TREX Operations Guidelines

Generic

Version 2.1

Date 28 March 2022

Table of Contents

Introduction

Threat Ranking and Extraction (TREX) is a software tool used for CCTV video analytics to detect moving objects that are outstanding in their environment.

TREX has the capability to learn the scene to focus on true targets and not on environmental factors that are common to the scene.

With the correct configuration, TREX is able to detect small (and very small) targets, which can translate to targets at great distance, for example, with a Flir-fb12-o thermal camera, TREX can detect a person walking at about 700m away among trees/bushes. Likewise, TREX can fulfil multi kilometre wide area surveillance tasks on the correct camera specification or narrow urban detection task with classification where speed of target acquisition is crucial.

TREX is designed to be deployed on stationary cameras. However, with the right set up it is capable of operating in cases where the camera experiences some slight movement such as camera vibration due to traffic, wind, or other factors.

Product Offering

TREX has been developed with a range of applications in mind. TREX is very efficient yet produces excellent results on a wide range of hardware.

It can run on small Arm-based CPU's found on a single board PC (such as Arm A7 (32bit)) to dual socket, server class Intel Xeon CPUs.

TREX is also developed with modern architecture/hardware optimisation so it can use modern GPUs for hardware acceleration when available. Due to its flexibility, there are three broad categories of TREX are on offer to suit different operational outcomes and environments, namely: **TREX Elite, TREX Premium, TREX Standard** and **TREX Lite**. The table beneath shows the main differences between them.

Feature	TREX Elite	TREX Premium	TREX Standard	CORE	Notes
Short range	Yes	Yes	Yes	Yes	Below 50m
Medium range	Yes	Yes	Yes	Yes	Up to ~100 m camera dependent
Long range	Yes	Yes	Yes ~	No	Up to ~200 m camera dependent
Very Long range	Yes	Yes	No	No	Up to 500 – 2000m+, camera and lens dependent

Feature	TREX Elite	TREX Premium	TREX Standard	CORE	Notes
Thermal classification	Yes	Yes	No	No	Only possible when objects are big or close enough, likely in the short range
Thermal noise reduction	Yes	Yes	No	No	Improve sensitivity and accuracy of thermal cameras
Suitable for slow CPU	No	No	Limited	Yes	Able to run on slow or low power devices.
Suitable for Patrolling PTZ cameras*	Yes	No	No	No	Able to avoid nuisance alerting on patrolling PTZ cameras.
Auto Tracking with PTZ slave	Yes	Yes	Yes	No	Only supported with compatible devices on Milestone
GPU acceleration preferred	Yes	Yes	Yes	Optional	GPU may be required for deep learning
Standard Object Classification	Yes	Yes	Yes	Yes	Only possible when objects are big or close enough
Loitering	Yes	Yes	Yes	Yes	Short range only
Left Object	Yes	Yes	Yes	Yes	Short to medium range
Directional Alert	Yes	Yes	Yes	Yes	Only on targets entering from a particular direction (e.g. outside in)
Very high sensitivity	Yes	Yes	No	No	Should be used for long range only or to detect very discrete targets in non-cluttered environments
Object age filtering by region	Yes	Yes	No	No	Minimum object age per region
Object size filtering by region	Yes	Yes	No	No	Minimum object size per region

*PTZ cameras must remain stationary for at least 30 seconds at each preset position.

TREX ELITE

TREX Elite, is the highest tier in the TREX line-up, and is focussed on very long-range detection, or detection of very discreet targets using patrolling PTZ cameras.

This version of TREX Is specifically designed for use with PTZ cameras, with an automated patrol pattern, for wide-area (360 degree) surveillance tasks (eg. open pit mining) or critical infrastructure safeguarding (eg. pipe lines or power lines).

TREX PREMIUM

TREX Premium, the second highest tier in the TREX line-up, is focussed on very long-range detection, or detection of very discreet targets.

Paired with appropriate cameras, (typically thermal) with high quality imagery, multi-kilometre detection ranges can be achieved.

This version of the TREX software is typically used for wide-area surveillance tasks (eg. open pit mining) or critical infrastructure safeguarding (eg. pipe lines or power lines).

TREX STANDARD

TREX Standard, is perhaps the most commonly used and practical of the TREX line-up. This version of the software is significantly more hardware efficient than TREX Premium, but still performs extremely well in long range applications if paired with an appropriate camera.

Typically, this version of the software is used for perimeter protection use cases up to 200m. Fast target acquisition (typically less than 500m) is a key feature of this software tier, and enables very efficient monitoring of narrow areas of intrusion, as typically found in urban and complex environments.

Camera Setup and Configuration

Correct configuration of the TREX iSentry system for the surveillance task required is key to operational outcomes. The system addresses the following use cases across its tiers and must be configured accordingly.

- Wide area surveillance
- Long range perimeter protection
- Traditional perimeter protection
- Narrow areas of detection where speed of detection and classification are crucial with the target only in view for a short amount of time.
- Object tracking and long term loiter functions

Recognising potential false positives and expected targets

TREX and iSentry should be set up to fulfil the desired surveillance task and has comprehensive configuration options.

Knowledge of the desired target profile will allow optimal setup and alerting performance. For example, when targeting human indoor activity at relatively close range, TREX can be configured to ignore small targets moving in the scene, targets that have been in the scene only for a certain length of time, the direction those targets are moving in etc.

In addition to knowing the expected targets, knowing the environment is equally important. For example, when operating in outdoor environments, there are certain types of objects that can cause unwanted alerts such as trees, running water fountain, flashing lights, fast moving water surfaces (pond/pool/beach) etc. If possible, detections from these areas should be excluded by using masks.

TREX has the following configuration options that will be elaborated below:

- Processing masks
- Sensitivity settings
- Sensitivity settings by region
- Directional alerting
- Object size in scene
- Object age in scene

Where camera resolution, object size and distance are within the performance envelope, the iSentry deep learning and rules engine can be utilised as a further layer of contextualisation to refine the initial configuration and get more precise and refined alerting performance. Options available include:

- Escalation or dismissal on multiple object classes (people, cars, animals, bicycles, motorbikes trucks, helmets, faces etc.)
- Escalation and dismissal on numbers of object classes i.e., groups of people versus a single person
- Combinations of object classes i.e., people and vehicle
- Integration with Milestone rules engine to access milestone rule functionalities, including mapping and camera auto slew to interrogate long distance targets by zone

TREX iSentry has a comprehensive embedded settings tool within the Milestone system as well as a standalone, web-based settings option.

TREX Settings

Mask Settings

In most of the cases, setting up a processing mask is a great way of achieving the desired outcome for TREX.

Whenever possible, it is recommended to set up masks for each time zone. However, in cases where it is not practical to set up a mask because doing so will render the tool unusable, like the example shown in Figure 1, TREX still can function well, by using correct sensitivity levels.

Sensitivity Levels

Low

This sensitivity level is designed to only detect targets with a high likelihood of being true targets. This level is designed to be used in an environment where false positives are intolerable. The LOW sensitivity level is best suited to such scenes as containing environmental interference, for example "tree branches waving in the wind" or "waves or white-caps at the beach".

Figure 1 A complex scene where false positive is costly and setting mask is not practical

Figure 1 shows an example of a scene where LOW could be used when false positive targets are unacceptable. While not recommended, LOW sensitivity can be used where setting up an area mask is not possible, as shown in Figure 1. If a mask were to be set up, it virtually has to cover most of the image (rendering the tool unusable) because tree branches are known to cause false positives, so they should be blocked out. Even with this sensitivity setting, false positives are still possible under some special circumstances.

Medium/Low

As the name suggests, this sensitivity level aims to fill the gap between the MEDIUM and LOW sensitivity levels. A higher priority is still given towards a low false positive detection rate, although more targets will be detected as compared to the low sensitivity setting. Like the case of LOW sensitivity, setting up a detection mask is highly recommended, when possible, although most environmental movement will still be effectively ignored.

Medium

This is the default sensitivity level where TREX aims at keeping the balance between false positive and misdetection for majority of the cases. For an unknown scene with little or no detail on the expected targets, it is recommended to use MEDIUM sensitivity before further fine tuning. The following figure shows a relatively straight forward scene where this sensitivity level would be an ideal starting point.

Figure 2 A relative straight forward scene where MEDIUM would work well

As shown in Figure 2, the scene doesn't appear to contain common false positive source (trees, water etc.). The camera viewing distance is limited, so the targets are of reasonable size, i.e.. small targets are not expected, so higher sensitivity is probably not required to start with.

Medium/High

As the name suggests, this mode is designed to bridge the gap between MEDIUM and HIGH sensitivity settings. This level should be used in structured environment where there are no obvious sources of false positives (or where they can be blocked out with a mask). The example shown in Figure 2 is also suitable for this sensitivity level.

High

This sensitivity level is aimed at longer distance use cases where missed detections will cause significant security risks. Although significant computing

effort is spent within the TREX system to distinguish between high value targets and noise (such as camera noise, compression artefacts or moving trees, clouds etc.), some false positives are expected when using this sensitivity. It is highly recommended for users to set up masks (one per time zone) to ignore irrelevant areas that might cause false positives.

Figure 3 A rather complicated & busy scene. A small sailboat at distance can be detected at HIGH sensitivity

For example, in Figure 3, if the aim is to detect the boat moving in and out of the area, a HIGH sensitivity level is likely the correct choice, since the boat is quite small (and slow moving). All uninteresting areas like the tree branches should be masked out.

Very High

This sensitivity level is generally not recommended, (and only available in some versions of TREX) unless missed targets are of great consequence or where very small, and not obvious targets are expected in the scene. Appropriate scenes for this sensitivity level should have minimal sources of false positive (environmental "noise"), or where the use of an appropriate mask will result in a "sterile" scene. This sensitivity level also requires a greater amount of computing power as well as memory (especially important when using hardware acceleration) resources.

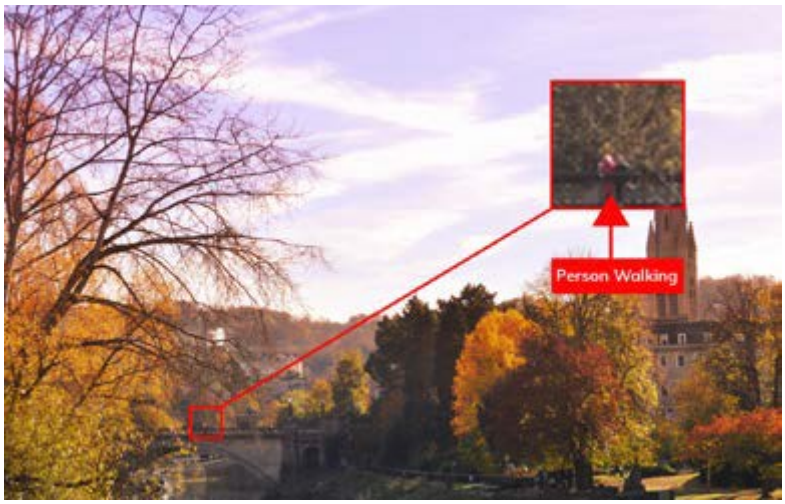

Figure 4 Example of a VERY HIGH sensitivity detection

This sensitivity, in general, is intended for long distance surveillance with the expectation of some false positives. The following example shows a challenging example because of the trees and structure and camera vibration, trees and structures are masked in the settings. As a result, TREX is capable of detecting small targets in difficult condition from about 2km away.

Sensitivity by Regions

This mechanism allows users to manually specify sensitivities for 64 equal regions in the camera view. Sensitivity regions are presented as 8x8 rectangular blocks, such as (red blocks) shown in Figure 5.

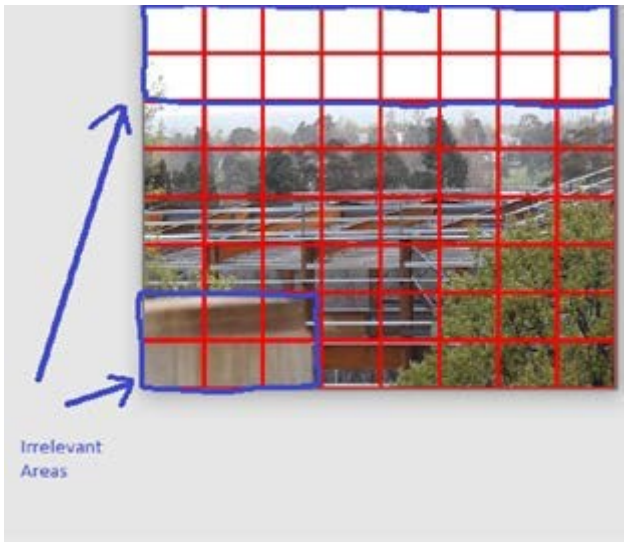

Figure 5 Example of Area sensitivity settings

This mechanism enables users to adapt sensitivity to match expected level of activity in different parts of the camera view. Areas where target detection is important, or targets are further away, a higher sensitivity level can be applied, and equally areas where environmental factors may cause false positive alerts, lower sensitivity levels can be used, allowing for optimal configuration to suite the particular scene. Sensitivity by region can be used on its own or in

conjunction with a processing mask. It is highly advisable to use both mechanisms, where sensitivity by region will apply general sensitivity to different areas and a processing mask is used to specifically ignore known sources of false positive alerts. Additionally, sensitivity by region can also helpful to reduce the required computing need by disabling areas that are completely irrelevant to the intended application, such as part of the sky or office roof or irrelevant areas caused by obstructions in front of the camera, an example is shown in Figure 5.

Directional Alerting

Directional alerting acts as a lens within TREX, focusing on or prioritizing targets based on their general direction and velocity. Typically, critical infrastructure such as a power generation facility or control room would be marked as the "area of interest". Any target moving towards an area of interest, will generate an alert before the area is ever entered. In this way any object simply passing by this area will be ignored, resulting is reduced false positive alerting.

This module requires a separate processing mask to mark any "areas of interest" within the camera view.

This mask is used in conjunction with the main TREX processing mask and sensitivity regions.

If it is desirable to focus on only incoming targets, while ignoring objects/targets leaving or already inside the facility, then the "area of interest" mask must cover the entire area, not just the perimeter, and the inclusion/ exclusion flag must be set.

It is particularly important to note that this module uses the estimated future position of each current target based on its current trajectory, with the assumption that the target stays its course. As it is an estimate, there will be a certain degree of error. The longer into the future the prediction (i.e. number of seconds used in forward prediction), the more error-prone the prediction will be, so when this mechanism is configured, only reasonably short prediction settings should be used.

Object size

Object size is another way to fine tune the TREX alerting profile. By setting minimum and maximum pixel sizes of objects to be detected, a lot of uninteresting alerts can be avoided. For example, where it is known that only human sized objects need to be detected at a certain fixed distance away from the camera, a valid size range for human sized targets can be specified. In doing so, larger objects like vehicles, and smaller objects like pets can be very effectively ignored.

As with all settings, care should be taken to not unintentionally exclude objects of interest, particularly when taking different target distances from the camera into account.

Small vs Large/Obvious Targets

Normally, TREX will track both small discrete targets as well as large and obvious ones. However, either of these object classes can be ignored if desired. The most typical scenario where it may be helpful to turn off obvious/large targets detection is for TREX Premium long-range applications.

As this application intended for small targets, obvious target detections can be switched off to reduce false alerts (especially in busy/cluttered environments). Similarly, in cases such as close-range monitoring, small targets can be ignored to reduce false alerts caused by discrete objects like insects or small birds.

Object Age

Object age can also very effectively increase the quality of alerts, when only objects remaining in the field of view for more than a certain threshold are alerted on. For example, the object age can be used to filter out false alerts caused by random nuisance environment factors such as waiving tree branches/grass or camera noise, which typically appear in short bursts.

Raising the object age requirement may also increase the delay in raising an alert. Therefore, care should be taken when using this feature. Typically, this feature should be used in combination with others to achieve the desired outcome.

Object Age and Size by Region Filtering

With the features, as described earlier, global object/target size and age can be applied to detected objects to further improve the quality of detection. However, there are cases where setting these global values for the entire image is impossible because of the large differences in distance from the target in different areas of the view. For example, in Figure 6, a person in the bottom part of the image (green box) would be close to the camera and a completely different size as compared to the person further away at the top of the image (red box). To solve this problem, object size by region feature of TREX Premium is available where different size filters can be applied to different areas of the view.

Figure 6 Due to different distances from the camera, the same target would appear hugely different in different areas of the view

Similarly, objects further away will appear to move much slower than those closer to the camera even though they may move at the same speed in reality. Object age by region feature of TREX Premium is available for fine tuning enabling the best possible outcome.

Other Features

DeFence

This feature is for high security areas where any enter/exit/crossing activities, including high speed objects throwing, jumping, running, needed to be looked at by security controller. This module can run independently of the regular TREX module. This module requires a separate processing mask, which is especially important when the area of interest is small, such as a small fence line.

Some targets may move at high velocity in the camera view, such as objects being thrown, therefore camera frame rate is particularly important. As a rule of thumb, the higher the speed of the object, the higher camera frame rate that is required. The requirement varies depend on type of objects being targeted.

TREX Loitering mode

This mode is configured to specifically target close range objects and prioritises consistent tracking over detection range. This feature is useful for areas where pass through is expected but loitering or stopping behaviour is explicitly banned. When, for instance, only persons loitering in a scene for more than 30 seconds is of interest to the operator, then all persons and other objects that quickly enter and leave the scene, will be ignored.

TREX Debris Detection

TREX is also able to detect when an object is introduced to/removed from the scene. With the correct configuration and camera, very discreet objects can be

identified, such as debris left behind on an airport runway. The same mechanism, however, can be used to alert on stationary objects such as suitcases, laptops, vehicles, persons etc..

GPU Acceleration

TREX is able to make use of GPU hardware acceleration when available. This can provide a significant performance increase, which then allows a processing server to process many cameras simultaneously. However, when using GPU, depending on input resolution, TREX requires a certain amount of dedicated GPU memory. The higher the sensitivity and input resolution, the more memory and CPU is required.

Therefore, if the GPU is unable to support all cameras at once, high resolution and/or any cameras that is set to TREX on HIGH or VERY/HIGH sensitivity should use CPU for processing (assuming enough system memory installed).

This way dedicated GPU memory is available for many other lower resolution/ sensitivity cameras. Please refers to the benchmark testing section for sample system specifications.

It is advisable, however, to stay with one mode of computation in deployment where possible.

Alert Enrichment and Qualification

As with all the analytics in the iSentry ecosystem, every generated TREX alert is subjected to a qualification process, whereby known alarm conditions can be automatically escalated, and equally, known false positives ignored. This alert qualification process is achieved through data enrichment by multiple deep learning networks, as well as through logic testing with the iSentry rules engine.

Deep Learning

Several image frames, around the time and duration of the alert, is sent to the iSentry deep learning server for enrichment. Several deep learning networks exist and can be applied depending on the environment and outcomes required. During this process, any objects recognised in the image data (eg. persons, pets, bicycles, vehicles, fire/smoke, helmets etc.) will be added to the alert metadata to help contextualise each alert.

The iSentry deep learning server is typically Nvidia GPU based to achieve full functionality and speed, but several versions exist with support for Intel Open Vino, and CPU mode, supporting both X86 and ARM architectures.

Rules Engine

The iSentry rules engine can act as an operator by making automated decisions on the importance of alerts, based on the type, time and specific region of the alert source, as well as the object type and number causing the alert. Several mechanisms are applied within this system to ensure a high-quality outcome, while maintaining complete flexibility and ease of use. By combining logic for several rules, extraordinarily complex rules can be created, allowing the solution of complex problems.

Rules can all be configured for specific areas in the view as well as for specific time zones. The ability to apply rules where their outcome is assured, is key in mitigating for risk when alerts are escalated or dismissed, while allowing for the application of many simultaneous rules in a single scene. Similarly, time zones allow the application of different analytics for different times of day, and light conditions, ensuring maximum alert quality.

Hardware Benchmark Testing

Overview

This document compares the number of TREX cameras (streams) different servers/PCs can process. All reported numbers are recorded manually and are prone to human error. As such, reported performance shown in this document should only be used as a guide rather than an absolute ground truth.

In all of the reported experiments/tests, the CPU load is limited to approximately 50% in total, reported by Windows task manager. This cap leaves enough CPU for Milestone processes and the occasional deep learning process when events are detected.

All test were carried out using the Milestone version of iSentry, which is windows based. Apart from the required video decoding component and TREX, no other iSentry components are activated. (i.e. all Milestone recording, database components etc. are on another PC).

As per default architecture of the recent Milestone release, deep learning processing is done on the same server running the load test. That removes the need for a separate powerful server dedicated for performing deep learning processing. However, it also adds significant load to the server during alerting time.

Even though TREX is generally a CPU intensive component (apart from video decoding), it is expected that if additional iSentry components are activated, the total number of cameras that can be processed will reduce. The details of those are out of scope for this document.

All the reported numbers are under simulated conditions which are aimed to be as close as possible to real life operational conditions. However, there might be some variations that may affect the outcome depending on the actual camera view/scene or the camera settings (bit rate/profile/compression settings etc.). It

is our view that some reserved buffer, in terms of processing capability, should be set aside for unexpected events.

All videos used for the experiments are playback video via network streaming simulator to represent real working conditions as close as possible. The detailed settings of these h264 videos are as followed:

Resolution	Frame Rate	Bitrate (Mbit/s)
1920 x 1080	12	4
1270 x 720	12	2
640 x 480	10-12	1-1.5

iSentry supports a range of Nvidia GPUs from GTX and RTX series card. To enable GPU support, a compatible Nvidia driver is required to be installed on host system, both on Windows and Linux operating systems. The following table shows some example of consumer level GPUs being support. The equivalent Enterprise level (Quadro series) GPUs are supported.

GPU Model	Memory (CUDA Cores)	Power Usage	Equivalent GPU	Notes
GTX 1030	2GB (384)	34W		Deep learning should not be used, No external power required (power via PCIe)
GTX 1050ti	4GB (768)	75W		Discontinued. No external power required (power via PCIe)
GTX 1650	4GB (896)	75W		No external power required (power via PCIe)
GTX 1070	8GB (1,920)	150W		Discontinued
GTX 1660	6GB (1,408)	120W		
GTX 1660ti	6GB (1,536)	120W		
RTX 2060	6GB (1,920)	160W		
RTX 2060 Super	8GB (2,176)	175W		
RTX 2070 Super	8GB (2,560)	215W	RTX 4000	
RTX 2080 Super	8GB (3,072)	250W	RTX 5000	
RTX 2080ti	11GB (4,352)	500W	RTX 6000	
RTX 3070	8GB (5,888)	200W		
RTX 3080	10GB (8,704)	320W		
RTX 3090	24GB (10,496)	350W		

Performance on specific server hardware

All the reported numbers are based on Trex GPU mode (where applicable). Using CPU only (such as on virtual machines) for Trex will reduce the number of

cameras that can be processed significantly. Therefore, it is advisable to use GPU mode where possible.

Intel i7-9700K + NVIDIA 1660 GPU - "Medium Server"

Model : Full size ATX standard build PC (~2018)

CPU : Intel i7-9700k (8 core/8 threads): Base Clock 3.60 GHz

Ref. CPU Score :

Memory : 32 GB

GPU : Nvidia GTX 1660 (6GB memory, 1048 CUDA cores)

Performance : Numbers of cameras

Resolution	PREMIUM	STANDARD	LITE
1920 x 1080	9	12	18
1280 x 720	12	18	26
640 x 512 visual	13	25	35
640 x 512 Thermal	17	30	40

AMD 3900x + NVIDIA RTX2070 - "Large Server"

Model : Full size ATX standard build PC (~2019)

CPU : AMD 3900X (12 core/24 threads): Base Clock 3.80 GHz

Memory : 32 GB

GPU : NVIDIA RTX 2070 (8GB memory, 1048 CUDA cores)

Performance : Numbers of cameras

Resolution	PREMIUM	STANDARD	LITE
1920 x 1080	30	32	40
1280 x 720	35	40	50
640 x 512 - Visual	38	55	70
640 x 512 - Thermal	42	60	75

Intel i5-6500 + NVIDIA 1050ti (4GB) - "Small Server"

Model : Dell Optiplex 7040 – Small Form Factor (~2015)

CPU : Intel i5-6500 (4 core/4 threads) – 3.2GHz

Memory : 8 GB (4+4)

GPU : NVIDIA GTX 1050ti (4GB memory, 768 CUDA cores)

Performance : Numbers of cameras

Resolution	PREMIUM	STANDARD	LITE
1920 x 1080	3	5	7
1280 x 720	4	6	8
640 x 512 Visual	4	6	15
640 x 512 Thermal	4	6	15

-END OF DOCUMENT -