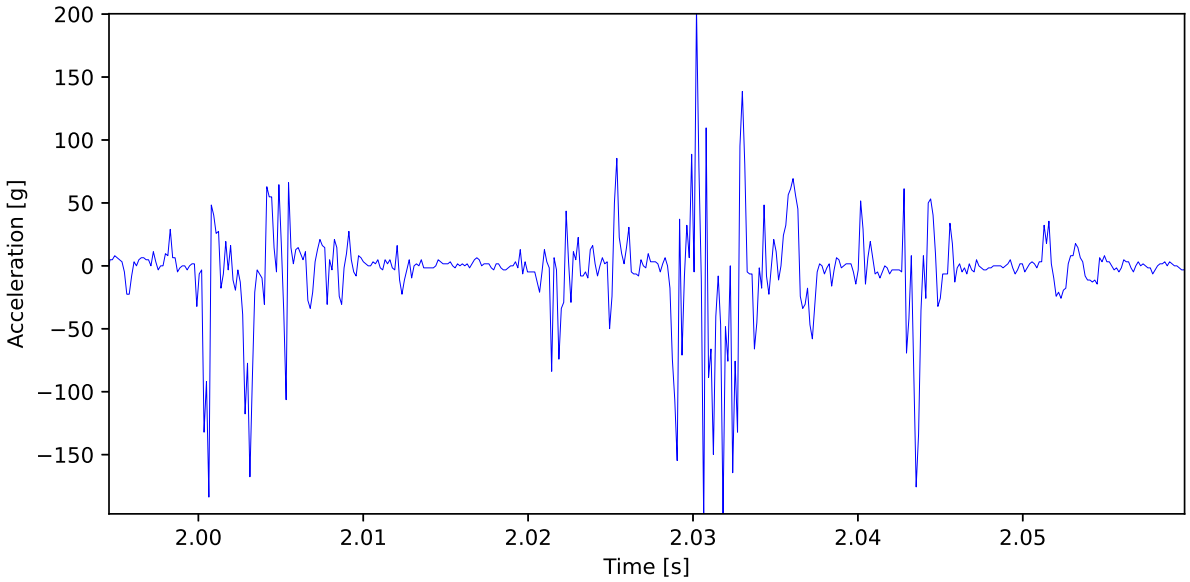
As we saw in Section 1.3.2, gunshots exhibit a wide range of accelerations, with significantly different scales at different steps of the firing cycle. While we argued that an approach based on shock duration and time separation is more robust than relying on shock intensity, the relative intensity between different parts of the firing cycle remains useful in ambiguous situations. Additionally, reducing the acceleration range too much will make distinguishing the individual shocks more difficult. A visual example is provided in Figure 8.4. As a result, successful shot counters generally comprise a high-g accelerometer.[1]

In Figure 8.5, we show the result of the EDGAR technique when the input dataset is artificially saturated at ±16 g. Due to the lower range of values, the candidate selection thresholds need to be adapted accordingly. Through a rough estimation with classical techniques, values of $T_H = 38$ and $T_L = 30$ were used (without fine-tuning). The M2 and MAG weapons were selected for their use of relatively high energy ammunition, representing a worst-case scenario.

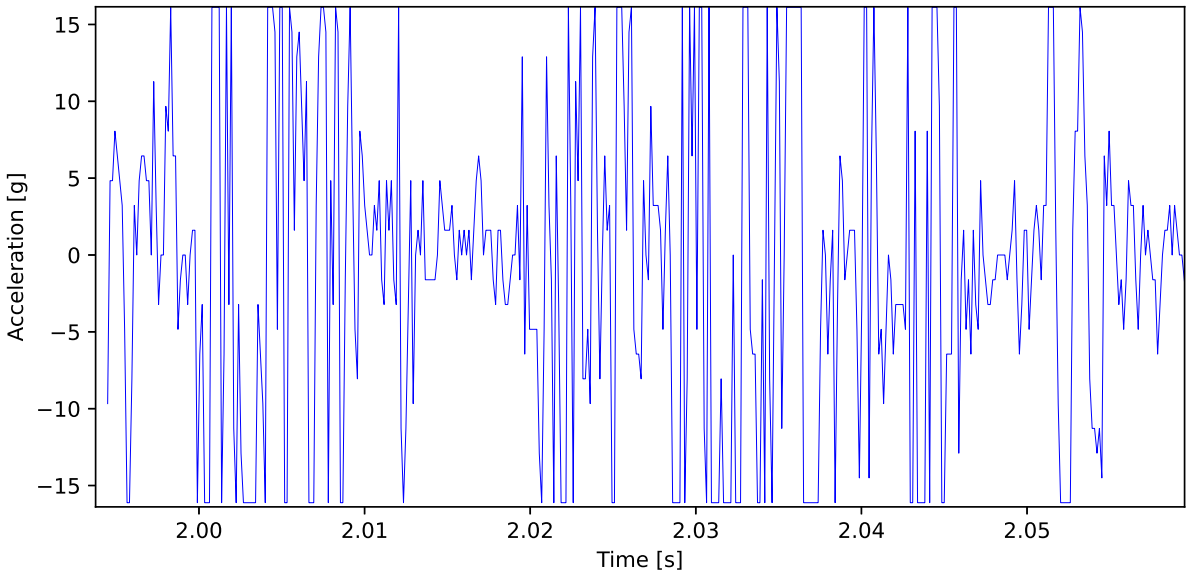
As expected, the error rate shows a significant relative increase. However, the MAG case suggests that when the absolute error is sufficiently low, the counting error of the ±16 g version remains well within the acceptable range. We assume that the difference could be reduced through fine-tuning of the hyperparameters. Actual data acquisition with a ±16 g accelerometer would also provide increased resolution, which might further reduce the error rate. These results suggest that the EDGAR technique is able to produce models for input spaces which were previously considered too difficult to work with.

Producing a model capable of accurate shot counting with low-g accelerometers is of practical interest. These sensors are generally less expensive and more power-efficient than their high-g counterparts. Since ±16 g accelerometers are widely used in existing devices,

[1]The work by Asbach and Canty [52] shows an example of shot counting development beginning with a 16 g accelerometer which was later supplemented by a 200 g accelerometer for increased accuracy.



(a) Full ±200 g scale



(b) 200 g scale saturated at ±16 g

Figure 8.4: Comparison between full-scale ±200 g and a simulated ±16 g time series.

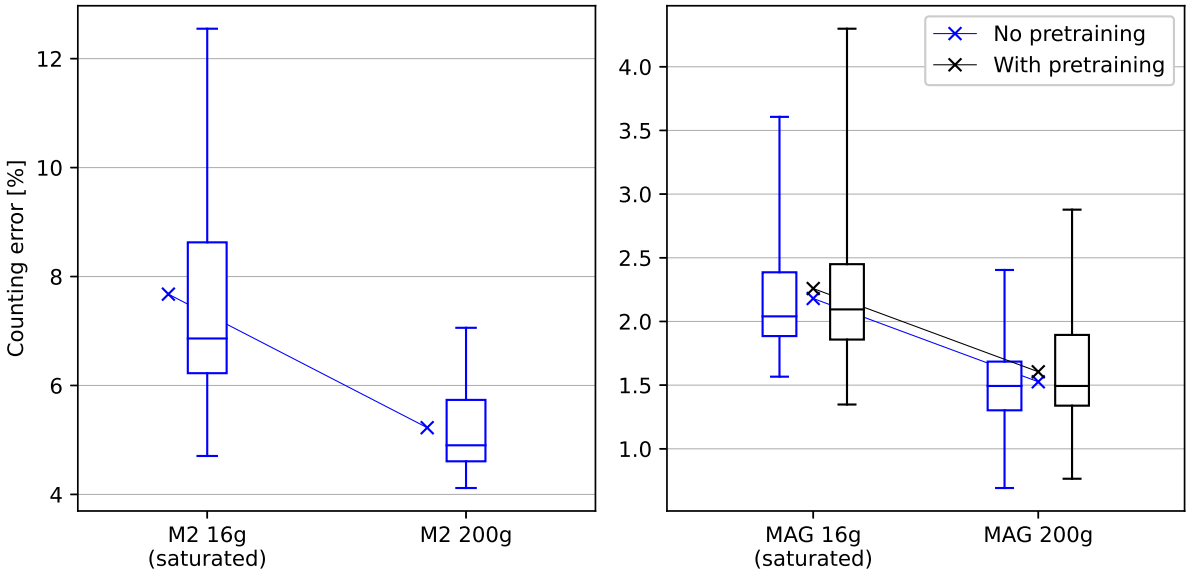


Figure 8.5: Evaluating the performance difference between the full 200 g-scale dataset and a 16 g-saturated version for both M2 backplate and MAG. Evaluation is performed on the test dataset. Hyperparameter tuning was not performed, except for a necessary rough adjustment of the candidate selection thresholds.

supporting this range also allows one to add shot counting capabilities to this equipment through only a firmware update.

Classical techniques would have required a significant time investment from an expert to prototype a shot-counting algorithm, confirming or ruling out deployment feasibility with this type of sensor. However, the use of machine learning allowed us to rapidly assess feasibility, limiting human intervention to a rough estimation of new candidate selection thresholds.

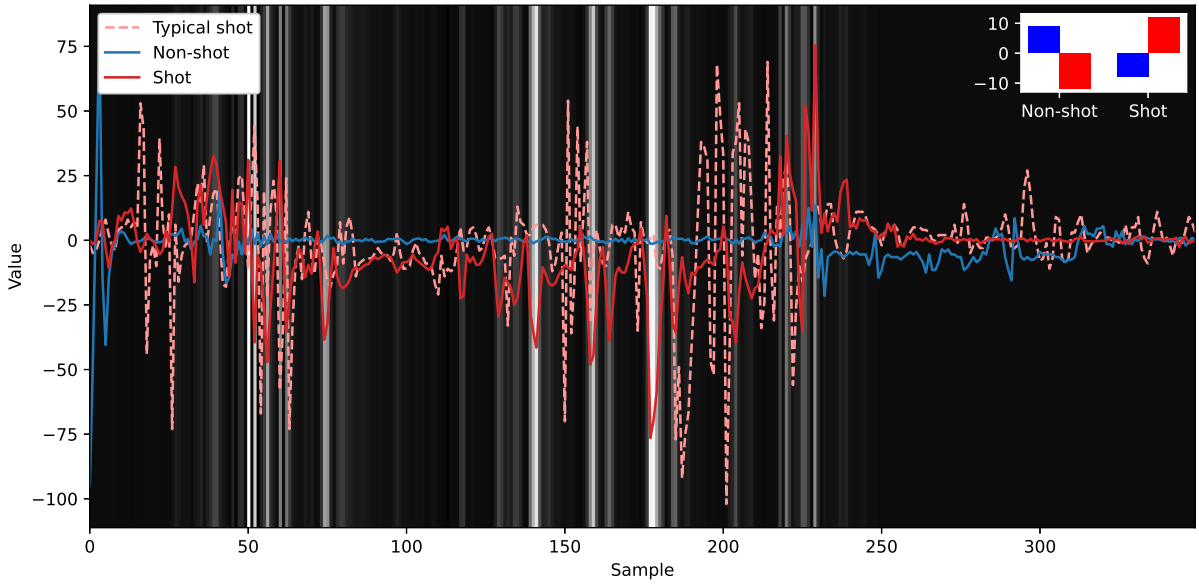
8.4 Interpretability via saliency mapping

Another white-box application of trained counting and discrimination neural networks involves examining the inputs that maximise specific outputs. Starting from a random initial input, we apply gradient descent to optimise the input vector such that it maximises the targeted output while simultaneously minimising the other. This approach offers insights into the characteristic features that the network associates with particular outputs. We can then go further and create a saliency map by integrating the gradients between an all-zero input and the maximizing input. An example for MAG is given in Figure 8.6a and Figure 8.6b. Figures for all considered weapons are provided in Section E. In each Figure, the input vector which maximises each of the two prediction is given, along with the corresponding output neuron activations in the top right. The value of the integrated gradient for each sample index is represented as its background colour, with darker values denoting the highest gradients and white the lowest. We also searched the dataset for inputs with the strongest predictions, and include them in dashed lines as "typical" inputs.

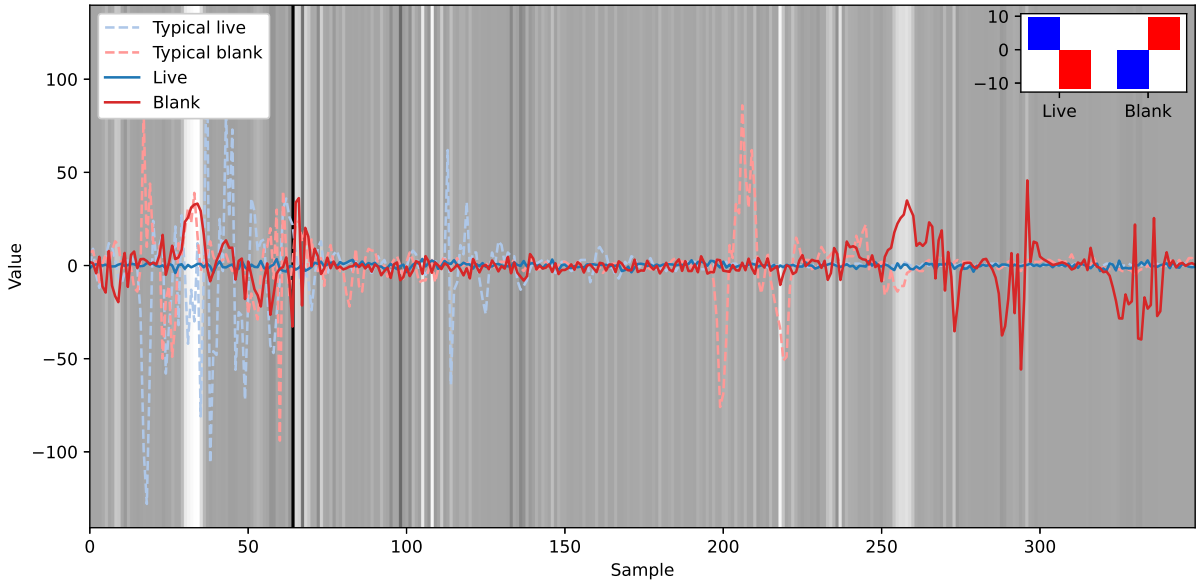
We notice that both problems exploit the full firing cycle. However, there are some sections which are strongly ignored which tend to be located during high intensity events. We assume that the undersampled, highly oscillatory behaviour in these areas is unreliable. For most weapons, the discrimination problem differs from the counting problem in that it focuses strongly on very localised features of the input. This information, which can be computed in a matter of minutes, can allow us to efficiently tune the input vector size ($|x|$), which has a significant influence on the resulting neural network size and inference time.[2] In addition, it also allows us to compare the different types of input, and gain insight into the underlying physical process.

This information could also be used to strategically downsample the input vector. Downsampling heavily the low-gradient features and preserving the high-gradient ones, the dimensionality of the input could be reduced, thus reducing the overall network size and inference time. However, the studied cases have shown that the counting network pays attention to most of the input cycle. Moreover, when using the shared activations structure from Section 7.3, only the area of low importance to both problems can be downsampled, further limiting the possibilities. Another drawback is the increased training time, as the saliency map would need to be determined from a network trained on the full input, before performing a second training phase on the reduced input.

[2]Figure A.36 provides a good example case.



(a) Counting model.



(b) Discrimination model.

Figure 8.6: Saliency maps for MAG.

Chapter 9

Experiments

9.1 Methodology

9.1.1 EDGAR

Randomly initialized neural networks are then trained in groups of 20 for counting networks and 200 for discrimination networks. Optimization is performed through stochastic gradient descent with a fixed 0.9 Nesterov momentum. The learning rate is reduced by a factor of 2 for every 20 epochs without improvement larger than 10^{-5} on the validation loss. A learning phase is stopped if 40 epochs occur without any improvement in the validation loss. A hard minimum of 60 epochs per phase is imposed, along with a maximum of 2000 epochs. The dropout rate for the dense layers is fixed at 35%. Learning occurs in three phases: pretraining on a wider dataset (see Section 6.9), normal training on the target firearm dataset, and quantization-aware training on the same dataset. After each phase, the network parameters yielding the best counting/discrimination performance on the validation set, then the lowest validation loss in case of a tie, are selected to proceed to the next phase or final evaluation. The resulting network is quantized to 8 bits, and runs on the target platform through the TFLite for Microcontrollers framework accelerated by the CMSIS-NN [71] library.

The final hyperparameters used for performance evaluation and deployment are reported in Table 9.1.

9.1.2 Baselines

When available, the primary baselines are the classical FN SmartCore® shot counters [10]. These share the same hardware and sensor and run hand-crafted algorithms which have been tuned on the same input datasets. Their shot counting and discrimination technique were detailed in Section 3.1. These devices have been commercially available since 2012 and have seen continuous performance improvement since then. They are part of the maintenance workflow of several armies worldwide and are available for a wide range of machine guns and assault rifles.

Hyperparameter	Minimi 5.56	Minimi 7.62	MAG
$\|x\|$	232	232	350
w	5ms	5ms	5ms
T_H	126	126	114
T_L	119	119	90
Filters	18	18	8
VAT ϵ	5	5	5
Counting LR	2.01e−3	2.01e−3	3.21e−3
Counting batch size	1	1	32
Min. cycle time	40ms	40ms	40ms
Counting PT	M762,M2P,MAG	M556	M556,M762,M2P
Discrim. LR	2.01e−3	2.01e−3	8.03e−4
Discrim. batch size	8	8	8
D	4	4	4
E	0	0	2
Voted discrim. time	60s	60s	60s
Discrim. PT	M762,M2,MAG	M556	None

Hyperparameter	M2P (EDGAR)	M2P (hybrid)	M2R	M134
$\|x\|$	360	360	360	110
w	5 ms	N/A	10 ms	2 ms
T_H	114	N/A	114	440
T_L	90	N/A	90	360
Filters	16	16	16	18
VAT ϵ	5	N/A	5	5
Counting LR	3.21e−3	N/A	2.57e−3	2.57e−3
Counting batch size	1	N/A	1	1
Counting PT	M556,M762	N/A	M2P,M556, M762,MAG	None
Min. cycle time	40 ms	40 ms	40 ms	13 ms
Discrim. LR	3.21e−3	3.21e−3	N/A	N/A
Discrim. batch size	1	8	N/A	N/A
D	0	N/A	N/A	N/A
E	0	N/A	N/A	N/A
Voted discrim. time	60s	60s	N/A	N/A
Discrim. PT	M556,M762	None	N/A	N/A

Table 9.1: Chosen preprocessing and EDGAR hyperparameters.

Hyperparameter	Minimi 5.56	Minimi 7.62	MAG
$\|x\|$	232	232	368
w	5 ms	5 ms	5 ms
T_H	126	126	114
T_L	119	119	90
Distance metric	CID	CID	CID
Cluster init.	kmeans	kmeans	kmeans
α	17.5	17.5	17.5
γ	90	90	90
Batch size	64	64	32
Pool size	4	4	16
Kernel size	96	96	192
Filters	50	50	50
Pretraining	M762,MAG,M2P	M556	M556,M762,M2P
Hyperparameter	M2P	M2R	M134
$\|x\|$	360	360	110
w	5 ms	10 ms	2 ms
T_H	114	114	440
T_L	90	90	360
Distance metric	CID	CID	CID
Cluster init.	kmeans	kmeans	kmeans
α	1	1	1
γ	1	1	1
Batch size	64	64	64
Pool size	1	1	1
Kernel size	48	48	48
Filters	50	50	50
Pretraining	M556,M762	M556,M762, MAG,M2P	None

Table 9.2: Chosen preprocessing and DTC hyperparameters.

Another interesting baseline is a classifier that predicts shots randomly according to the relative frequency of shots to non-shots in the learning set, which would be the most trivial exploitation of the weak labels over a random predictor. While such a model will report a fairly accurate total count for identical input distributions, it will vary wildly on individual inputs. Such a model will also produce a large number of false positives during non-shot usage, which is highly undesirable.

While the domain of time series classification is well studied, the lack of labels for individual shots prevents us from using supervised techniques. As a baseline, we include for comparison a method of unsupervised clustering on the candidates based on deep learning: Deep Temporal Clustering (DTC) as described by Madiraju et al. [72]. The hyperparameters used are reported in Table 9.2, and further tuning details in Section A.4. The resulting networks are, on average, 60% larger than our proposed models. A comparison is given in Table 9.3.

	EDGAR	DTC
Minimi 5.56	33242	46287
Minimi 7.62	33242	46287
MAG	21298	51113
M2 backplate	44610	44443
M2 barrel	44610	44443
M134	18266	43943

Table 9.3: Network parameters count for EDGAR and DTC models.

9.1.3 False positives

The real-world distribution of shots and non-shots will heavily depend on the end user and their doctrine. For example, some users will frequently manipulate the weapon without shooting during training, perform dry firings and safety checks before or after shooting, and transport it in vehicles generating significant vibrations. Other users will leave the weapon mostly in storage, and do a minimal number of manipulations around the shooting.

In this chapter, we will thus evaluate the counting error on shot data inputs, and evaluate false positives on inputs containing only non-shot data separately. Shot data inputs still comprise mechanical events happening before, during, and after the shots as well as the necessary manipulations and safety checks, and are representative of active usage of the firearm. Non-shot inputs are representative of passive usage and general manipulation of the firearm. The number of erroneous shot detections on these inputs will be reported as FP (False Positives) separately from the main counting error.

9.2 Counting performance

In the experimental results presented in this section, we also report the raw counting result. This metric is ultimately used by the armourer and offers the possibility of compensating for counting errors, which is often beneficial in practice.

9.2.1 Offline benchmark

The results of the best EDGAR model and baselines on the relevant benchmarks are presented in Tables 9.4 to 9.9. The complete benchmark results are available in Section B.1.

Validation set

	EDGAR	Human	DTC	Random
$\bar{E}_{C_{\text{norm}}}$	0	0.054	0.777	0.951
$\bar{E}_C$	0%	1.23%	20.49%	24.75%
FP	0	0	27	32
Count	1292	1275	1552	1255

Ground-truth count: 1292

Table 9.4: Counting benchmark for Minimi 7.62.

Test set

	EDGAR	Human	DTC	Random
$\bar{E}_{C_{\text{norm}}}$	0.022	0.150	0.935	0.754
$\bar{E}_C$	0.56%	3.86%	24.15%	19.46%
FP	2	3	147	174
Count	2736	2616	3112	2759

Ground-truth count: 2745

Table 9.5: Counting benchmark for MAG.

Validation set

	EDGAR	Human	DTC	Random
$\bar{E}_{C_{\text{norm}}}$	0.001	0.051	0.446	1.071
$\bar{E}_C$	0.03%	1.32%	11.52%	27.67%
FP	0	30	40	72
Count	2760	2752	3064	2839

Ground-truth count: 2759

Test set

	EDGAR	Human	DTC	Random
$\bar{E}_{C_{\text{norm}}}$	0.098	0.111	0.984	1.013
$\bar{E}_C$	1.38%	1.57%	13.92%	14.32%
FP	0	0	0	0
Count	2904	2910	3282	2632

Ground-truth count: 2950

Table 9.6: Counting benchmark for Minimi 5.56.

Validation set

	EDGAR	Human	DTC	Random
$\bar{E}_{C_{\text{norm}}}$	0.094	0.000	1.414	0.887
$\bar{E}_C$	2.95%	0.00%	46.24%	28.12%
FP	2	0	394	111
Count	1153	1168	1825	1259

Ground-truth count: 1168

Test set

	EDGAR	Human	DTC	Random
$\bar{E}_{C_{\text{norm}}}$	0.303	0.057	1.245	0.738
$\bar{E}_C$	9.58%	1.80%	39.38%	23.33%
FP	0	0	35	11
Count	462	501	559	485

Ground-truth count: 510

Table 9.7: Counting benchmark for M2 backplate.

	Test set		
	EDGAR	DTC	Random
$E_{C_{\text{norm}}}$	0.011	0.528	0.992
E_C	0.34%	11.72%	21.72%
FP	1	26	31
Count	292	286	306
Ground-truth count: 290			

Table 9.8: Counting benchmark for M134.

	Test set		
	EDGAR	DTC	Random
$E_{C_{\text{norm}}}$	0	0.659	0.703
E_C	0%	27.78%	29.63%
FP	0	54	46
Count	270	333	310
Ground-truth count: 270			

Table 9.9: Counting benchmark for M2 barrel.

We observe that the DTC baseline, working from unsupervised data, performs only slightly better than the weighted random predictor that exploits the global labels proportions. However, the EDGAR technique is able to fully exploit the weak labels and shows an average 94% improvement over the unsupervised baseline.

The technique also shows a significant 85% improvement over the classical FN SmartCore devices for the MAG (Table 9.5). While a similar improvement is seen in the validation set for the Minimi 5.56, the test set only shows a 12% improvement because some configurations present in the test dataset were not included in the learning dataset (Table 9.6). The data presented in Section B.1 show that when the learning set adequately represents the test set (as is the case for MAG, M2 barrel, and M134), the difference between the validation and test $E_{C_{\text{norm}}}$ is at most 0.011. A perfect score is obtained on the Minimi 7.62 validation set (Table 9.4).

The M2 backplate version is the only case where the EDGAR technique performs worse than the baselines, particularly on the test set; however, it remains within the acceptable range for commercial viability of FN SmartCore products, judged at $E_{C_{\text{norm}}} < 0.5$. We attribute the poor performance on the test set, acquired separately, to a bias in the training data. Overall, performance for this version could likely be improved with further tuning. This is supported by the perfect performance obtained with the more recent barrel version of the same weapon. The M134 does not have an FN SmartCore baseline, but shows close to perfect counting performance.

Overall, very few false positives are reported for all weapons. In particular, a large improvement is seen in the Minimi 5.56 case, where the EDGAR model is much less susceptible to false positives on dry firings, which result in a significant number of errors for the classical FN SmartCore on the learning and validation sets. (Dry firings were not present in the test set.)

9.2.2 Field performance

The technique was deployed and evaluated through a series of field tests during which the EDGAR model performed real-time counting. Where applicable, it also performed live/blank discrimination (see Section 9.3.2). The results include both tests performed under controlled conditions and reports from third-party testing. In particular, the Minimi 5.56 (Table 9.12) counter was tested with multiple weapons and a range of different suppressors. It also underwent a prolonged endurance test in which the reported ground-truth count may be approximate. A more detailed presentation of these results is available in Section B.2.

These tests were conducted during development and include results obtained with earlier models and implementations that are inferior to the current system. Nevertheless, the observed field performance remains close to the results reported in offline benchmarks, providing evidence for the generalisation capability of the model.

$\bar{E}_{C_{\text{norm}}}$	0.003
$\bar{E}_C$	0.08%
FP	$^{1}/_{>100}$
Count	1172
Ground-truth count: 1170	

Table 9.10: Counting benchmark for Minimi 7.62.

$\bar{E}_{C_{\text{norm}}}$	0.049
$\bar{E}_C$	1.12%
FP	$^{0}/_{>400}$
Count	3457
Ground-truth count: 3463	

Table 9.11: Counting benchmark for MAG.

$\bar{E}_{C_{\text{norm}}}$	0.048
$\bar{E}_C$	1.25%
FP	$^{0}/_{>600}$
Count	48065
Ground-truth count: 47229	

Table 9.12: Counting benchmark for Minimi 5.56.

$E_{C_{\text{norm}}}$	0.081
E_C	1.26%
FP	$^{0}/_{>750}$
Count	714
Ground-truth count: 715	

Table 9.13: Counting benchmark for M134.

9.3 Discrimination performance

The results of the best EDGAR model and classical FN SmartCore baselines on the relevant benchmarks are presented in Tables 9.14 to 9.17. The complete benchmarks are available in Section B.3. Instance-level labels are known for offline benchmarks. For field performance tests, discrimination results were always checked before changing ammunition type. Thus, individual discrimination errors can be fully identified, and the use of normalized error metrics is not necessary. Instead, a classic F_1 score can be provided.

Classical discrimination algorithms are paired with their respective classical counting algorithms, and EDGAR discrimination models are paired with EDGAR counting models. The only exception is the M2 case, where the EDGAR discrimination model is paired with the classical counting algorithm. The majority voting technique discussed in Section 7.4 is used to improve final performance. In practical use, this would typically be performed on a time basis; however, majority voting is applied here only within each acceleration curve, which represents a more restricted usage. The results are presented with and without this technique, enabling the separate evaluation of its impact and providing insight into single-shot discrimination performance.

Considering that the datasets largely over-represent extreme cases, classical algorithms show acceptable performance for practical discrimination on the Minimi platforms, particularly in 7.62 (Table 9.15 and Table 9.16). The MAG (Table 9.14) and M2 (Table 9.17) cases, however, exhibit much higher error rates, making their use unreliable. In contrast, EDGAR discrimination achieves perfect performance on all datasets when using majority voting. Additional insight can be gained by examining individual predictions, which remain important for sporadic usage. The Minimi 5.56 shows the highest reported individual error rate, with $\bar{E}_D$ at 3.74%, due to the inclusion of configurations not represented in the training data. However, this remains well below the 10% error rate commonly accepted for practical discrimination purposes, and represents an 89% improvement over the classical FN SmartCore error rate. A similar improvement of 96% is observed in the validation set for the 7.62 variant.

Even greater improvements are observed in the MAG and M2 use cases, with respective improvements of 98 and 96%, and error rates falling to around 1% before voting. On average, the EDGAR technique shows a 95% reduction in individual error rate. This not only makes deployment of the ammunition discrimination feature possible, but also promises a high degree of reliability, even in the case of isolated shots.

9.3.1 Offline benchmark

Test set

Classical (raw)						
$\bar{\mathbf{E}}_\mathbf{D}$	**40.13%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**60.02%**
E_{D_B}	23.38%	d_{Bj}	1245	380	F_{1_B}	72.55%
E_{D_K}	56.88%	d_{Kj}	562	426	F_{1_K}	47.49%
Classical (voted)						
$\bar{\mathbf{E}}_\mathbf{D}$	**38.48%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**61.75%**
E_{D_B}	22.40%	d_{Bj}	1261	364	F_{1_B}	73.64%
E_{D_K}	54.55%	d_{Kj}	539	449	F_{1_K}	49.86%
EDGAR (raw)						
$\bar{\mathbf{E}}_\mathbf{D}$	**0.73%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**99.33%**
E_{D_B}	0.34%	d_{Bj}	1739	6	F_{1_B}	99.51%
E_{D_K}	1.11%	d_{Kj}	11	978	F_{1_K}	99.14%
EDGAR (voted)						
$\bar{\mathbf{E}}_\mathbf{D}$	**0.00%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**100%**
E_{D_B}	0.00%	d_{Bj}	1745	0	F_{1_B}	100%
E_{D_K}	0.00%	d_{Kj}	0	989	F_{1_K}	100%

Table 9.14: Discrimination benchmark for MAG.

Validation set

Classical (raw)

$\bar{\mathbf{E}}_\mathbf{D}$	**25.52%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**72.61%**
E_{D_B}	23.74%	d_{Bj}	1439	448	F_{1_B}	80.98%
E_{D_K}	27.31%	d_{Kj}	228	607	F_{1_K}	64.23%

Classical (voted)

$\bar{\mathbf{E}}_\mathbf{D}$	**17.06%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**82.86%**
E_{D_B}	10.76%	d_{Bj}	1684	203	F_{1_B}	89.43%
E_{D_K}	23.35%	d_{Kj}	195	640	F_{1_K}	76.28%

EDGAR (raw)

$\bar{\mathbf{E}}_\mathbf{D}$	**0.60%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**99.40%**
E_{D_B}	0.36%	d_{Bj}	1914	7	F_{1_B}	99.64%
E_{D_K}	0.83%	d_{Kj}	7	832	F_{1_K}	99.17%

EDGAR (voted)

$\bar{\mathbf{E}}_\mathbf{D}$	**0.00%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**100%**
E_{D_B}	0.00%	d_{Bj}	1921	0	F_{1_B}	100%
E_{D_K}	0.00%	d_{Kj}	0	839	F_{1_K}	100%

Test set

Classical (raw)

$\bar{\mathbf{E}}_\mathbf{D}$	**33.55%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**66.40%**
E_{D_B}	27.25%	d_{Bj}	1279	479	F_{1_B}	73.17%
E_{D_K}	39.84%	d_{Kj}	459	693	F_{1_K}	59.64%

Classical (voted)

$\bar{\mathbf{E}}_\mathbf{D}$	**10.81%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**90.63%**
E_{D_B}	0.00%	d_{Bj}	1758	0	F_{1_B}	93.39%
E_{D_K}	21.61%	d_{Kj}	249	903	F_{1_K}	87.88%

EDGAR (raw)

$\bar{\mathbf{E}}_\mathbf{D}$	**3.74%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**96.22%**
E_{D_B}	3.13%	d_{Bj}	1702	55	F_{1_B}	97.01%
E_{D_K}	4.36%	d_{Kj}	50	1097	F_{1_K}	95.43%

EDGAR (voted)

$\bar{\mathbf{E}}_\mathbf{D}$	**0.00%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**100%**
E_{D_B}	0.00%	d_{Bj}	1757	0	F_{1_B}	100%
E_{D_K}	0.00%	d_{Kj}	0	1147	F_{1_K}	100%

Table 9.15: Discrimination benchmark for Minimi 5.56.

Validation set

Classical (raw)						
$\bar{\mathbf{E}}_{\mathbf{D}}$	**8.89%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_{\mathbf{1}}$	**90.45%**
E_{D_B}	6.25%	d_{Bj}	870	58	F_{1_B}	94.67%
E_{D_K}	11.53%	d_{Kj}	40	307	F_{1_K}	86.24%
Classical (voted)						
$\bar{\mathbf{E}}_{\mathbf{D}}$	**3.23%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_{\mathbf{1}}$	**94.35%**
E_{D_B}	6.47%	d_{Bj}	868	60	F_{1_B}	96.66%
E_{D_K}	0.00%	d_{Kj}	0	347	F_{1_K}	92.04%
EDGAR (raw)						
$\bar{\mathbf{E}}_{\mathbf{D}}$	**0.39%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_{\mathbf{1}}$	**99.61%**
E_{D_B}	0.21%	d_{Bj}	941	2	F_{1_B}	99.79%
E_{D_K}	0.57%	d_{Kj}	2	347	F_{1_K}	99.43%
EDGAR (voted)						
$\bar{\mathbf{E}}_{\mathbf{D}}$	**0.00%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_{\mathbf{1}}$	**100%**
E_{D_B}	0.00%	d_{Bj}	943	0	F_{1_B}	100%
E_{D_K}	0.00%	d_{Kj}	0	349	F_{1_K}	100%

Table 9.16: Discrimination benchmark for Minimi 7.62.

Validation set

Classical (raw)

$\bar{\mathbf{E}}_\mathbf{D}$	**39.54%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**60.46%**
E_{D_B}	33.78%	d_{Bj}	443	226	F_{1_B}	66.22%
E_{D_K}	45.29%	d_{Kj}	226	273	F_{1_K}	54.71%

Classical (voted)

$\bar{\mathbf{E}}_\mathbf{D}$	**30.90%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**69.23%**
E_{D_B}	23.92%	d_{Bj}	509	160	F_{1_B}	74.47%
E_{D_K}	37.88%	d_{Kj}	189	310	F_{1_K}	63.98%

EDGAR (raw)

$\bar{\mathbf{E}}_\mathbf{D}$	**1.00%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**99.04%**
E_{D_B}	0.60%	d_{Bj}	665	4	F_{1_B}	99.18%
E_{D_K}	1.40%	d_{Kj}	7	492	F_{1_K}	98.89%

EDGAR (voted)

$\bar{\mathbf{E}}_\mathbf{D}$	**0.00%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**100%**
E_{D_B}	0.00%	d_{Bj}	669	0	F_{1_B}	100%
E_{D_K}	0.00%	d_{Kj}	0	499	F_{1_K}	100%

Test set

Classical (raw)

$\bar{\mathbf{E}}_\mathbf{D}$	**45.05%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**53.21%**
E_{D_B}	62.31%	d_{Bj}	98	162	F_{1_B}	46.12%
E_{D_K}	27.80%	d_{Kj}	67	174	F_{1_K}	60.31%

Classical (voted)

$\bar{\mathbf{E}}_\mathbf{D}$	**54.74%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**43.79%**
E_{D_B}	69.23%	d_{Bj}	80	180	F_{1_B}	36.61%
E_{D_K}	40.25%	d_{Kj}	97	144	F_{1_K}	50.97%

EDGAR (raw)

$\bar{\mathbf{E}}_\mathbf{D}$	**1.58%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**98.58%**
E_{D_B}	0.38%	d_{Bj}	259	1	F_{1_B}	98.85%
E_{D_K}	2.78%	d_{Kj}	5	175	F_{1_K}	98.31%

EDGAR (voted)

$\bar{\mathbf{E}}_\mathbf{D}$	**0.00%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**100%**
E_{D_B}	0.00%	d_{Bj}	260	0	F_{1_B}	100%
E_{D_K}	0.00%	d_{Kj}	0	180	F_{1_K}	100%

Table 9.17: Discrimination benchmark for M2 backplate.

9.3.2 Field performance

The results presented below show the field discrimination performance obtained during the same field tests described in Section 9.2.2.

We observe very good performance on the MAG (Table 9.18), with only a single 15-round burst being misclassified. Perfect discrimination performance was obtained with the Minimi 7.62 (Table 9.20). Performance on the Minimi 5.56 (Table 9.19) is noticeably lower, primarily due to the use of a prototype discrimination model that was developed several years earlier and is inferior to the current version.

EDGAR (voted)

$\bar{\mathbf{E}}_\mathbf{D}$	**1.59%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**99.07%**
E_{D_B}	0.00%	d_{Bj}	3492	0	F_{1_B}	99.75%
E_{D_K}	3.18%	d_{Kj}	15	457	F_{1_K}	98.39%

Table 9.18: Field performance in discrimination for MAG.

EDGAR (voted)

$\bar{\mathbf{E}}_\mathbf{D}$	**10.84%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**93.23%**
E_{D_B}	0.05%	d_{Bj}	46576	22	F_{1_B}	99.76%
E_{D_K}	21.64%	d_{Kj}	198	717	F_{1_K}	86.70%

Table 9.19: Field performance in discrimination for Minimi 5.56.

EDGAR (voted)

$\bar{\mathbf{E}}_\mathbf{D}$	**0%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**100%**
E_{D_B}	0%	d_{Bj}	540	0	F_{1_B}	100%
E_{D_K}	0%	d_{Kj}	0	541	F_{1_K}	100%

Table 9.20: Field performance in discrimination for Minimi 7.62.

9.3.3 Suppressors

In Section 7, we studied live/blank discrimination. Another interesting application is to discriminate ammunition fired with a suppressor. In a preliminary experiment, we acquired data from 465 shots fired on a Minimi 5.56 equipped with an Oceania Defense BF600 suppressor. The discrimination network was then trained to classify three classes: live rounds, blank rounds, and live rounds with suppressor. The results are included in Table 9.21. With majority voting, suppressed shots are perfectly discriminated. However, since the sample size was small, with low variance in weapon configurations and only a single type of suppressor, we cannot be certain that the network performed this discrimination based specifically on the suppressor, rather than on another external factor.

(a) Without majority voting

		Predicted label		
		Live	Blank	Suppr.
True label	Live	1805	8	2
	Blank	6	742	0
	Suppr.	9	2	50

(b) With majority voting

		Predicted label		
		Live	Blank	Suppr.
True label	Live	1815	0	0
	Blank	0	748	0
	Suppr.	0	0	61

Table 9.21: Confusion matrices for Minimi 5.56 suppressed shots discrimination.

A larger, albeit still preliminary, dataset was acquired for the FN EVOLYS®. For the 7.62 variant, 2145 shots were recorded with a suppressor and 2415 without. For the 5.56 variant, 2425 shots were recorded with a suppressor and 4994 without. Results on the corresponding test set are presented in Table 9.22. The model achieved perfect discrimination prior to majority voting. Further investigation is planned to confirm these results through an expanded testing campaign, including field trials.

(a) EVOLYS 7.62

		Predicted label		
		Live	Blank	Suppr.
True label	Live	300	0	0
	Blank	0	0	0
	Suppr.	0	0	360

(b) EVOLYS 5.56

		Predicted label		
		Live	Blank	Suppr.
True label	Live	300	0	0
	Blank	0	345	0
	Suppr.	0	0	300

Table 9.22: Confusion matrices for EVOLYS 5.56 and EVOLYS 7.62 suppressed shots discrimination (without majority voting).

9.4 Resource consumption

9.4.1 EDGAR model footprint

Table 9.23 provides the resource requirements of the different EDGAR models on the target hardware (64 MHz ARM Cortex-M4F). The preprocessing time reported is the average time per sample.

	Minimi 5.56	Minimi 7.62	MAG	M2P	M2R	M134
Preprocessing [µs]	1.5	1.5	1.5	N/A	5.2	4.3[1]
Counting [ms]	61	61	34	N/A	81	27
Discrimination [ms]	18	18	26	84	N/A	N/A
RAM [kB]	12	12	11	14	14	7

Table 9.23: Inference time and RAM consumption of EDGAR models on 64 MHz ARM Cortex-M4F.

9.4.2 Shot counter energy consumption and lifespan

By adopting certain assumptions, it becomes possible to estimate the practical lifespan of an electronic shot counter. Regarding shot data, we assume that the device experiences 14,000 bursts, each averaging 5 shots. Concerning non-shot data, we assume the weapon is active approximately 100 days annually, with significant acceleration events occurring 60 times per day, each sufficient to wake up the device. These values are informed by empirical field-return data. Furthermore, communication and data retrieval are assumed to occur weekly. Accounting for typical battery degradation under these conditions, the device lifespan is estimated to be approximately 18 years with the considered battery solution.

Figure 9.1 presents the energy consumption distribution for a real device under these operational assumptions. It reveals that the vast majority of energy consumption results from a background current averaging 2.9 µA. When active, the primary energy usage arises from recording data following bursts of activity, due to significant current draw associated with flash memory write operations. Notably, only approximately 12% of the available energy is consumed for signal analysis.

Under conditions that do not employ a dedicated sleep mode, provided the model inference time remains within the burst rate constraints of the firearm (see Section 6.5), the inference process does not induce additional power consumption.

[1] M134 implementation used an older, non-optimised candidate selection algorithm.

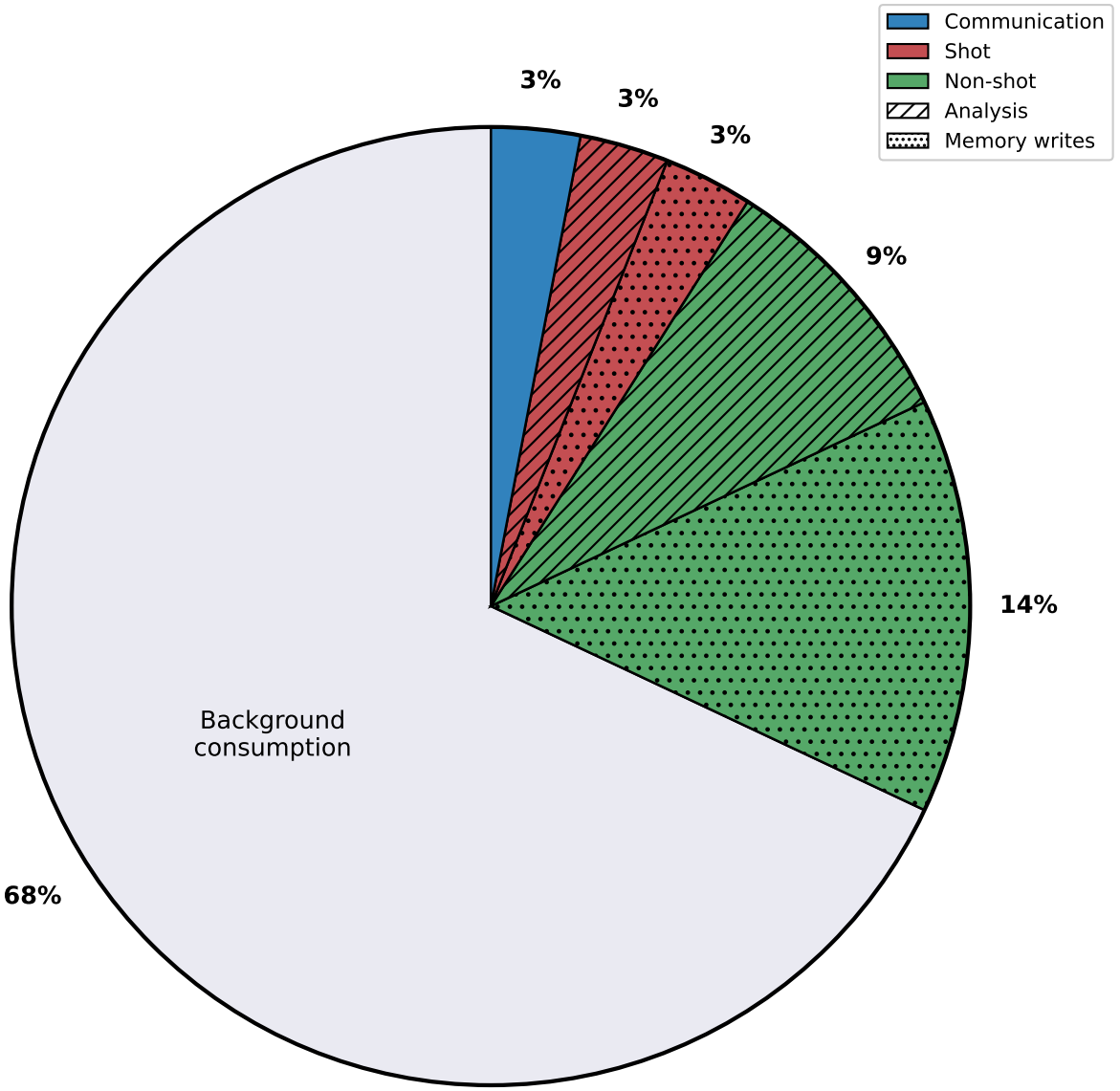


Figure 9.1: Distribution of energy expenditure for a typical usage scenario of a shot counter, excluding battery degradation.

Chapter 10

Conclusion and future work

10.1 Conclusion

We started this thesis by establishing the economic and operational benefits of electronic shot counters for army logistics. We examined the different types of commercially available shot counters, and then reviewed the literature on the topic, noting a recent surge of interest as embedded neural networks became a viable solution. The use of machine learning techniques makes it practical to develop algorithms accounting for a broader range of configurations, allows faster deployment, improved accuracy, and greater robustness in counting performance. We also noted the increasing interest in live/blank discrimination, and suppressor usage detection.

However, the collection of comprehensive, fully-labelled datasets is highlighted by multiple sources as particularly challenging in this field. As a result, currently available approaches either require significant involvement from human experts or are limited to relatively small datasets. Yet, in Section 8.2 we demonstrated that a minimum of 1000–2000 diversified shots are necessary to capture the behaviour of a modern firearm, and that the problem responds well to the inclusion of additional data.

To address this, we designed a machine learning technique capable of working directly on data labelled only with the total numbers of events in a time series, while still providing individual shot detection and localisation. In contrast to full supervision, these weak labels are obtained cheaply by annotating the time series with the ammunition containers used, which correspond to known numbers of rounds. This enables full automation of the process of creating shot counting algorithms. Moreover, by leveraging a continuously growing database comprising over 40,000 shots made usable by our technique, we are able to achieve shot counting performance comparable to or exceeding the current state of the art.

By restructuring the neural network and leveraging the similarities between the counting and discrimination sub-problems, we then showed how the technique can be efficiently extended to provide ammunition type discrimination or suppressor usage detection with only an 8% computational overhead, while achieving classification performance far above

the current state of the art. This approach also simplifies training, and provides control over the relevant trade-offs. By showing a 95% average improvement in individual error rate over the state of the art, our technique enables deployment on platforms where practical, reliable ammunition discrimination was previously considered infeasible. To our knowledge, this is not only the first comprehensive description of a machine learning technique for this domain of application, but also the first truly generic technique applicable to a broader range of weapon configuration discrimination tasks.

After performing an extensive study of the effect of the different hyperparameters involved in our technique, we observed that their optimal values for different firearms lie within a relatively narrow range with a single minimum. Future developments can thus be finely tuned in a few steps by starting from the recommended values and using the recommended tuning procedure based on orthogonal array testing. We also described how to evaluate the exact inference time budget of the model and advised how to tune it to fit within these constraints.

We demonstrated how the model can be used as a tool to evaluate data acquisition scenarios. In particular, we used it to show that the performance of both the counting and discrimination sub-problems relative to dataset size can be modelled by a power law. We also observed that discrimination is initially harder, but also more responsive to larger datasets, which explains the strong performance of our approach on live/blank discrimination. We then explored the conditions under which this law can be used for extrapolation, allowing one to predict the number of rounds required to reach a desired level of counting performance. We are also able to detect when the available dataset saturates the model. This improves the organisational aspects of the frequent process of dataset creation.

Using a white-box approach and the computation of saliency maps, we demonstrated how the model can be used to investigate the physical differences between the different acceleration events, or between ammunition types, that make classification possible. This information can be used both for model tuning and to gain insight into the physical processes occurring within the firearm.

Finally, we showed that our technique can yield models achieving this level of performance in real-time on microcontrollers while using less than 12 kB of RAM and 80 ms per inference on a 64 MHz Cortex-M4F. This corresponds to an energy consumption of approximately 1.36 mJ on our test hardware. By integrating our models in production-ready devices, we confirmed their real-time predictive performance under operational conditions. Through this field testing, we studied the critical hardware-related challenges faced by embedded shot counters, particularly the selection of batteries resistant to extreme acceleration and the accurate characterisation of accelerometer behaviour under firearm-specific conditions, such as unpredictable early wake-ups and variable sampling frequencies.

With its potential for adaptability and low error rates, this approach holds promise to set a new standard for electronic shot counter capabilities, and is already deployed in multiple commercial applications. This will enable armourers worldwide to conduct tailored maintenance, as the system provides precise, automated recommendations based on ammunition counts, reducing costs, preventing incidents, and increasing operational availability.

10.2 Future work

10.2.1 Shared convolution results

We have seen in Section 6.3 that candidates closely following a confirmed shot can be filtered out, leading to reduced overall computational requirements for inference. However, in general it is possible to need to evaluate several candidates that are close to each other without being able to perform any filtering, for example when there is no actual shot or when it is the last candidate to be evaluated. In Section 6.4, we have seen that the suggested network starts with a series of convolutional layers using filters of relatively small kernel size. The first two blocks of convolution and pooling operations account for the vast majority (about 95% for practical input sizes, see Figure 7.5) of the inference computational cost.

In the case of close candidates, the corresponding output vectors will show a significant overlap. The result of the convolution operations on this overlap (barring some relatively small padding effects at the edges) will be identical for all candidates. For the kind of embedded platforms considered in this work, the processing power is a much more limiting factor than memory footprint. We can thus perform a space/speed trade-off by caching the results of convolutional operations to be reused by a potential overlapping candidate. If the available memory does not allow caching the entire results, a possible compromise is to cache only partial results, starting with those at the end of the input vector, which have the best chance of overlap.

10.2.2 Early out-of-distribution detection

A 2021 paper by Sumaiya et al. [73] proposed a technique aiming to reduce the number of times an edge device uploads sensor data to a cloud-based server to be evaluated by a CNN. This is done by exporting only the shallow layers of the CNN to edge devices, and detecting whether the input is out-of-distribution (OOD) based on a subset of the features produced by those layers. While our approach runs entirely on the edge device, a similar method could be applied to perform OOD detection after the shallow layers of the model. If a candidate is detected as OOD the inference can then be aborted early, reducing overall computational requirements.

10.2.3 Shooting plan optimisation

In Chapter 8, we used the generated models to establish the minimum dataset size needed for representativeness and target performance under the assumption that every weapon configuration is equally sampled. However, some external factors likely exert far less influence than others. Going forward, we will quantify the individual impact of each variable and pinpoint those whose under or non-representation does not impair accuracy. By pruning these low-impact variables from the shooting plan, we can simplify data acquisition and streamline the entire process.

10.2.4 Predictive maintenance

This thesis focuses mainly on detecting events for preventive maintenance purposes. In the future, we aim to study how we can extend this to predictive maintenance, where we detect anomalies in the signal indicating that a weapon is in urgent need of maintenance before a failure occurs.

From shot detection, we can easily derive a burst rate. Tracking the evolution of the burst rate is already commonly used for predictive maintenance. Maintenance is warranted when the burst rate trend indicates that it might go either above or below acceptable bounds, either of which can lead to failures.

Shot discrimination could also be used for this purpose. In this application, the network would discriminate between nominal and degraded shots. The drawback of this approach is the difficulty of acquiring degraded input data, as weapon failures are both rare and costly to induce. We intend to explore this possibility in the coming year by acquiring data from barrels of varying age, to assess whether the network can reliably distinguish between newer and more worn barrels.

10.2.5 Other applications

Although this thesis has focused on electronic shot counters, the fundamental method of learning instance-level detectors from weak labels extends to any domain where total counts are easily obtained but per-event annotation is impractical. By using only overall tallies, the approach recovers individual detections without manual time-stamping.

For example, in manufacturing quality control, batch-level defect counts can train a model to flag each faulty unit during production without bounding-box annotations. In wildlife acoustic surveys, simple call tallies enable learning detectors that timestamp each animal vocalization. In microseismic monitoring, daily tremor counts allow the method to localize individual events in continuous seismic traces.

In all these scenarios, readily available aggregate counts serve as effective weak labels, while the resulting models deliver scalable, fine-grained, on-device event detection eliminating the need for exhaustive manual labelling.

Chapter 11

List of acronyms

- **AI**: Artificial Intelligence
- **BLE**: Bluetooth Low Energy
- **BMS**: Battlefield Management System
- **BR**: Burst Rate
- **CID**: Complexity-Invariant Similarity
- **CMSIS**: Cortex Microcontroller Software Interface Standard
- **CNN**: Convolutional Neural Network
- **COR**: Correlation-based Similarity
- **DT**: Decision Tree
- **DTC**: Deep Temporal Clustering
- **DTW**: Dynamic Time Warping
- **EDGAR**: Embedded Detection of Gunshots by AI in Real-time
- **FIR**: Finite Impulse Response
- **FP**: False Positive
- **FSM**: Finite State Machine
- **GAN**: Generative Adversarial Network
- **GPU**: Graphics Processing Unit
- **GRU**: Gated Recurrent Unit

- **IoT**: Internet of Things
- **LR**: Learning Rate
- **LSTM**: Long Short-Term Memory
- **M134**: M134 Minigun
- **M2**: FN® M2HB-QCB
- **M2P**: FN® M2HB-QCB backPlate sensor
- **M2R**: FN® M2HB-QCB baRrel sensor
- **M556**: FN Minimi® 5.56
- **M762**: FN Minimi® 7.62
- **MAG**: FN MAG®
- **MEMS**: Micro-ElectroMechanical Systems
- **OOD**: Out-of-Distribution
- **PT**: PreTraining
- **QAT**: Quantization-Aware Training
- **RAM**: Random Access Memory
- **RNN**: Recurrent Neural Network
- **SIMD**: Single Instruction Multiple Data
- **SVM**: Support Vector Machine
- **TDF**: Time-Dilation Factor
- **TNN**: Transformer Neural Network
- **VAT**: Virtual Adversarial Training

Appendix A: Hyperparameter tuning

Section 6.11 outlines the procedure for fine-tuning model hyperparameters, beginning with the recommended settings listed in Table 9.1. These values were identified through an extensive optimisation process detailed in this chapter, which involved iteratively analysing how model performance responds to variations in individual hyperparameters. This appendix provides a comprehensive account of that process to support future fine-tuning efforts. It shows that most hyperparameters exhibit a single global minimum and highlights the sensitivity of performance to their respective values.

A.1 ReLU maximum

This section presents supplementary figures associated with Section 6.6.

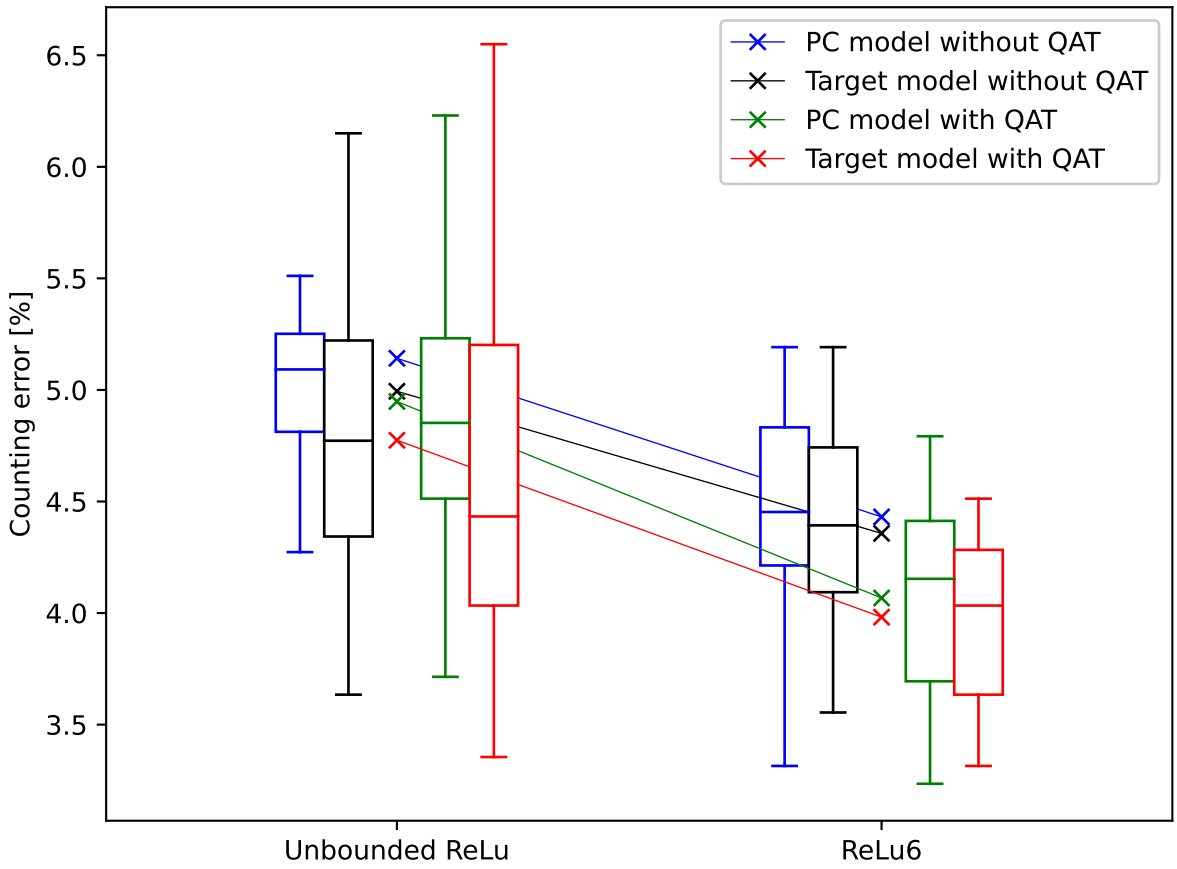


Figure A.1: Comparison between unbounded and bounded ReLU activations on prediction performance.

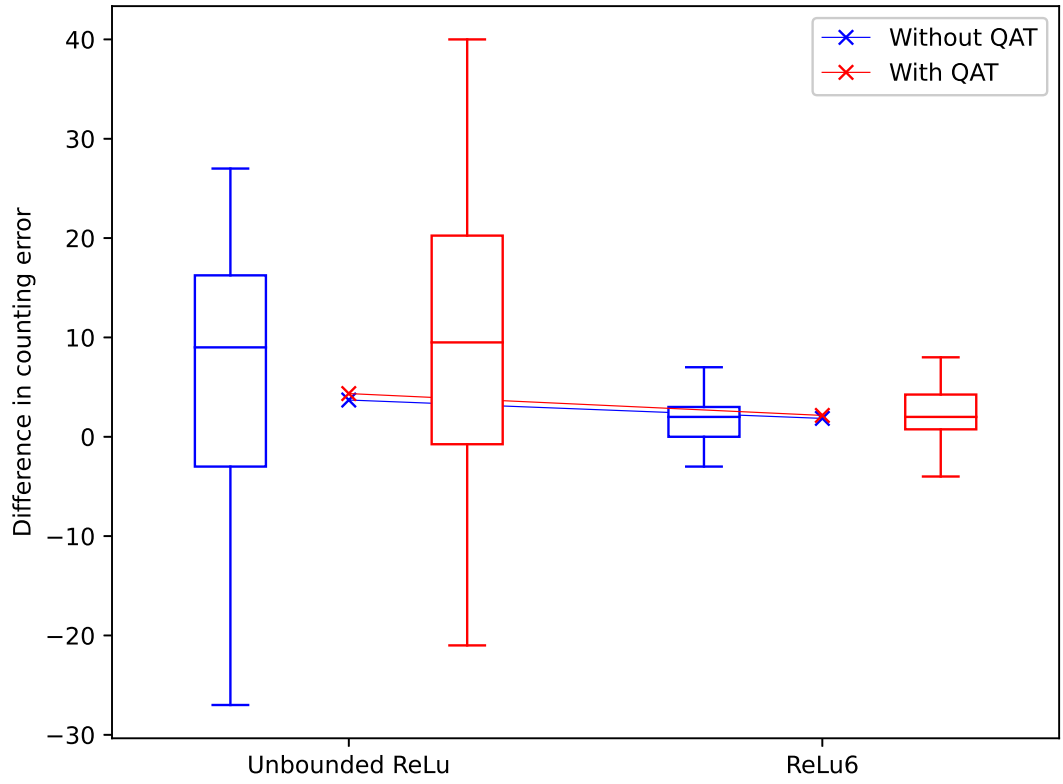


Figure A.2: Comparison between unbounded and bounded ReLU activations on prediction consistency. The effect is measured in the difference in counting error between the unquantized and quantized models, computed on the target device. Bounded activations show much more consistent results, meaning that quantization has less impact on models with bounded activations.

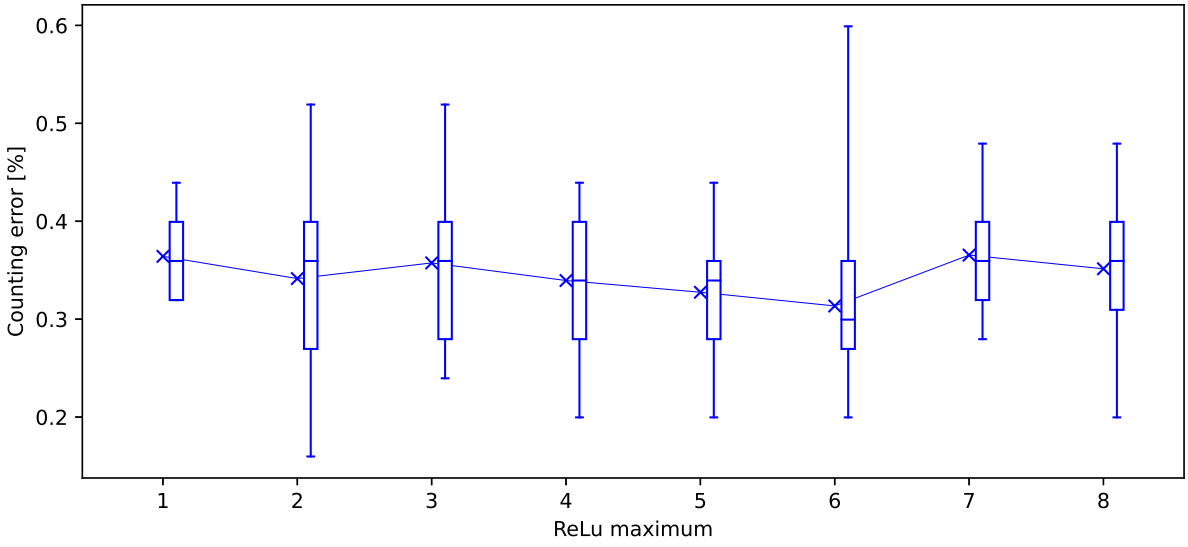


Figure A.3: Maximum value of the ReLU activation. Best results are obtained with 6.

A.2 Loss function

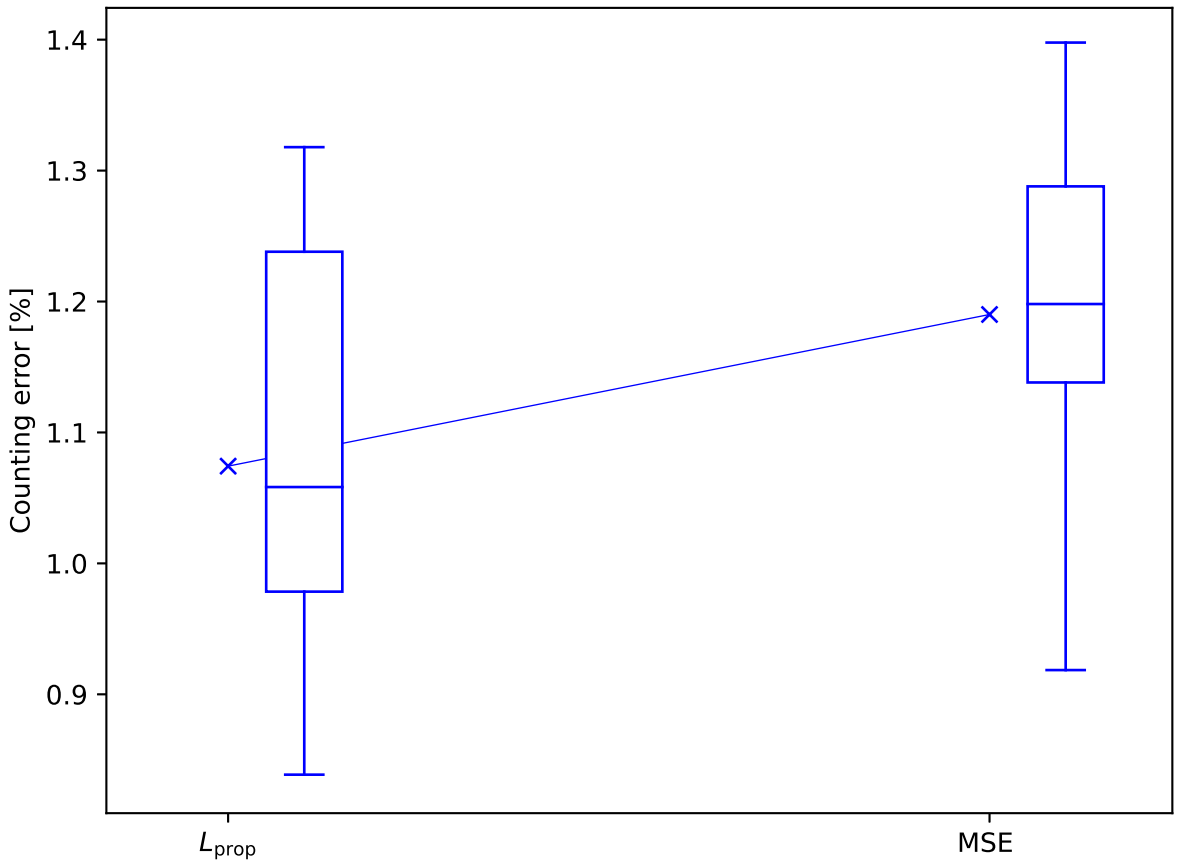


Figure A.4: Comparing the $\mathcal{L}_{\text{prop}}$ formulation of Equation 6.6 and a mean-square error (MSE) between $\mathbf{p}_i$ and $\hat{\mathbf{p}}_i$.

A.3 EDGAR hyperparameters

A.3.1 Counting

Learning Rate (LR)

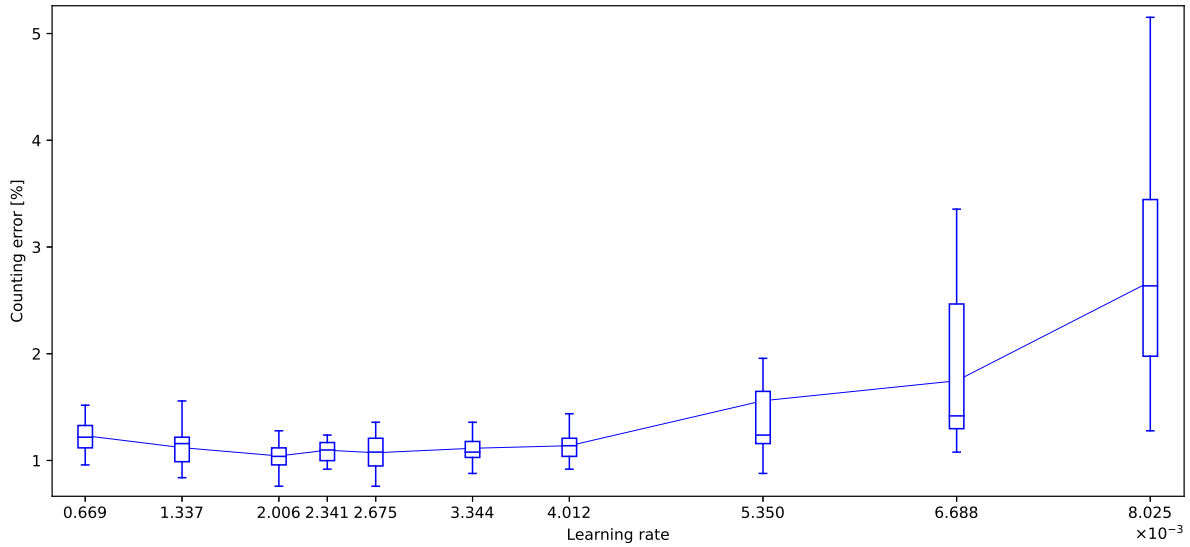


Figure A.5: Exploring the learning rate (LR) parameter space for Minimi 5.56.

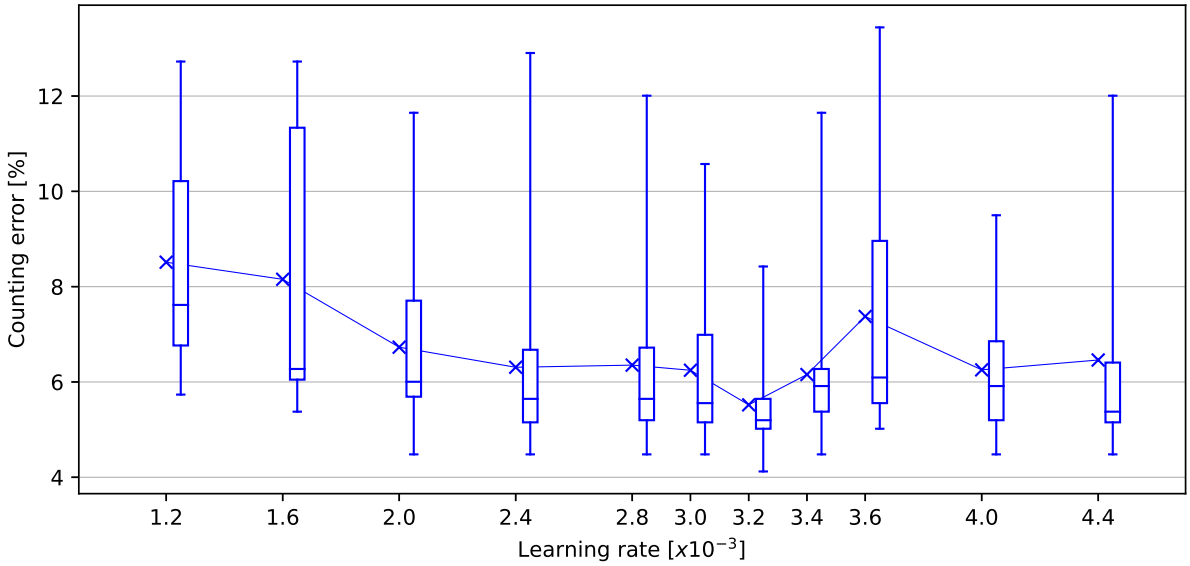


Figure A.6: Exploring the learning rate (LR) parameter space for M2 backplate.

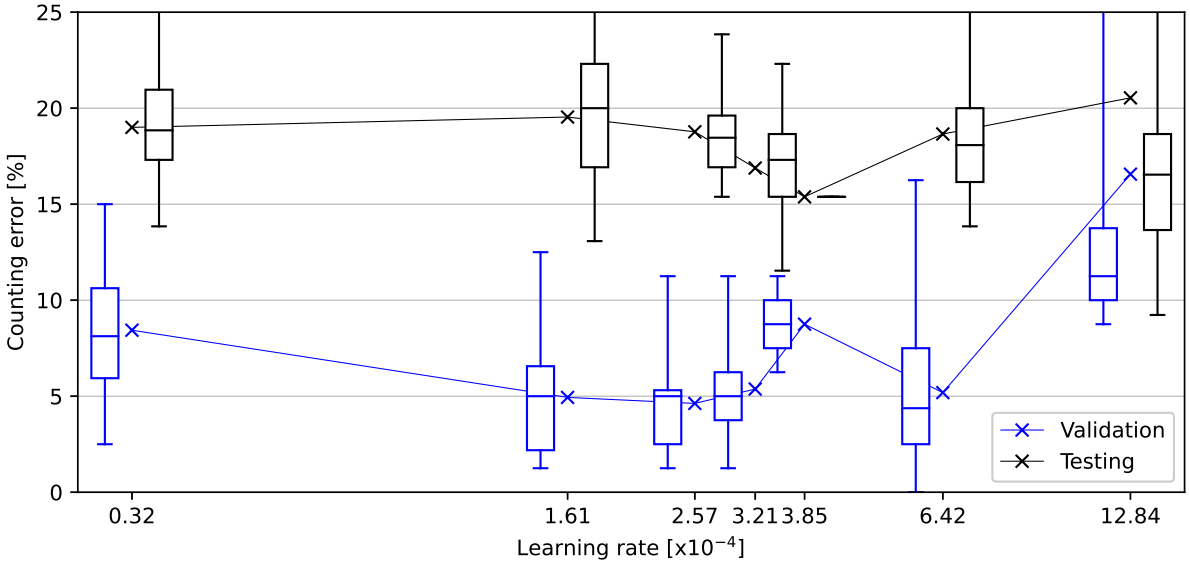


Figure A.7: Exploring the learning rate (LR) parameter space for M2 barrel.

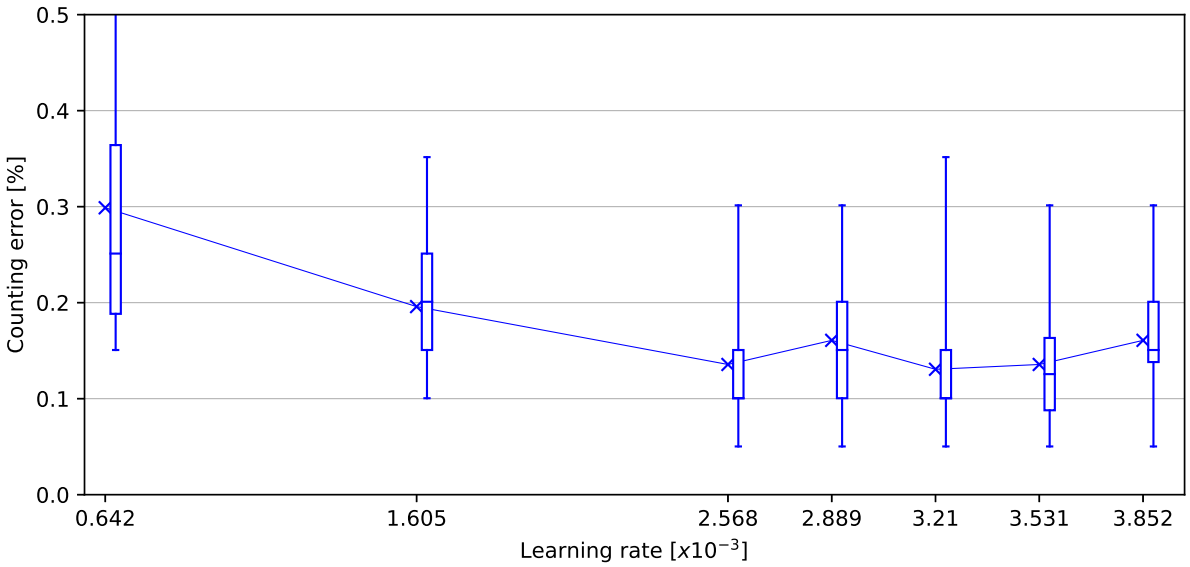


Figure A.8: Exploring the learning rate (LR) parameter space for MAG.

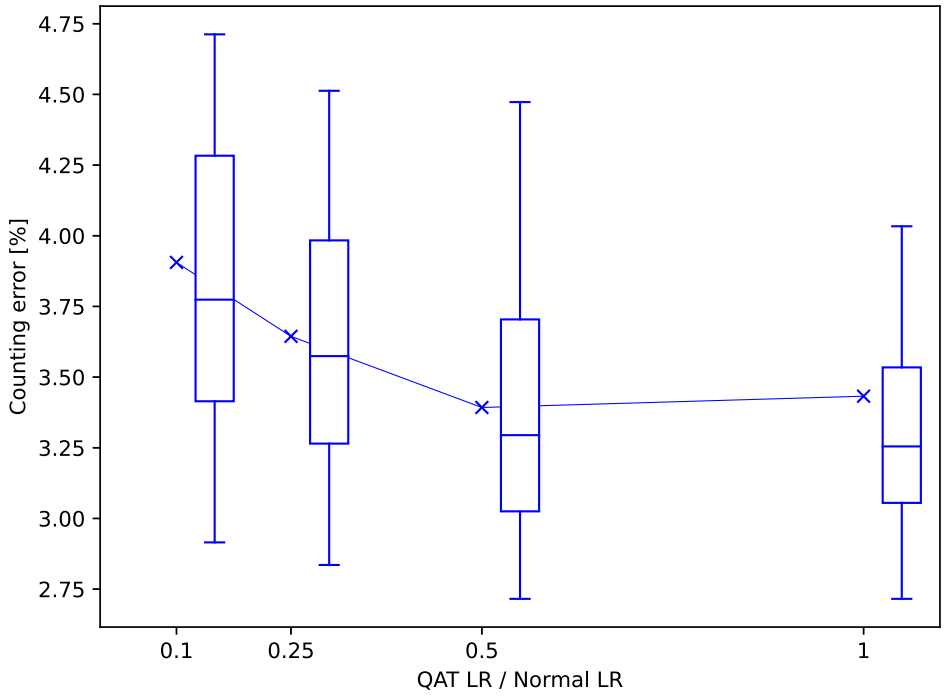


Figure A.9: Experiments lowering the QAT learning rate (LR) for Minimi 5.56. Best performance is actually obtained by keeping LR at a value close to the initial LR.

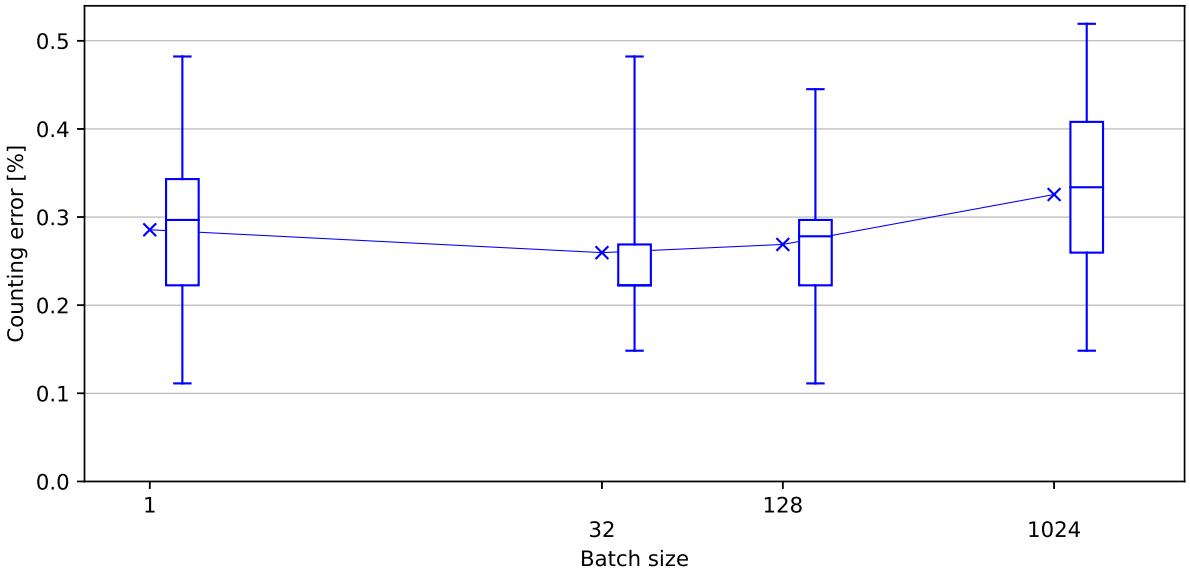


Figure A.10: Exploring the batch size parameter space for MAG. A batch size around 32 seems optimal, but the difference is not significant.

Input size ($|x|$)

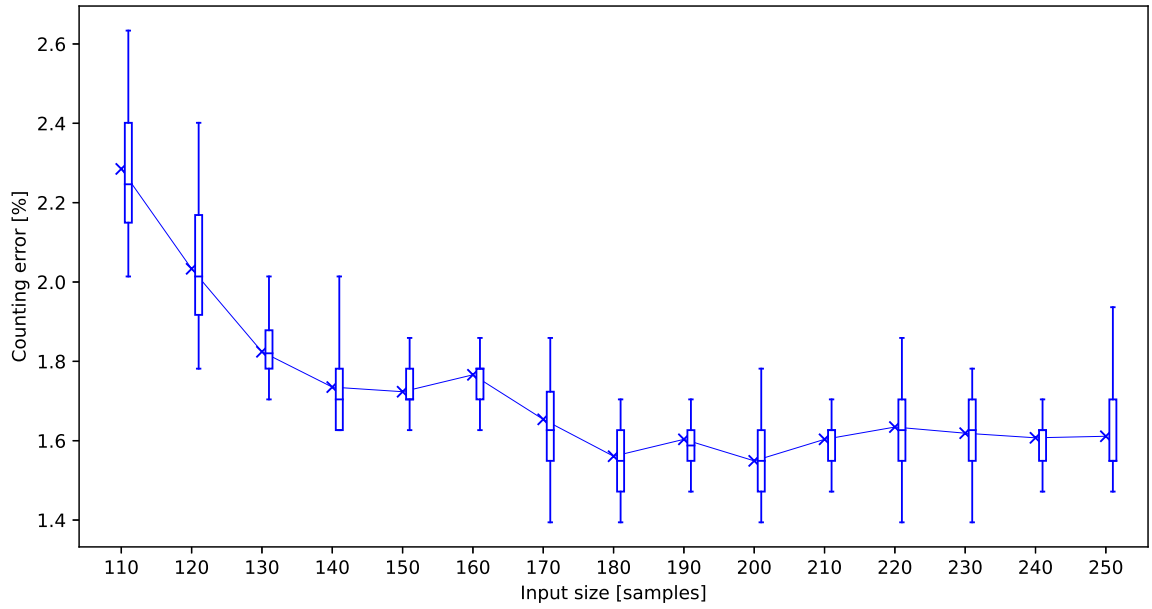


Figure A.11: Exploring the input size ($|x|$) parameter space for Minimi 7.62. Performance improves until the 200 samples (31ms) range, after which no significant improvement is shown. As the neural network size is proportional to the input size, the smallest sufficiently performant value of $|x|$ will be preferred.

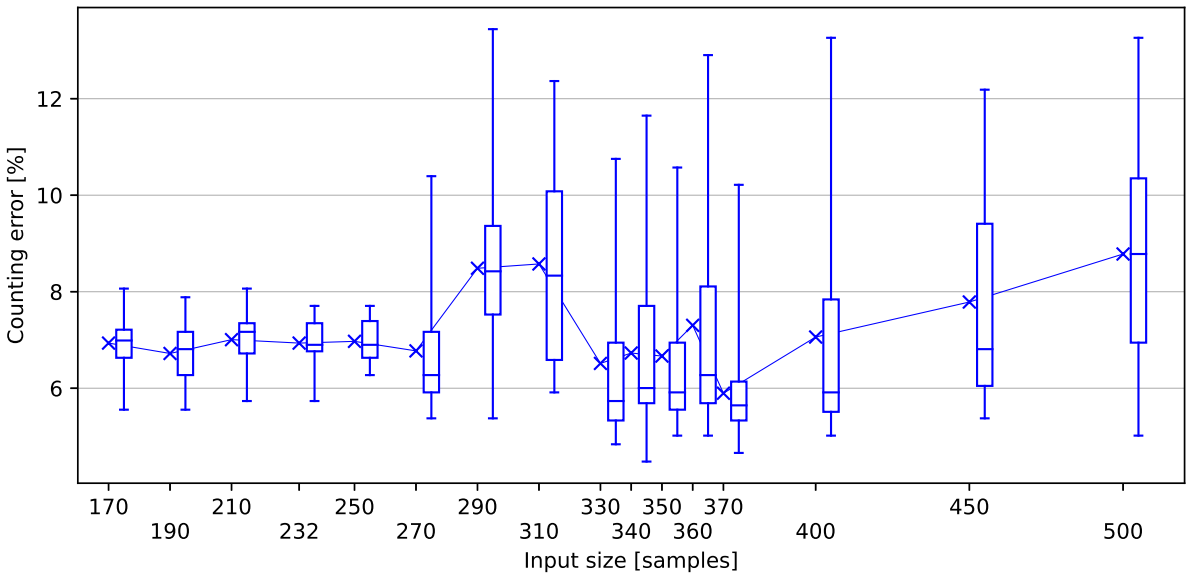


Figure A.12: Exploring the input size ($|x|$) parameter space for M2 backplate. Best performance is shown around 350 samples (55ms). However, it presents multiple local minima. While not optimal, small input sizes can present a worthwhile performance/inference time tradeoff.

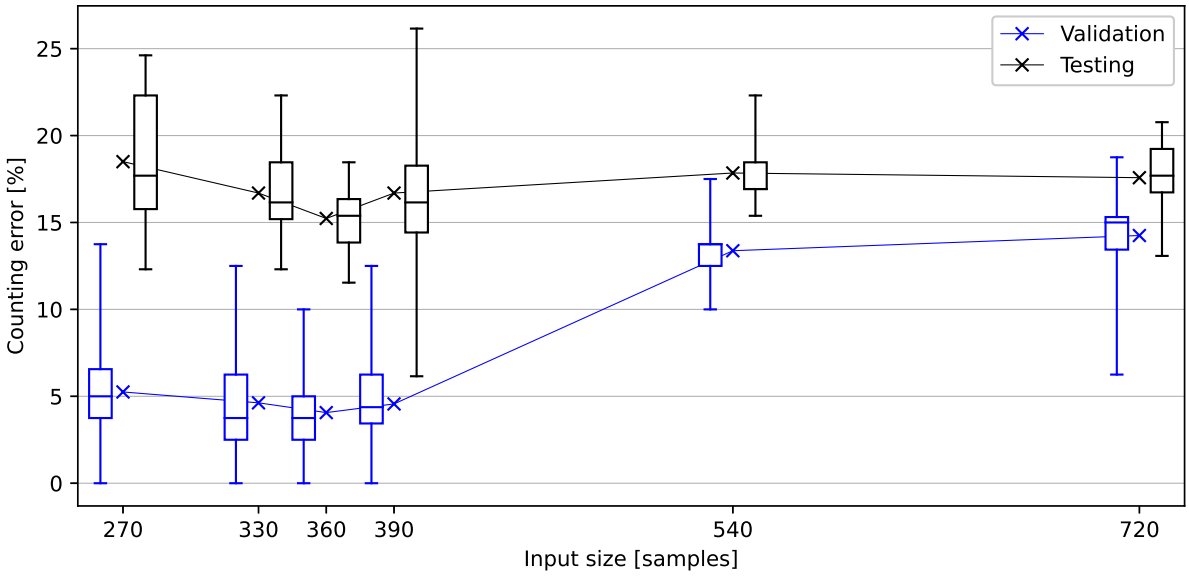


Figure A.13: Exploring the input size ($|x|$) parameter space for M2 barrel. Best performance is obtained around $|x| = 360$, similar to the backplate dataset.

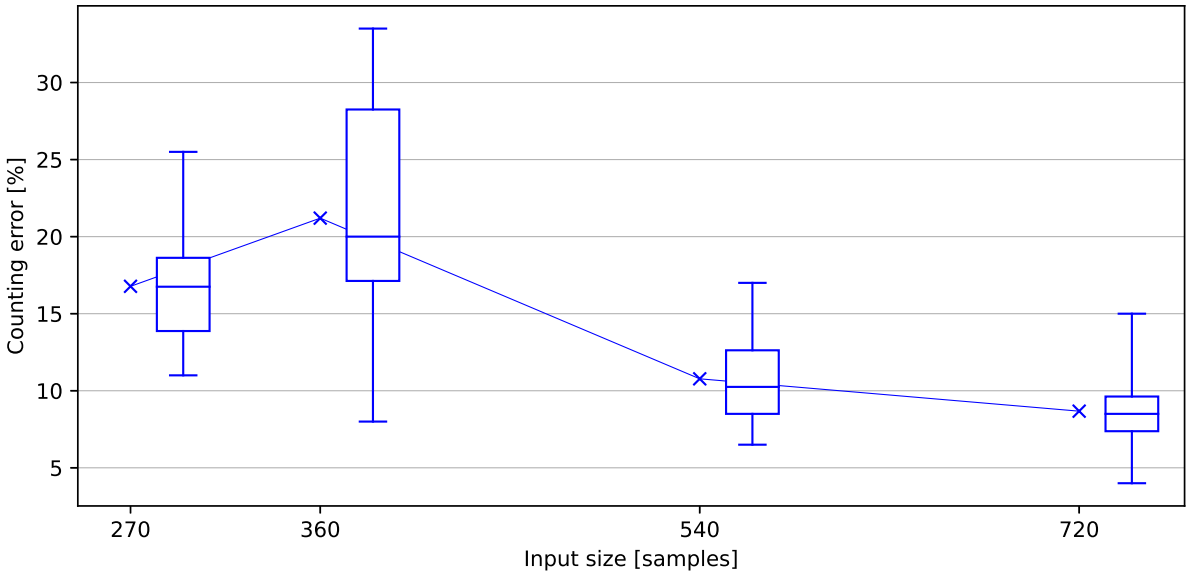


Figure A.14: Exploring the input size ($|x|$) parameter space for M2 barrel, when dataset includes semi-automatic mode. Semi-automatic mode interrupts the firing cycle halfway through, and a manual release of the mobile parts is needed. This firing method is only used for weapon tuning and is not considered part of normal usage. We observe that in this case, larger input sizes are beneficial.

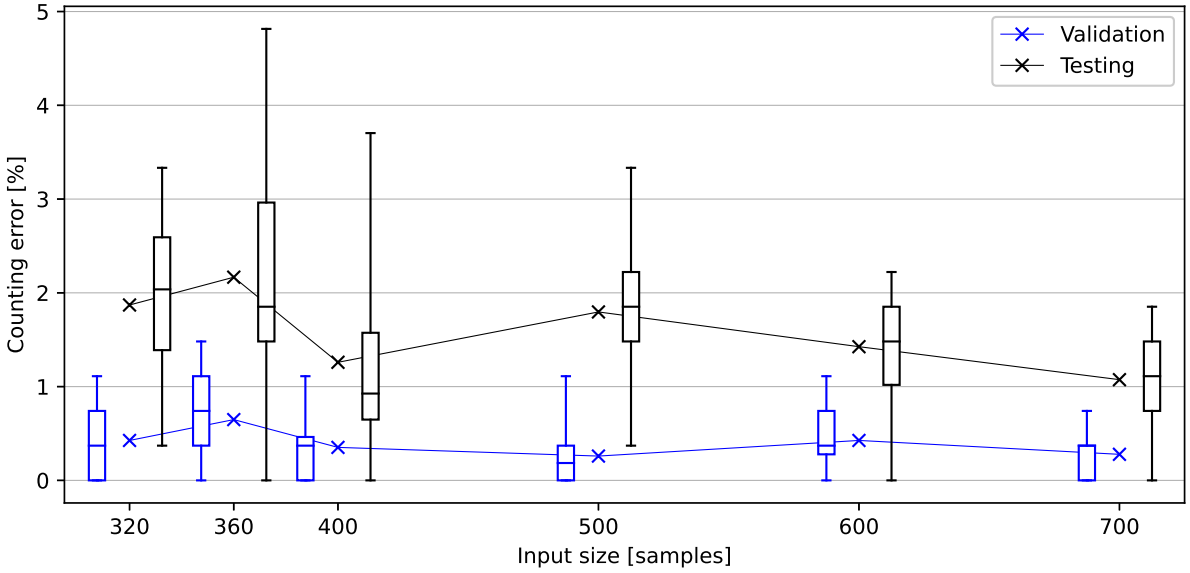


Figure A.15: Exploring the input size ($|x|$) parameter space for M2 barrel after band-stop filtering is applied.

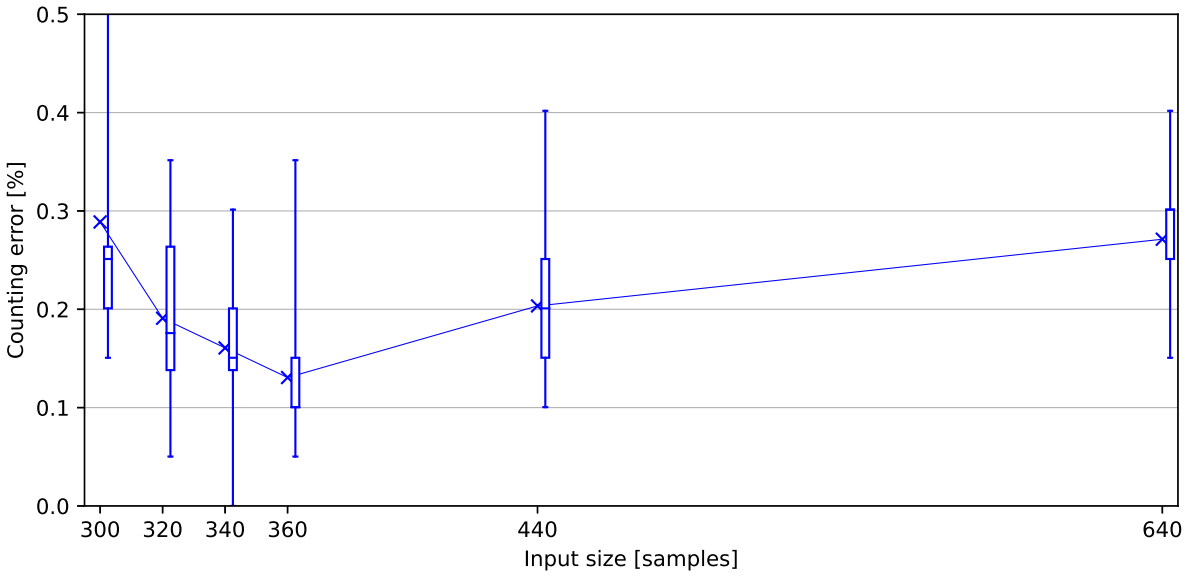


Figure A.16: Exploring the input size ($|x|$) parameter space for MAG.

Candidate selection parameters

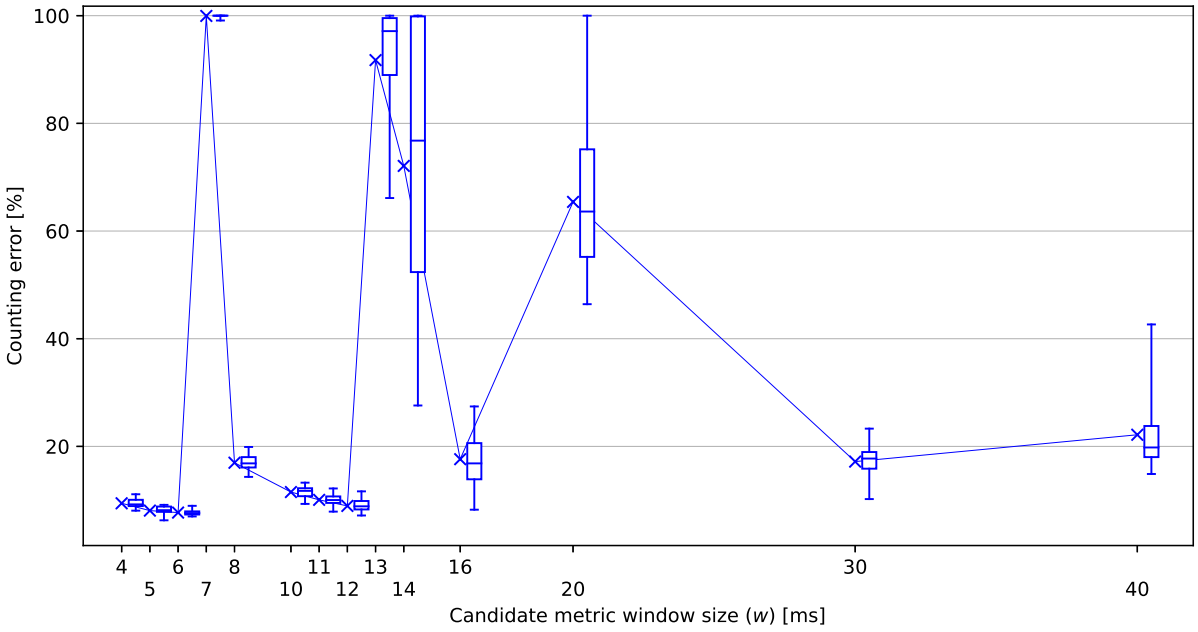


Figure A.17: Exploring the candidate metric window size w parameter space for M2 backplate. This hyperparameter also shows multiple local minima. Among these, the smallest windows leads to both best prediction and computational performance.

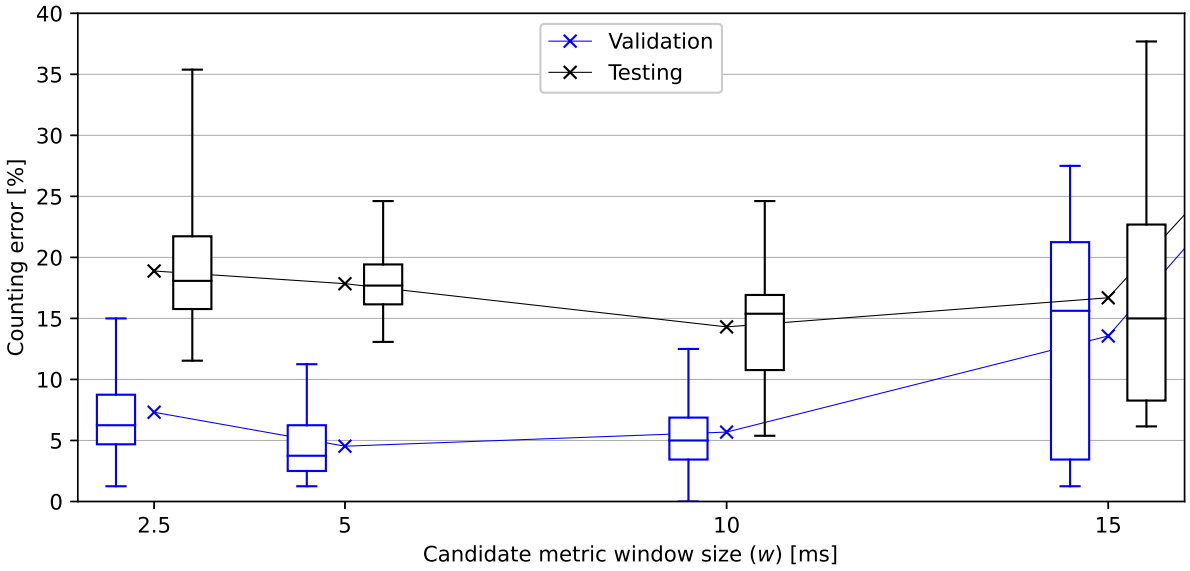


Figure A.18: Exploring the candidate metric window size w parameter space for M2 barrel.

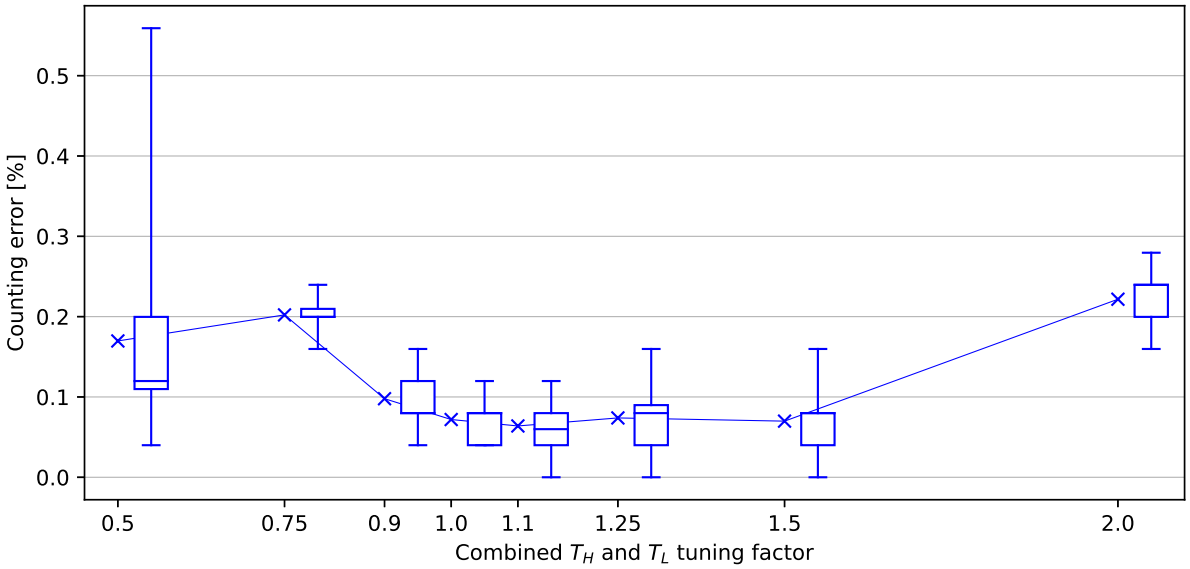


Figure A.19: Exploring the combined candidate selection thresholds parameter space for Minimi 5.56. Adjustments are made proportionally to both thresholds, starting with $T_L = 90$ and $T_H = 114$.

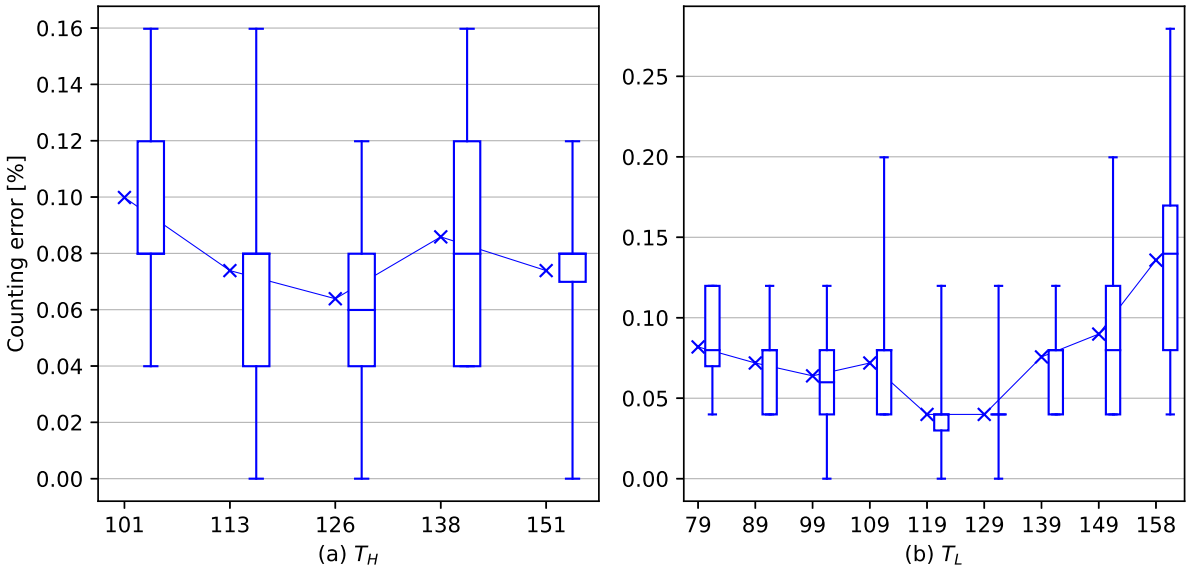


Figure A.20: Exploring the individual candidate selection thresholds parameter space for Minimi 5.56.

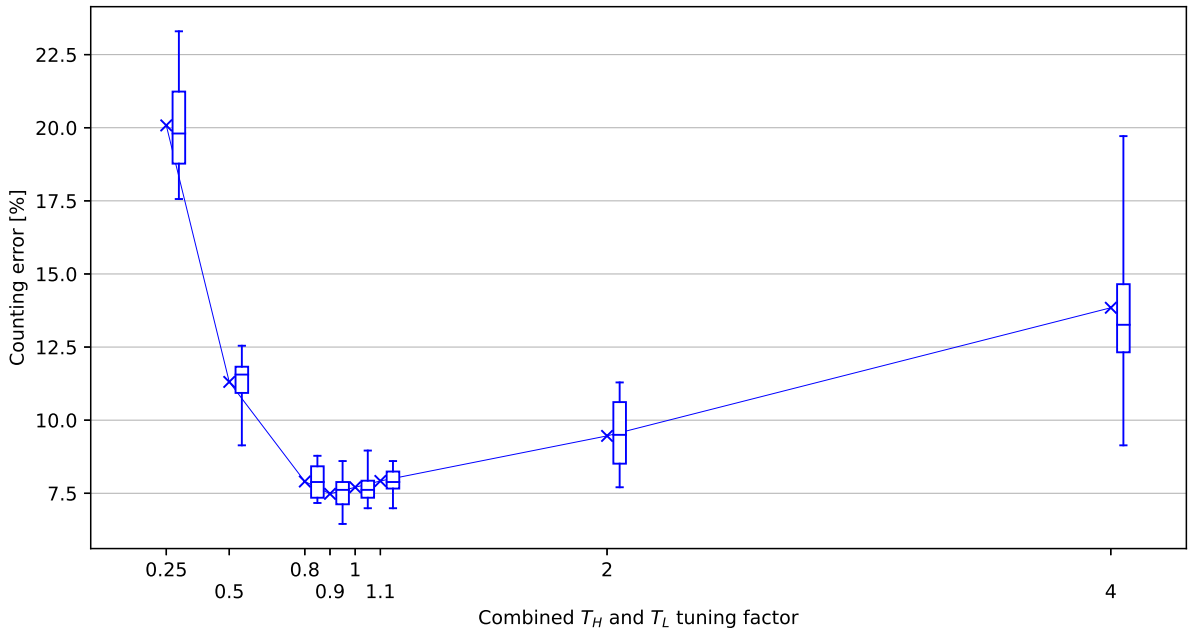


Figure A.21: Exploring the combined candidate selection thresholds parameter space for M2 backplate. Adjustments are made proportionally to both thresholds, starting with $T_L = 100$ and $T_H = 121$.

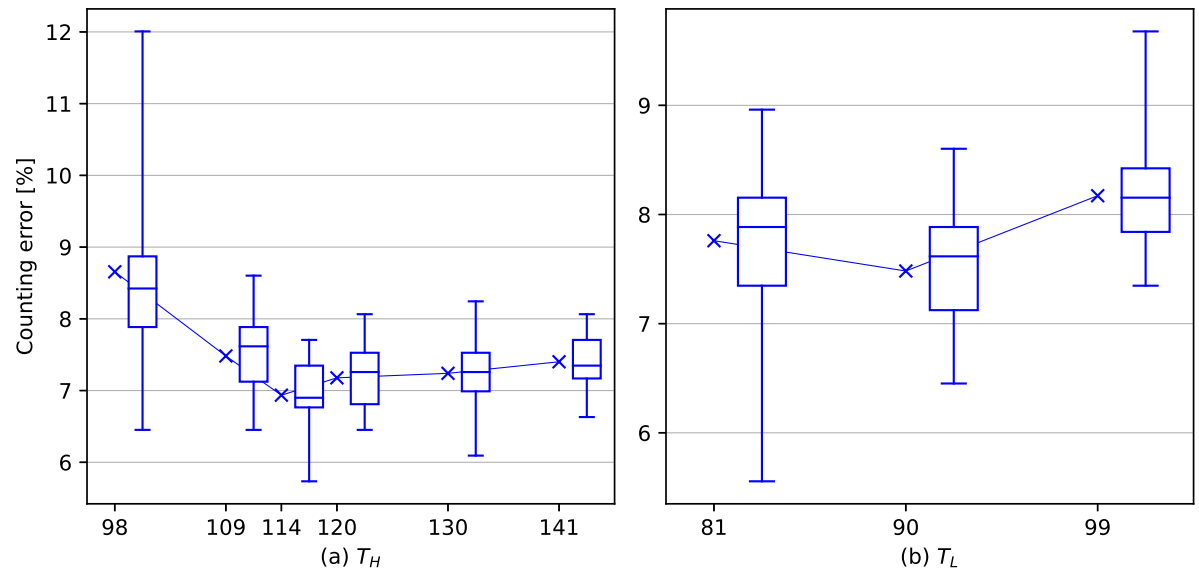


Figure A.22: Exploring the individual candidate selection thresholds parameter space for M2 backplate.

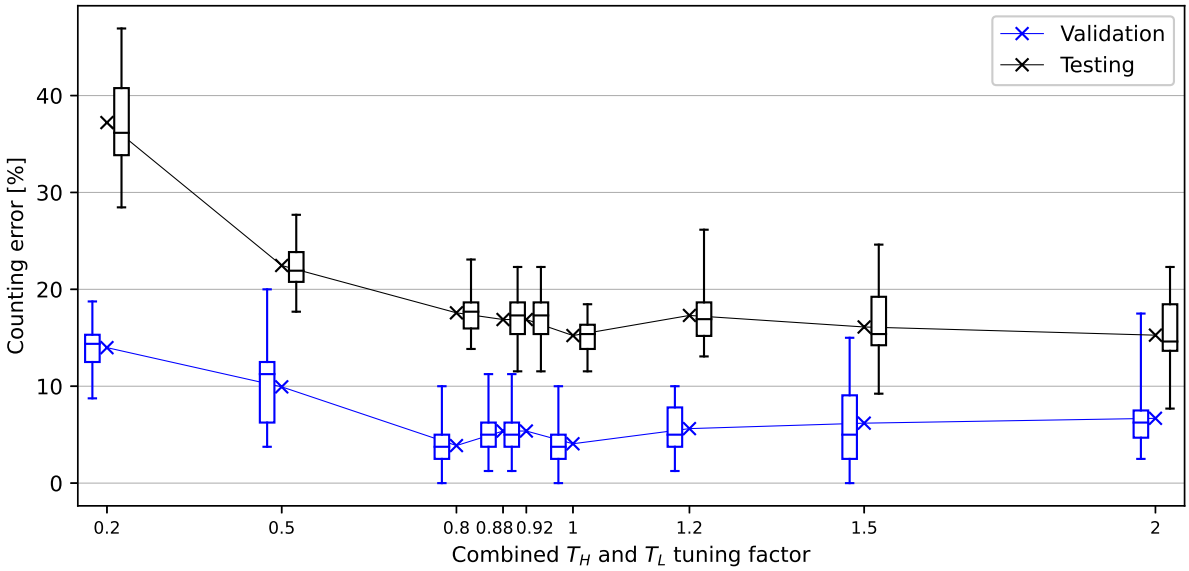


Figure A.23: Exploring the combined candidate selection thresholds parameter space for M2 backplate. Adjustments are made proportionally to both thresholds, starting with $T_L = 90$ and $T_H = 114$.

Network structure

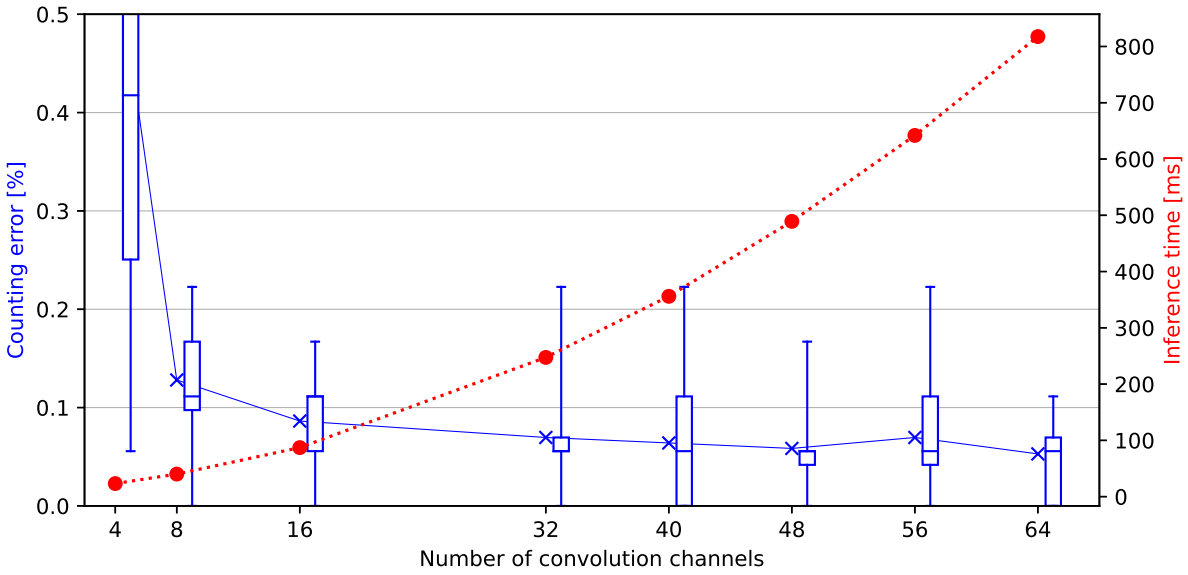


Figure A.24: Exploring the parameter space for the number of convolution filters for MAG, against the inference time. This is an alternative representation of the subfigure in Figure 6.7.

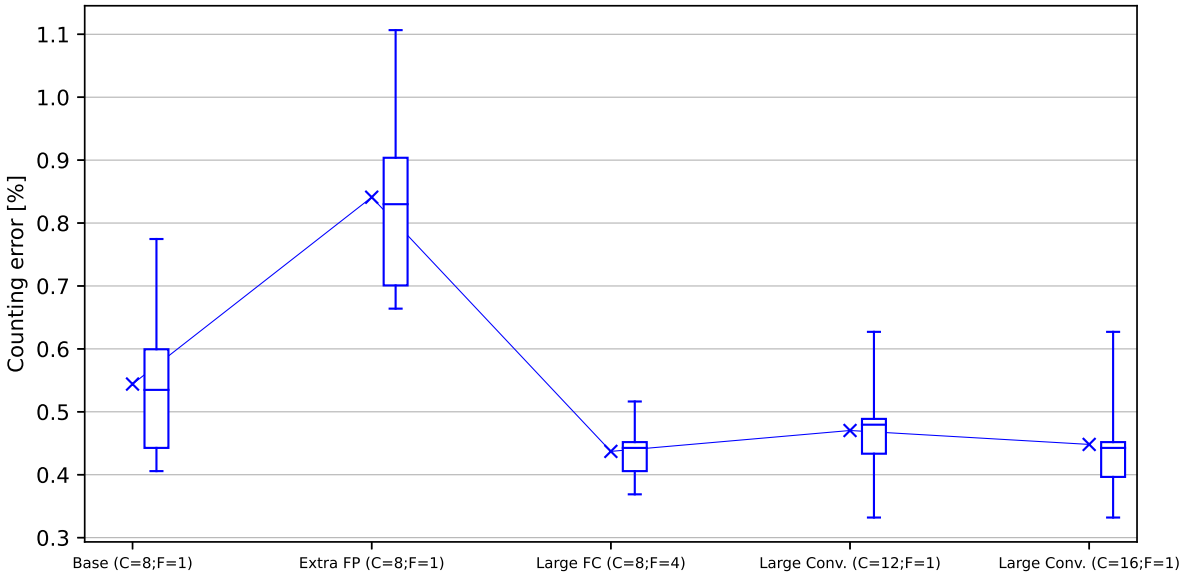


Figure A.25: MAG experiments. In "Extra FP", the learning dataset (post pretraining) is supplemented with non-shot data from the pretraining datasets. This is slightly detrimental. "Large FC" increases the number of neurons in the fully connected layer by a factor of 4. "Large Conv." increases the number of convolution channels (C). All show a slight improvement over the base values, at the cost of increased inference time.

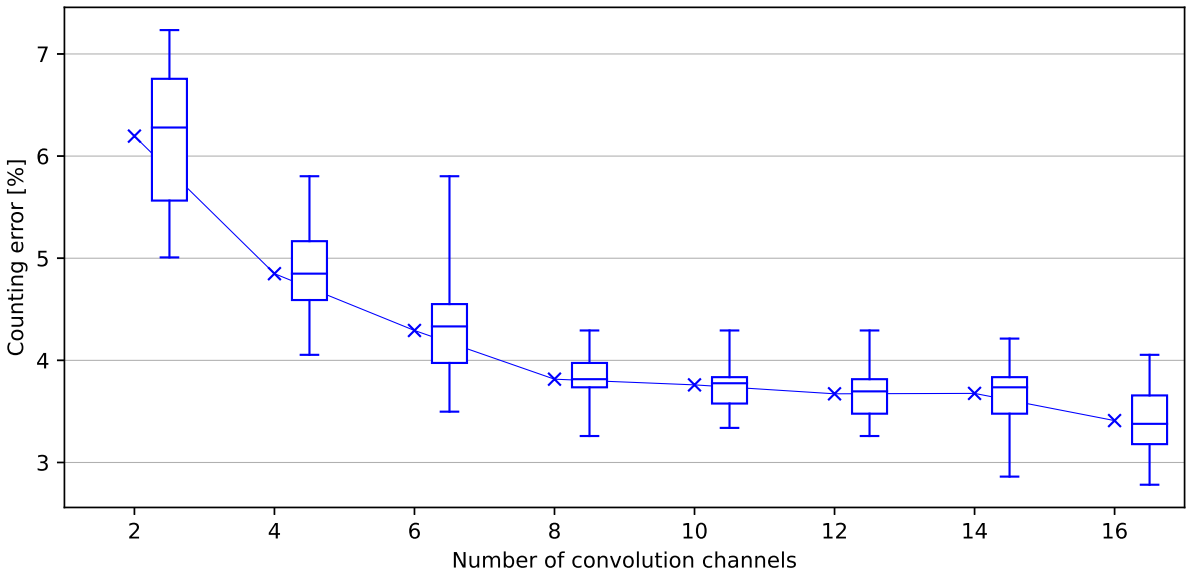


Figure A.26: Exploring the parameter space for the number of convolution filters for M2 backplate. More channels provide increased performance, with diminishing returns.

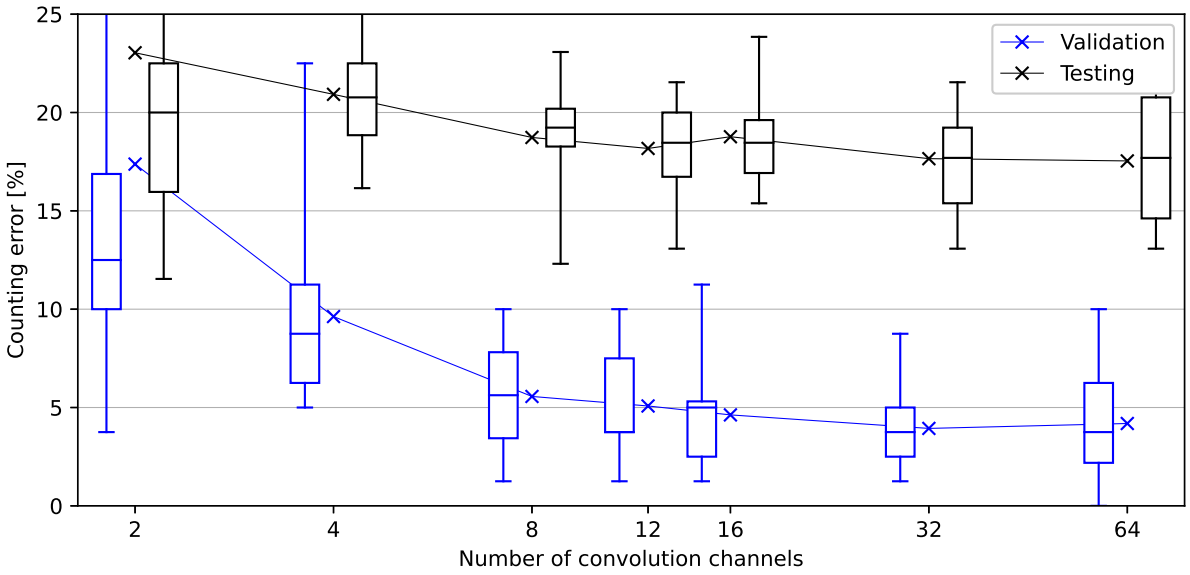


Figure A.27: Exploring the parameter space for the number of convolution filters for M2 barrel. Behaviour is similar to the backplate version.

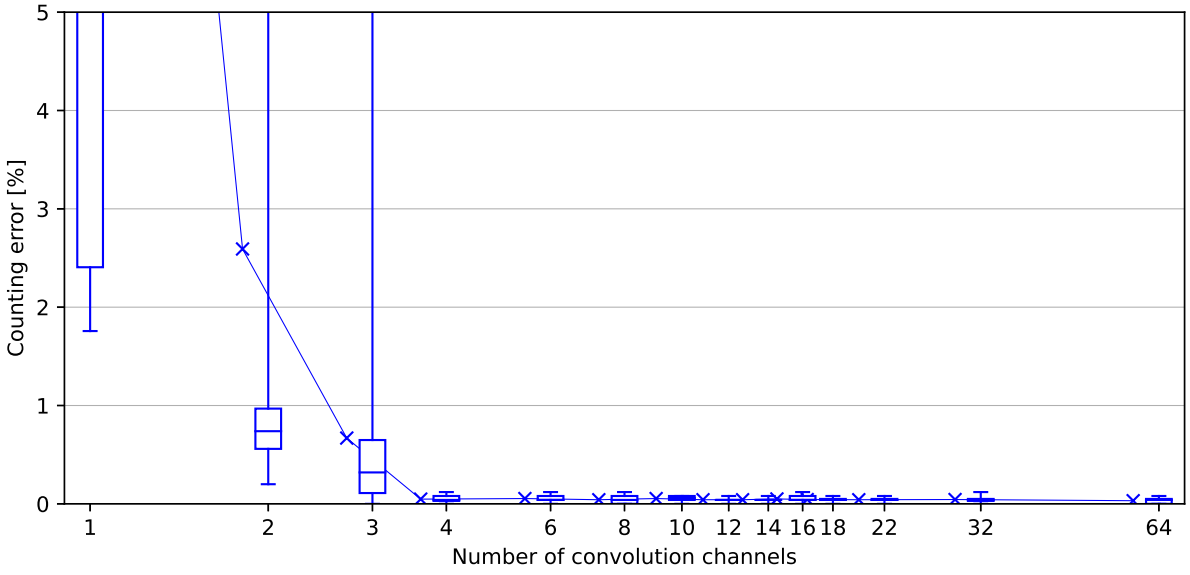


Figure A.28: Exploring the parameter space for the number of convolution filters for Minimi 5.56. This weapon shows convergence with an unusually low number of necessary filters.

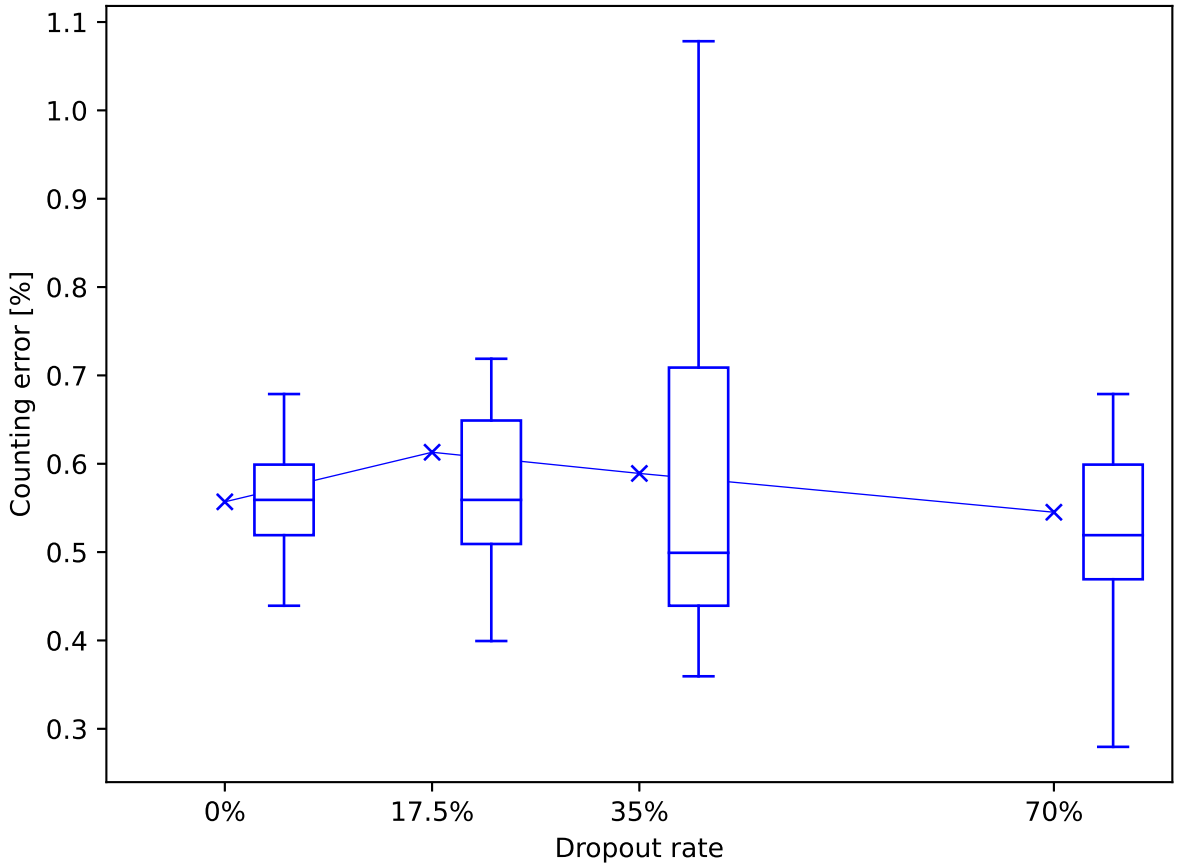


Figure A.29: Exploring dropout rate parameter space for Minimi 5.56. Higher values show a slight positive effect on the median counting error rate, as well as an increase in variance leading to potentially better best-outcome networks.

VAT parameters

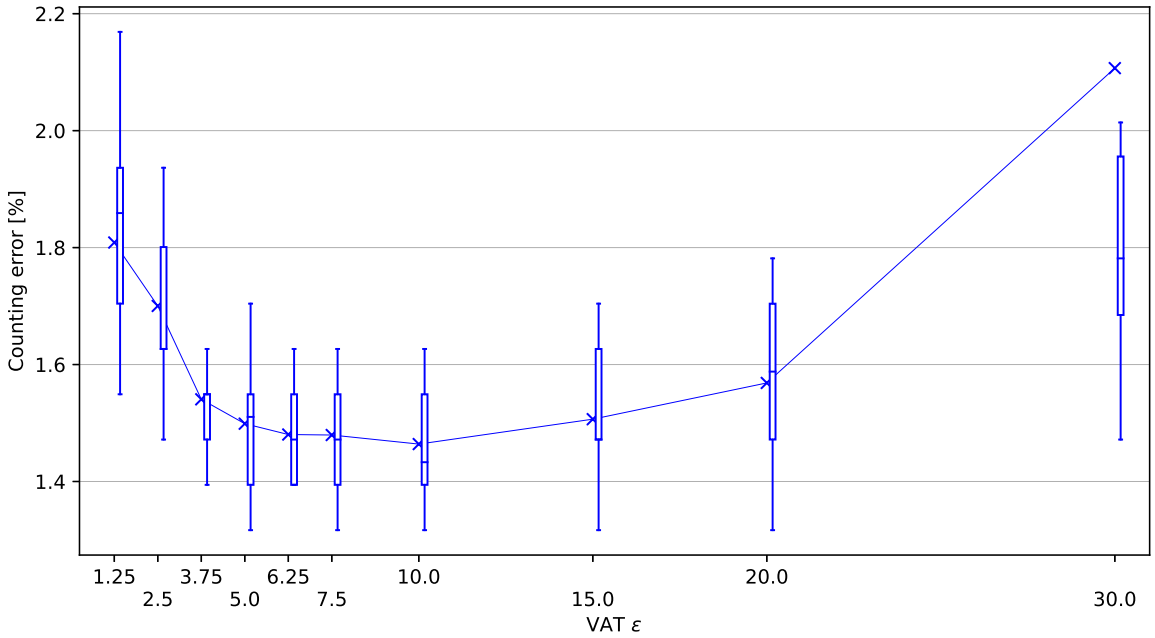


Figure A.30: Exploring the VAT perturbation size (ϵ) parameter space for Minimi 7.62. Parameter shows a significant effect.

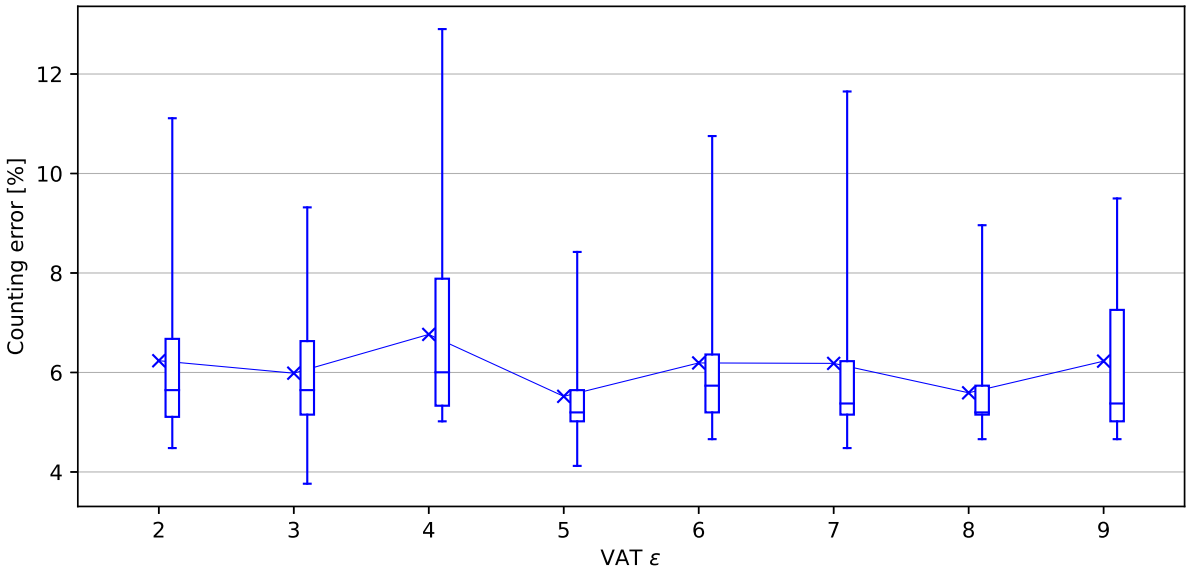


Figure A.31: Exploring the VAT perturbation size (ϵ) parameter space for M2 backplate. No significant difference is found on this dataset.

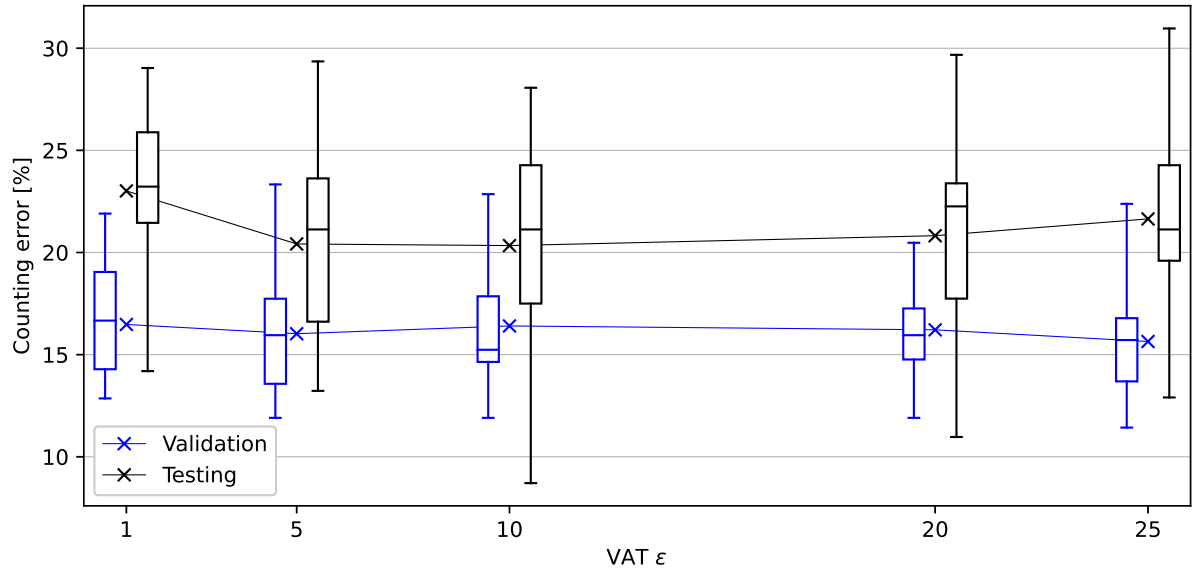


Figure A.32: Exploring the VAT perturbation size (ϵ) parameter space for M2 barrel. Similarly to the backplate dataset, no significant effect is found.

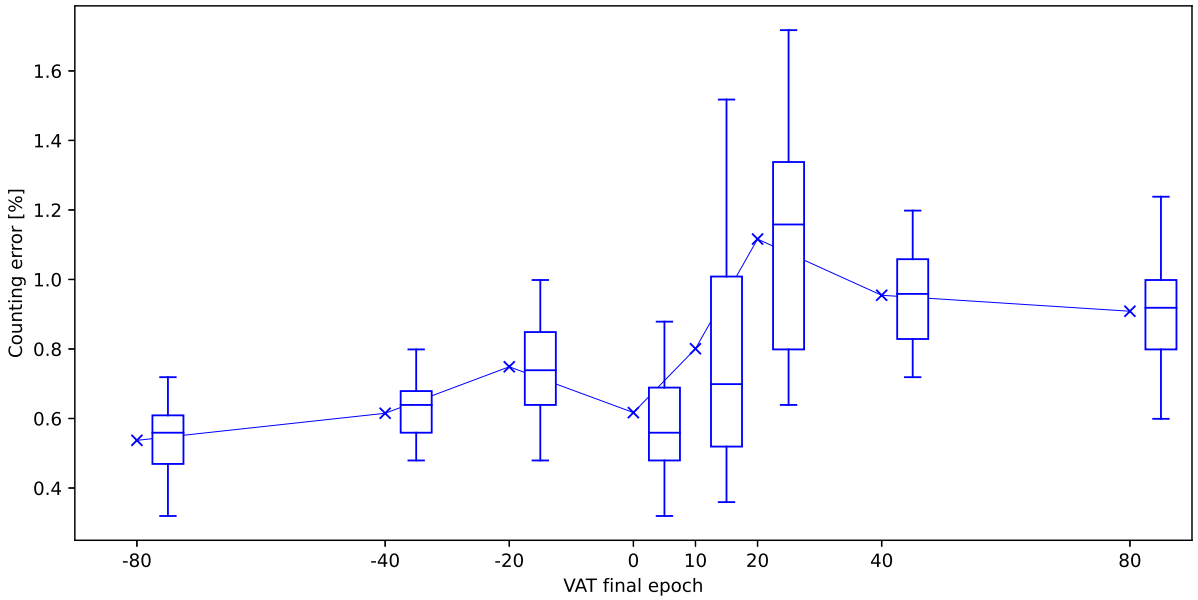


Figure A.33: Exploring the VAT ramping (ϵ) parameter space for Minimi 5.56. For positive values, VAT is ramped up linearly until the specified epoch. The original proposition by Tsai et al. [63] is 80. For negative values, VAT is introduced fully at epoch 0, then linearly ramped down to 0 at the specified epoch.

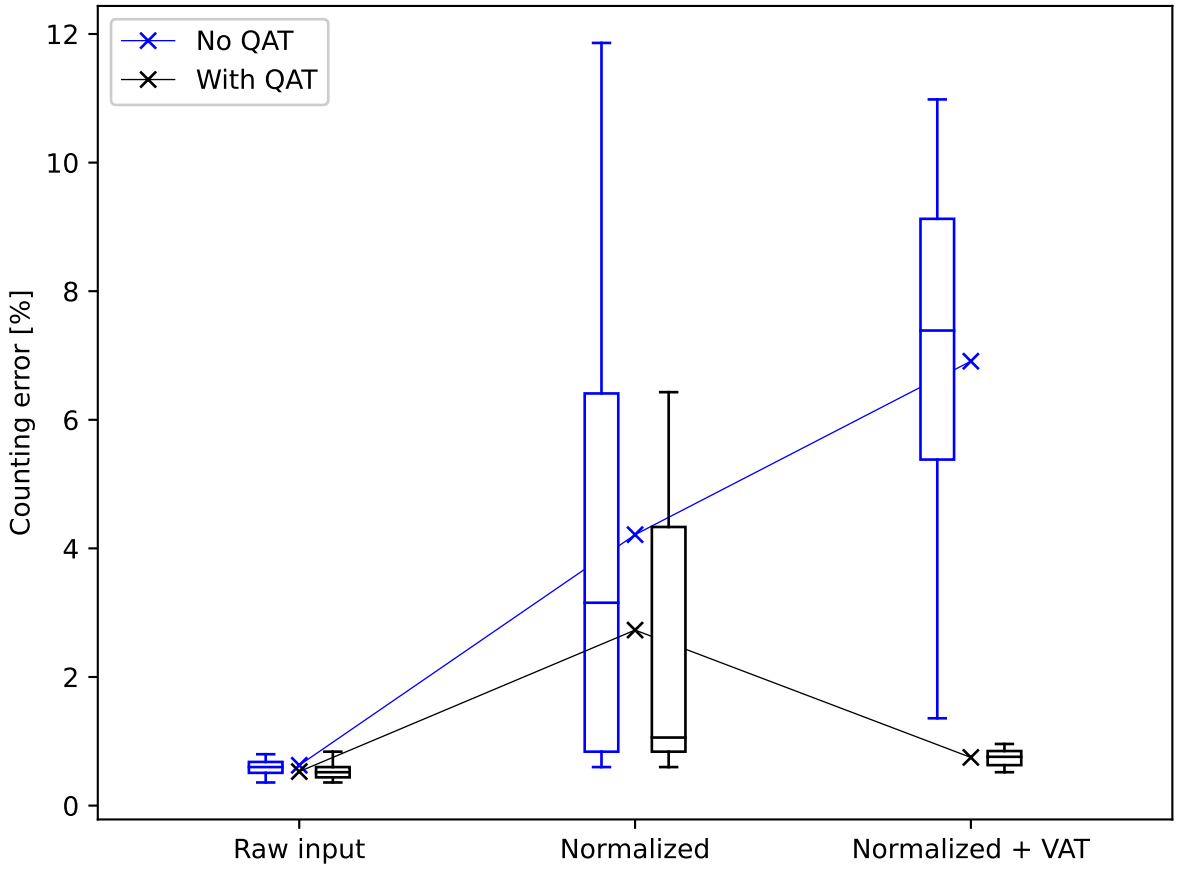


Figure A.34: Attempt to reduce the input space from raw integer values [-128, +127] to floating-point values [-1.0, 1.0]. The transformation has a detrimental effect, with notable interactions observed between QAT and VAT.

A.3.2 Discrimination

Learning Rate (LR)

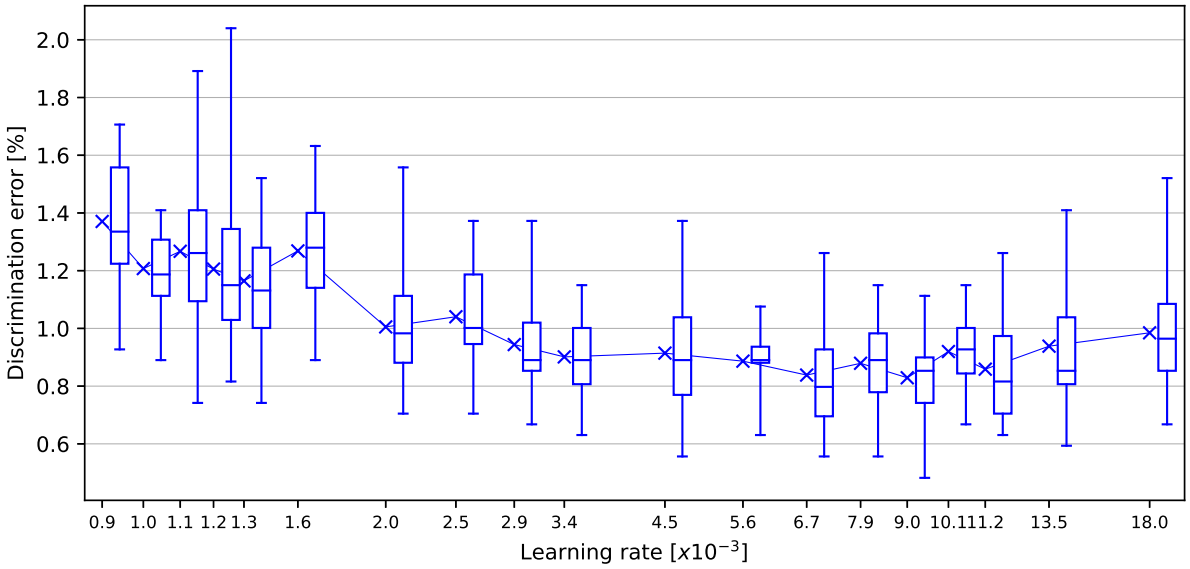


Figure A.35: Exploring the learning rate (LR) parameter space for MAG for discrimination with LLP loss. A slight increase from the optimal value for the counting problem is beneficial (see Figure A.8). However, applying the counting LR would not result in significantly decreased performance.

Input size ($|x|$)

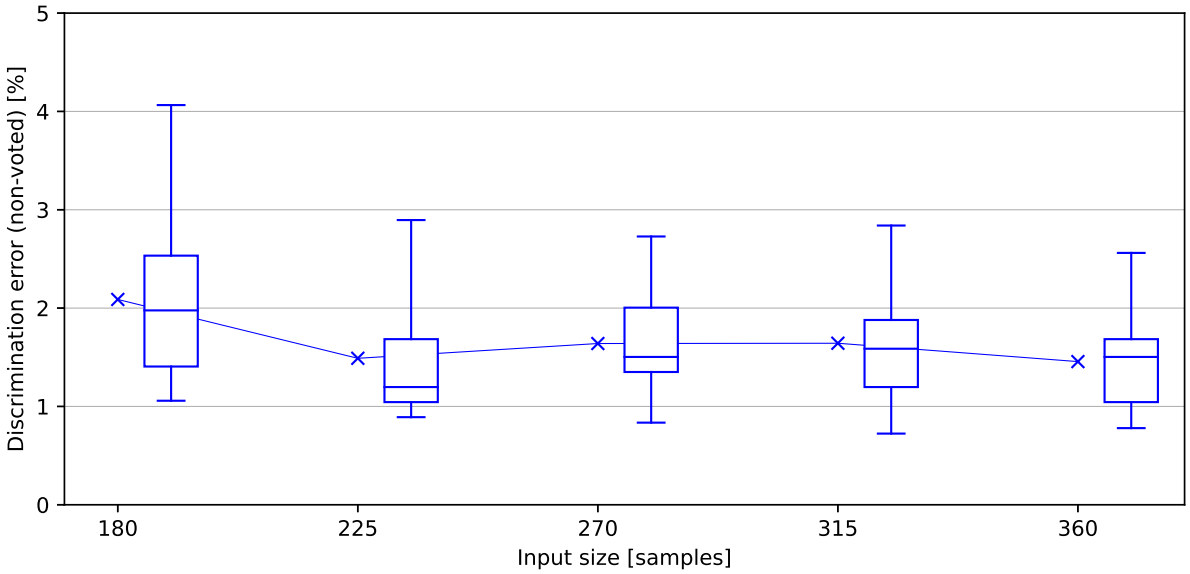


Figure A.36: Exploring the effect of input size ($|x|$) on discrimination error for MAG. Reducing $|x|$ does not affect discrimination error as much as the counting error (see Figure A.16). This suggests that the discrimination information is more concentrated around the shot than the counting information. This matches intuition, as the classical method for discrimination seen in Section 3.1.2 focuses on the presence or absence or reaction from the projectile ejection around the shot event. Meanwhile, to solve the counting problem one needs to observe the full firing cycle. This matches with the findings of Figure 8.6b, in which we observe a low density of high-gradient features after index 150.

Network structure

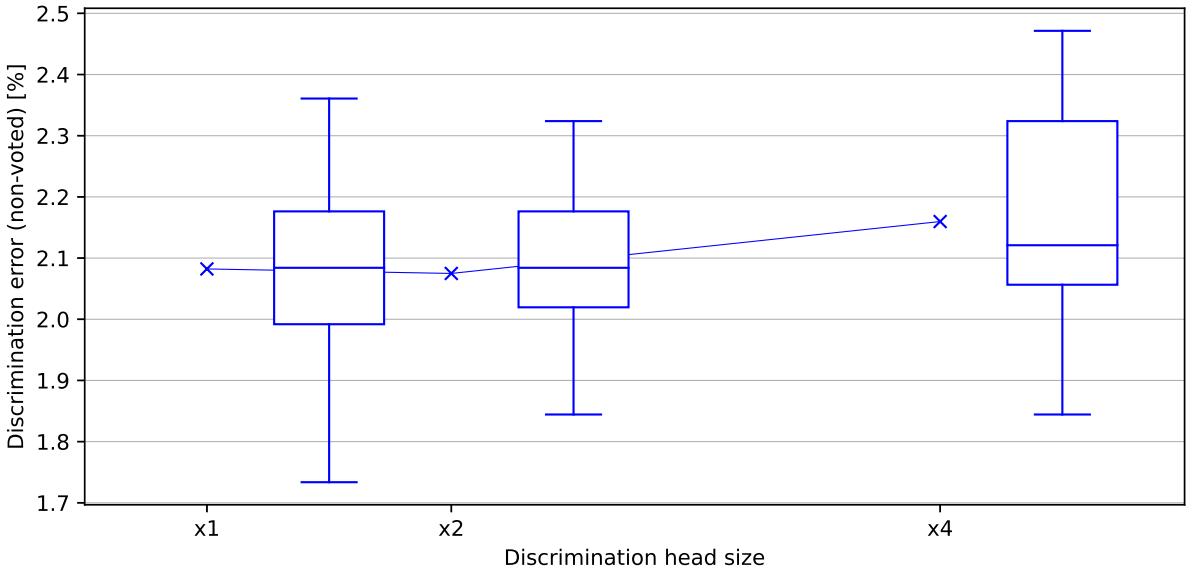


Figure A.37: Exploring the effect of increasing the number of neurons in the fully-connected discrimination head for MAG in the case of shared activations. A slight detrimental effect is shown, suggesting the information from the CNN is already fully exploited, and capacity should be allocated there instead.

VAT parameters

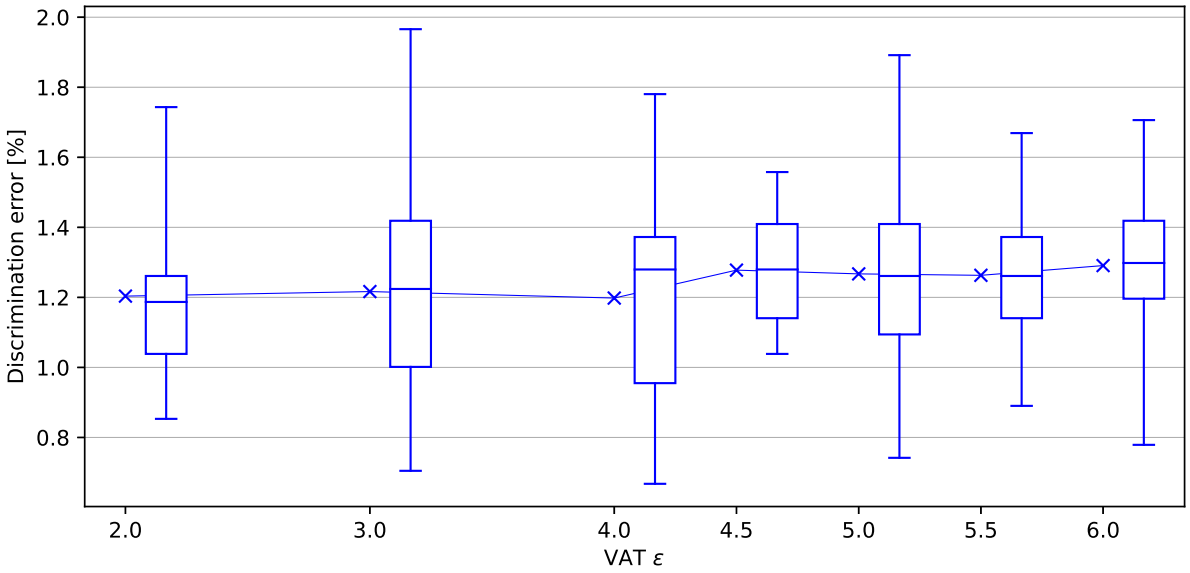


Figure A.38: Exploring the VAT perturbation size (ϵ) parameter space for MAG discrimination with LLP loss. No significant effect is found.

Batch size

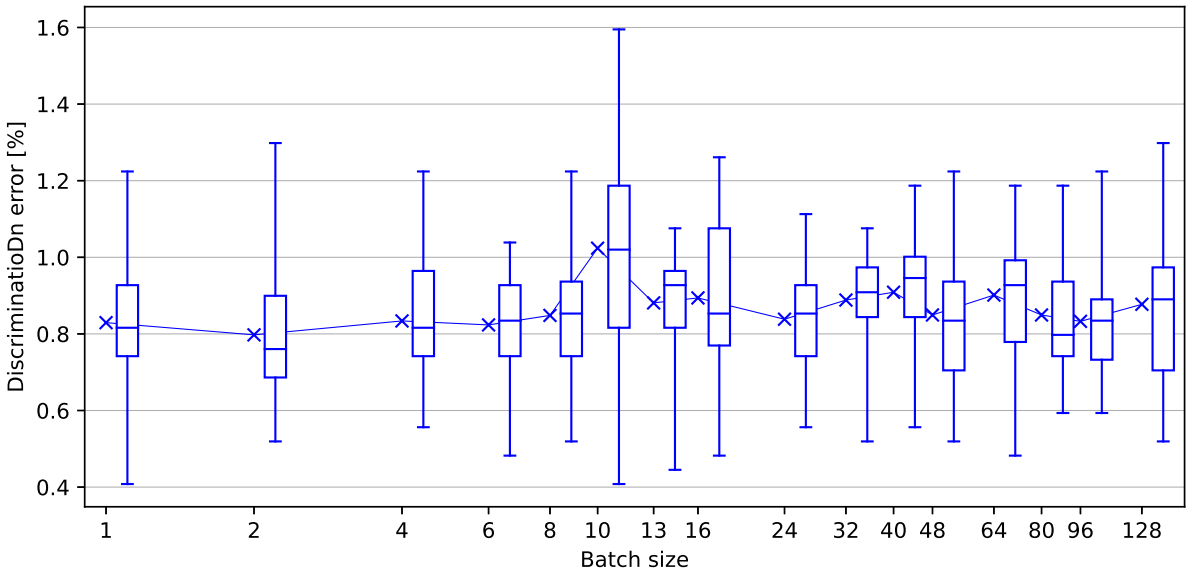


Figure A.39: Exploring the batch size parameter space for MAG for discrimination with LLP loss. No significant effect is found.

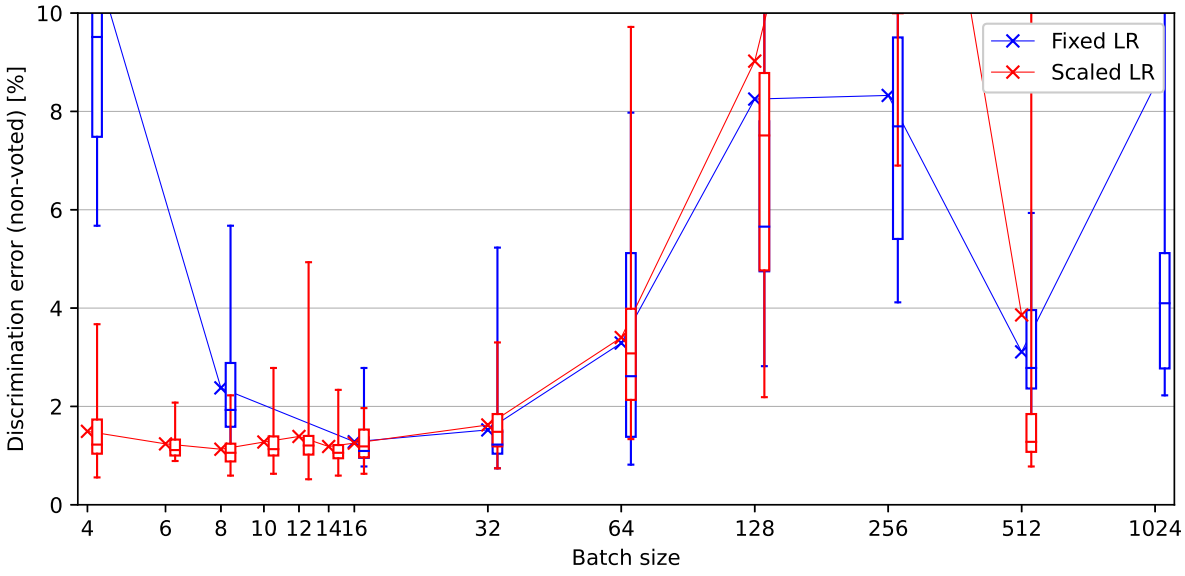


Figure A.40: Exploring the batch size parameter space on MAG discrimination with CE loss and shared activations. We compare with learning rate (LR) both fixed and scaled with the batch size. In this case, the discrimination problem shows a much higher sensitivity to this parameter than the counting problem (see Figure A.10). We thus suggest tuning them independently.

A.4 DTC hyperparameters

This section details the hyperparameter tuning process for the Deep Temporal Clustering (DTC) baseline, as described in Section 9.1.2. For further information on the meaning of the hyperparameters, please refer to the original publication by Madiraju et al. [72].

An extensive fine-tuning was performed for the Minimi 5.56 in order to fully explore the behaviour of each hyperparameters. For other firearms, the optimal parameters for Minimi 5.56 were used as a basis, and a local optimisation around them was performed (as illustrated for the MAG in Section A.4).

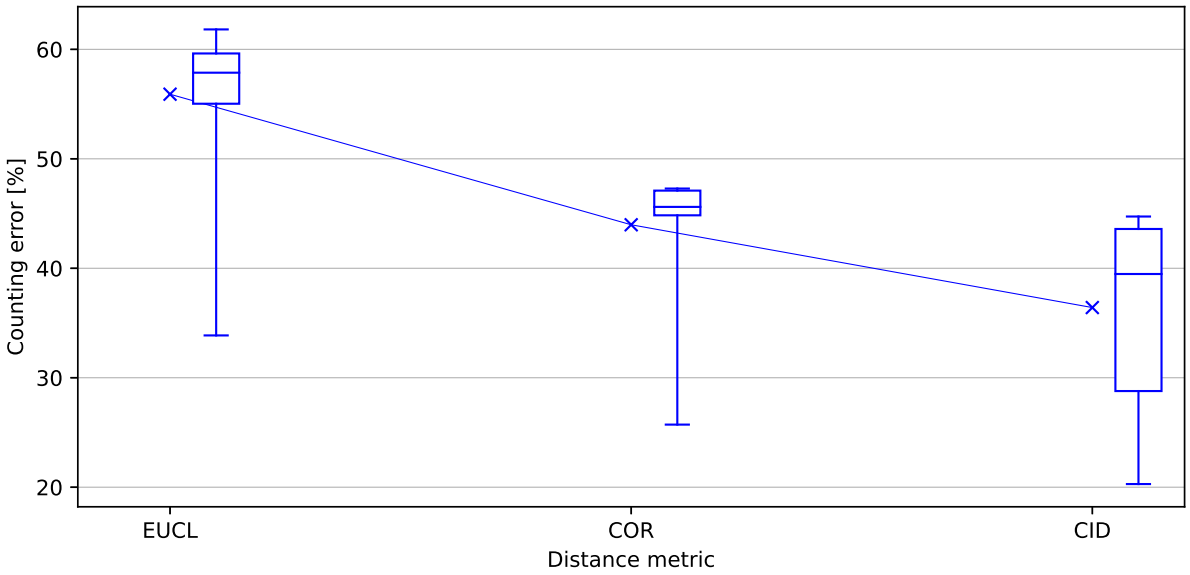


Figure A.41: DTC distance metric comparison for Minimi 5.56. Best result is obtained with Complexity Invariant Similarity (CID) followed by Correlation based Similarity (COR). Fine-tuning for both of these distance metrics is explored in the following subsections.

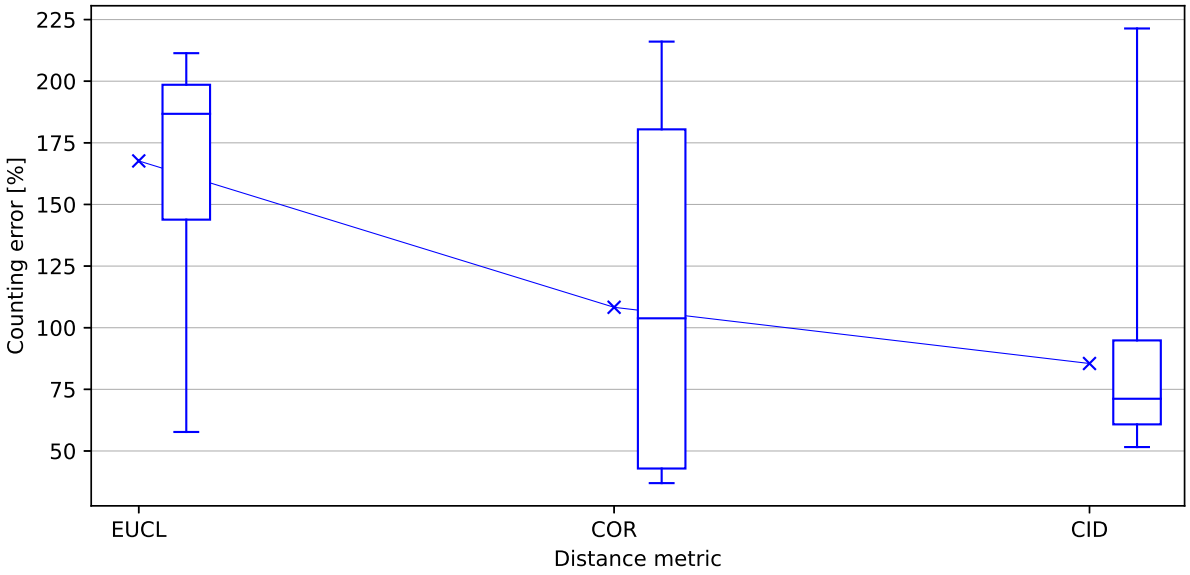


Figure A.42: DTC distance metric comparison for M2 backplate. The CID metric also provides overall better results, although the COR metric provided the best individual result.

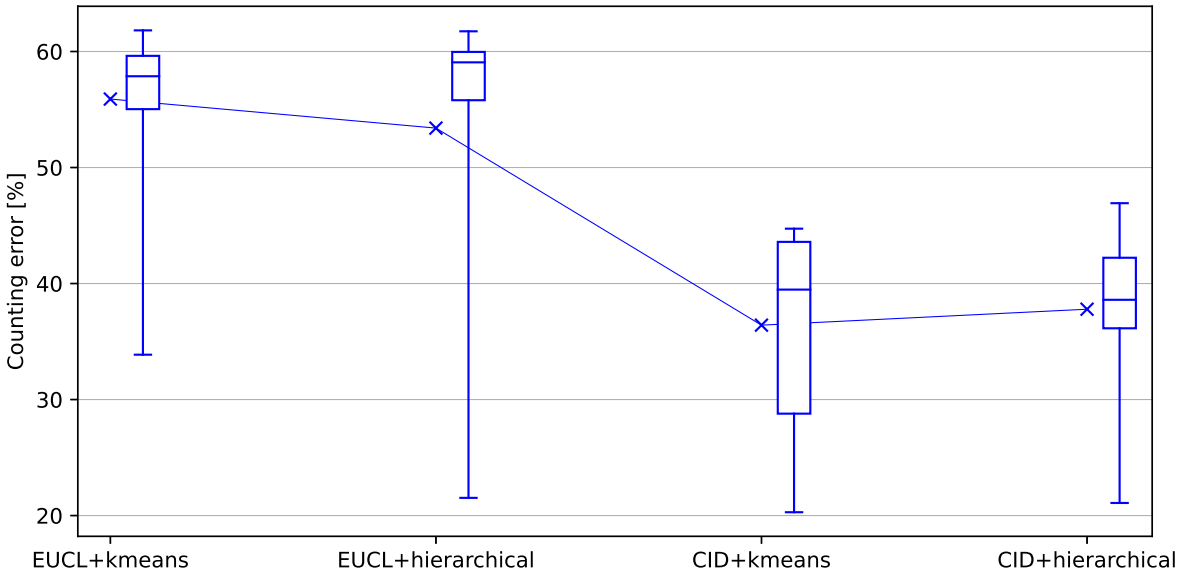


Figure A.43: DTC cluster initialisation method comparison for Minimi 5.56. No significant difference was found.

COR distance

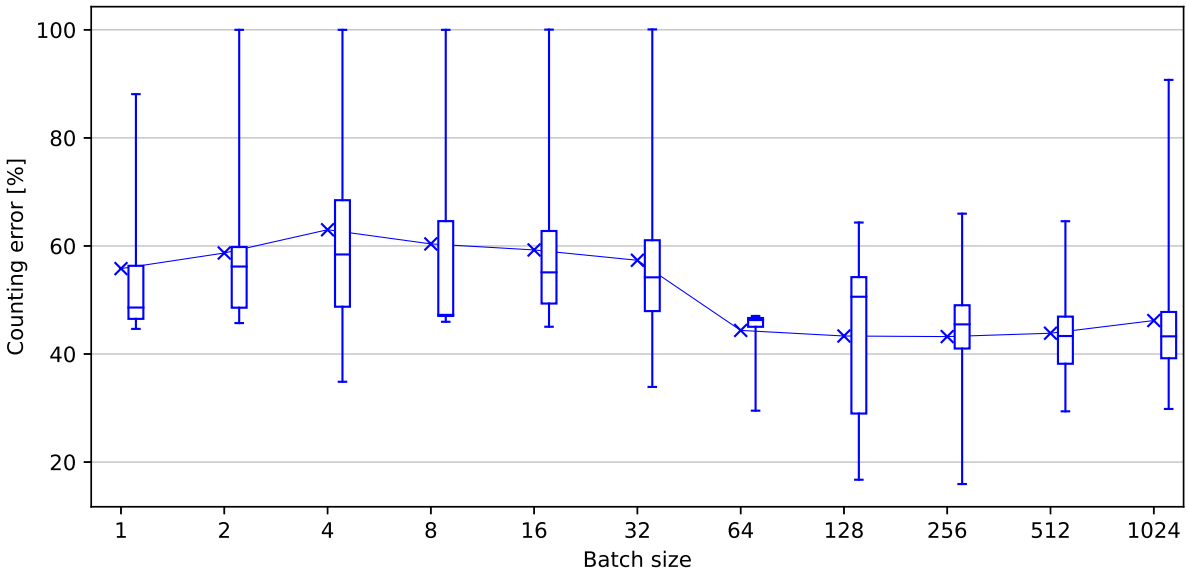


Figure A.44: Batch size parameter space for Minimi 5.56. Default DTC value is 64. Larger batch sizes seem to have a positive influence, especially at 64 onwards.

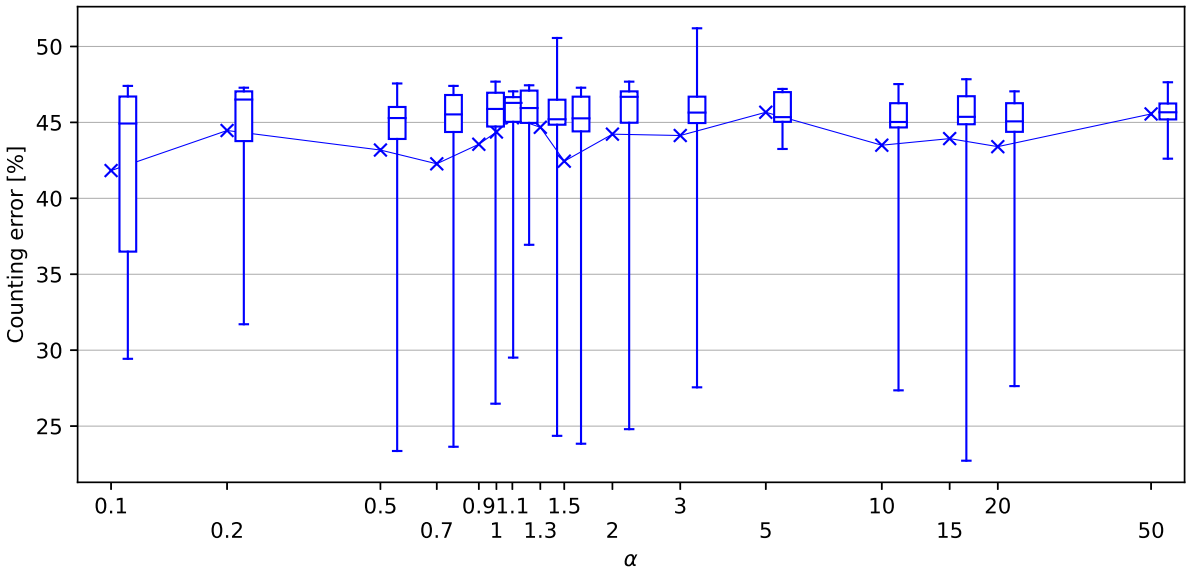


Figure A.45: DTC α parameter space for Minimi 5.56. Default DTC value is 1. Tuning does not result in any significant difference.

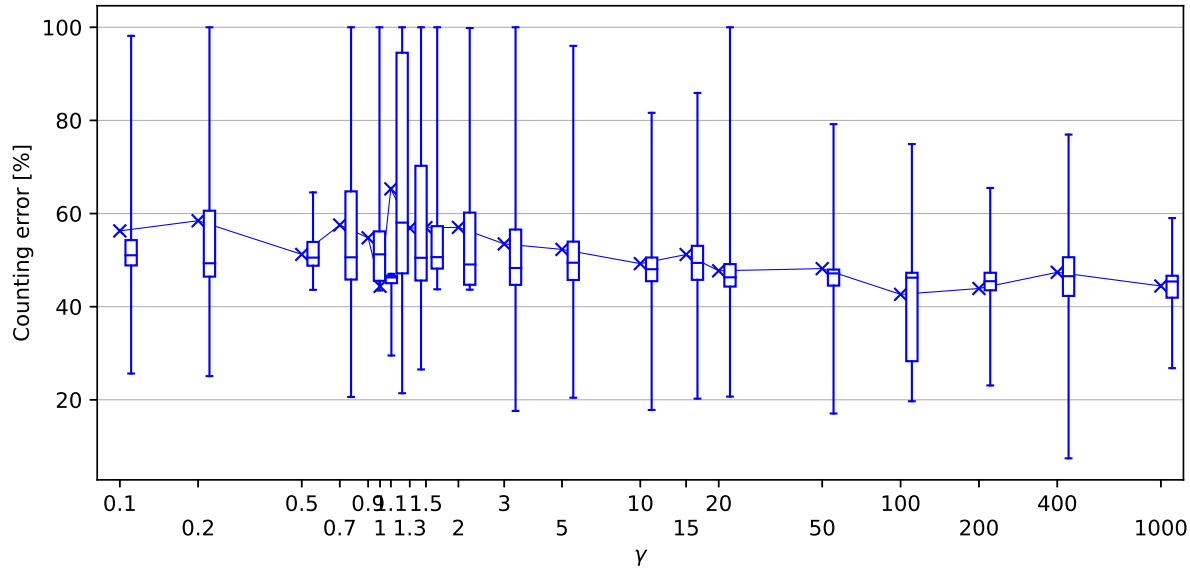


Figure A.46: DTC γ parameter space for Minimi 5.56. Default DTC value is 1. Higher values show slightly improved performance.

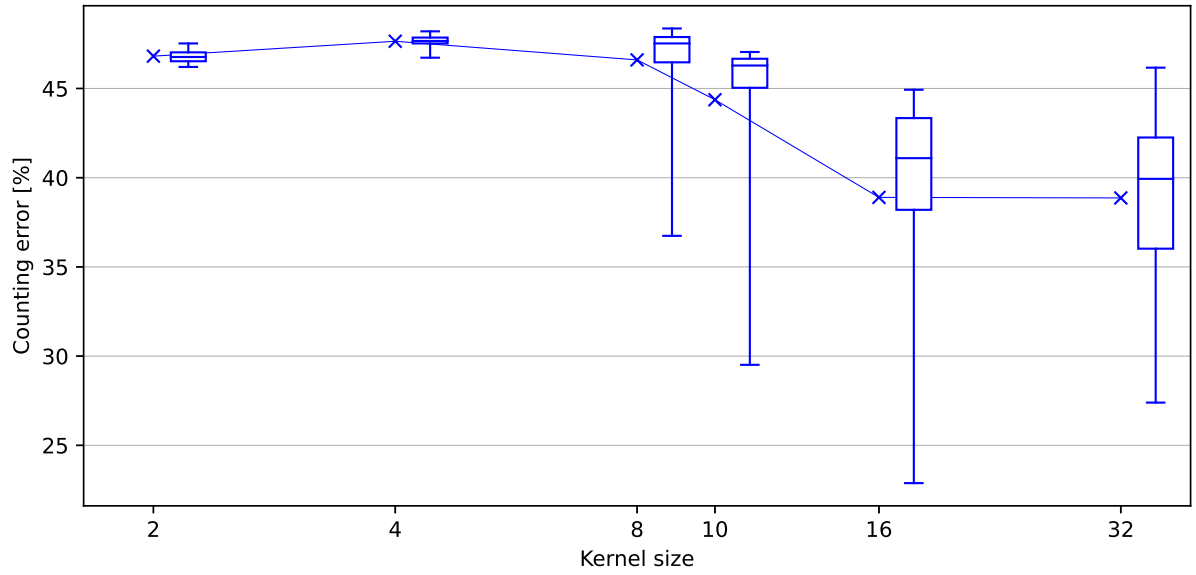


Figure A.47: DTC kernel size parameter space for Minimi 5.56. Default DTC value is 10. Higher values have a positive effect.

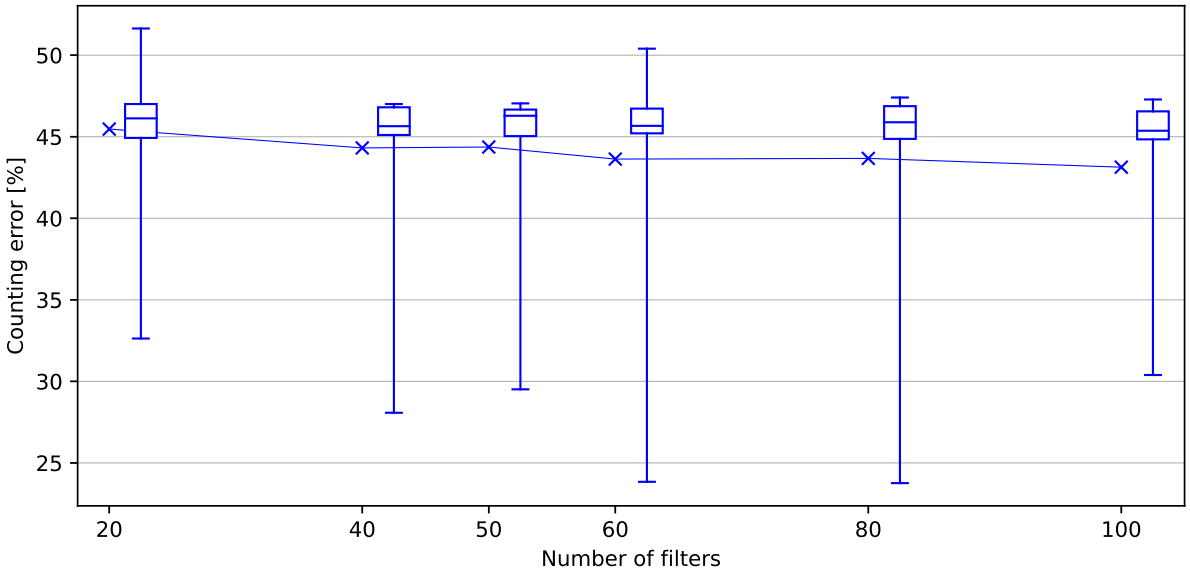


Figure A.48: DTC filter count parameter space for Minimi 5.56. Default DTC value is 50. Larger filter counts yield no significant improvement.

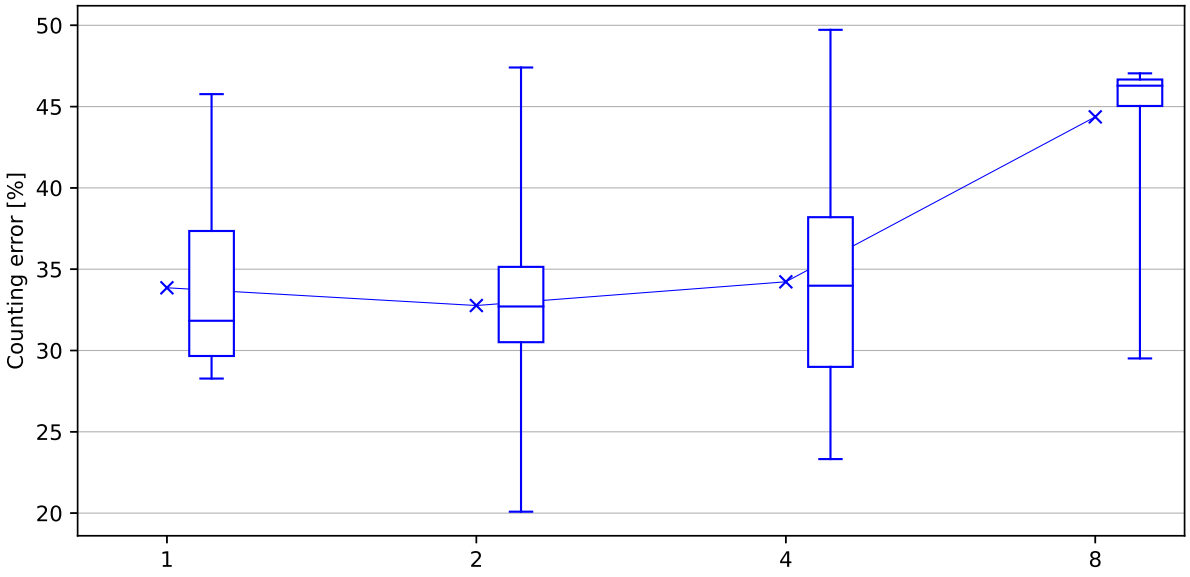


Figure A.49: DTC maxpool size parameter space for Minimi 5.56. Default DTC value is 8. Smaller maxpool sizes improve performance.

CID distance

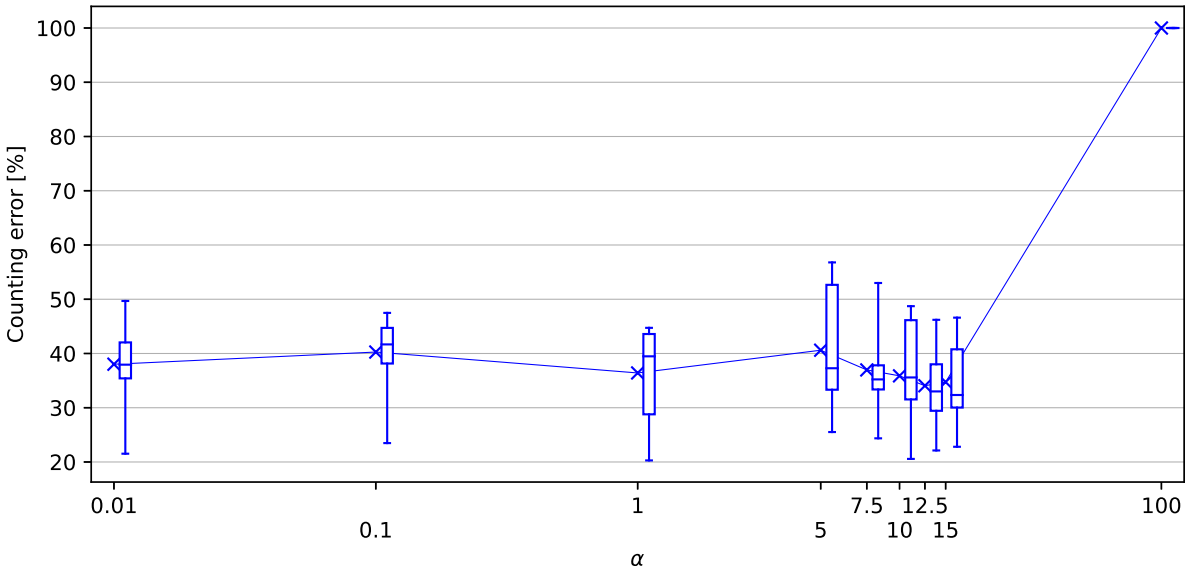


Figure A.50: DTC α parameter space for Minimi 5.56. Default DTC value is 1. An increase to higher than default values yields a slight performance improvement. High values lead to non-convergence.

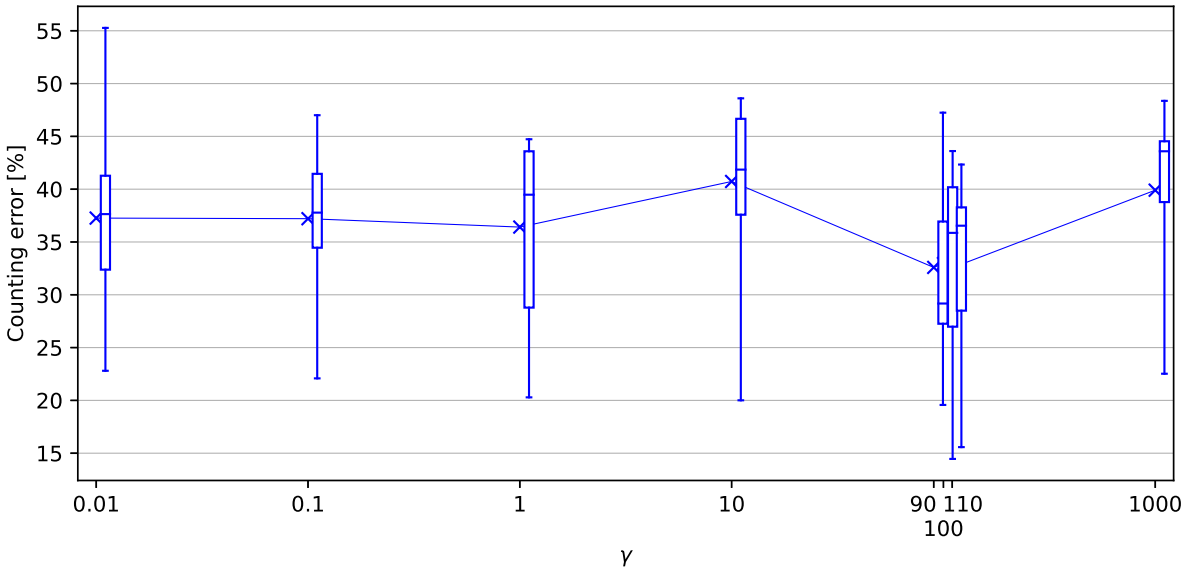


Figure A.51: DTC γ parameter space for Minimi 5.56. Default DTC value is 1. A minimum is found around $\gamma = 100$.

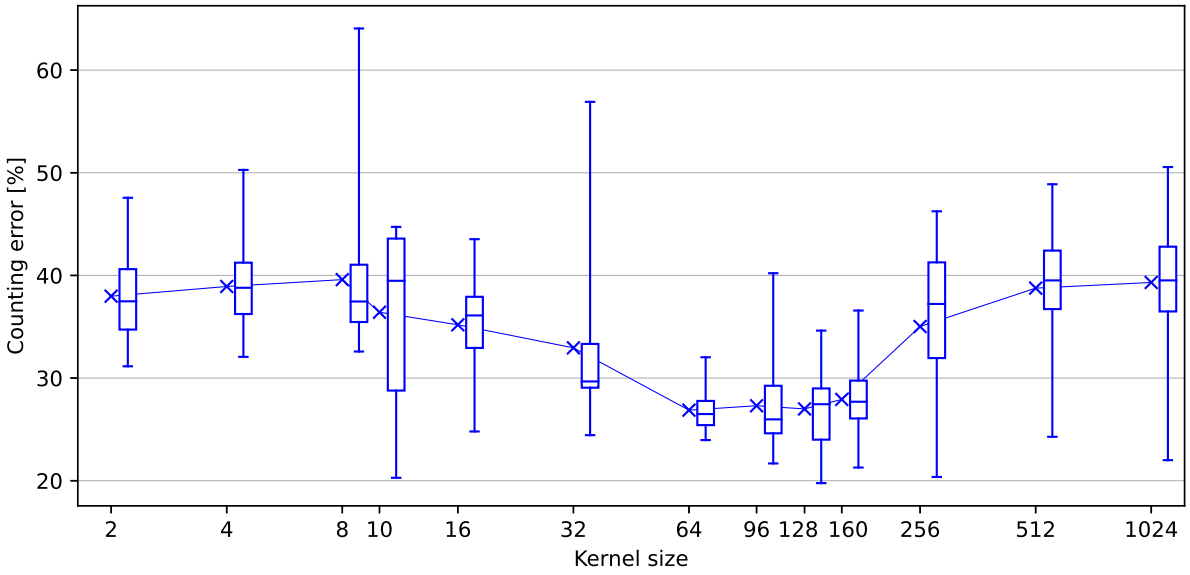


Figure A.52: DTC kernel size parameter space for Minimi 5.56. Default DTC value is 10. A minimum for counting error is shown around 96.

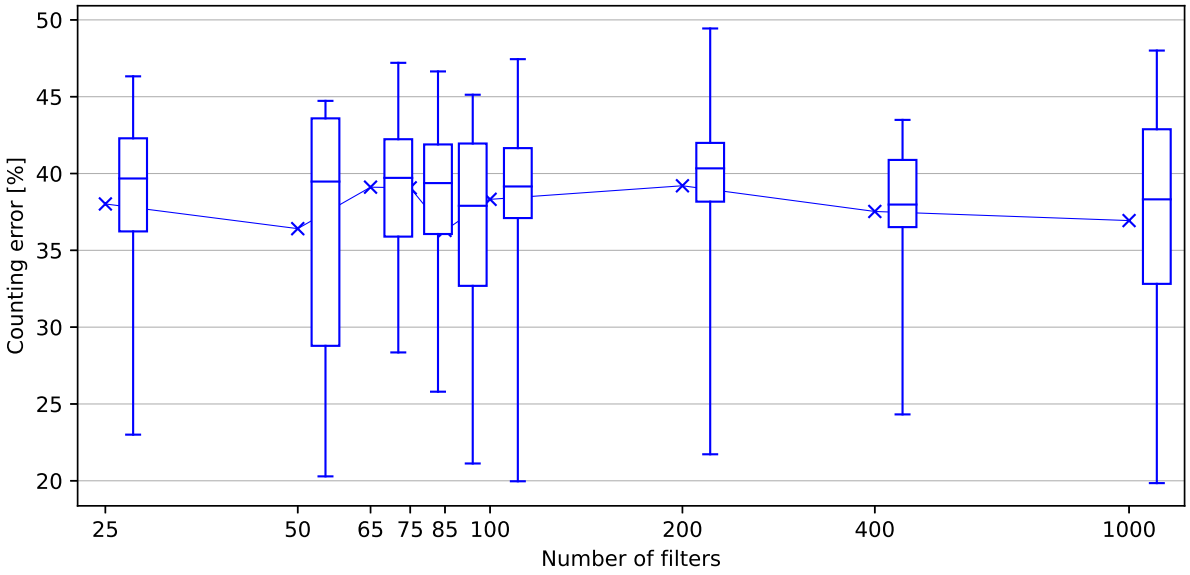


Figure A.53: DTC filter count parameter space for Minimi 5.56. Default DTC value is 50. Larger filter counts yield no significant improvement.

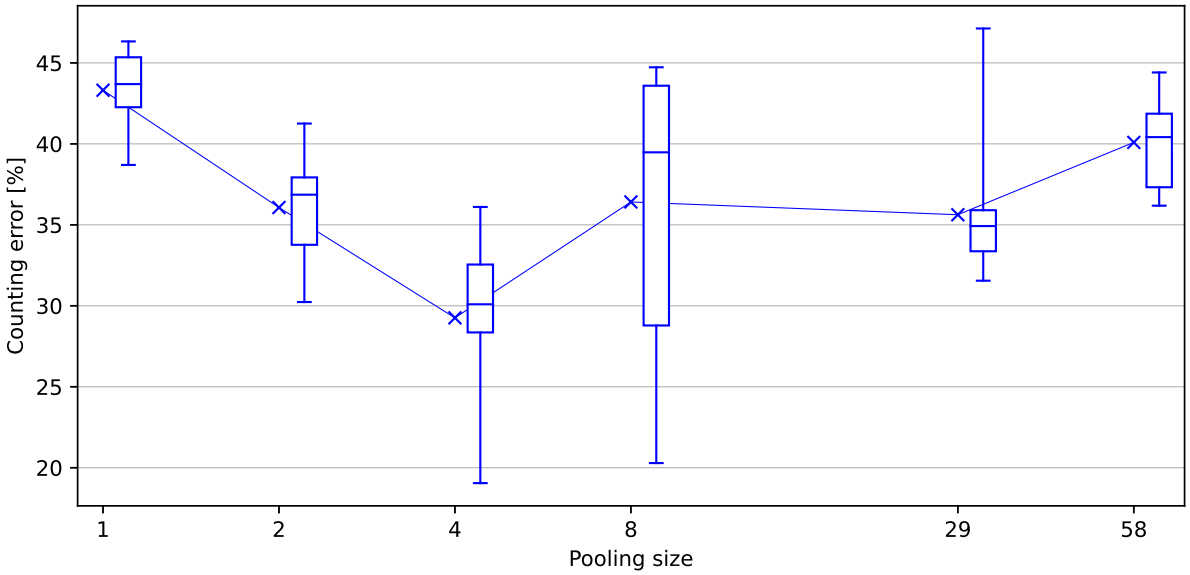


Figure A.54: DTC maxpool size parameter space for Minimi 5.56. Default DTC value is 8. A smaller pool size of 4 improves performance.

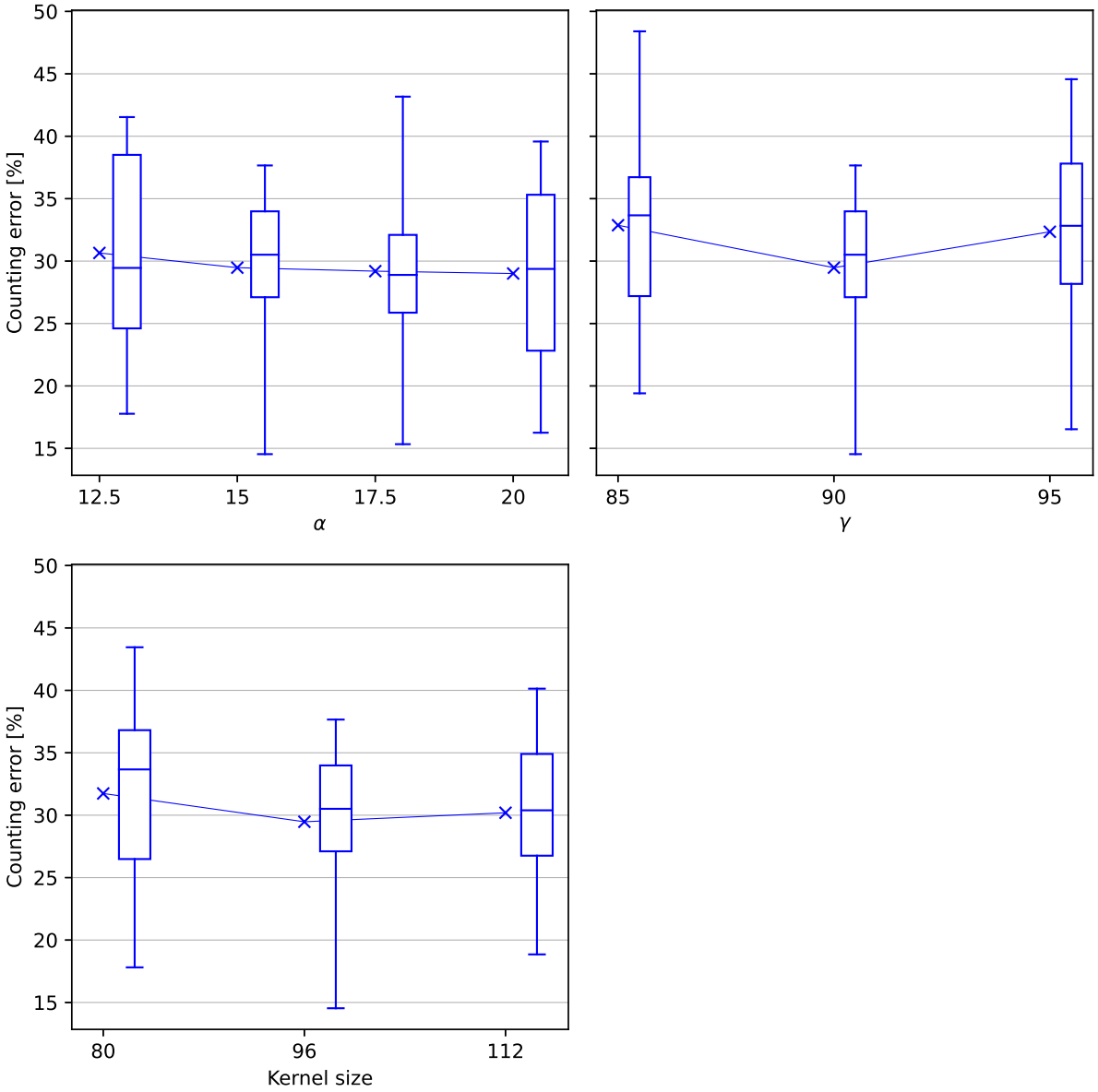


Figure A.55: Fine-tuning DTC parameters for Minimi 5.56. Optimal parameters determined through individual tuning are combined, and their immediate neighbourhood is explored.

MAG

MAG uses the Minimi 5.56 best parameters as basis

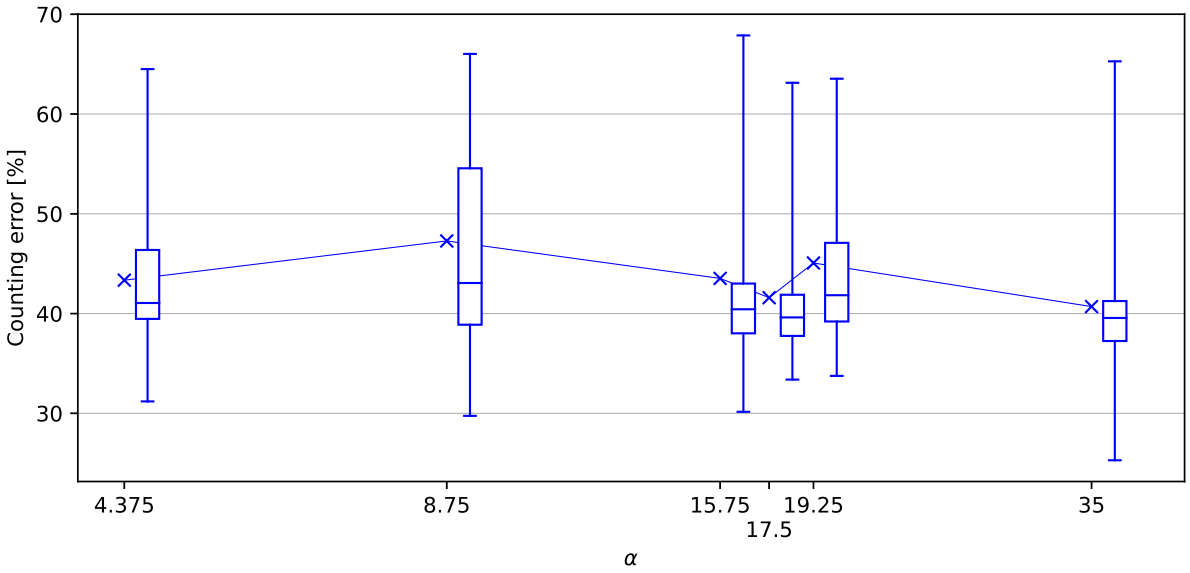


Figure A.56: DTC α parameter space for MAG. Effect is weak, with a similar minimum region to the Minimi 5.56.

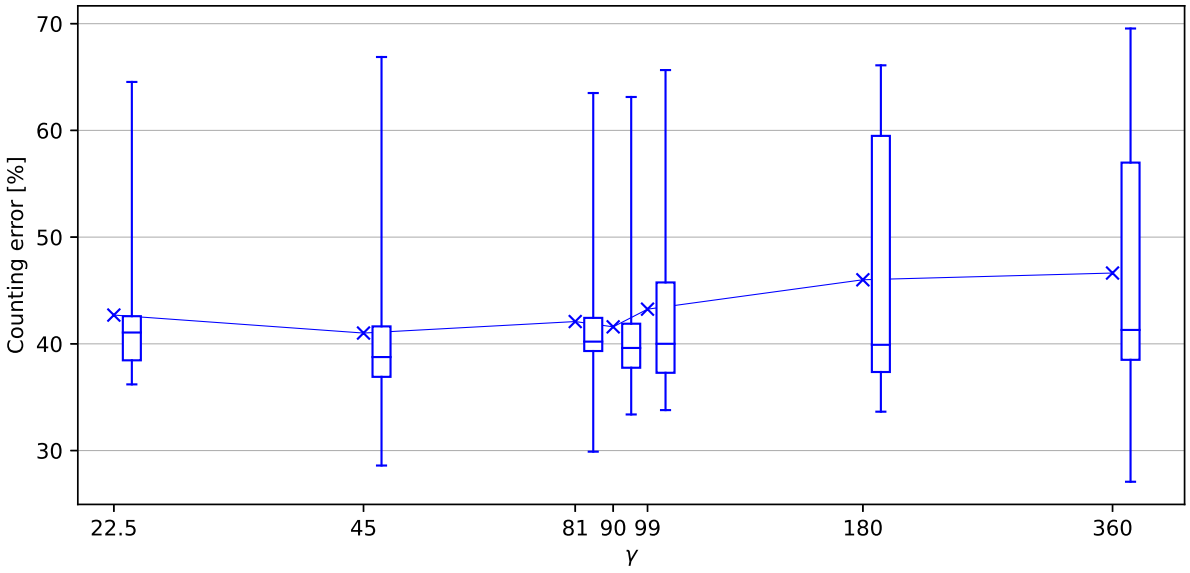


Figure A.57: DTC γ parameter space for MAG. Effect is weak, with a similar minimum region to the Minimi 5.56.

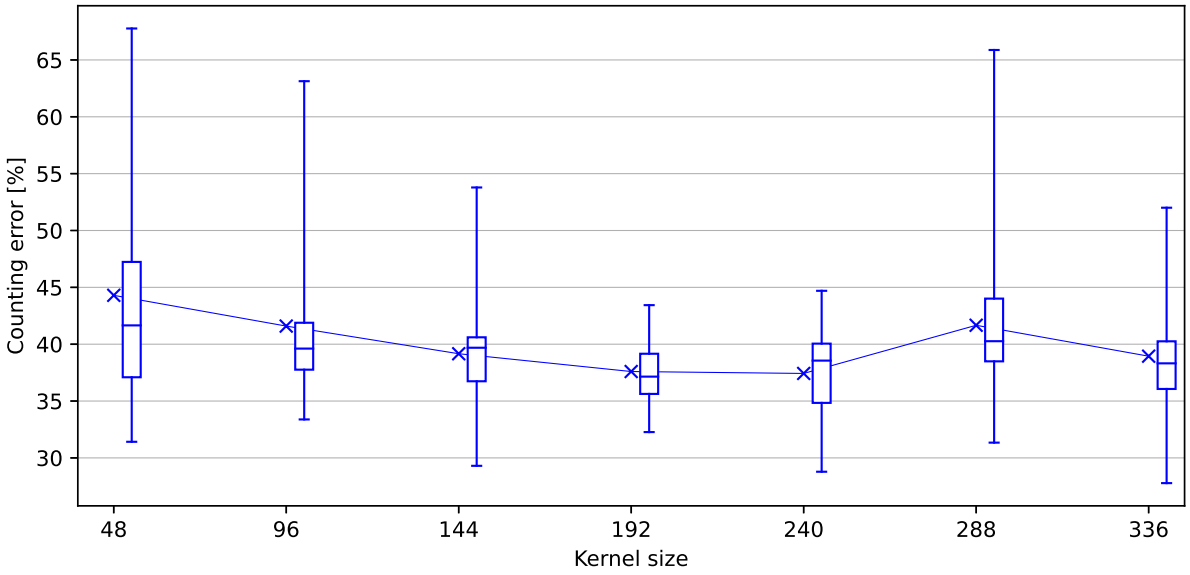


Figure A.58: DTC kernel size parameter space for MAG. A slightly higher value is beneficial.

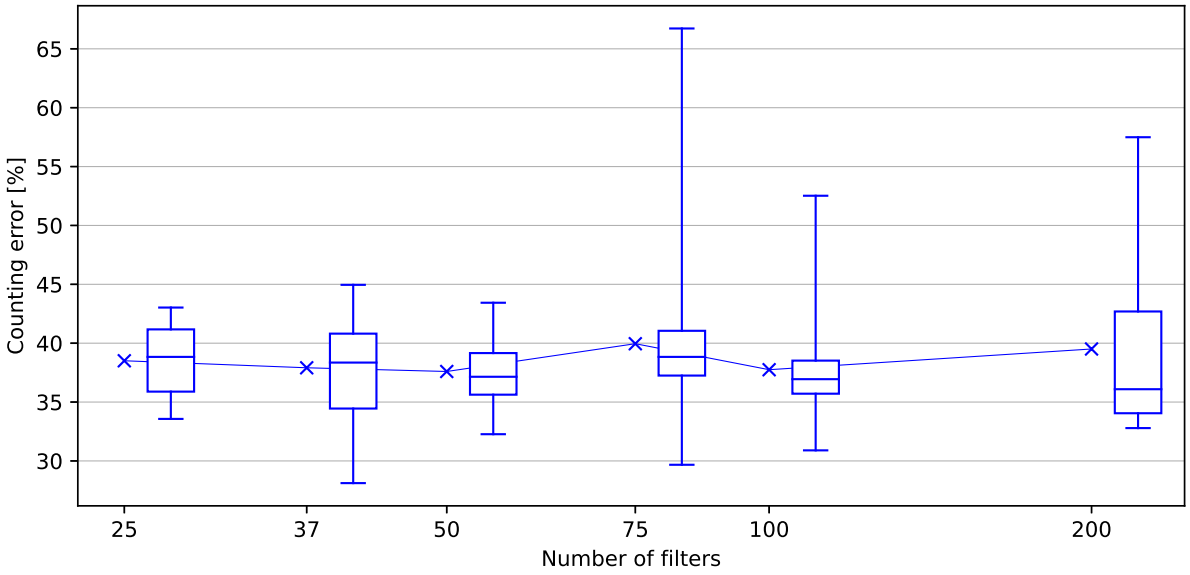


Figure A.59: DTC filter count parameter space for MAG. Larger filter counts yield no significant improvement.

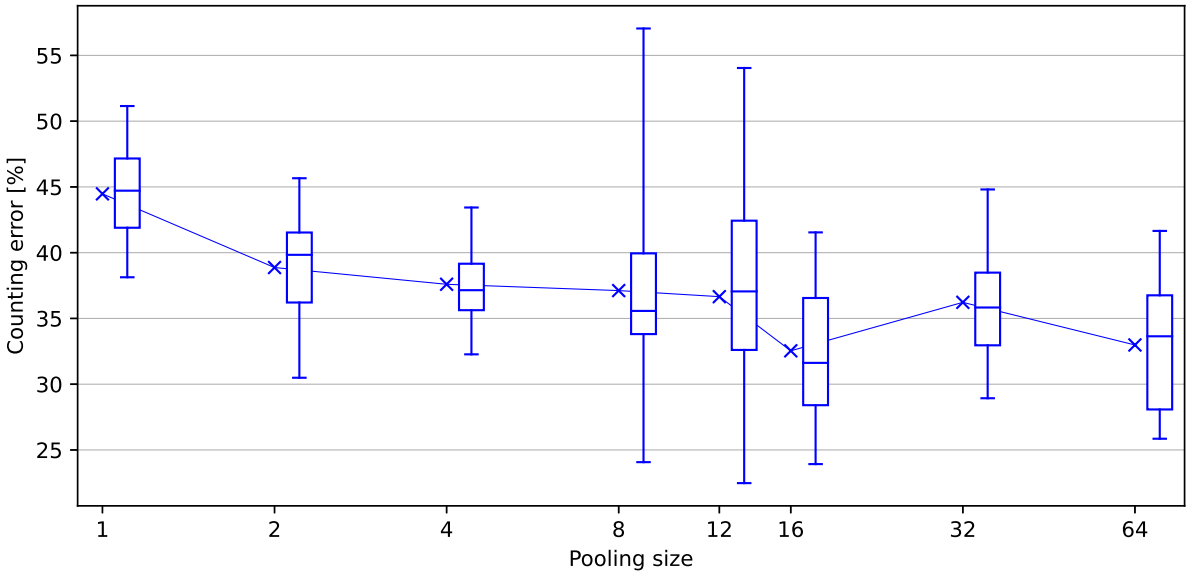


Figure A.60: DTC maxpool size parameter space for MAG. A pooling size significantly larger than for the Minimi 5.56 is beneficial.

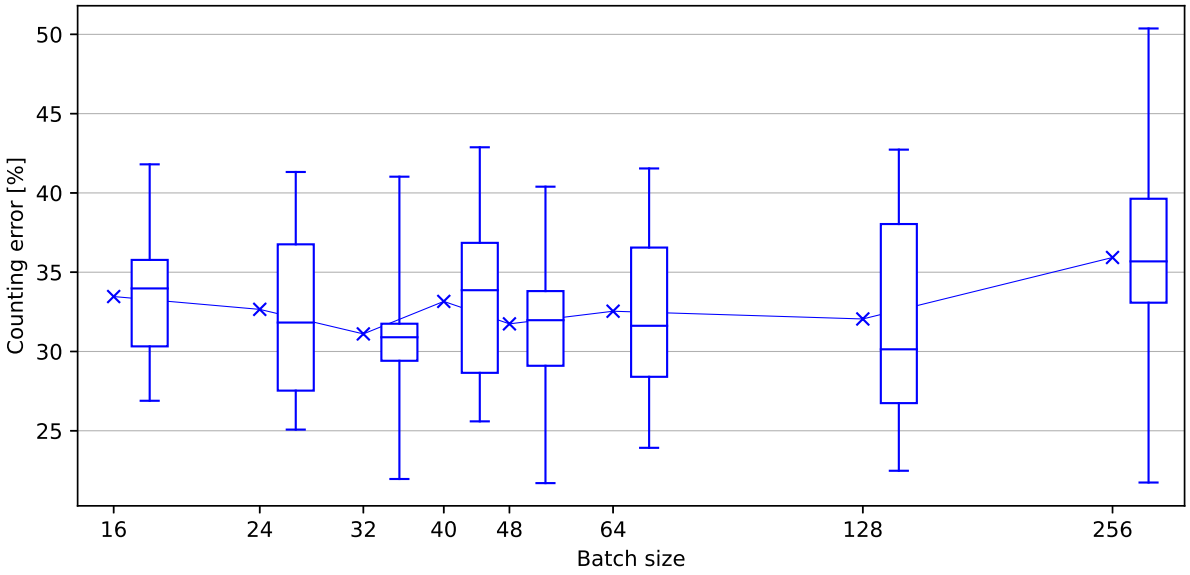


Figure A.61: DTC batch size parameter space for MAG. A local minimum is found around 32.

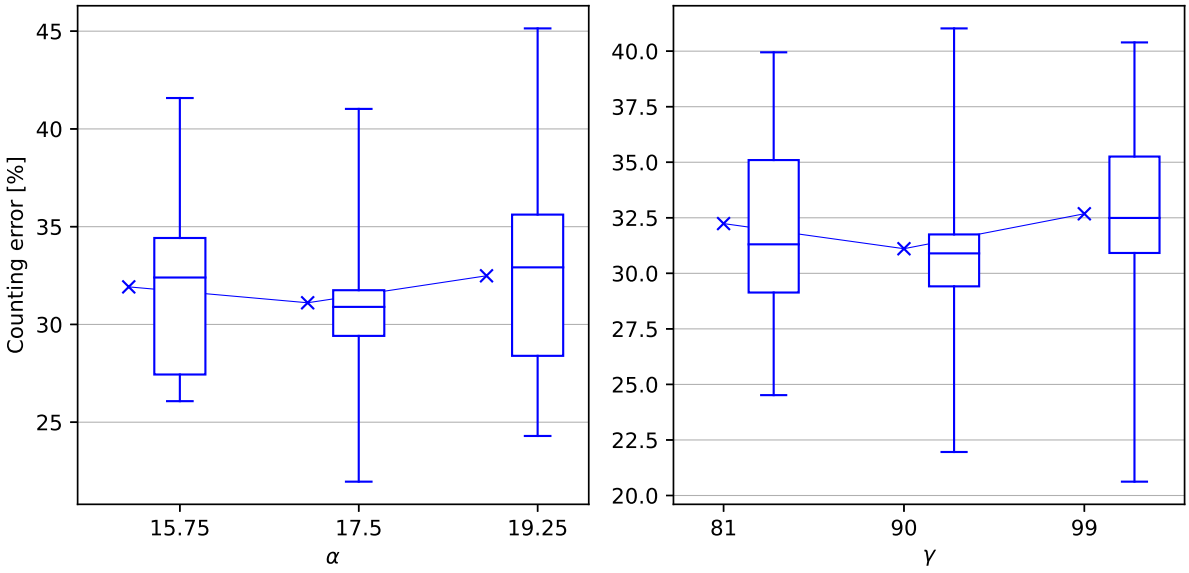


Figure A.62: Final local fine-tuning of α and γ after the optimal values found through individual tuning were applied. The local minimum for these values was confirmed.

A.5 Pretraining

This section presents more in-depth results on the effects of the pretraining process described in Section 6.9, for both counting and discrimination tasks.

A.5.1 Counting

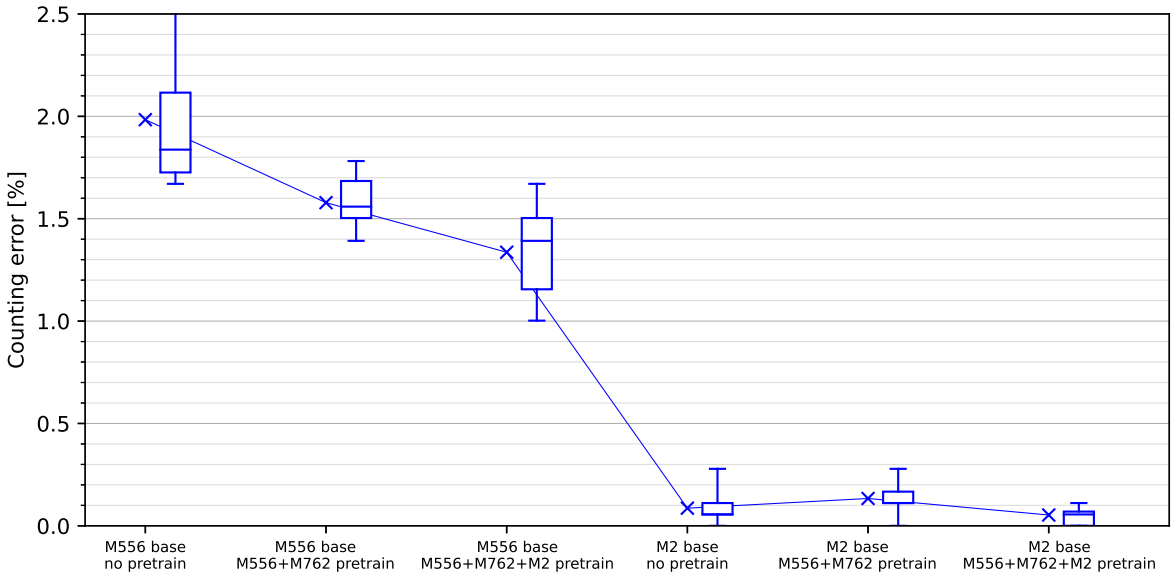


Figure A.63: Investigating the best baseline hyperparameters for MAG. Minimi 5.56 and M2 hyperparameter sets are considered. The second shows significantly better performance. The effect of pretraining is also compared at this stage. We compare no pretraining, pretraining with Minimi 5.56 and Minimi 7.62 data (which share a similar firing cycle), and finally with M2 data, which uses a very different ammunition and firing cycle. Results show that more input data, even with a different firing cycle, is beneficial.

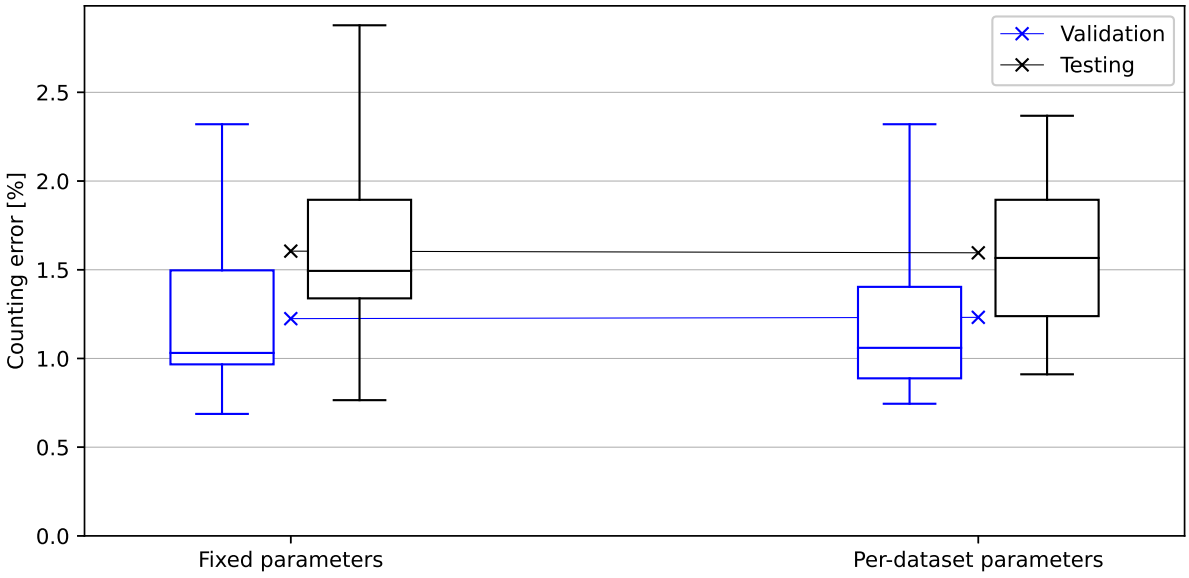


Figure A.64: MAG pretraining settings for the counting problem. We compare between fixed preprocessing parameters for all pretraining datasets, and the per-dataset parameters used by the associated best counting model. No significant difference is found, likely due to the small difference in parameters.

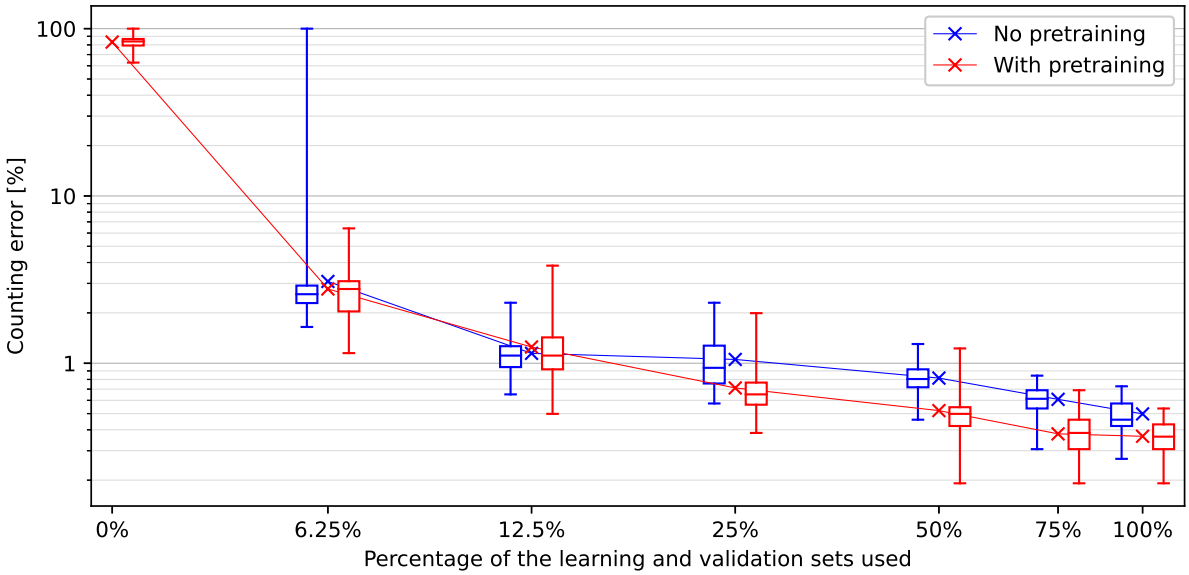


Figure A.65: Effect of pretraining for MAG for the counting problem. Pretraining is performed on the Minimi 5.56, Minimi 7.62 and M2 backplate datasets, and different fractions of the full MAG dataset. Fractioning was performed using two different random samplings and results were merged. We observe that performance with 0% MAG-specific data is extremely poor, underlining the need for target-specific data. Pretraining is however beneficial at for all considered fractions.

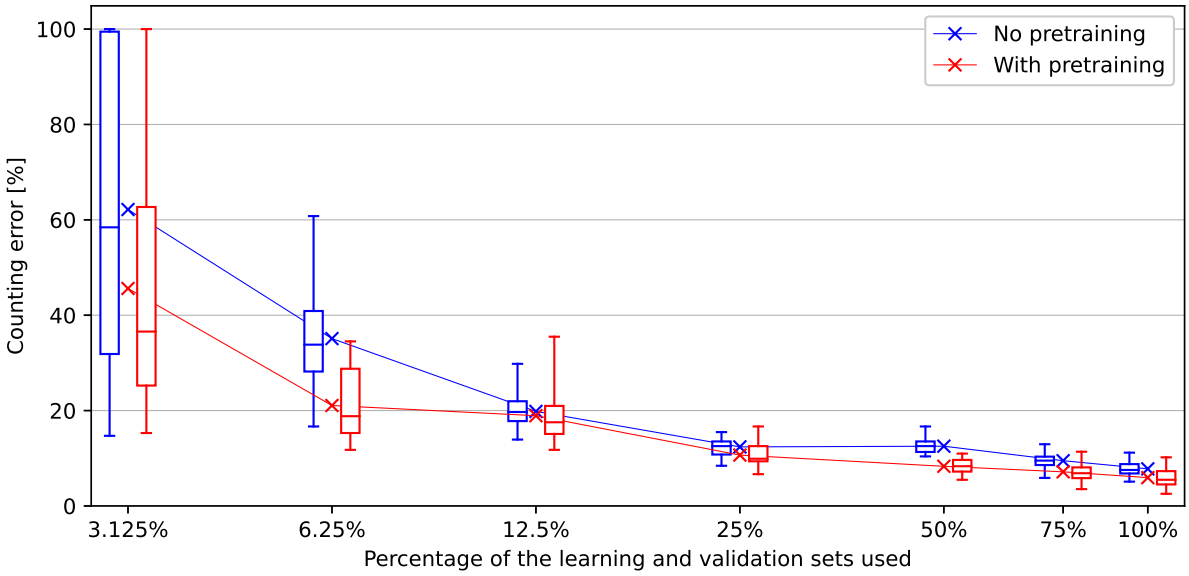


Figure A.66: Effect of pretraining for M2 backplate for the counting problem. Pretraining is performed with the Minimi 5.56 and Minimi 7.62 datasets, and different fractions of the full M2 backplate dataset. Fractioning was performed using two different random samplings and results were merged.

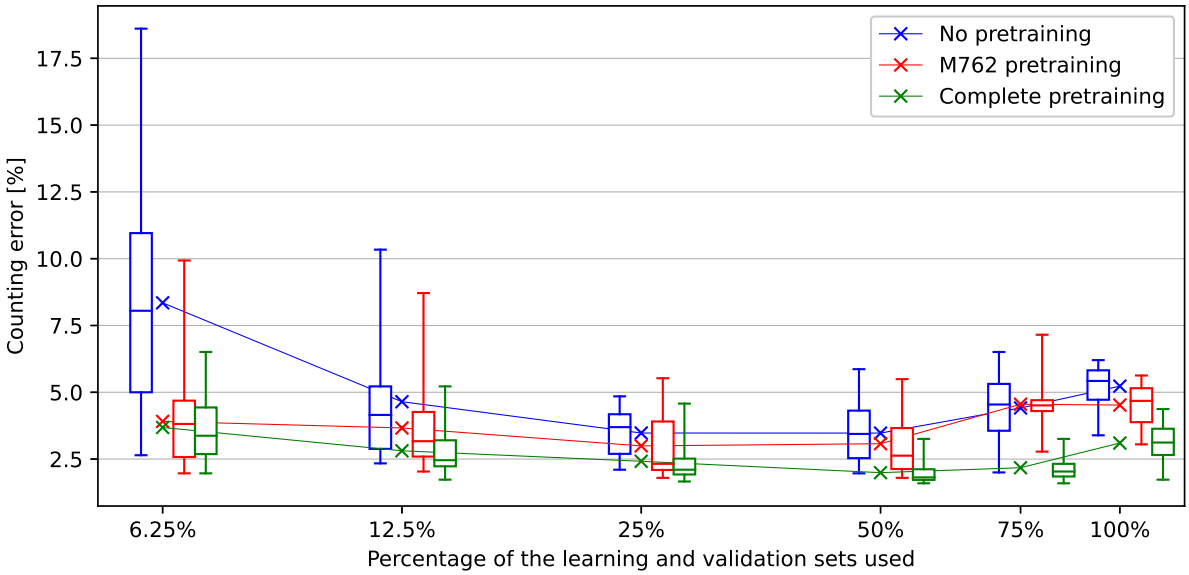


Figure A.67: Effect of pretraining for Minimi 5.56 for the counting problem. "Complete" pretraining is performed with the Minimi 7.62, MAG and M2 backplate datasets, and different fractions of the full Minimi 5.56 dataset. "M762" pretraining uses only the Minimi 7.62 and fractional Minimi 5.56 datasets. The performance degradation at higher fractions is due to the test set presenting a different distribution. We observe that while the performance benefit given by the M762 and Complete pretraining is initially similar, the Complete version generalises significantly better with a higher fraction of the Minimi 5.56 dataset.

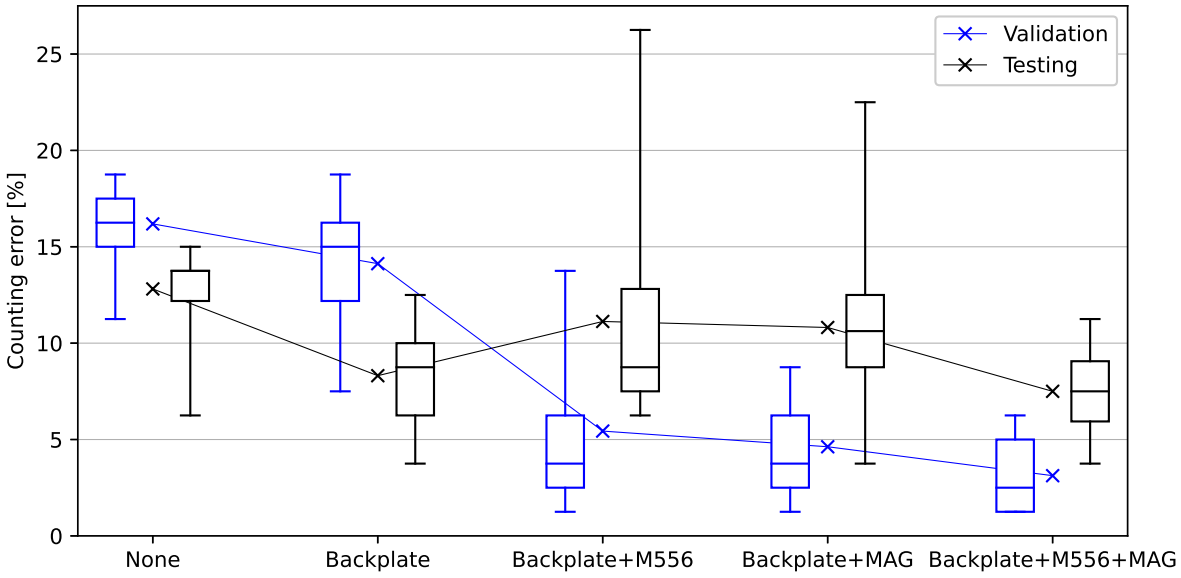


Figure A.68: Comparing pretraining datasets for M2 barrel for the counting problem. Usage of the adjacent M2 backplate dataset is significantly beneficial. The individual addition of the Minimi 5.56 and MAG datasets is beneficial on the validation set but not the test set. However, the complete pretraining provides the best performance on both datasets.

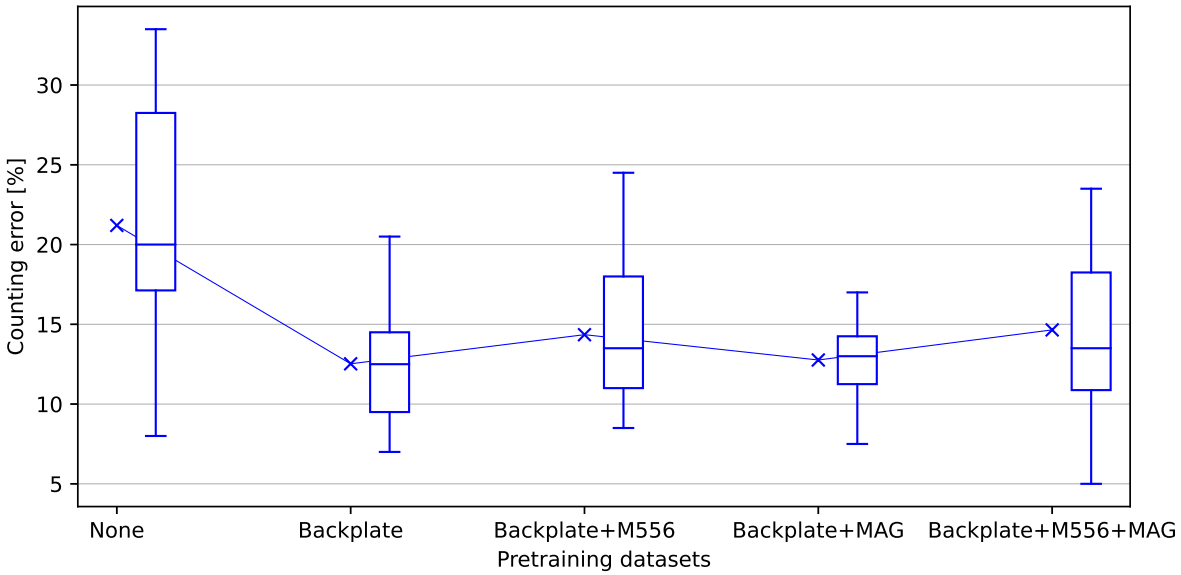


Figure A.69: Comparing pretraining datasets for M2 barrel for the counting problem. Usage of the adjacent M2 backplate dataset is significantly beneficial. Additional datasets only show marginal benefits. The largest pretraining provides the best performance.

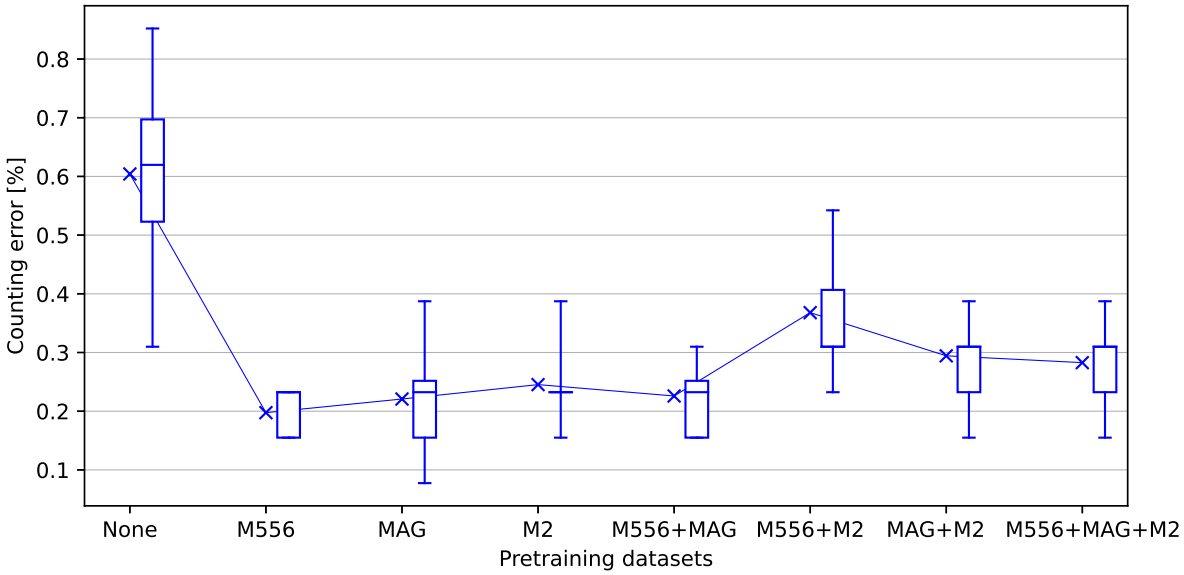


Figure A.70: Comparing pretraining datasets for Minimi 7.62 for the counting problem (evaluated on the validation set as a test set was not available). All combinations of Minimi 5.56 (M556), MAG and M2 datasets are explored. The use of any pretraining significantly improves performance. Only a slight difference exists between the different pretrainings, with the use of only Minimi 5.56 data being the most effective.

A.5.2 Discrimination

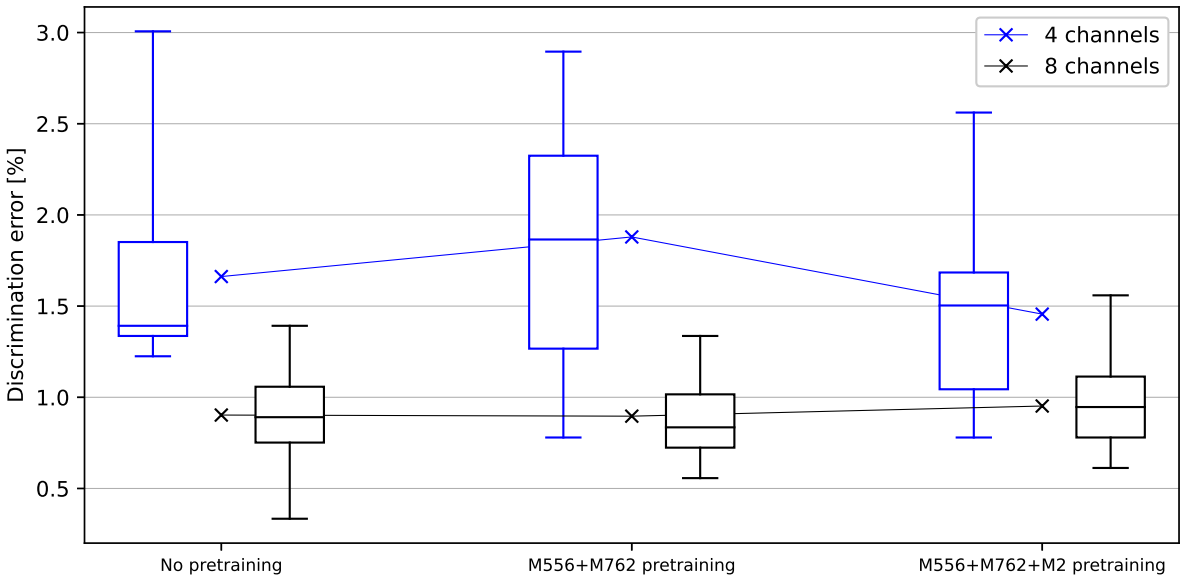


Figure A.71: Pretraining effect on MAG discrimination in the case of split networks ($D = 0$). We compare no, partial (Minimi 5.56 and Minimi 7.62) and full (with the addition of M2 backplate) pretraining. This is evaluated on discrimination networks with either 4 or 8 convolution channels.

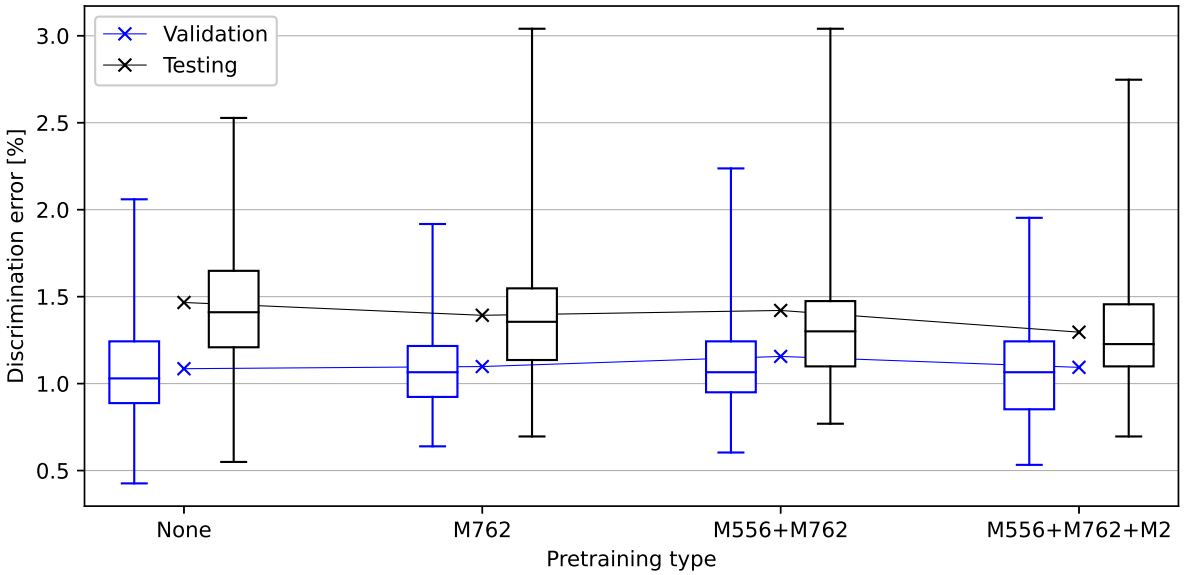


Figure A.72: Pretraining effect on MAG discrimination with shared activations ($D = 4$). We compare no, partial (Minimi 7.62 only, then combined with and Minimi 5.56) and full (with the addition of M2 backplate) pretraining. The MAG dataset used is larger than the one used in Figure A.71.

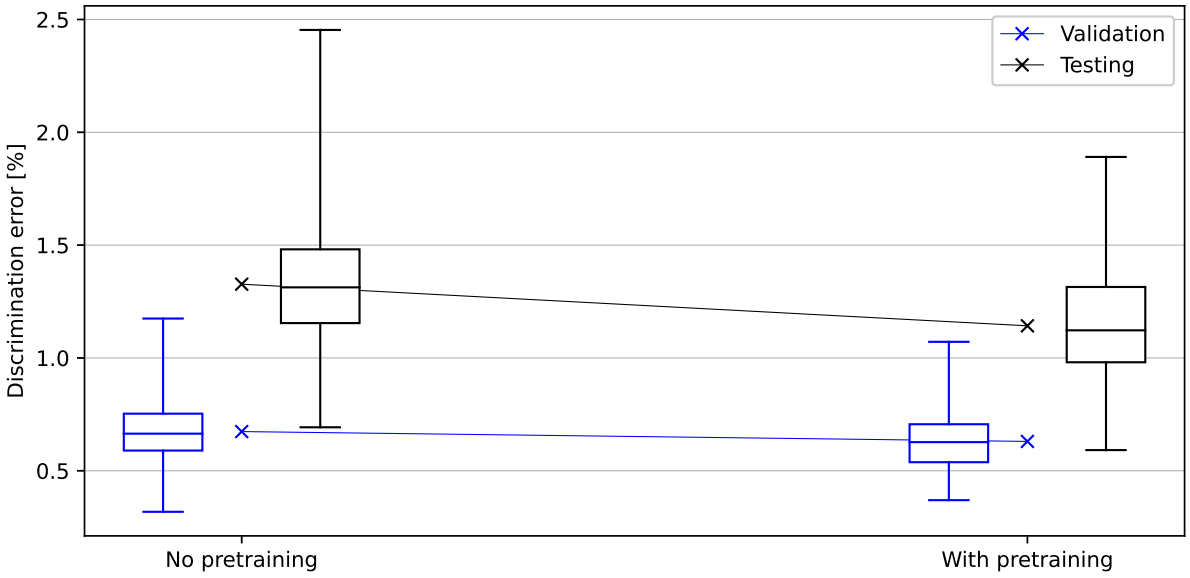


Figure A.73: Pretraining effect on MAG discrimination with shared activations ($D = 4$). Pretraining is performed with Minimi 5.56, Minimi 7.62 and M2 backplate. The datasets and parameters are more recent than Figure A.72.

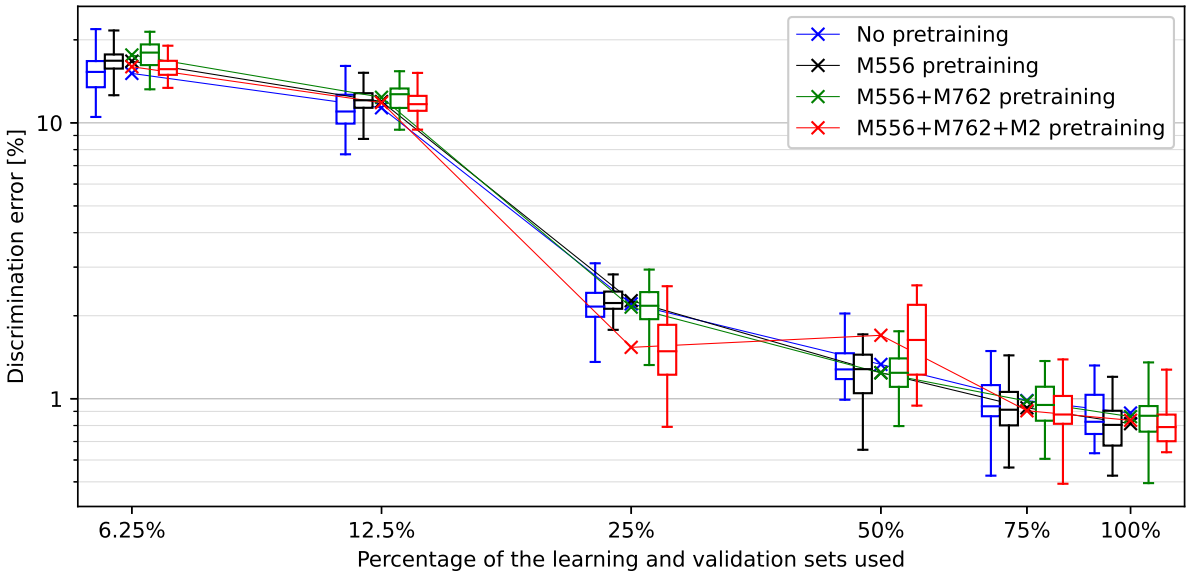


Figure A.74: Effect of pretraining for MAG for the discrimination problem. Pretraining is on datasets including successively Minimi 5.56, Minimi 7.62, M2 backplate, and different fractions of the full MAG dataset. Fractioning was performed using two different random samplings and results were merged. Results show a non-significant or even detrimental effect at lower fractions, but a beneficial effect at higher fractions.

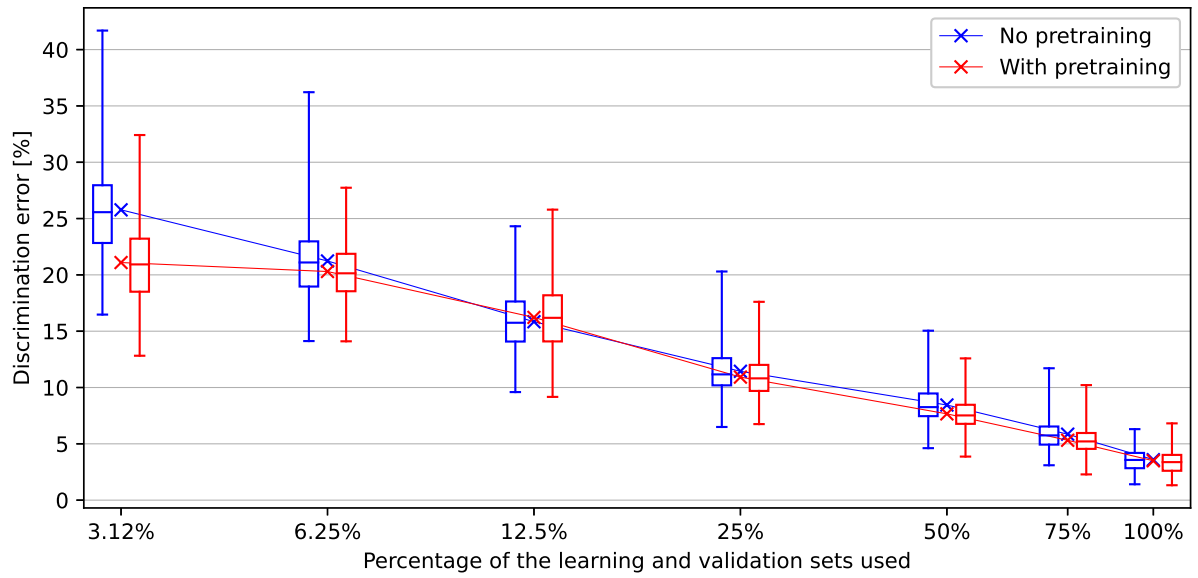


Figure A.75: Effect of pretraining for M2 backplate for the discrimination problem. Pretraining is performed on Minimi 5.56 and Minimi 7.62 datasets, and different fractions of the full M2 backplate dataset. Fractioning was performed using two different random samplings and results were merged. The effect is either neutral, or slightly beneficial at all fractions.

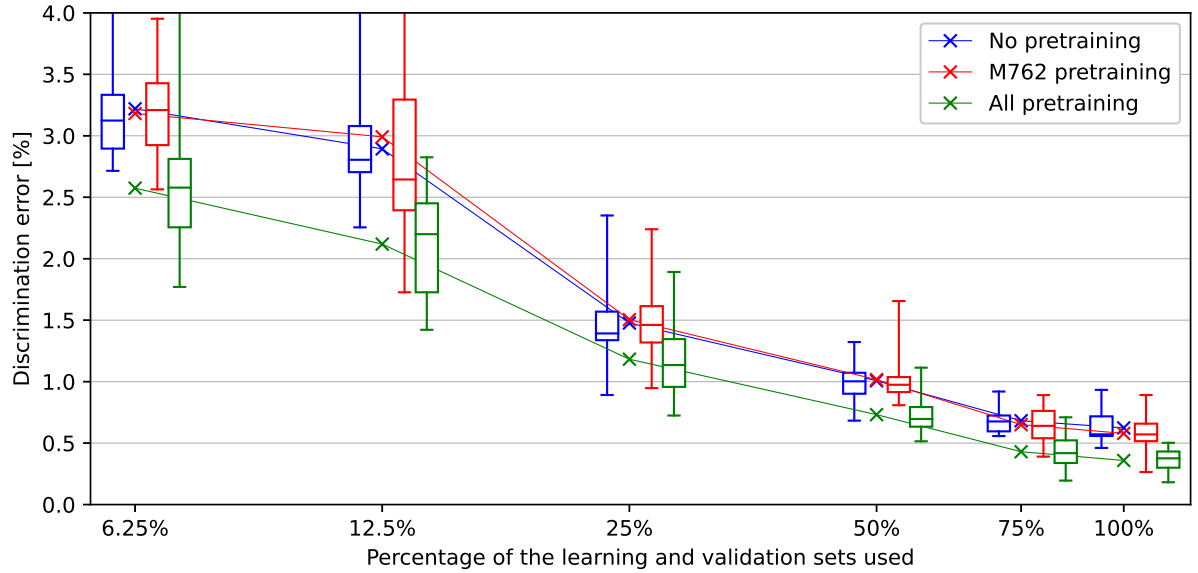


Figure A.76: Effect of pretraining for Minimi 5.56 for the discrimination problem on the **validation** dataset. "All pretraining" is performed with the Minimi 7.62, MAG and M2 backplate datasets, and different fractions of the full Minimi 5.56 dataset. "M762" pretraining uses only the Minimi 7.62 and fractional Minimi 5.56 datasets. While Minimi 7.62 pretraining only shows a slight effect, the complete pretraining shows a significantly beneficial effect.

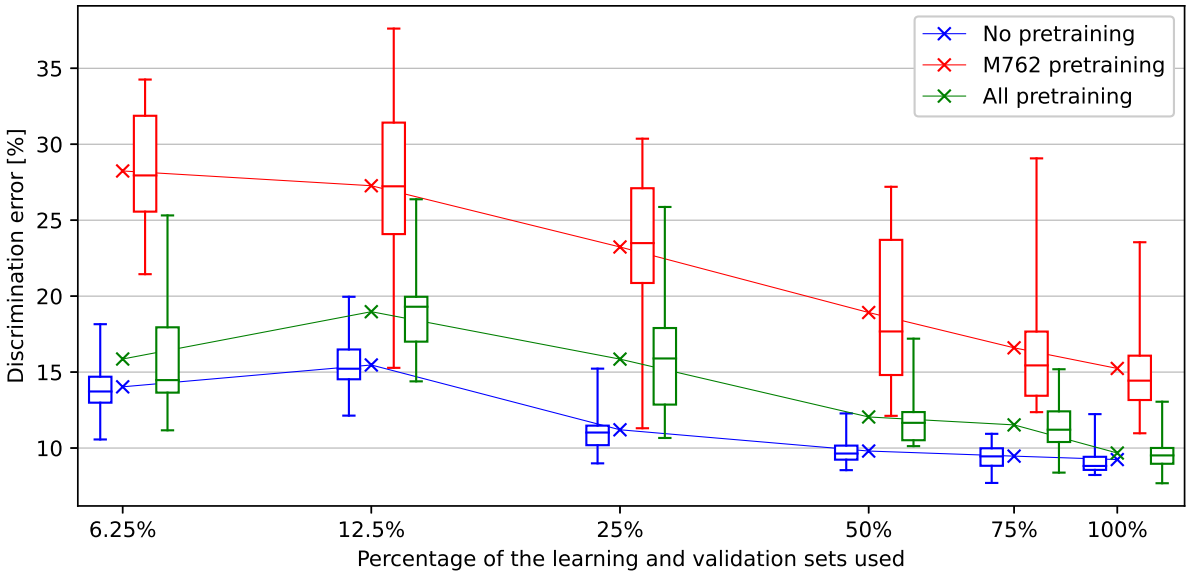


Figure A.77: Effect of pretraining for Minimi 5.56 for the discrimination problem on the **test** dataset. "All pretraining" is performed with the Minimi 7.62, MAG and M2 backplate datasets, and different fractions of the full Minimi 5.56 dataset. "M762" pretraining uses only the Minimi 7.62 and fractional Minimi 5.56 datasets. In this experiment, pretraining generally performs worse than the non-pretrained version. This is due to the test dataset having a different distribution from the learning and validation datasets. However, complete pretraining with the full Minimi 5.56 dataset still achieves performance competitive with the best non-pretrained cases.

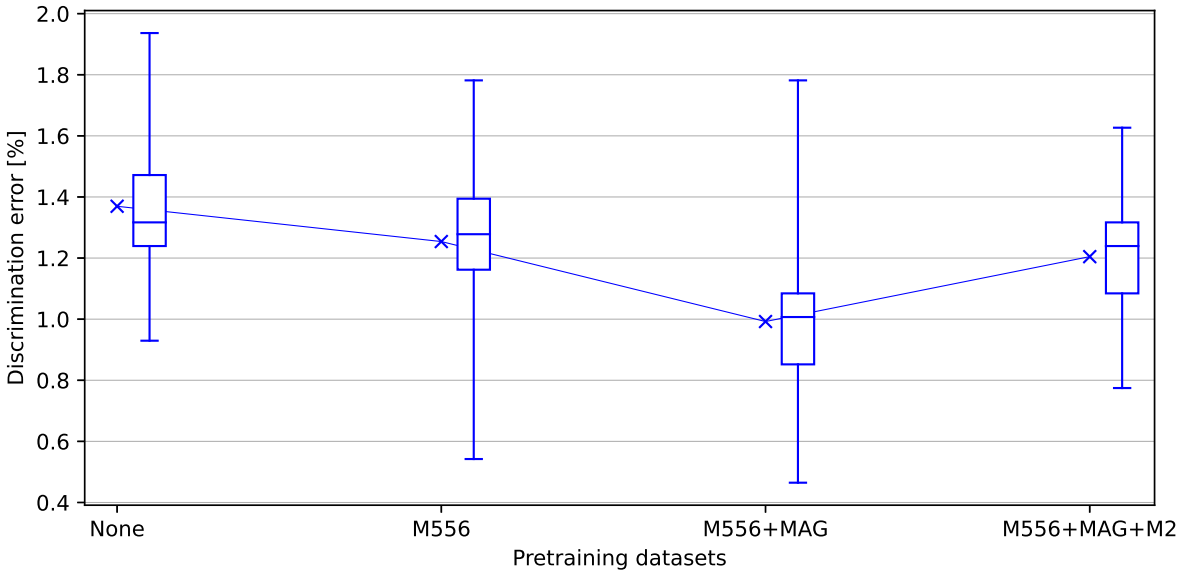


Figure A.78: Comparing pretraining datasets for Minimi 7.62 for the discrimination problem (evaluated on the validation set as a test set was not available). The use of Minimi 5.56 (M556) and MAG datasets, which share a similar firing cycle, are both beneficial. However, the use of the M2 dataset with a different firing cycle is detrimental.

Appendix B: Extended benchmark results

Chapter 9 presented the key experimental results on counting and discrimination benchmarks. This appendix complements those findings by providing extended results, including the full set of defined performance metrics across all considered datasets.

B.1 Counting offline benchmarks

Learning set				
	EDGAR	Human	DTC	Random
E_C	0.31%	1.08%	20.76%	27.19%
$\bar{E}_C$	**0.33%**	**0.76%**	**21.92%**	**27.69%**
E_{C_B}	0.28%	1.47%	19.33%	26.56%
E_{C_K}	0.39%	0.06%	24.5%	28.83%
$E_{C_{\text{norm}}}$	0.012	0.042	0.786	1.029
$\bar{E}_{C_{\text{norm}}}$	**0.013**	**0.029**	**0.849**	**1.073**
$E_{C_{\text{norm}_B}}$	0.011	0.057	0.749	1.028
$E_{C_{\text{norm}_K}}$	0.015	0.002	0.949	1.117
FP	**0**	**0**	**63**	**60**
Count	6497	6441	7893	6534
Ground-truth count: 6507				
Validation set				
	EDGAR	Human	DTC	Random
E_C	0%	1.39%	18.19%	25.46%
$\bar{E}_C$	**0%**	**1.23%**	**20.49%**	**24.75%**
E_{C_B}	0%	1.59%	15.48%	26.3%
E_{C_K}	0%	0.86%	25.5%	23.21%
$E_{C_{\text{norm}}}$	0	0.058	0.667	0.938
$\bar{E}_{C_{\text{norm}}}$	**0**	**0.054**	**0.777**	**0.951**
$E_{C_{\text{norm}_B}}$	0	0.063	0.599	1.018
$E_{C_{\text{norm}_K}}$	0	0.046	0.956	0.884
FP	**0**	**0**	**27**	**32**
Count	1292	1275	1552	1255
Ground-truth count: 1292				

Table B.1: Full counting benchmark for Minimi 7.62.

Learning set				
	EDGAR	Human	DTC	Random
E_C	0.16%	1.79%	11.34%	30.61%
$\bar{E}_C$	**0.21%**	**1.62%**	**13%**	**30.4%**
E_{C_B}	0.04%	2.18%	7.81%	31.05%
E_{C_K}	0.38%	1.05%	18.19%	29.75%
$E_{C_{\text{norm}}}$	0.005	0.070	0.358	0.965
$\bar{E}_{C_{\text{norm}}}$	**0.008**	**0.063**	**0.503**	**1.175**
$E_{C_{\text{norm}_B}}$	0.002	0.084	0.302	1.198
$E_{C_{\text{norm}_K}}$	0.015	0.041	0.704	1.152
FP	**0**	**188**	**272**	**424**
Count	6957	7037	7934	6957
Ground-truth count: 6966				
Validation set				
	EDGAR	Human	DTC	Random
E_C	0.04%	1.56%	10.33%	27.73%
$\bar{E}_C$	**0.03%**	**1.32%**	**11.52%**	**27.67%**
E_{C_B}	0.05%	1.93%	8.49%	27.81%
E_{C_K}	0%	0.72%	14.54%	27.53%
$E_{C_{\text{norm}}}$	0.001	0.060	0.372	0.998
$\bar{E}_{C_{\text{norm}}}$	**0.001**	**0.051**	**0.446**	**1.071**
$E_{C_{\text{norm}_B}}$	0.002	0.075	0.329	1.077
$E_{C_{\text{norm}_K}}$	0	0.028	0.564	1.066
FP	**0**	**30**	**40**	**72**
Count	2760	2752	3064	2839
Ground-truth count: 2759				
Test set				
	EDGAR	Human	DTC	Random
E_C	1.63%	1.76%	13.02%	15.12%
$\bar{E}_C$	**1.38%**	**1.57%**	**13.92%**	**14.32%**
E_{C_B}	2.5%	2.44%	9.83%	17.94%
E_{C_K}	0.26%	0.70%	18%	10.7%
$E_{C_{\text{norm}}}$	0.115	0.125	0.92	1.069
$\bar{E}_{C_{\text{norm}}}$	**0.098**	**0.111**	**0.984**	**1.013**
$E_{C_{\text{norm}_B}}$	0.177	0.173	0.695	1.269
$E_{C_{\text{norm}_K}}$	0.018	0.049	1.273	0.756
FP	**0**	**0**	**0**	**0**
Count	2904	2910	3282	2632
Ground-truth count: 2950				

Table B.2: Full counting benchmark for Minimi 5.56.

Learning set				
	EDGAR	Human	DTC	Random
E_C	0.49%	2.90%	22.16%	21.27%
$\bar{E}_C$	**0.41%**	**2.19%**	**20.8%**	**21.15%**
E_{C_B}	0.63%	4.27%	24.77%	21.51%
E_{C_K}	0.2%	0.11%	16.84%	20.79%
$E_{C_{\text{norm}}}$	0.018	0.112	0.807	0.774
$\bar{E}_{C_{\text{norm}}}$	**0.016**	**0.085**	**0.805**	**0.818**
$E_{C_{\text{norm}_B}}$	0.024	0.165	0.959	0.833
$E_{C_{\text{norm}_K}}$	0.008	0.004	0.651	0.804
FP	**5**	**20**	**1089**	**1219**
Count	13996	13623	16474	14138
Ground-truth count: 14009				
Validation set				
	EDGAR	Human	DTC	Random
E_C	0.6%	3.58%	24.35%	21.43%
$\bar{E}_C$	**0.52%**	**2.91%**	**23.31%**	**21.38%**
E_{C_B}	0.88%	5.75%	27.74%	21.57%
E_{C_K}	0.15%	0.07%	18.88%	21.2%
$E_{C_{\text{norm}}}$	0.022	0.139	0.896	0.789
$\bar{E}_{C_{\text{norm}}}$	**0.02**	**0.113**	**0.902**	**0.828**
$E_{C_{\text{norm}_B}}$	0.034	0.222	1.073	0.834
$E_{C_{\text{norm}_K}}$	0.006	0.003	0.731	0.821
FP	**3**	**7**	**178**	**179**
Count	3481	3373	4009	3459
Ground-truth count: 3491				
Test set				
	EDGAR	Human	DTC	Random
E_C	0.69%	4.88%	25.5%	19.71%
$\bar{E}_C$	**0.56%**	**3.86%**	**24.15%**	**19.46%**
E_{C_B}	1.03%	7.52%	29%	20.34%
E_{C_K}	0.1%	0.20%	19.29%	18.59%
$E_{C_{\text{norm}}}$	0.025	0.189	0.922	0.713
$\bar{E}_{C_{\text{norm}}}$	**0.022**	**0.150**	**0.935**	**0.754**
$E_{C_{\text{norm}_B}}$	0.04	0.291	1.123	0.788
$E_{C_{\text{norm}_K}}$	0.004	0.008	0.747	0.72
FP	**2**	**3**	**147**	**174**
Count	2736	2616	3112	2759
Ground-truth count: 2745				

Table B.3: Full counting benchmark for MAG.

Learning set				
	EDGAR	Human	DTC	Random
E_C	2.88%	0.10%	37.29%	27.15%
$\bar{E}_C$	**2.92%**	**0.11%**	**37.59%**	**27.06%**
E_{C_B}	2.54%	0.08%	35.16%	27.83%
E_{C_K}	3.31%	0.13%	40.01%	26.28%
$E_{C_{\text{norm}}}$	0.077	0.003	1.008	0.731
$\bar{E}_{C_{\text{norm}}}$	**0.092**	**0.003**	**1.195**	**0.856**
$E_{C_{\text{norm}_B}}$	0.08	0.003	1.124	0.882
$E_{C_{\text{norm}_K}}$	0.105	0.004	1.265	0.831
FP	**5**	**0**	**3134**	**901**
Count	8631	8757	13157	8779
Ground-truth count: 8762				
Validation set				
	EDGAR	Human	DTC	Random
E_C	2.83%	0.00%	46.32%	28.42%
$\bar{E}_C$	**2.95%**	**0.00%**	**46.24%**	**28.12%**
E_{C_B}	2.09%	0.00%	46.79%	30.19%
E_{C_K}	3.81%	0.00%	45.69%	26.05%
$E_{C_{\text{norm}}}$	0.076	0.000	1.194	0.759
$\bar{E}_{C_{\text{norm}}}$	**0.094**	**0.000**	**1.414**	**0.887**
$E_{C_{\text{norm}_B}}$	0.067	0.000	1.385	0.952
$E_{C_{\text{norm}_K}}$	0.121	0.000	1.442	0.823
FP	**2**	**0**	**394**	**111**
Count	1153	1168	1825	1259
Ground-truth count: 1168				
Test set				
	EDGAR	Human	DTC	Random
E_C	9.41%	1.76%	39.22%	23.33%
$\bar{E}_C$	**9.58%**	**1.80%**	**39.38%**	**23.33%**
E_{C_B}	0.77%	0.00%	30.77%	23.46%
E_{C_K}	18.4%	3.60%	48%	23.2%
$E_{C_{\text{norm}}}$	0.281	0.056	1.171	0.697
$\bar{E}_{C_{\text{norm}}}$	**0.303**	**0.057**	**1.245**	**0.738**
$E_{C_{\text{norm}_B}}$	0.024	0.000	0.973	0.742
$E_{C_{\text{norm}_K}}$	0.582	0.114	1.518	0.734
FP	**0**	**0**	**35**	**11**
Count	462	501	559	485
Ground-truth count: 510				

Table B.4: Full counting benchmark for M2 backplate.

Learning set			
	EDGAR	DTC	Random
E_C	1.1%	24.26%	25.1%
$E_{C_{\text{norm}}}$	0.029	0.646	0.669
FP	**0**	**206**	**150**
Count	1551	1916	1528
Ground-truth count: 1550			
Validation set			
	EDGAR	DTC	Random
E_C	0%	24.81%	32.59%
$E_{C_{\text{norm}}}$	0	0.623	0.818
FP	**0**	**49**	**42**
Count	270	332	273
Ground-truth count: 270			
Test set			
	EDGAR	DTC	Random
E_C	0%	27.78%	29.63%
$E_{C_{\text{norm}}}$	0	0.659	0.703
FP	**0**	**54**	**46**
Count	270	333	310
Ground-truth count: 270			

Table B.5: Full counting benchmark for M2 barrel.

Learning set			
	EDGAR	DTC	Random
E_C	0.3%	12.02%	15.47%
$E_{C_{\text{norm}}}$	0.013	0.419	0.526
FP	**1**	**129**	**138**
Count	1013	1062	1034
Ground-truth count: 1015			
Validation set			
	EDGAR	DTC	Random
E_C	0%	12.76%	20.69%
$E_{C_{\text{norm}}}$	0	0.574	0.841
FP	**0**	**23**	**28**
Count	290	299	276
Ground-truth count: 290			
Test set			
	EDGAR	DTC	Random
E_C	0.34%	11.72%	21.72%
$E_{C_{\text{norm}}}$	0.011	0.528	0.992
FP	**1**	**26**	**31**
Count	292	286	306
Ground-truth count: 290			

Table B.6: Full counting benchmark for M134.

B.2 Counting field performance

E_C	1.95%
$\bar{E}_C$	**1.25%**
E_{C_B}	1.99%
E_{C_K}	0.50%
$E_{C_{\text{norm}}}$	0.061
$\bar{E}_{C_{\text{norm}}}$	**0.048**
$E_{C_{\text{norm}_B}}$	0.072
$E_{C_{\text{norm}_K}}$	0.023
FP	$^0/_{>600}$
Count	48065
Ground-truth count: 47229	

Table B.7: Counting benchmark for Minimi 5.56.

E_C	1.33%
$\bar{E}_C$	**1.12%**
E_{C_B}	1.41%
E_{C_K}	0.84%
$E_{C_{\text{norm}}}$	0.057
$\bar{E}_{C_{\text{norm}}}$	**0.049**
$E_{C_{\text{norm}_B}}$	0.064
$E_{C_{\text{norm}_K}}$	0.033
FP	$^0/_{>400}$
Count	3457
Ground-truth count: 3463	

Table B.8: Counting benchmark for MAG.

E_C	0.09%
$\bar{E}_C$	**0.08%**
E_{C_B}	0.00%
E_{C_K}	0.16%
$E_{C_{\text{norm}}}$	0.003
$\bar{E}_{C_{\text{norm}}}$	**0.003**
$E_{C_{\text{norm}_B}}$	0.000
$E_{C_{\text{norm}_K}}$	0.006
FP	$^1/_{>100}$
Count	1172
Ground-truth count: 1170	

Table B.9: Counting benchmark for Minimi 7.62.

E_C	1.26%
$E_{C_{\text{norm}}}$	0.081
FP	$^0/_{>750}$
Count	714
Ground-truth count: 715	

Table B.10: Counting benchmark for M134.

B.3 Discrimination offline benchmarks

Classical (raw)

$\bar{\mathbf{E}}_\mathbf{D}$	**26.94%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**71.97%**
E_{D_B}	24.80%	d_{Bj}	3384	1116	F_{1_B}	79.00%
E_{D_K}	29.08%	d_{Kj}	683	1666	F_{1_K}	64.94%

Classical (voted)

$\bar{\mathbf{E}}_\mathbf{D}$	**19.21%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**80.07%**
E_{D_B}	16.49%	d_{Bj}	3758	742	F_{1_B}	85.67%
E_{D_K}	21.92%	d_{Kj}	515	1834	F_{1_K}	74.48%

EDGAR (raw)

$\bar{\mathbf{E}}_\mathbf{D}$	**0.00%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**100%**
E_{D_B}	0.00%	d_{Bj}	4594	0	F_{1_B}	100%
E_{D_K}	0.00%	d_{Kj}	0	2363	F_{1_K}	100%

EDGAR (voted)

$\bar{\mathbf{E}}_\mathbf{D}$	**0.00%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**100%**
E_{D_B}	0.00%	d_{Bj}	4594	0	F_{1_B}	100%
E_{D_K}	0.00%	d_{Kj}	0	2363	F_{1_K}	100%

Table B.11: Full discrimination benchmark for Minimi 5.56 - learning set.

Classical (raw)						
$\bar{\mathbf{E}}_{\mathbf{D}}$	**25.52%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_{\mathbf{1}}$	**72.61%**
E_{D_B}	23.74%	d_{Bj}	1439	448	F_{1_B}	80.98%
E_{D_K}	27.31%	d_{Kj}	228	607	F_{1_K}	64.23%
Classical (voted)						
$\bar{\mathbf{E}}_{\mathbf{D}}$	**17.06%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_{\mathbf{1}}$	**82.86%**
E_{D_B}	10.76%	d_{Bj}	1684	203	F_{1_B}	89.43%
E_{D_K}	23.35%	d_{Kj}	195	640	F_{1_K}	76.28%
EDGAR (raw)						
$\bar{\mathbf{E}}_{\mathbf{D}}$	**0.60%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_{\mathbf{1}}$	**99.40%**
E_{D_B}	0.36%	d_{Bj}	1914	7	F_{1_B}	99.64%
E_{D_K}	0.83%	d_{Kj}	7	832	F_{1_K}	99.17%
EDGAR (voted)						
$\bar{\mathbf{E}}_{\mathbf{D}}$	**0.00%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_{\mathbf{1}}$	**100%**
E_{D_B}	0.00%	d_{Bj}	1921	0	F_{1_B}	100%
E_{D_K}	0.00%	d_{Kj}	0	839	F_{1_K}	100%

Table B.12: Full discrimination benchmark for Minimi 5.56 - validation set.

Classical (raw)						
$\bar{\mathbf{E}}_{\mathbf{D}}$	**33.55%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_{\mathbf{1}}$	**66.40%**
E_{D_B}	27.25%	d_{Bj}	1279	479	F_{1_B}	73.17%
E_{D_K}	39.84%	d_{Kj}	459	693	F_{1_K}	59.64%
Classical (voted)						
$\bar{\mathbf{E}}_{\mathbf{D}}$	**10.81%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_{\mathbf{1}}$	**90.63%**
E_{D_B}	0.00%	d_{Bj}	1758	0	F_{1_B}	93.39%
E_{D_K}	21.61%	d_{Kj}	249	903	F_{1_K}	87.88%
EDGAR (raw)						
$\bar{\mathbf{E}}_{\mathbf{D}}$	**3.74%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_{\mathbf{1}}$	**96.22%**
E_{D_B}	3.13%	d_{Bj}	1702	55	F_{1_B}	97.01%
E_{D_K}	4.36%	d_{Kj}	50	1097	F_{1_K}	95.43%
EDGAR (voted)						
$\bar{\mathbf{E}}_{\mathbf{D}}$	**0.00%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_{\mathbf{1}}$	**100%**
E_{D_B}	0.00%	d_{Bj}	1757	0	F_{1_B}	100%
E_{D_K}	0.00%	d_{Kj}	0	1147	F_{1_K}	100%

Table B.13: Full discrimination benchmark for Minimi 5.56 - test set.

Classical (raw)						
$\bar{\mathbf{E}}_{\mathbf{D}}$	**10.42%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_{\mathbf{1}}$	**87.52%**
E_{D_B}	10.73%	d_{Bj}	4144	498	F_{1_B}	92.42%
E_{D_K}	10.12%	d_{Kj}	182	1617	F_{1_K}	82.63%
Classical (voted)						
$\bar{\mathbf{E}}_{\mathbf{D}}$	**5.76%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_{\mathbf{1}}$	**91.56%**
E_{D_B}	9.03%	d_{Bj}	4223	419	F_{1_B}	94.79%
E_{D_K}	2.50%	d_{Kj}	45	1754	F_{1_K}	88.32%
EDGAR (raw)						
$\bar{\mathbf{E}}_{\mathbf{D}}$	**0.00%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_{\mathbf{1}}$	**100%**
E_{D_B}	0.00%	d_{Bj}	4702	0	F_{1_B}	100%
E_{D_K}	0.00%	d_{Kj}	0	1795	F_{1_K}	100%
EDGAR (voted)						
$\bar{\mathbf{E}}_{\mathbf{D}}$	**0.00%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_{\mathbf{1}}$	**100%**
E_{D_B}	0.00%	d_{Bj}	4702	0	F_{1_B}	100%
E_{D_K}	0.00%	d_{Kj}	0	1795	F_{1_K}	100%

Table B.14: Full discrimination benchmark for Minimi 7.62 - learning set.

Classical (raw)						
$\bar{\mathbf{E}}_{\mathbf{D}}$	**8.89%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_{\mathbf{1}}$	**90.45%**
E_{D_B}	6.25%	d_{Bj}	870	58	F_{1_B}	94.67%
E_{D_K}	11.53%	d_{Kj}	40	307	F_{1_K}	86.24%
Classical (voted)						
$\bar{\mathbf{E}}_{\mathbf{D}}$	**3.23%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_{\mathbf{1}}$	**94.35%**
E_{D_B}	6.47%	d_{Bj}	868	60	F_{1_B}	96.66%
E_{D_K}	0.00%	d_{Kj}	0	347	F_{1_K}	92.04%
EDGAR (raw)						
$\bar{\mathbf{E}}_{\mathbf{D}}$	**0.39%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_{\mathbf{1}}$	**99.61%**
E_{D_B}	0.21%	d_{Bj}	941	2	F_{1_B}	99.79%
E_{D_K}	0.57%	d_{Kj}	2	347	F_{1_K}	99.43%
EDGAR (voted)						
$\bar{\mathbf{E}}_{\mathbf{D}}$	**0.00%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_{\mathbf{1}}$	**100%**
E_{D_B}	0.00%	d_{Bj}	943	0	F_{1_B}	100%
E_{D_K}	0.00%	d_{Kj}	0	349	F_{1_K}	100%

Table B.15: Full discrimination benchmark for Minimi 7.62 - validation set.

Classical (raw)						
$\bar{\mathbf{E}}_{\mathbf{D}}$	**35.77%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_{\mathbf{1}}$	**63.92%**
E_{D_B}	27.03%	d_{Bj}	6567	2432	F_{1_B}	74.56%
E_{D_K}	44.50%	d_{Kj}	2049	2555	F_{1_K}	53.28%
Classical (voted)						
$\bar{\mathbf{E}}_{\mathbf{D}}$	**34.76%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_{\mathbf{1}}$	**64.95%**
E_{D_B}	25.98%	d_{Bj}	6661	2338	F_{1_B}	75.41%
E_{D_K}	43.55%	d_{Kj}	2005	2599	F_{1_K}	54.48%
EDGAR (raw)						
$\bar{\mathbf{E}}_{\mathbf{D}}$	**0.00%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_{\mathbf{1}}$	**100%**
E_{D_B}	0.00%	d_{Bj}	9387	0	F_{1_B}	100%
E_{D_K}	0.00%	d_{Kj}	0	4604	F_{1_K}	100%
EDGAR (voted)						
$\bar{\mathbf{E}}_{\mathbf{D}}$	**0.00%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_{\mathbf{1}}$	**100%**
E_{D_B}	0.00%	d_{Bj}	9387	0	F_{1_B}	100%
E_{D_K}	0.00%	d_{Kj}	0	4604	F_{1_K}	100%

Table B.16: Full discrimination benchmark for MAG - learning set.

Classical (raw)						
$\bar{\mathbf{E}}_{\mathbf{D}}$	**36.56%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_{\mathbf{1}}$	**63.60%**
E_{D_B}	25.59%	d_{Bj}	1512	520	F_{1_B}	72.38%
E_{D_K}	47.53%	d_{Kj}	634	700	F_{1_K}	54.82%
Classical (voted)						
$\bar{\mathbf{E}}_{\mathbf{D}}$	**34.29%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_{\mathbf{1}}$	**65.99%**
E_{D_B}	21.36%	d_{Bj}	1598	434	F_{1_B}	75.02%
E_{D_K}	47.23%	d_{Kj}	630	704	F_{1_K}	56.96%
EDGAR (raw)						
$\bar{\mathbf{E}}_{\mathbf{D}}$	**0.43%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_{\mathbf{1}}$	**99.57%**
E_{D_B}	0.33%	d_{Bj}	2136	7	F_{1_B}	99.67%
E_{D_K}	0.52%	d_{Kj}	7	1328	F_{1_K}	99.48%
EDGAR (voted)						
$\bar{\mathbf{E}}_{\mathbf{D}}$	**0.00%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_{\mathbf{1}}$	**100%**
E_{D_B}	0.00%	d_{Bj}	2143	0	F_{1_B}	100%
E_{D_K}	0.00%	d_{Kj}	0	1335	F_{1_K}	100%

Table B.17: Full discrimination benchmark for MAG - validation set.

Classical (raw)

$\bar{\mathbf{E}}_\mathbf{D}$	**40.13%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**60.02%**
E_{D_B}	23.38%	d_{Bj}	1245	380	F_{1_B}	72.55%
E_{D_K}	56.88%	d_{Kj}	562	426	F_{1_K}	47.49%

Classical (voted)

$\bar{\mathbf{E}}_\mathbf{D}$	**38.48%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**61.75%**
E_{D_B}	22.40%	d_{Bj}	1261	364	F_{1_B}	73.64%
E_{D_K}	54.55%	d_{Kj}	539	449	F_{1_K}	49.86%

EDGAR (raw)

$\bar{\mathbf{E}}_\mathbf{D}$	**0.73%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**99.33%**
E_{D_B}	0.34%	d_{Bj}	1739	6	F_{1_B}	99.51%
E_{D_K}	1.11%	d_{Kj}	11	978	F_{1_K}	99.14%

EDGAR (voted)

$\bar{\mathbf{E}}_\mathbf{D}$	**0.00%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**100%**
E_{D_B}	0.00%	d_{Bj}	1745	0	F_{1_B}	100%
E_{D_K}	0.00%	d_{Kj}	0	989	F_{1_K}	100%

Table B.18: Full discrimination benchmark for MAG - test set.

Classical (raw)

$\bar{\mathbf{E}}_\mathbf{D}$	**40.63%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**59.31%**
E_{D_B}	37.60%	d_{Bj}	3072	1851	F_{1_B}	63.54%
E_{D_K}	43.66%	d_{Kj}	1674	2160	F_{1_K}	55.07%

Classical (voted)

$\bar{\mathbf{E}}_\mathbf{D}$	**33.68%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**66.12%**
E_{D_B}	33.21%	d_{Bj}	3288	1635	F_{1_B}	69.08%
E_{D_K}	34.14%	d_{Kj}	1309	2525	F_{1_K}	63.17%

EDGAR (raw)

$\bar{\mathbf{E}}_\mathbf{D}$	**0.00%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**100%**
E_{D_B}	0.00%	d_{Bj}	4915	0	F_{1_B}	100%
E_{D_K}	0.00%	d_{Kj}	0	3789	F_{1_K}	100%

EDGAR (voted)

$\bar{\mathbf{E}}_\mathbf{D}$	**0.00%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**100%**
E_{D_B}	0.00%	d_{Bj}	4915	0	F_{1_B}	100%
E_{D_K}	0.00%	d_{Kj}	0	3789	F_{1_K}	100%

Table B.19: Full discrimination benchmark for M2 backplate - learning set.

Classical (raw)

$\bar{\mathbf{E}}_\mathbf{D}$	**39.54%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**60.46%**
E_{D_B}	33.78%	d_{Bj}	443	226	F_{1_B}	66.22%
E_{D_K}	45.29%	d_{Kj}	226	273	F_{1_K}	54.71%

Classical (voted)

$\bar{\mathbf{E}}_\mathbf{D}$	**30.90%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**69.23%**
E_{D_B}	23.92%	d_{Bj}	509	160	F_{1_B}	74.47%
E_{D_K}	37.88%	d_{Kj}	189	310	F_{1_K}	63.98%

EDGAR (raw)

$\bar{\mathbf{E}}_\mathbf{D}$	**1.00%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**99.04%**
E_{D_B}	0.60%	d_{Bj}	665	4	F_{1_B}	99.18%
E_{D_K}	1.40%	d_{Kj}	7	492	F_{1_K}	98.89%

EDGAR (voted)

$\bar{\mathbf{E}}_\mathbf{D}$	**0.00%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**100%**
E_{D_B}	0.00%	d_{Bj}	669	0	F_{1_B}	100%
E_{D_K}	0.00%	d_{Kj}	0	499	F_{1_K}	100%

Table B.20: Full discrimination benchmark for M2 backplate - validation set.

Classical (raw)

$\bar{\mathbf{E}}_\mathbf{D}$	**45.05%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**53.21%**
E_{D_B}	62.31%	d_{Bj}	98	162	F_{1_B}	46.12%
E_{D_K}	27.80%	d_{Kj}	67	174	F_{1_K}	60.31%

Classical (voted)

$\bar{\mathbf{E}}_\mathbf{D}$	**54.74%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**43.79%**
E_{D_B}	69.23%	d_{Bj}	80	180	F_{1_B}	36.61%
E_{D_K}	40.25%	d_{Kj}	97	144	F_{1_K}	50.97%

EDGAR (raw)

$\bar{\mathbf{E}}_\mathbf{D}$	**1.58%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**98.58%**
E_{D_B}	0.38%	d_{Bj}	259	1	F_{1_B}	98.85%
E_{D_K}	2.78%	d_{Kj}	5	175	F_{1_K}	98.31%

EDGAR (voted)

$\bar{\mathbf{E}}_\mathbf{D}$	**0.00%**		d_{iB}	d_{iK}	$\bar{\mathbf{F}}_\mathbf{1}$	**100%**
E_{D_B}	0.00%	d_{Bj}	260	0	F_{1_B}	100%
E_{D_K}	0.00%	d_{Kj}	0	180	F_{1_K}	100%

Table B.21: Full discrimination benchmark for M2 backplate - test set.

Appendix C: Evaluation on the DGUWA dataset

In Section 3.2.3, Chen et al. [48] evaluate their technique on Loeffler's [41] dataset (which they refer to as the "DGUWA" dataset). More specifically, they use the test dataset for which the ground-truth shot count is available, and claim an accuracy of 100% (although erroneously assuming a shot count of 359 instead of 358).

While Chen et al. comment on the dataset containing 68 hours of non-shot accelerometer data acquired during daily life and construction, their reported number of 598 non-gunshot candidates seems more in line with only the non-gunshot data of the test set, for which Loeffler reports 693 non-gunshot candidates. We thus similarly restricted our experiment to these datasets, and used a similar split of 70% of samples being allocated to the learning set and 30% to the test set. The candidates after EDGAR preprocessing are reported in Table C.1. The number of non-shot candidates is over one order of magnitude larger. This is likely due to our candidate selection method being more generic and more suited to usage with a high-g accelerometer. Chen et al. make use of Reese's [44] candidate selection method, which assumes calm aiming period before and after gunshots, and employ multiple computationally-intensive filtering techniques. In addition, we performed no tuning of the candidate selection parameters and used those of the MAG dataset, which were determined for very different input data (different sampling frequency, range, weapons, sensor mounting, etc.). We assume that tuning these parameters would significantly reduce the number of non-shot candidates and thus improve performance.

	Non-shot (X)	Non-shot (Y)	Non-shot (Z)	Shot
Learning	29603	31759	22200	250
Test	12254	14228	9496	108

Table C.1: Number of candidates on the DGUWA test dataset. Shots are detected identically for all 3 axes and are only reported once.

Although the EDGAR technique can be used on multidimensional input, our current practical implementation assumes unidimensional data. We thus processed each acceleration axis independently and report the results in Figure C.1 and Table C.2. The hyperparameters of the MAG network were used without any tuning. Since this dataset was acquired with wrist-mounted sensors and handguns, we observe that best performance is obtained with the Z axis (vertical axis). This differs military applications studied in the rest of this thesis where information is concentrated in the X axis (firing axis).

The best performance on the Z axis is a counting error of 2.8%, or 3 errors on 108 gunshots. However, these errors are all false-positives, detected on daily life and con-

struction data. Perfect accuracy is obtained on the shooting range data. With a similar candidate selection system, the EDGAR technique should also achieve perfect accuracy on this dataset while being able to run in real-time on a weapon-embedded device, whereas MobileNetTime is computed offline.

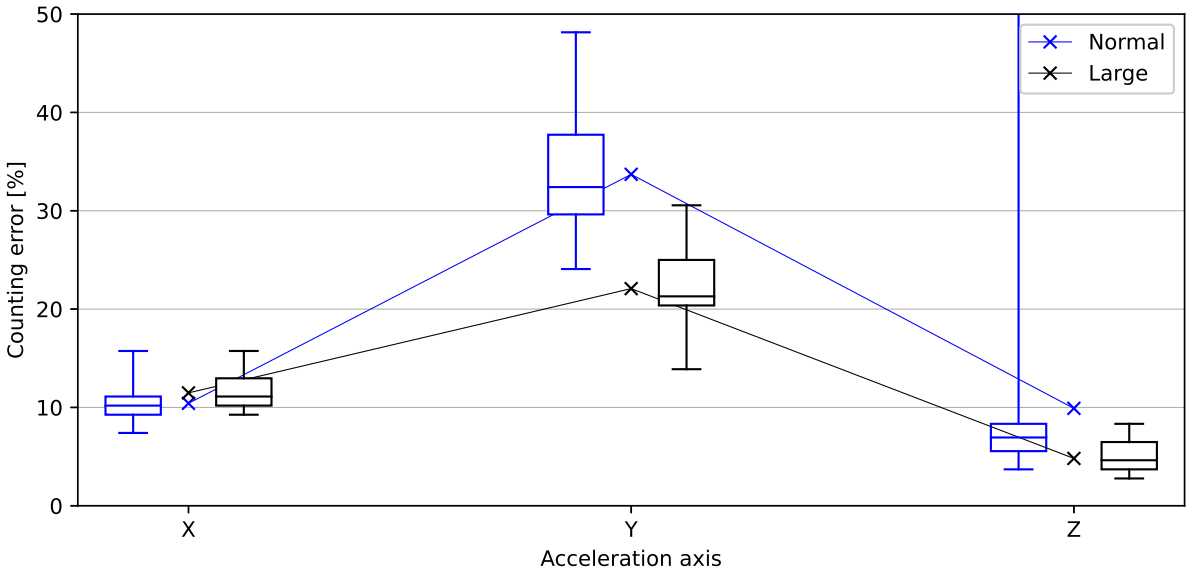


Figure C.1: Evaluating the EDGAR technique on the individual axes of the 30% test split derived from the DGUWA test dataset [41]. "Normal" networks use 18 convolution channels and 64/32 neurons in the dense layers, similar to the deployed embedded EDGAR real-time networks. "Large" networks increase these values by a factor of 10.

(a) EDGAR (Z axis)

		Predicted label	
		Non-shot	Shot
True label	Non-shot	9493	3
	Shot	0	108

(b) MobileNetTime [48]

		Predicted label	
		Non-shot	Shot
True label	Non-shot	183	0
	Shot	0	105

Table C.2: Confusion matrices on the 30% test split derived from the DGUWA test dataset.

Appendix D: Candidate densities

Section 6.5 presented an example distribution of candidate densities for the Minimi 5.56, as shown in Figure 6.8. This appendix provides additional density profiles for all other weapons considered in this study.

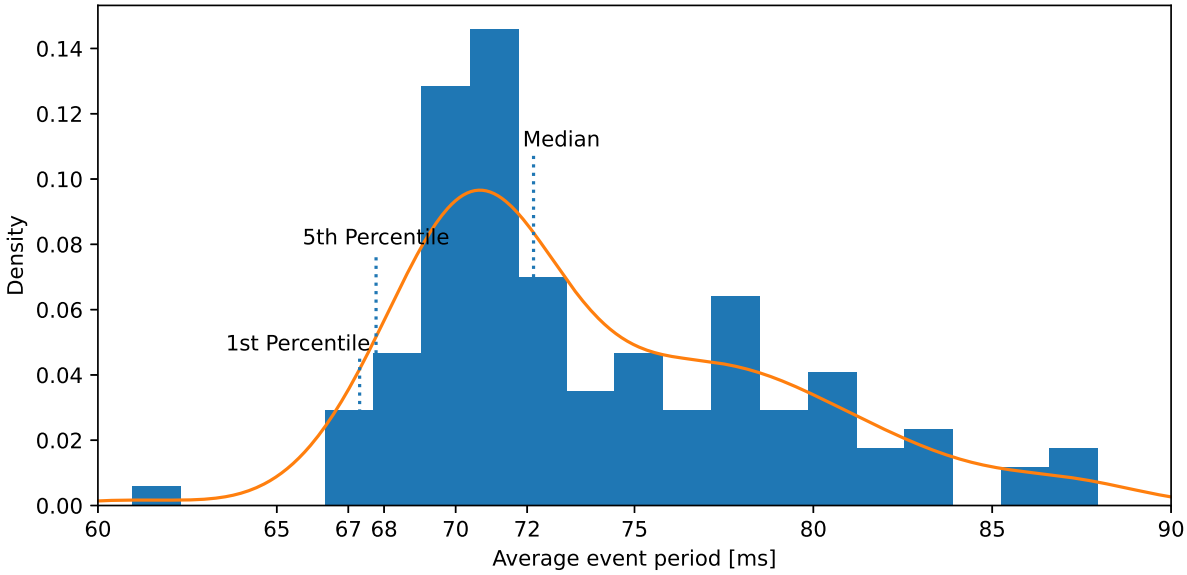


Figure D.1: Filtered candidate density distribution on the learning set for Minimi 5.56.

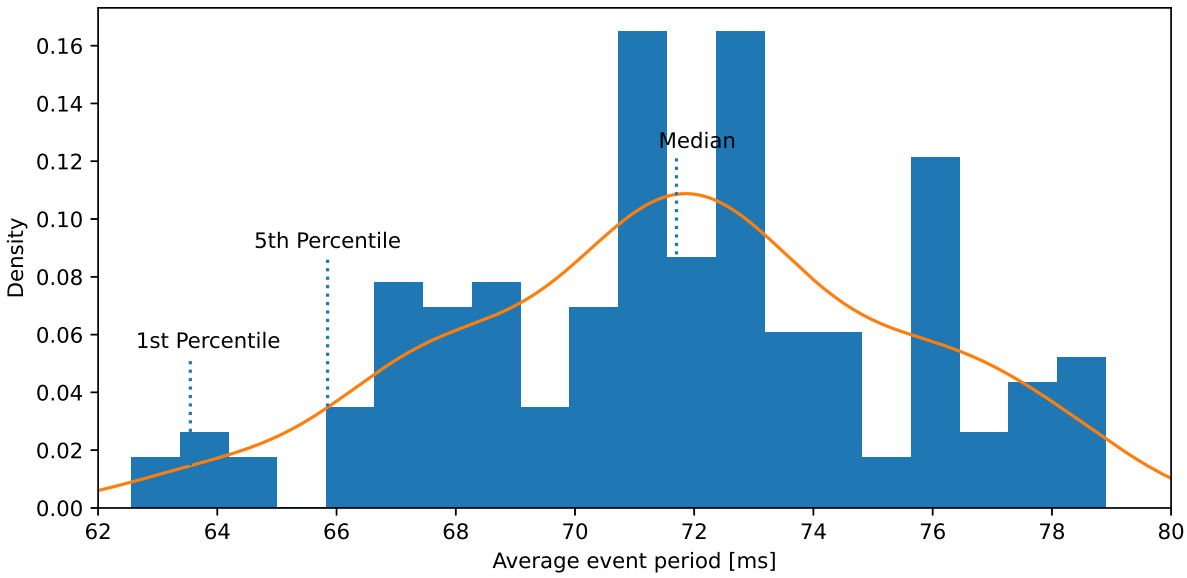


Figure D.2: Filtered candidate density distribution on the learning set for Minimi 7.62.

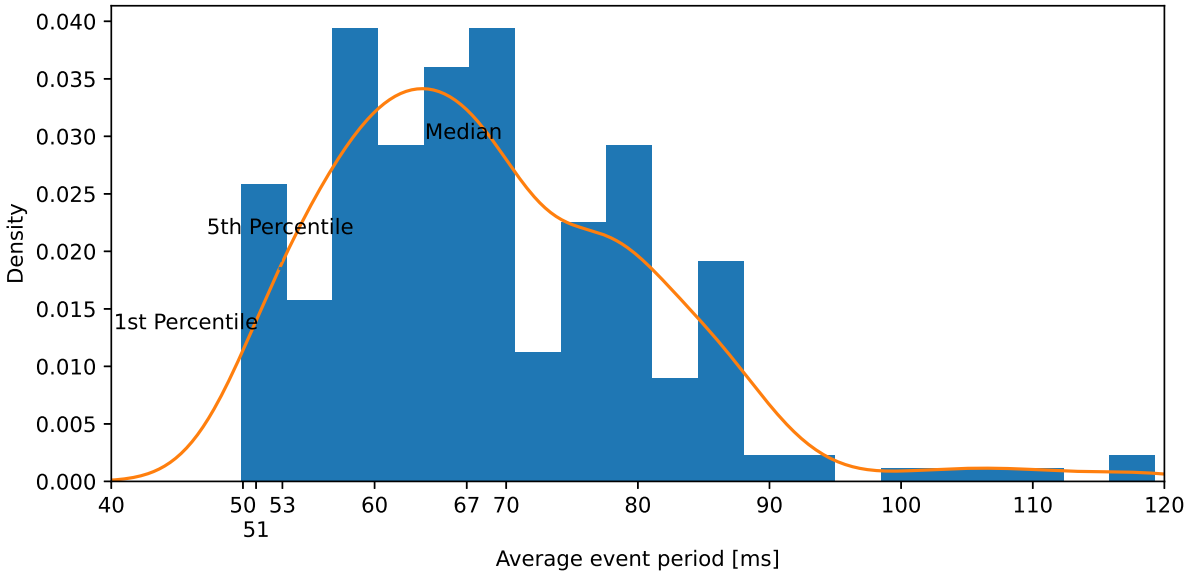


Figure D.3: Filtered candidate density distribution on the learning set for MAG.

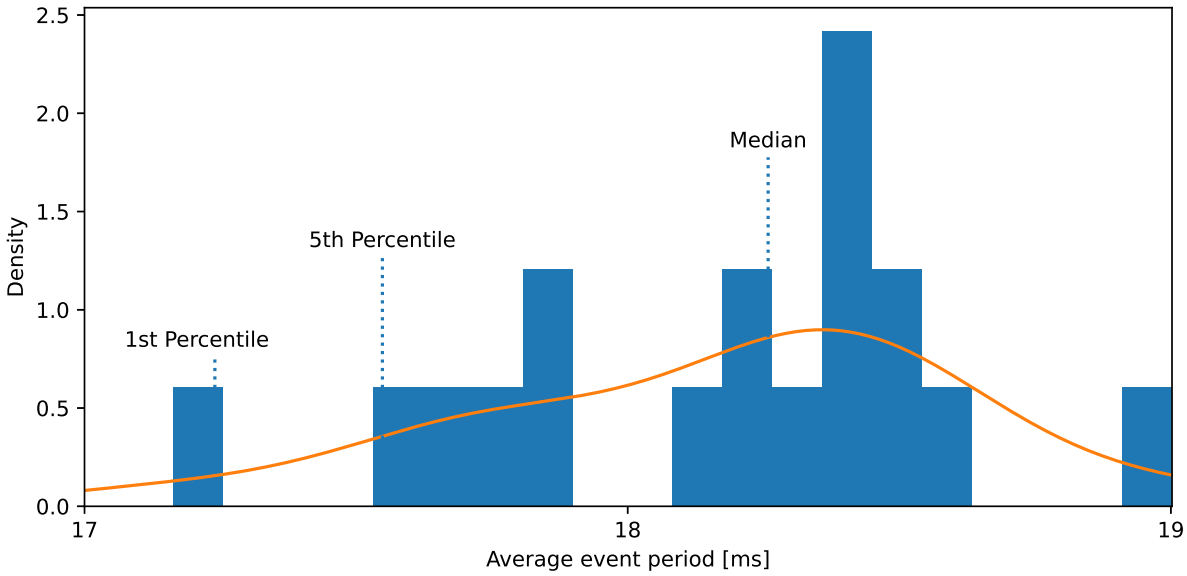


Figure D.4: Candidate density distribution on the learning set for M134.

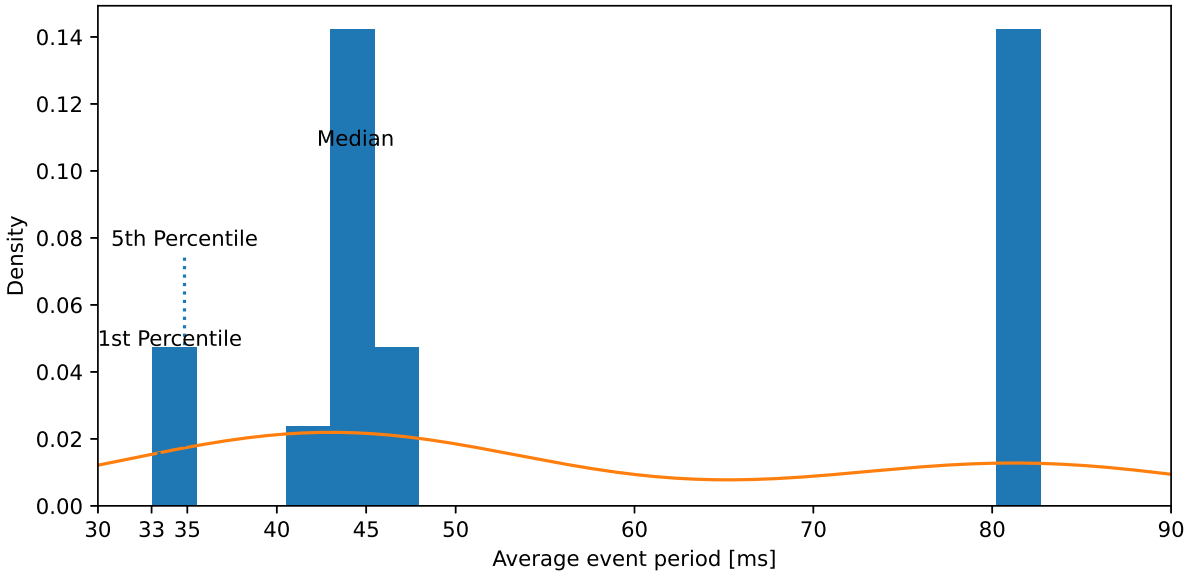


Figure D.5: Filtered candidate density distribution on the learning set for M2 backplate.

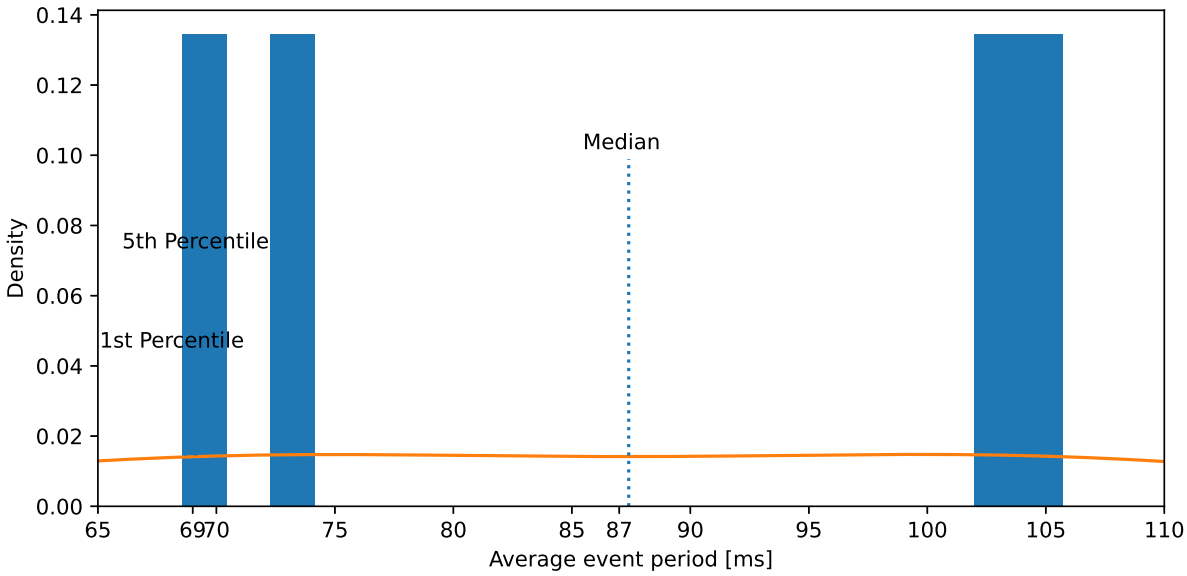


Figure D.6: Filtered candidate density distribution on the learning set for M2 barrel.

Appendix E: Saliency maps

Section 8.4 illustrated a saliency map for the MAG in Figure 8.6. This appendix includes corresponding maps for the remaining weapons.

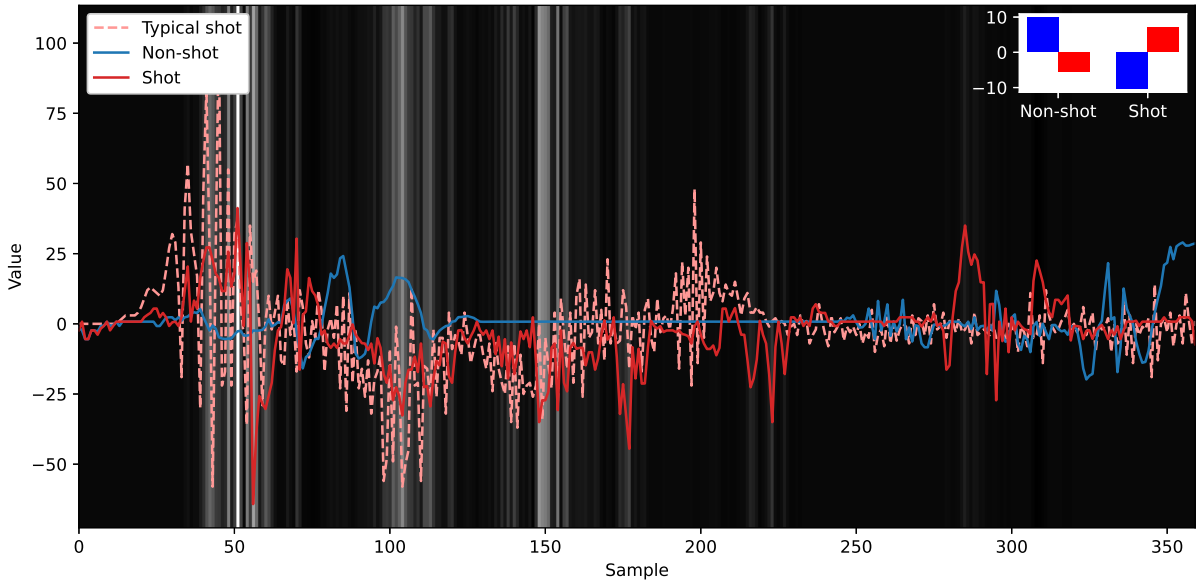


Figure E.1: Saliency map of the counting model for M2 barrel.

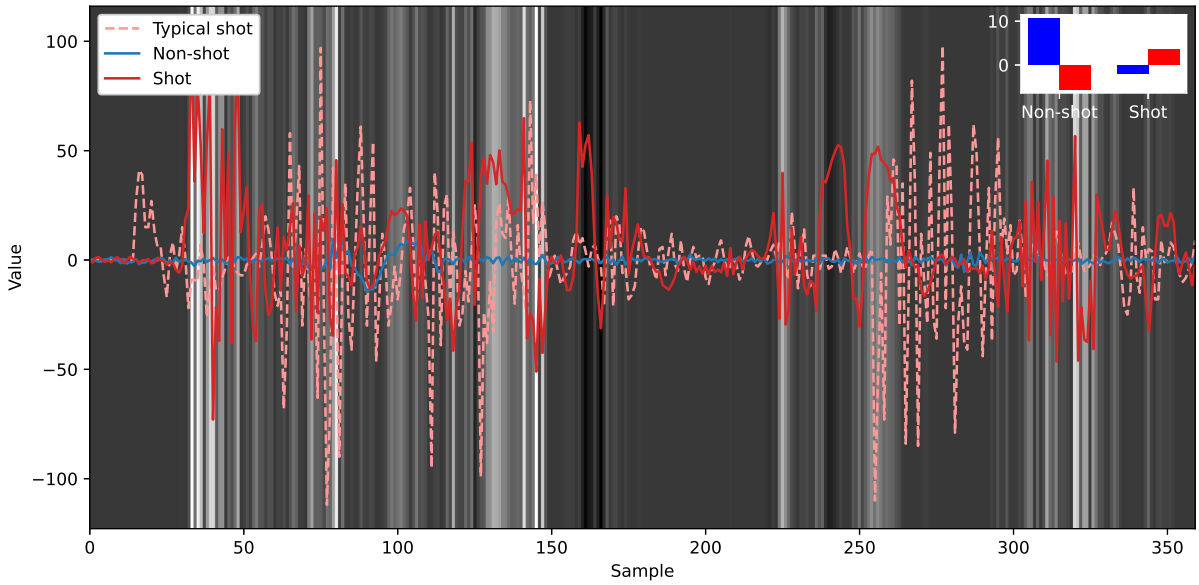


Figure E.2: Saliency map of the counting model for M2 backplate.

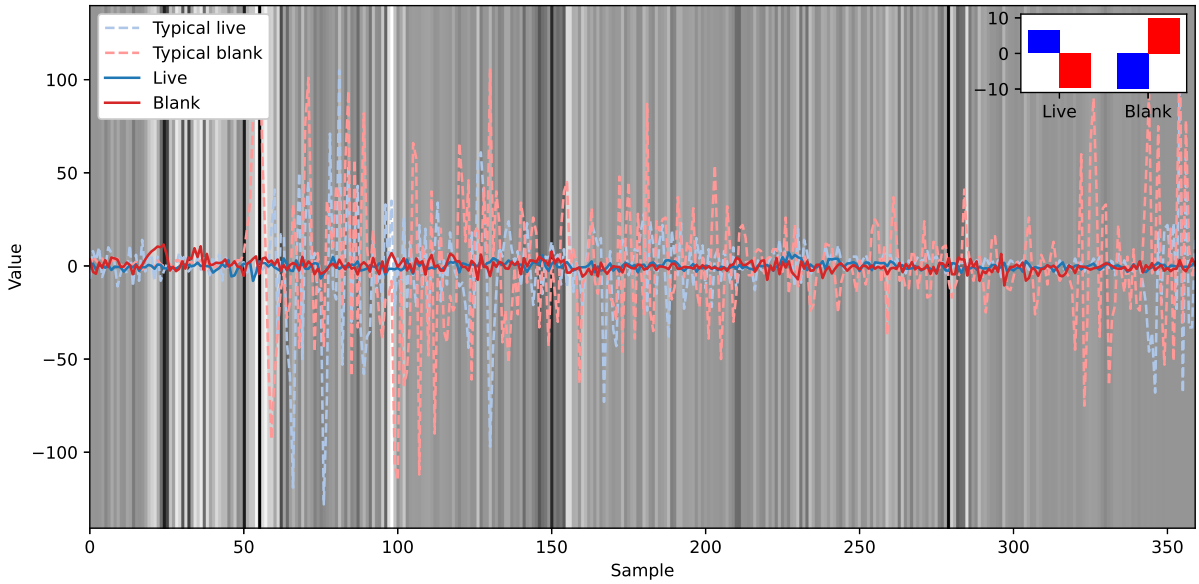


Figure E.3: Saliency map of the discrimination model for M2 backplate.

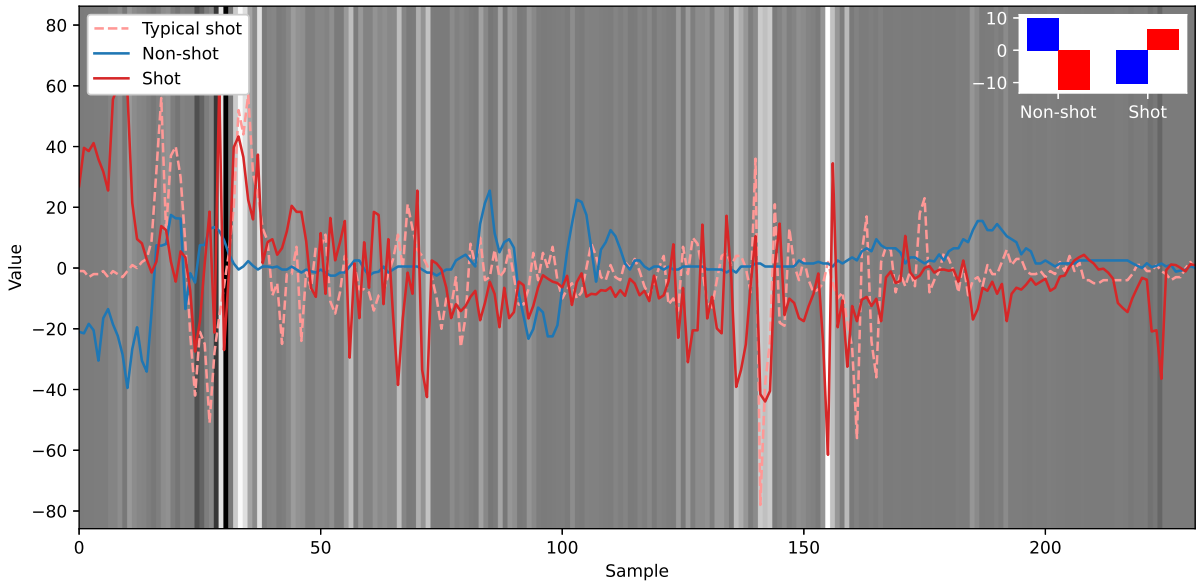


Figure E.4: Saliency map of the counting model for Minimi 5.56.

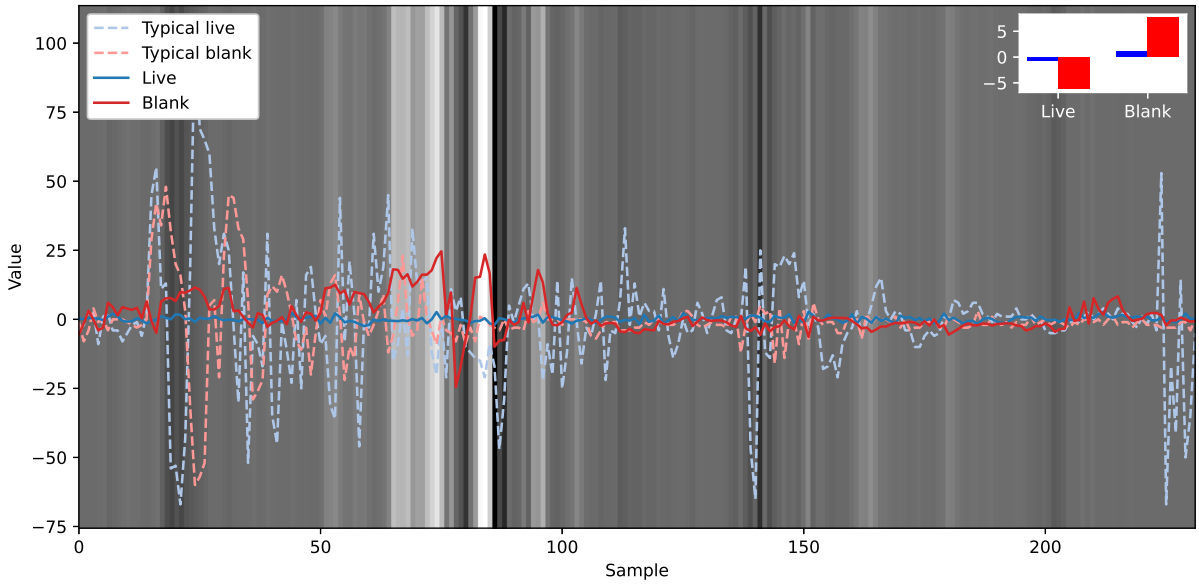


Figure E.5: Saliency map of the discrimination model for Minimi 5.56.

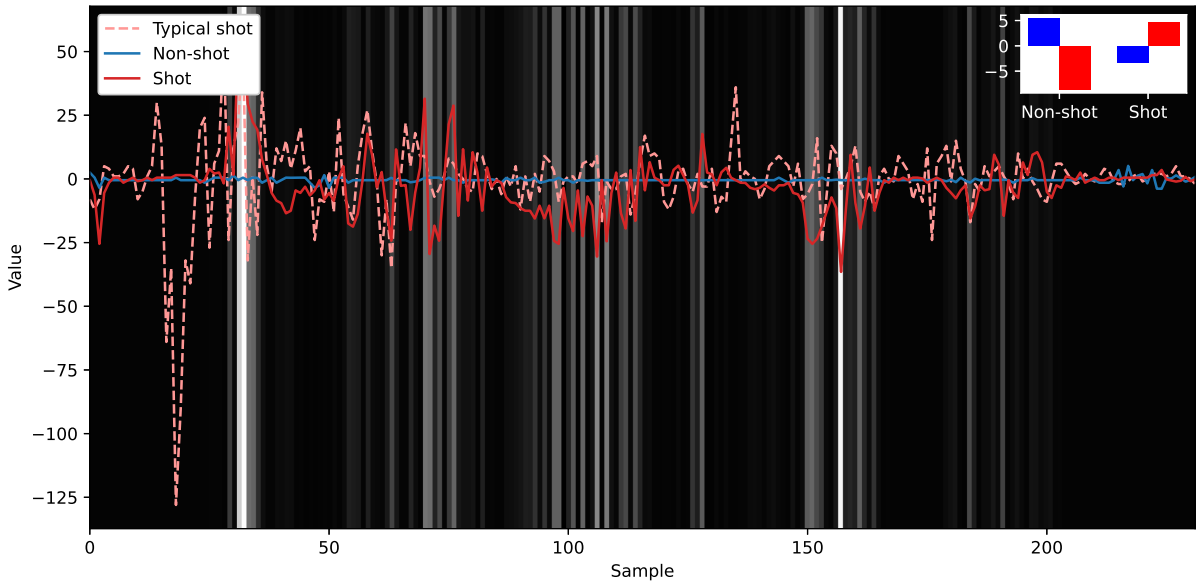


Figure E.6: Saliency map of the counting model for Minimi 7.62.

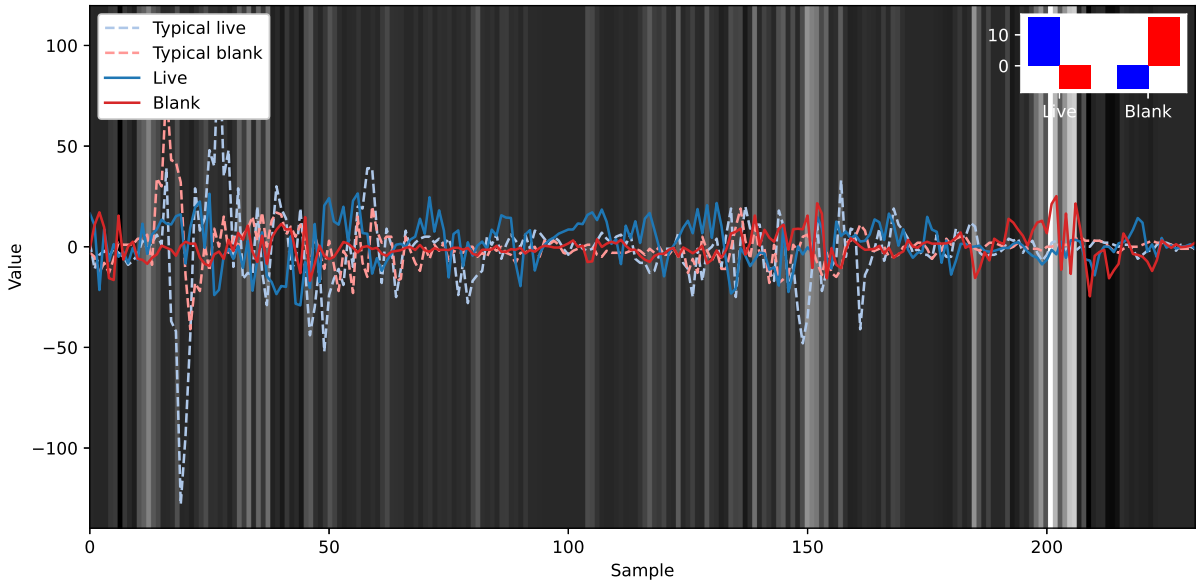


Figure E.7: Saliency map of the discrimination model for Minimi 7.62.

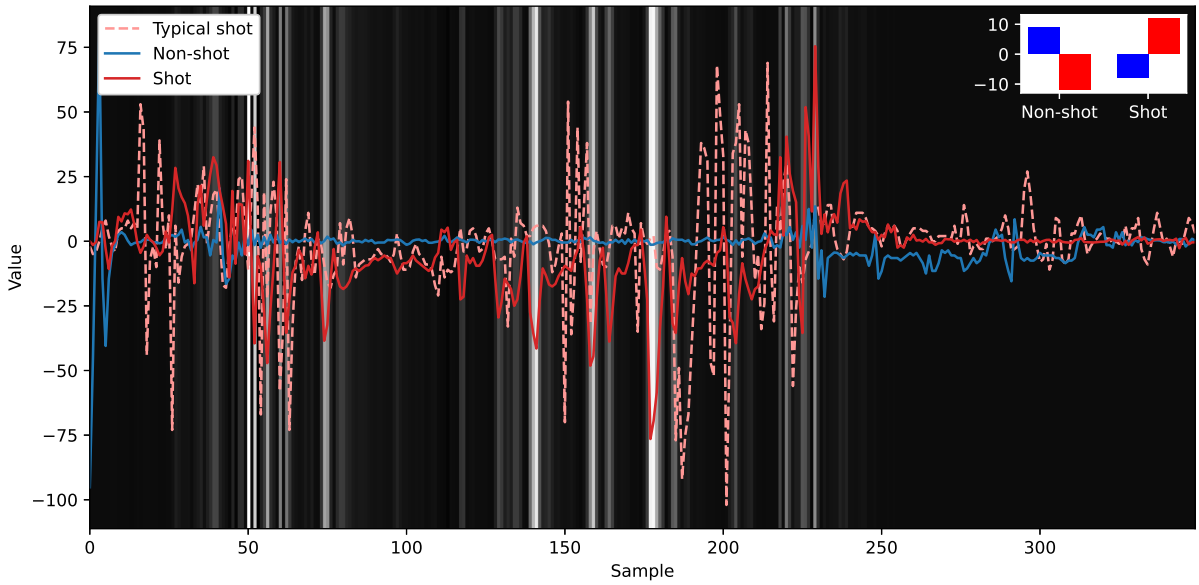


Figure E.8: Saliency map of the counting model for MAG.

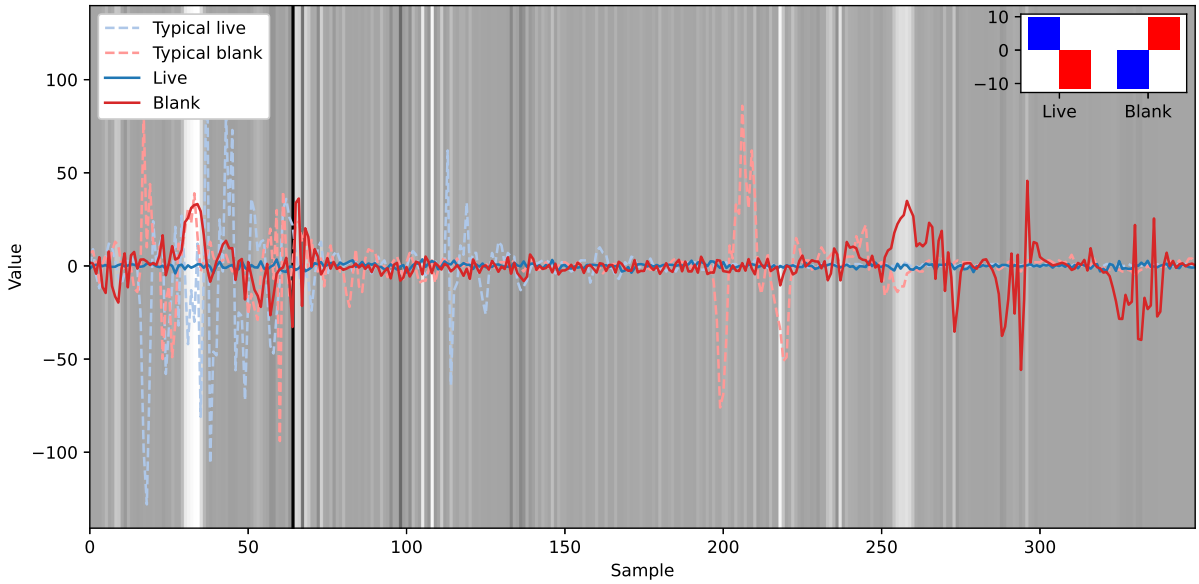


Figure E.9: Saliency map of the discrimination model for MAG.

Bibliography

[1] FN Herstal, S.A., *Small Arms Management: Break free of paper chains with a smart, digital armory*, Sep. 2020. [Online]. Available: https://fnenovation.eu/products/small-arms-management/fn-smart-armory/ (cit. on pp. 9, 12).

[2] FN Herstal, S.A., *FN® Smart Armoury solution - Product information file*, Apr. 4, 2023 (cit. on pp. 10, 16, 19).

[3] *CANIK 2025 Product Catalog*, Mar. 2025. Accessed: Mar. 11, 2025. [Online]. Available: https://www.me-canik.com/pdf/canik_tr.pdf (cit. on p. 13).

[4] Armée de Terre, director, *Auxylium Battlefield, les soldats connectés*, Mar. 29, 2018. Accessed: Jan. 24, 2025. [Online]. Available: https://www.youtube.com/watch?v=gmRygXZuAuY (cit. on p. 20).

[5] P. Colman. "MKIV XP Championship Timer/Chronograph", PACT Shooting Products, Accessed: Nov. 21, 2024. [Online]. Available: https://pact.com/products/shooting-timers/mkiv-xp/ (cit. on p. 21).

[6] K. Gartz, "Apparatus for monitoring the firing stress of a weapon barrel", U.S. Patent 5918304A, Jun. 29, 1999 (cit. on p. 21).

[7] A. Gering, "Device for counting live shots, blank shots and dry shots", U.S. Patent 10866048B2, Dec. 15, 2020 (cit. on pp. 24, 50, 54, 71).

[8] M. Roth, "Heckler & Koch Bolt Motion Sensor System (BMSS)", *Polizeipraxis*, pp. 12–22, Jan. 2024 (cit. on pp. 24, 25).

[9] M. Staiger, M. Scheuermann, J. A. Kopf, D. Gebert, and D. Rimpf, "Firearm analysis devices", U.S. Patent 11802747B2, Oct. 31, 2023 (cit. on pp. 24, 50, 54, 71).

[10] FN Herstal. "FN SmartCore® Shot Counter", FN e-novation, Accessed: Apr. 9, 2025. [Online]. Available: https://fnenovation.eu/products/small-arms-management/fn-smartcore/ (cit. on pp. 25, 26, 144).

[11] Radetec, *RISCpro Installation and User Manual*, Oct. 16, 2024. [Online]. Available: https://www.radetecusa.com/wp-content/uploads/2024/10/RISC-Pro-User-Manual-Version-10162024.pdf (cit. on p. 25).

[12] Senseer. "Weaponlogic by Senseer", Accessed: Dec. 10, 2024. [Online]. Available: https://www.linkedin.com/posts/senseerai_ai-weapon-mortar-activity-7216871310302314496-Ai2m (cit. on pp. 26, 27).

[13] LMT Advanced Technologies, *Benefits of PAL*, Aug. 29, 2022. Accessed: Apr. 9, 2025. [Online]. Available: https://lmt-at.com/wp-content/uploads/2021/07/PAL-spec-sheet.pdf (cit. on pp. 26, 28).

[14] ShotPM. "NO. 1 Active RFID Shot Counter — ShotPM — Phoenix", shotpm, Accessed: Nov. 3, 2021. [Online]. Available: https://www.shotpm.com (cit. on pp. 26, 28, 69).

[15] "Lancer Systems L5 Advanced Warfighter Magazine, 20 Round 5.56mm/.223cal, Translucent Dark Earth", MGW, Accessed: Apr. 18, 2025. [Online]. Available: https://www.midwestgunworks.com/page/mgwi/prod/999-000-2320-14 (cit. on p. 37).

[16] "Mil-Tec US M2A1 50 Cal Ammo Box Steel", Military Kit, Accessed: Apr. 18, 2025. [Online]. Available: https://www.militarykit.com/products/mil-tec-us-m2a1-50-cal-ammo-box-steel (cit. on p. 37).

[17] A. M. Martínez-Heredia and S. Ventura, "Weak Supervision: A Survey on Predictive Maintenance", *WIREs Data Mining and Knowledge Discovery*, vol. 15, no. 2, e70022, 2025, ISSN: 1942-4795. DOI: 10.1002/widm.70022 (cit. on p. 38).

[18] R. Joannes, J.-P. Delcourt, and P. Heins, "Device for detecting and counting shots fired by an automatic or semi-automatic firearm, and firearm equipped with such a device", U.S. Patent 7669356B2, Mar. 2, 2010 (cit. on pp. 38, 42, 43, 50, 52, 55, 57, 65, 69, 71).

[19] N. Morsa, "Automated system and method for a projectile launcher monitoring", European pat. 4269931A1, Nov. 1, 2023 (cit. on pp. 40, 64, 68, 91).

[20] N. Morsa, "EDGAR: Embedded detection of gunshots by AI in real-time", in *Advanced Analytics and Learning on Temporal Data*, T. Guyet, G. Ifrim, S. Malinowski, A. Bagnall, P. Shafer, and V. Lemaire, Eds., Cham: Springer International Publishing, 2023, pp. 148–166, ISBN: 978-3-031-24378-3. DOI: 10.1007/978-3-031-24378-3 (cit. on pp. 40, 64).

[21] N. Morsa, "Extending Electronic Shot Counters With Machine Learning-Based Ammunition Discrimination for Targeted Maintenance", *Journal of Defense Analytics and Logistics*, May 28, 2025, ISSN: 2399-6439. DOI: 10.1108/JDAL-02-2025-0004 (cit. on p. 40).

[22] R. B. Calhoun, S. Lamkin, and D. Rodgers, "Systems and methods involving creation and/or utilization of image mosaic in classification of acoustic events", U.S. Patent 10424048B1, Sep. 24, 2019 (cit. on p. 53).

[23] A. Raza, F. Rustam, B. Mallampati, P. Gali, and I. Ashraf, "Preventing Crimes Through Gunshots Recognition Using Novel Feature Engineering and Meta-Learning Approach", *IEEE Access*, vol. 11, pp. 103 115–103 131, 2023, ISSN: 2169-3536. DOI: 10.1109/ACCESS.2023.3316695 (cit. on p. 53).

[24] S. B. Nesar, B. M. Whitaker, and R. C. Maher, "Machine Learning Analysis on Gunshot Recognition", in *2024 Intermountain Engineering, Technology and Computing (IETC)*, Logan, UT, USA: IEEE, May 13, 2024, pp. 249–254, ISBN: 979-8-3503-7307-3. DOI: 10.1109/IETC61393.2024.10564263 (cit. on p. 53).

[25] Z. Chen et al., "A Wireless Gunshot Recognition System Based on Tri-Axis Accelerometer and Lightweight Deep Learning", *IEEE Internet of Things Journal*, vol. 10, no. 19, pp. 17 450–17 464, Oct. 2023, ISSN: 2327-4662. DOI: 10.1109/JIOT.2023.3273859 (cit. on pp. 53, 64, 71).

[26] K. F. Packer, A. D. Wilks, P. J. Schubert, and T. E. Long, "Shot-counting device for a firearm", U.S. Patent 8046946B2, Nov. 1, 2011 (cit. on p. 54).

[27] D. Ferrarini, L. Fioretti, M. Coccoli, and F. Caravaggi, "Device for counting shots for firearms", U.S. Patent 8418388B2, Apr. 16, 2013 (cit. on p. 54).

[28] R. D. Acarreta, "Device for controlling ammunition consumption in real time", European pat. 2151658B1, Jan. 15, 2014 (cit. on p. 54).

[29] R. D. Acarreta, "Counting device", U.S. Patent 9303937B2, Apr. 5, 2016 (cit. on p. 54).

[30] A. Gering, "Dispositif de comptage de tirs reels, a blanc et a sec", European pat. 3268689A1, Jan. 17, 2018 (cit. on p. 54).

[31] I. Lamparter, F. Scheuermann, and M. Schumacher, "Batterieloser schusszähler", European pat. 3140605A1, Mar. 15, 2017 (cit. on p. 54).

[32] C. D. Johnson and H. E. Weidner, "Round counter", U.S. Patent 4001961A, Jan. 11, 1977 (cit. on p. 55).

[33] K. L. Brinkley, "Firearm monitoring device", U.S. Patent 5566486A, Oct. 22, 1996 (cit. on p. 55).

[34] E. Johnson and J. Kulesza, "Device for collecting statistical data for maintenance of small-arms", U.S. Patent 20050155420A1, Jul. 21, 2005 (cit. on pp. 55, 56, 59).

[35] FN Herstal SA, "Dispositif pour la detection et le comptage des coups tires par une arme automatique ou semi-automatique et arme equipee d'un tel dispositif", pat. BE1017549A3, Dec. 2, 2008 (cit. on p. 55).

[36] R. B. I. Clark, D. Gessel, and P. A. Leitner-Wise, "Processes and Systems for Monitoring Usage of Projectile Weapons", U.S. Patent 20090277065A1, Nov. 12, 2009 (cit. on p. 57).

[37] G. Glock, "Handgun", U.S. Patent 8387295B2, Mar. 5, 2013 (cit. on p. 57).

[38] A. Bar-David and Y. Spector, "System and method for automated gun shot measuring", U.S. Patent 8571815B2, Oct. 29, 2013 (cit. on p. 57).

[39] "Pres. Business court of Brussels, 18 December 2020", *Ingénieur-Conseil*, vol. 2020/4, pp. 769–804, Dec. 18, 2020 (cit. on p. 57).

[40] R. Ufer and K. L. Brinkley, "Self calibrating weapon shot counter", U.S. Patent 8826575B2, Sep. 9, 2014 (cit. on pp. 57, 63, 68).

[41] C. E. Loeffler, "Detecting gunshots using wearable accelerometers", *PLOS ONE*, vol. 9, no. 9, pp. 1–6, Sep. 3, 2014, ISSN: 1932-6203. DOI: 10.1371/journal.pone.0106664 (cit. on pp. 60, 61, 63, 64, 71, 93, 223, 224).

[42] I. P. E. S. Putra, J. Brusey, E. Gaura, and R. Vesilo, "An Event-Triggered Machine Learning Approach for Accelerometer-Based Fall Detection", *Sensors*, vol. 18, p. 20, Dec. 22, 2017. DOI: 10.3390/s18010020 (cit. on p. 61).

[43] G. L. Santos, P. T. Endo, K. H. de Carvalho Monteiro, E. da Silva Rocha, I. Silva, and T. Lynn, "Accelerometer-Based Human Fall Detection Using Convolutional Neural Networks", *Sensors*, vol. 19, p. 1644, Apr. 6, 2019. DOI: 10.3390/s19071644 (cit. on pp. 61, 98).

[44] K. A. Reese, "A situational-awareness system for networked infantry including an accelerometer-based shot-identification algorithm for direct-fire weapons", M.S. thesis, Naval Postgraduate School, Sep. 2016, 136 pp. (cit. on pp. 61, 64, 65, 71, 223).

[45] D. Welsh and N. Roy, "Smartphone-based mobile gunshot detection", in *2017 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, Mar. 2017, pp. 244–249. DOI: 10.1109/PERCOMW.2017.7917566 (cit. on pp. 62, 63).

[46] M. A. A. H. Khan, D. Welsh, and N. Roy, "Firearm Detection Using Wrist Worn Tri-Axis Accelerometer Signals", in *2018 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, Mar. 2018, pp. 221–226. DOI: 10.1109/PERCOMW.2018.8480345 (cit. on pp. 62, 64, 65, 71).

[47] D. Welsh, A. Z. M. Faridee, and N. Roy, "Hybrid Distance-Based Framework for Classification of Embedded Firearm Recoil Data", in *2021 IEEE International Conference on Pervasive Computing and Communications Workshops and Other Affiliated Events (PerCom Workshops)*, Kassel, Germany: IEEE, Mar. 22, 2021, pp. 50–55, ISBN: 978-1-6654-0424-2. DOI: 10.1109/PerComWorkshops51409.2021.9431020 (cit. on pp. 63, 68, 71).

[48] Z. Chen, H. Zheng, L. Wu, J. Huang, and Y. Yang, "Deep Transfer Learning Based Intelligent Gunshot Detection and Firearm Recognition Using Tri-Axial Acceleration", *IEEE Internet of Things Journal*, pp. 1–1, 2024, ISSN: 2327-4662. DOI: 10.1109/JIOT.2024.3489963 (cit. on pp. 64, 71, 223, 224).

[49] W. Boettcher, X. Jia, D. Evans, B. Mayall, and S. Hill, "High precision shot detection system", pat. WO2024113022A1, Jun. 6, 2024 (cit. on p. 65).

[50] U. Amit, Z. Shneorson, H. Sender, and D. Haitin, "An apparatus and method for counting rounds fired from a firearm", European pat. 2905570B1, Jun. 13, 2018 (cit. on pp. 66, 69).

[51] I. Weiss and I. Hyden, "Device system and method for projectile launcher operation monitoring", U.S. Patent 11566861B2, Jan. 31, 2023 (cit. on pp. 67, 70, 71).

[52] J. Asbach and M. Canty, "Weapon usage monitoring system having discharge event monitoring", U.S. Patent 12007185B1, Jun. 11, 2024 (cit. on pp. 68, 71, 139).

[53] M. Chedid, M. Forselius, S. Dahlqvist, A. Isaksson, and I. Emricson, "Method and device for detecting a fire shot event in a weapon", European pat. 2026030A1, Feb. 18, 2009 (cit. on pp. 68–70).

[54] APB360. "ShotPM FAQ – APB360.com", Accessed: Nov. 3, 2021. [Online]. Available: https://apb360.com/weapon-shot-counter/shotpm-faq/16799/ (cit. on p. 69).

[55] J. Deu-Ngoc and D. Bujak, "Method and apparatus for automatic counting of projectiles shot from a projectile launcher", pat. WO2020186343A1, Sep. 24, 2020 (cit. on p. 69).

[56] D. Reed, D. Dodge, and T. Thomas, "Weapon simulation systems", U.S. Patent 11644278B2, May 9, 2023 (cit. on pp. 70, 71).

[57] M. Filipovic and R. Sostar, "Device, system and method of detecting and counting shots", pat. WO2023088554A1, May 25, 2023 (cit. on p. 70).

[58] K. T. Kloepfer, B. W. Dwyer, J. H. Thiesen, and C. J. Owens, "Systems and techniques for identifying gun events", U.S. Patent 11920881B1, Mar. 5, 2024 (cit. on p. 70).

[59] *nRF52840 Product Specification*, manual, Rev. 1.7, Nordic Semiconductor, Sep. 2021. Accessed: Apr. 26, 2022. [Online]. Available: https://infocenter.nordicsemi.com/pdf/nRF52840_PS_v1.7.pdf (cit. on p. 72).

[60] *Micropower, 3-Axis, ±200 g Digital Output, MEMS*, manual, Rev. B, Analog Devices, 2018. Accessed: Apr. 26, 2022. [Online]. Available: https://www.analog.com/media/en/technical-documentation/data-sheets/adxl372.pdf (cit. on pp. 72, 74, 77).

[61] *150-mA, Ultra-Low Quiescent Current, 1-µA I_Q Low-Dropout Linear Regulator*, manual, SBVS138, Texas Instruments, Dec. 2009. Accessed: Apr. 26, 2022. [Online]. Available: https://www.ti.com/lit/ds/symlink/tps782-q1.pdf (cit. on p. 72).

[62] *AB08X5 Real-Time Clock Family*, manual, Rev. C, Abracon, Oct. 16, 2014. Accessed: Apr. 26, 2022. [Online]. Available: https://abracon.com/Precisiontiming/AB08X5-RTC.PDF (cit. on p. 72).

[63] K.-H. Tsai and H.-T. Lin, "Learning from label proportions with consistency regularization", *CoRR*, vol. abs/1910.13188, Oct. 29, 2019. DOI: 10.48550/arXiv.1910.13188 (cit. on pp. 95, 105, 183).

[64] J. Ma, *A higher-level neural network library on microcontrollers (NNoM)*, version v0.4.2, Zenodo, Oct. 2020. DOI: 10.5281/zenodo.4158710 (cit. on p. 104).

[65] T. Miyato, S.-I. Maeda, M. Koyama, and S. Ishii, "Virtual Adversarial Training: A Regularization Method for Supervised and Semi-Supervised Learning", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 8, pp. 1979–1993, Aug. 2019, ISSN: 1939-3539. DOI: 10.1109/TPAMI.2018.2858821 (cit. on p. 105).

[66] S. Laine and T. Aila, "Temporal Ensembling for Semi-Supervised Learning", Mar. 15, 2017. DOI: 10.48550/arXiv.1610.02242 (cit. on p. 105).

[67] J. Hestness et al., "Deep Learning Scaling is Predictable, Empirically", Dec. 1, 2017. DOI: 10.48550/arXiv.1712.00409, pre-published (cit. on p. 136).

[68] J. Kaplan et al., "Scaling Laws for Neural Language Models", Jan. 22, 2020. DOI: 10.48550/arXiv.2001.08361, pre-published (cit. on p. 136).

[69] T. Henighan et al., "Scaling Laws for Autoregressive Generative Modeling", Nov. 5, 2020. DOI: 10.48550/arXiv.2010.14701, pre-published (cit. on p. 136).

[70] J. S. Rosenfeld, A. Rosenfeld, Y. Belinkov, and N. Shavit, "A Constructive Prediction of the Generalization Error Across Scales", Dec. 20, 2019. DOI: 10.48550/arXiv.1909.12673, pre-published (cit. on p. 138).

[71] L. Lai, N. Suda, and V. Chandra, "CMSIS-NN: Efficient Neural Network Kernels for Arm Cortex-M CPUs", Jan. 19, 2018. DOI: 10.48550/arXiv.1801.06601 (cit. on p. 144).

[72] N. S. Madiraju, S. M. Sadat, D. Fisher, and H. Karimabadi, "Deep Temporal Clustering : Fully Unsupervised Learning of Time-Domain Features", Feb. 3, 2018. DOI: 10.48550/arXiv.1802.01059 (cit. on pp. 147, 189).

[73] S. T. Nimi, A. Arefeen, Y. S. Uddin, and Y. Lee, "EARLIN: Early out-of-distribution detection for resource-efficient collaborative inference", in *Machine Learning and Knowledge Discovery in Databases. Research Track*, N. Oliver, F. Pérez-Cruz, S. Kramer, J. Read, and J. A. Lozano, Eds., Cham: Springer International Publishing, 2021, pp. 635–651, ISBN: 978-3-030-86486-6. DOI: 10.1007/978-3-030-86486-6_39 (cit. on p. 162).