# Threat Image Projection (TIP) into X-ray images of cargo containers for training humans and machines

Thomas W. Rogers
*Dept. Security & Crime Sciences, and Dept. Computer Science*
*University College London*

Nicolas Jaccard, and Emmanouil D. Protonotarios
*Dept. Computer Science*
*University College London*

James Ollier, and Edward J. Morton
*Rapiscan Systems Ltd.*
*Stoke-on-Trent, UK*

Lewis D. Griffin
*Dept. Computer Science*
*University College London*
*L.Griffin@cs.ucl.ac.uk*

***Abstract*—We propose a framework for Threat Image Projection (TIP) in cargo transmission X-ray imagery. The method exploits the approximately multiplicative nature of X-ray imagery to extract a library of threat items. These items can then be projected into real cargo. We show using experimental data that there is no significant qualitative or quantitative difference between real threat images and TIP images. We also describe methods for adding realistic variation to TIP images in order to robustify Machine Learning (ML) based algorithms trained on TIP. These variations are derived from cargo X-ray image formation, and include: (i) translations; (ii) magnification; (iii) rotations; (iv) noise; (v) illumination; (vi) volume and density; and (vii) obscuration. These methods are particularly relevant for representation learning, since it allows the system to learn features that are invariant to these variations. The framework also allows efficient addition of new or emerging threats to a detection system, which is important if time is critical.**

**We have applied the framework to training ML-based cargo algorithms for (i) detection of loads (empty verification), (ii) detection of concealed cars (ii) detection of Small Metallic Threats (SMTs). TIP also enables algorithm testing under controlled conditions, allowing one to gain a deeper understanding of performance. Whilst we have focused on robustifying ML-based threat detectors, our TIP method can also be used to train and robustify human threat detectors as is done in cabin baggage screening.**

## I. Introduction

A major challenge for obtaining high human performance at visual screening tasks, such as detecting Small Metallic Threats (SMTs) in X-ray baggage scans, is the rarity of real threats. Studies have shown that humans perform much better in terms of detection and false alarm rates if threat items have high prevalence [1]. This prompted research into Threat Image Projection (TIP) techniques, mostly in Cabin Baggage Screening (CBS), whereby threat items are realistically projected into baggage imagery to increase threat prevalence during live screening operations. TIP is also used in Computer Based Training (CBT) [2, 3], and for evaluating operator performance and vigilance [4].

Most TIP methods insert Fictional Threat Items (FTI) from a threat database into the image [5]. Researchers have focused on determining realistic placement locations (voids) in baggage and generating threat noise and artefacts that are consistent with the rest of the baggage [6–8], so as to reduce visual cues for operators. To our knowledge there have been no academic publications on TIP methods for cargo. Authors have commented on possible cues caused by superposition-based TIP methods for single-view X-ray baggage [8]. We follow a similar superposition approach, but demonstrate, experimentally, that it does not lead to any obvious visual cues.

Researchers also face a similar threat prevalence issue when training Machine Learning (ML) based Automated Threat Detection (ATD) algorithms. There is often a large imbalance between the *innocuous* and *threat* classes. This often leads to learning algorithms that are biased towards the *innocuous* class and therefore detection performance on the *threat* class suffers. This observation is similar, and possibly analogous, to the one found in humans. Class imbalance can also affect performance evaluation, particularly accuracy measures, in what is known as the "accuracy paradox" [9].

To remedy the class imbalance problem, researchers often consider: (i) dataset re-sampling [10, 11]; (ii) reformulating the problem as one-class [12]; (iii) adjusting the algorithm cost function [13]; or (iv) generating or collecting more data. Recently, with the development of end-to-end ML methods such as Convolutional Neural Networks (CNNs), which require very large amounts of training data, dataset augmentation [14, 15] has become increasingly the focus of attention. In dataset augmentation, class-preserving transformations are made to existing training data to expose the ML algorithm to natural variation, which reduces overfitting and improves generalisation to unseen examples. Such transformations often include rotations, translations, reflections, and changes in illumination and noise.

In cargo screening, we are faced with a major class imbalance problem since threats are extremely rare in the wild. It is also expensive and time consuming to collect large numbers of realistic *staged* threat examples. To this end, we have developed a TIP framework for cargo. The framework allows generation of realistic synthetic threat images and the injection of realistic variation derived from the characteristics of X-ray cargo image formation. These variations include: (i) translations; (ii) rotations; (iii) pixel noise; (iv) magnification; (v) illumination; (vi) volume and density; and (vii) obscuration. Whilst TIP is beneficial in training ML-based algorithms, it is also useful for gaining a deeper understanding of algorithm performance by controlling particular aspects in testing. We evaluate the threat extraction and projection

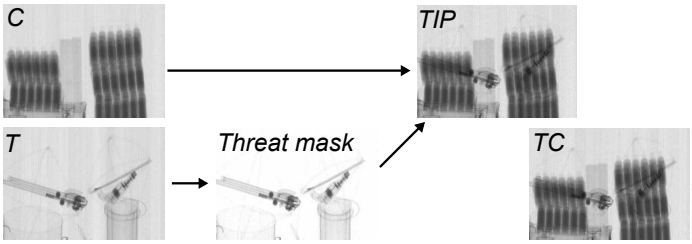


Fig. 1. Illustration of TIP used in validation. Three images were captured of: (i) cargo only ($C$); (ii) threat (and support structures) only ($T$); and (iii) threat and cargo ($TC$). The background was removed from $T$ by dividing by a background estimate leaving the threat attenuation mask. The threat was then projected onto $C$ by multiplication. For evaluation the TIP image can be compared to the treat threat image $TC$. Note that the threat support structures have been included as threat in this experiment, but would be removed in practice.

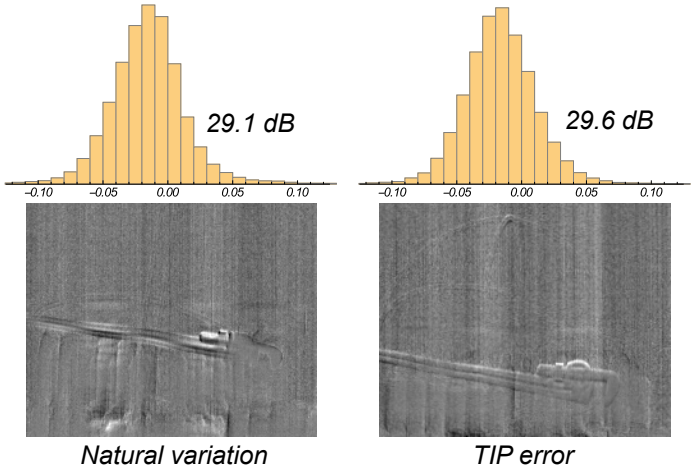


Fig. 3. Comparison of natural system variation (left) with TIP pixel errors (right). Natural variation was computed as the deviation between repeat scans of identical cargo and threat ($TC$). TIP error was computed as the deviation between the TIP image and a $TC$ image. Images show these deviations, rescaled, so that they are visible. Histograms show distribution of deviations. For each case, the Peak Signal-to-Noise Ratio (PSNR) is given in decibels (dB). TIP does not lead to large errors relative to natural image variation and does not change the distribution of deviations.

method on experimental data, and give examples of use cases in training ML-based detectors for cargo imagery.

## II. Threat Item Extraction and Projection

We assume that X-ray image formation obeys the Beer-Lambert rule so that the pixel value $I_{xy}$ at image location $\{x,y\}$ is given by

$$I_{xy} = I_0 \exp\left(-\int \mu_{xy}(z)\,dz\right), \tag{1}$$

where $I_0$ is the beam intensity, $x$ are horizontal image coordinates, $y$ are vertical image coordinates, $z$ are depth coordinates, and $\mu$ is the affective attenuation coefficient of the objects composing the scene.

The pixel value can be split into contributions from the threat $T$ and its background $B$

$$\begin{aligned} I_{xy} &= I_0 \exp\left(-\int_T \mu_{xy}(z)\,dz\right) \exp\left(-\int_B \mu_{xy}(z)\,dz\right), \\ &= I_0 T_{xy} B_{xy}, \end{aligned} \tag{2}$$

Therefore by estimating $I_0 B_{xy}$, one can estimate the *threat mask* $T_{xy} \in [0,1]$. The threat mask can then be projected into X-ray images by multiplication.

Unlike TIP for baggage Computed Tomography (CT), one does not have to compute plausible threat locations, except in the case that the threat occupies a very large container volume. Threat extraction and projection is shown in Fig. 1. In this case the background is approximated by averaging across columns in a small image patch directly above the threat. This simple approach is possible due to the uniform appearance of the container in the image verticals. In more complicated cases, the threat and other structures can be manually delineated before background division.

It is important that TIP imagery is realistic, in particular the TIP process should not generate any cues that may be learnt by a ML algorithm, especially if testing is performed on TIP imagery. To this end, the TIP method was validated experimentally, using a Rapiscan®Eagle M60 operating in interlaced dual-energy mode using Bremsstrahlung X-rays with 4 MeV and 6 MeV cut-offs for low and high energy, respectively. We scanned containers, containing: (i) *threat only* ($T$); (ii) *threat and other cargo* ($TC$); and (iii) *other cargo only* ($C$). Industrial tools (pipe wrench, electric drill, pipe bender) were used as threat models and plastic cylinders and hula-hoops were used to support the threats in place. For the purposes of this experiment, we have included the support structures as part of the threat.

Threats were extracted from the $T$ images and projected onto $C$ images to create TIP image. Visual comparison of TIP images and $TC$ images (Fig. 2) shows that TIP is realistic; one would not be able to distinguish which image is real and which is TIP without being told. Furthermore, the TIP error can be quantified by measuring the deviation between the TIP image and the $TC$ image and compared to natural system variation. We estimate natural variation by taking the deviation of repeat $TC$ scans. In both cases we can also study the distribution of deviations in a histogram and compute the Peak Signal-to-Noise Ratio (PSNR). This is shown in Fig. 3. We observe that TIP does not give rise to large errors relative to natural image variation (TIP error is less than natural variation in this case), and that the errors, in distribution and spatial arrangement, are very similar to natural variation. In addition, there are no obvious visual cues generated from the TIP process.

## III. Injection of Realistic Variation

We can inject variation into the threat appearance, using transformations that preserve the class of the threat. These transformations can be derived by considering the nature of X-ray image formation. Here we discuss several different types of transformations applicable to X-ray cargo imagery.

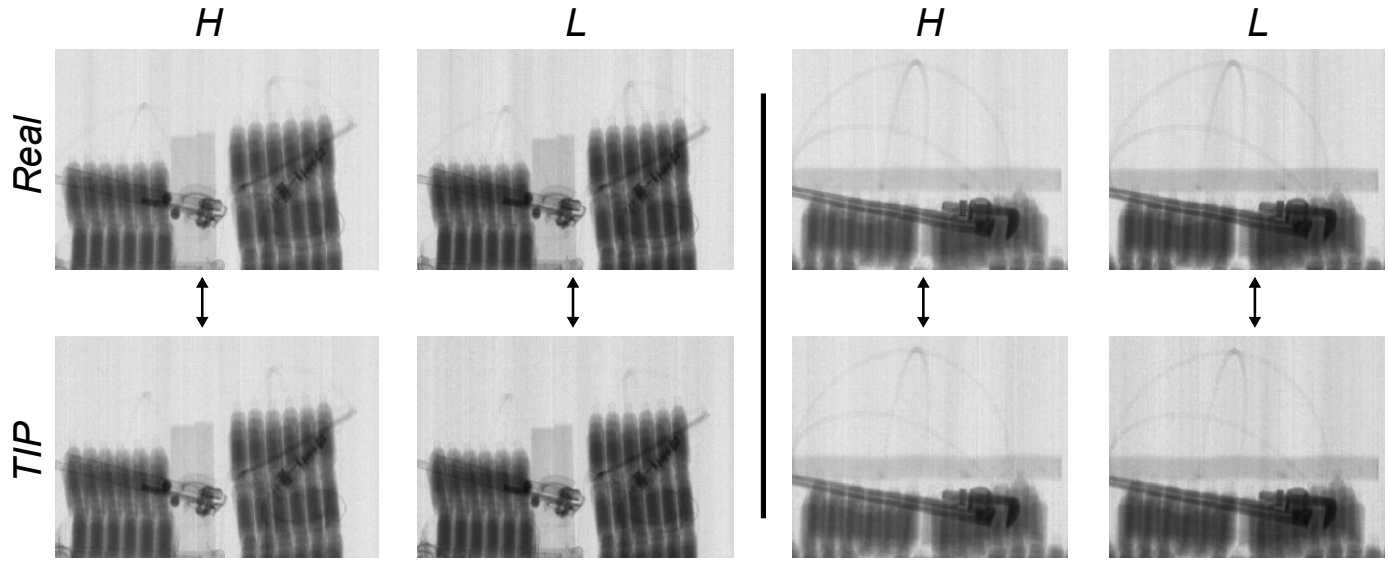


Fig. 2. Qualitative comparison of real threat images (top) and TIP images (bottom) for high (H) and low (L) energies. Industrial tools (pipe wrench, electric drill, pipe bender) have been used as a threat model. The images correspond to raw captured data and no additional processing (e.g. denoising) has been applied.

### A. Translations

Translation variation can be injected either by controlling the threat insertion position (Fig. 4), or by oversampling the threat item. Oversampling, samples multiple windows that overlap the threat, but with a small displacement. It is similar to random crops, which been used in the wider machine vision community [14] to encourage learning of small translation invariant features and to achieve class balance. However, TIP also allows us to vary larger scale placement of the threat (e.g. within a cargo container). It is possible to obtain full coverage of possible threat locations within the container.

Whilst TIP enables manipulation of threat placement, it also provides groundtruth labels for the threat ROI within an image. Such labels are essential for training and testing detection algorithms, and avoids the inconvenience of manually labeling threat regions of interest.

### B. Magnification

In X-ray cargo scanners, the fan-beam geometry means that photon paths are divergent, rather than parallel, and so the appearance of an object varies as a function of the distance from the source. When the object is close to the source it appears taller in the image than when it is placed further away. Therefore there is a natural variation in the vertical magnification of the object depending on its location. We approximate the magnification scale by

$$\alpha = 1 + d\left(\frac{l_f}{l_n} - 1\right), \tag{3}$$

where $d \in [0,1]$ is the distance away from the source normalised by container depth, $l_n$ and $l_f$ are the vertical lengths (in pixels) of the same object placed at nearest and furthest container wall from the source, respectively. The container walls, themselves, can be used to measure $l_n/l_f$ for a particular system. The parameter $d$ can sampled randomly when generating TIP examples for algorithm training.

Magnification verification is demonstrated in Fig. 4 (left).

### C. Rotations

A 3D rotation of a threat, has a corresponding 2D image appearance transformation, which is non-trivial to determine, particularly for out-of-plane rotations. However, in-plane rotations can be approximated by rotating the threat image in 2D. Adding random 2D rotations to threats during training encourages the learning of rotation-invariant features. Rotations are demonstrated in Fig. 4 (left).

### D. Noise

Cargo X-ray images are mostly affected by: (i) salt-and-pepper noise possibly from bit errors, dead pixels, or analogue-to-digital conversion; and (ii) Poisson noise originating from the number of photons emitted [16]. Both types of noise can be added to TIP imagery for training ML-based algorithms, so that they can become robust to such noise. It is particularly useful to vary the noise on a threat item that is used multiple times in training.

### E. Illumination

The illumination (mean number of X-ray photons emitted) can vary for different images due to slight differences between scanners. Illumination can also vary within images due to detector wobble in mobile configurations or due to X-ray source fluctuations [16].

We inject illumination variation into training data, by scaling the intensity of an image by some random factor (typically $1 \pm 0.05$). Variation due to source fluctuation is often removed in an image preprocessing step, but if required can be

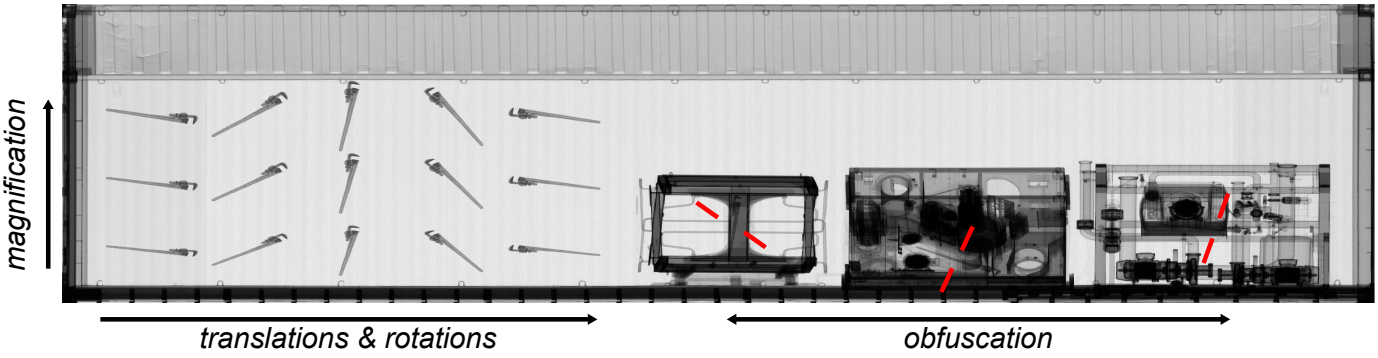


Fig. 4. Illustration of variation injection for TIP into an empty cargo container. A pipe wrench has been used as a threat model. On the left of container, and going bottom-top, the vertical dimension of the wrench has been stretched (magnified) to represent appearance variation as it is positioned at varying distances form the source. Going left-right, the wrench is being translated and rotated anti-clockwise. On the right of the container, the wrench has been obscured by different database objects before insertion into the container. The red lines indicate the location of the wrench.

generated by scaling the intensity of individual image columns by factors sampled randomly from a normal distribution.

Illumination variation due to detector wobble is more difficult to generate, as it varies as a function of image $x$ and $y$ coordinates. However, one could assume wobble is sinusoidal in $x$, and determine illumination variation in $y$ by the intersection of the detector array with the Gaussian cross-section of the fan-beam. This is essentially the reverse process to wobble correction in Ref. [16], but with wobble randomly generated rather than estimated.

### F. Volume and density

In some cases, volume transformations leave the threat class intact, for example with bulk powder or liquid narcotics. This is often not the case for weapons, where a shrunk sniper rifle does not look like a typical hand-held gun and possibly more like a toy. The relationship between volume and image appearance can be approximated by considering the Beer-Lambert law in Eq. 1. Scaling the volume $V$ equally in each dimension by some factor $\nu$ ($V \rightarrow \nu^3 V$) leads to the transformation

$$T_{xy} \rightarrow (T_{x'y'})^{\nu} \tag{4}$$

on the the threat mask. The new in-plane coordinate system $\{x'y'\}$ has also been scaled by $\nu$ in each dimension. When the volume decreases, the occupied image-area decreases and the threat simultaneously becomes brighter (less attenuating).

In even rarer cases it is useful to scale the density of the threat, such as when detecting container loads as a means of empty verification [17]. Adding density variations during training makes the algorithm more robust to the possible range of load densities. Scaling the density by $p$ ($\rho \rightarrow p\rho$) approximately transforms the threat as

$$T_{xy} \rightarrow (T_{xy})^{p}. \tag{5}$$

### G. Obscuration

When smuggling threats, criminals often attempt to obscure the threat with benign items to confuse inspectors when performing physical or image-based searches. obscuration can include (i) shielding by thick/dense materials so that the threat is barely visible in the image, or (ii) concealing the threat within complex, textured, cargo to make the resultant image very confusing. It is important that ML-based algorithms are exposed to such cases during training, as much as it is for a human. To achieve this one can project threats onto a very diverse range of real Stream-of-Commerce (SoC) images, or can use a database of extracted cargoes to project onto threat items during TIP. The later approach is demonstrated in Fig. 4 (right).

Controlling the attenuation and complexity of obscuration can be useful in both training and testing algorithms. For example, it can be ineffective to train an algorithm on threats that are so heavily attenuated that there is almost no information about the threat left. Additionally, one might identify that an algorithm is poor at distinguishing threats under certain obscuration complexities, and may want to encourage the algorithm to perform better in these cases by including more of them in the training data. The mean and variance of the obscuring attenuation may be suitable measures of difficulty, however we have not fully investigated this. Schwaninger *et al.* [18] have introduced similar metrics in baggage, which may be applicable.

## IV. Use cases

In our previous work on cargo, we have proposed algorithms for: (i) detection of loads (empty verification) [17, 19]; (ii) detection of concealed cars [20, 21]; and (iii) detection of "small metallic threats" (SMTs) [19, 22]. We have used aspects of out TIP framework for training and/or testing in each case. We here, give a brief overview of the algorithms and how TIP was employed.

### A. Load detection (empty verification)

Load detection was achieved by using a Random Forest (RF) of decision trees to classify image windows based on their coordinates, intensity moments, and oriented Basic Image Features (oBIF) histograms at multiple scales [17]. The algorithm was trained purely on TIP imagery and tested on both real SoC data and TIP imagery. Variation was injected into the

TIP training data using random flips and translations, whilst manipulating the volume and density of the loads (threats). To further increase variation, composite loads were created by combining up to five loads randomly selected from the database. The TIP framework is also beneficial because it is possible to sample image windows that definitely contain load, rather than having to manually delineate the loads in the SoC dataset.

When tested on the SoC dataset, the TIP-trained system achieves state-of-the-art performance, which is evidence that it is possible to train an algorithm purely on TIP imagery but generalise well to real imagery. TIP also allows one to gain a deeper understanding of this performance. For example, performance can be measured as a function of the position of the TIP load within the container, or as a function of the volume and density of the load. For the later, one can, for example, fix the TIP density to that of cocaine, and vary the TIP volume to measure performance at different masses of cocaine. The former is demonstrated in Fig. 5, which shows the false positives (incorrectly classified as load-containing) and false negatives (incorrectly classified as empty) as a function window position within the image. Analysing performance with TIP in this way shows that the performance is lower when loads are place in the top corners of the container, or when placed on the dark floor region. Such findings can be useful in the practical implementation of the system, for example the operator can be given a confidence rating that the image contains an adversarial load, which can be tuned by position. It can also be used to inject more samples from difficult positions in the training data so that improvements in those cases can be made.

### B. Car detection

The car detection algorithm [20] is based on a very deep 19-layer CNN [23] with 16 convolutional layers and 3 fully-connected layers. Network training was performed on stream-of-commerce X-ray cargo images. Car windows were over-sampled to create balanced car and non-car training sets. Window oversampling, is similar to random crops used in data augmentation, and can reduce CNN overfitting by encouraging the CNN to become robust to small translations.

The algorithm was tested on real data, however the TIP framework allows one to gain a deeper understanding of performance. For a particular test image, the level of obscuration on the car can be manipulated by randomly inserting database cargoes using TIP. One can measure the level of obscuration by taking the Mean Relative Attenuation (MRA), which we define as

$$\text{MRA} = \text{Mean}\left[\frac{\text{raw image} - \text{TIP image}}{\text{raw image}}\right] \quad (6)$$

$$= 1 - \text{Mean}\left[T\right], \quad (7)$$

where $T$ is the threat attenuation mask as in Eq. 2

By measuring the classification score as a function obscuration, one can determine the point at which the detector would fail under adversarial obscuration. Fig. 6 shows the classification score as a function of MRA. Cars are detected up until MRA$\sim$0.85, from which point the detector begins to misclassify them as non-car. For MRA$>$0.95, the classifier misses all cars, and it is indeed difficult for humans, even, to distinguish the car.

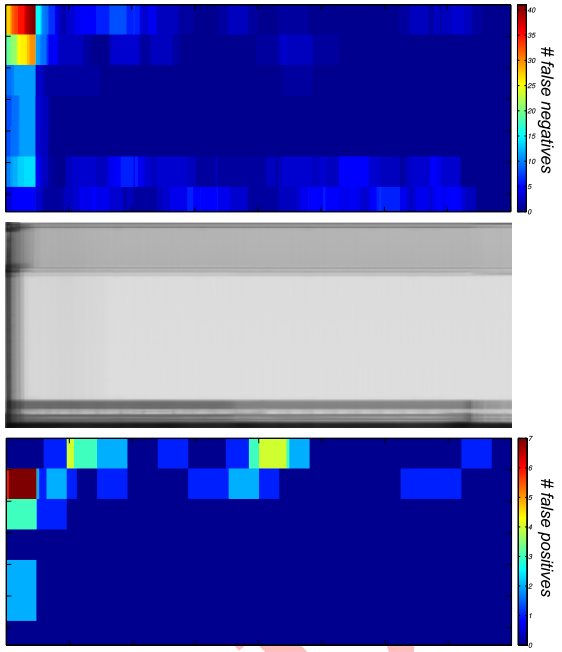


Fig. 5. Heat maps of false negatives (top) and false positives (bottom) as a function of $\{x, y\}$ position for loads similar to 1L of water. The mean across cargo container images (middle) is included for position reference. The majority of misclassified windows are placed at the top corners of the container where it is more difficult for the classifier to distinguish background from small load.

### C. SMT detection

Smuggled SMTs[1] pose a severe and persistent security threat. The development of automated detection approaches is thus critical for border and security agencies. However, the detection of SMTs in images is extremely challenging, both for security officers and algorithms: SMTs are small relative to the dimensions of X-ray cargo images, are easily concealed within legitimate cargo, are visually very similar to other load, and can be positioned in any pose. In order to train ML algorithms for the detection of SMTs, it is therefore necessary to use suitably large and diverse datasets. However, more so than other type of threats, SMTs are extremely rare in SoC images, and the staging of smuggling events for image acquisition is extremely challenging. As such, dataset augmentation through TIP is critical to the development of high performing SMT detection algorithms.

We have trained a 19-layer CNN for the detection of SMTs [19, 22]. Training is performed purely on TIP imagery. Variation is injected into the training set by projecting threats

[1]Note that we use the term 'Small Metallic Threats' (SMTs) to prevent the results being easily discoverable by keyword searching, but the objects in question are similar in appearance to industrial tools e.g. a hand drill.

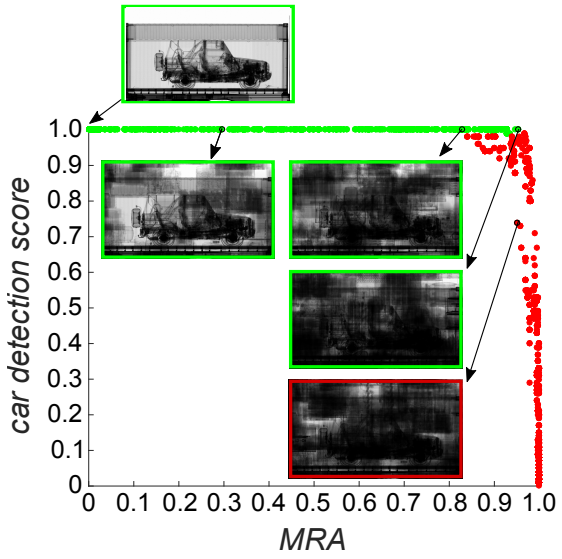


Fig. 6. The car detection score from CNN-based car classifier as a function of Mean Relative Attenuation (MRA) as the car is increasingly obscured using TIP. Green (red) dots indicate obscured examples that are correctly (incorrectly) classified as car. Thumbnails show the obscured car examples for varying MRA. Cars begin to be misclassified for MRA>0.85, at which point it is very difficult even for humans. [20]

onto a very large number of real background cargoes, with random vertical and horizontal flips, and random intensity (illumination) scaling. Initial findings suggest that this approach results in promising performance, potentially rivaling Human operators for a fraction of the typical inspection time. Further work will be required to determine the optimal size of the threat library and to assess whether further addition of realistic variation, such as out-of-plane rotation interpolation, would improve performance further.

## V. Conclusion

Training and testing Machine Learning (ML) based Automated Threat Detection (ATD) algorithms is complicated by the difficulty of obtaining large datasets and the major imbalance between the threat and innocuous classes. We propose a Threat Image Projection (TIP) framework to remedy this problem. The framework can be used to generate a very large number of training examples by adding realistic, random, variation during projection, including: (i) translations; (ii) magnification; (iii) rotations; (iv) noise; (v) illumination; (vi) volume and density; and (vii) obscuration. This framework allows generation of very large numbers of unique training images from a few images captured of a threat, thus enabling rapid addition of detection capability for emerging threats. In addition, it also allows one to form a deeper understanding of algorithm performance by carefully controlling aspects of the test data such as threat position or obscuration.

The threat extraction and projection methods were validated on experimental data and showed no significant qualitative or quantitative difference between TIP imagery and real threat imagery. In particular, there was no evidence that the TIP process created additional visual cues that could be exploited by humans or ML algorithms. We have presented three example use cases for TIP in automated cargo image analysis: (i) training and controlled testing of a load detector (empty verification); (ii) controlled testing of a car detector under increasing levels of adversarial obscuration; and (iii) training and testing a Small Metallic Threat (SMT) detector.

Future work will go towards verifying that classifiers trained purely on TIP imagery perform equally to those trained on purely real data. We will also investigate using 3D Computer-Aided Design (CAD) models [24], or Computed Tomography (CT) scans, of threats as a basis for generate realistic synthetic radiographs which can be used in TIP, and whether they can improve the performance of ML-based algorithms. This could also solve the problem of generating out-of-plane threat rotations and improve the speed with which new threats can be added to the detection capabilities of an ATD system.

Finally, whilst we have focused on training and testing ML-based threat detectors, we feel this TIP framework is equally applicable to training and evaluating human operators.

## Acknowledgment

This work was funded by Rapiscan Systems, and by EPSRC Grant no. EP/G037264/1 as part of UCL's Security Science Doctoral Training Centre.

## References

[1] J. M. Wolfe, T. S. Horowitz, and N. M. Kenner, "Cognitive psychology: rare items often missed in visual searches," *Nature*, vol. 435, no. 7041, pp. 439–440, 2005.

[2] A. Schwaninger, F. Hofer, and O. E. Wetter, "Adaptive computer-based training increases on the job performance of x-ray screeners," in *41st Annual IEEE International Carnahan Conference on Security Technology*, Oct 2007, pp. 117–124.

[3] A. Schwaninger, A. Bolfing, T. Halbherr, S. Helman, A. Belyavin, and L. Hay, "The impact of image based factors and training on threat detection performance in x-ray screening," in *Proceedings of the 3rd International Conference on Research in Air Transportation*, 2008, pp. 317–324.

[4] F. Hofer and A. Schwaninger, "Using threat image projection data for assessing individual screener performance," *WIT Transactions on the Built Environment*, vol. 82, 2005.

[5] M. Mitckes, "Threat image projection–an overview," 2003.

[6] N. Megherbi, T. P. Breckon, G. T. Flitton, and A. Mouton, "Fully automatic 3d threat image projection: Application to densely cluttered 3d computed tomography baggage images," *3rd Int. Conf. Image Process. Theory, Tools Appl*, pp. 153–159, 2012.

[7] ——, "Radon transform based automatic metal artefacts generation for 3d threat image projection," in *SPIE Security + Defence*. International Society for Optics and Photonics, 2013, pp. 89 010B–89 010B.

[8] Y. O. Yildiz, D. Q. Abraham, S. Agaian, and K. Panetta, "3D threat image projection," *Three-Dimensional Image Capture Appl.*, vol. 6805, no. 1, pp. 680 508–8, 2008.

[9] F. J. Valverde-Albacete and C. Peláez-Moreno, "100% classification accuracy considered harmful: The normalized information transfer factor explains the accuracy paradox," *PloS one*, vol. 9, no. 1, p. e84217, 2014.

[10] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.

[11] C. Drummond and R. C. Holte, "C4.5, class imbalance, and cost sensitivity: Why under-sampling beats over-sampling," 2003, pp. 1–8.

[12] N. Japkowicz and S. Stephen, "The class imbalance problem: A systematic study," *Intelligent data analysis*, vol. 6, no. 5, pp. 429–449, 2002.

[13] Z.-H. Zhou and X.-Y. Liu, "Training cost-sensitive neural networks with methods addressing the class imbalance problem," *IEEE Transactions on Knowledge and Data Engineering*, vol. 18, no. 1, pp. 63–77, 2006.

[14] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," *CoRR*, vol. abs/1405.3531, 2014.

[15] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[16] T. W. Rogers, J. Ollier, E. J. Morton, and L. D. Griffin, "Reduction of wobble artefacts in images from mobile transmission x-ray vehicle scanners," in *IEEE International Conference on Imaging Systems and Techniques Proceedings*, 2014, pp. 356–360.

[17] T. W. Rogers, N. Jaccard, E. J. Morton, and L. D. Griffin, "Detection of cargo container loads from x-ray images," in *The IET Conference on Intelligent Signal Processing*, 2015, pp. 6 .–6 .(1).

[18] A. Schwaninger, S. Michel, and A. Bolfing, "A statistical approach for image difficulty estimation in x-ray screening using image measurements," in *Proceedings of the 4th Symposium on Applied Perception in Graphics and Visualization*. ACM, 2007, pp. 123–130.

[19] N. Jaccard, T. W. Rogers, E. J. Morton, and L. D. Griffin, "Tackling the x-ray cargo inspection challenge using machine learning," in *SPIE Defense+ Security*. International Society for Optics and Photonics, 2016, pp. 98 470N–98 470N.

[20] ——, "Detection of concealed cars in complex cargo x-ray imagery using deep learning," *CoRR*, vol. abs/1606.0, pp. 1–15, 2016.

[21] N. Jaccard, T. W. Rogers, and L. D. Griffin, "Automated detection of cars in transmission x-ray images of freight containers," *IEEE Adv. Video Signal Based Surveill.*, pp. 387–392, 2014.

[22] N. Jaccard, T. W. Rogers, E. J. Morton, and L. D. Griffin, "Using deep learning on x-ray images to detect threats," in *1st Defence and Security Doctoral Symposium*. Cranfield University, 2015.

[23] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2014.

[24] Q. Gong, D. Coccarelli, R.-I. Stoian, J. Greenberg, E. Vera, and M. Gehm, "Rapid GPU-based simulation of x-ray transmission, scatter, and phase measurements for threat detection systems," International Society for Optics and Photonics, 2016.