

Efficient Design of THz Security Systems

Demian Tinkiel, Dr. David White

Department of Computing, School of Science
Institute of Technology Tallaght Dublin
Dublin, Ireland

demian.tinkiel@ittdublin.ie, david.white@ittdublin.ie

Dr. Neil A. Trappe

Department of Experimental Physics
National University of Ireland, Maynooth
Maynooth, Ireland
neal.a.trappe@nuim.ie

Abstract - The Terahertz (THz) frequency band has traditionally been of interest for the past 30 years in astronomy. However, recent advancements in technology in this band have sparked interest in other applications such as security. The THz frequency band is specially suited to this task, given that it has the penetration aspects of higher frequencies without the harmful ionizing effects. Although the area of airport security systems has seen new developments in hardware and commercial products, it is still largely underdeveloped in terms of efficient software. There are two potential considerations for efficiency in this area: personal privacy and detection speed. Both are addressed in our research.

Index Terms - Terahertz, airport, security, concealed objects, Fourier.

I. INTRODUCTION

The Terahertz (THz) waveband ranges between 300 GHz and 3 THz ($1\text{THz} = 10^{12}\text{Hz}$) ergo it has a wavelength of $\sim 300\ \mu\text{m}$ ($\sim 10^{-3}\text{m}$) and it is often referenced to belong to the sub-millimetre range. This places it above the microwave frequencies ($\sim 10^{-2}\text{m}$) but below the frequency of far-infrared light in the electromagnetic spectrum (10^{-5}m). Its wavelength allows it to be non-ionizing, as opposed to higher frequency bands such as gamma radiation (10^{-12}m) [1].

THz waves also share characteristics of its neighbouring wavebands in the fact that it has a low index of scattering and a high index of penetration. This is of particular interest for security applications because it can provide a good compromise between penetration and spatial resolution. Another aspect of THz waves is that they are reflected by metals and absorbed by water. All these properties make it difficult to generate and detect. Given that the THz waveband is seemingly inadequate for long range communications, it has been of limited commercial interest [2]. Until relatively recently, the astronomy community has been the main driver for its research. As sources and detectors have become more cost efficient and capable of reaching higher resolutions, other potential areas of applications have emerged.

Currently, only single pixel sensors are commercially available which increases detection time as well as costs, although active THz cameras are under development [3] [4] [5]. Nonetheless, there is still a clear niche for optimised software that makes efficient use of sensors. For security applications, the properties of the THz band allow for its use in an imaging system capable of detecting concealed objects. Current passive systems use the blocking properties of metals to detect dark areas on a person. An operator can then see if the shapes of these dark areas resemble a weapon (similarly to x-ray

scanners) as illustrated in fig. 1. Throughout this paper we will investigate 3 areas:

1. The use of the Fourier Transforms as a mean to model propagation.
2. The effects that an object intersecting the THz beam has on the FT output.
3. A methodology to automatically recognise the resulting patterns. Additionally certain properties of the FT will be used to increase efficiency of the approach.

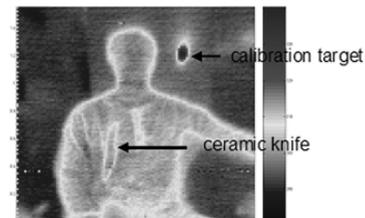


Fig. 1: Passive millimeter-wave image of concealed contraband [6].

II. MODELLING THZ

A. Optical Approach

Optical devices present several advantages over electronic devices, as they are immune to interference by external electromagnetic sources. Optical methods also have the advantage that the use of collimators and other focussing elements extend the distances over which they can be used with acceptable signal noise [7]. The complex mathematical nature of diffraction modelling methods calls for the heavy computational processing use. Furthermore, modern optical systems are quite expensive to produce, thus the use of modelling software to predict the behaviour of such equipment is widely applied in both research and commercial applications. The modelling method presented in this paper is the Fourier Transform (FT). It effectively changes a periodic function in the time domain to the frequency domain [8]. Our optical system is based on the Gaussian ($4f$) telescope (see fig. 2). This setup is highly predictable as the output will match the input, provided all distances (f) between the elements are maintained equal. Furthermore the centre of the setup is given a focussed illumination from THz source and the system output plane corresponds to the FT of the centre [9].

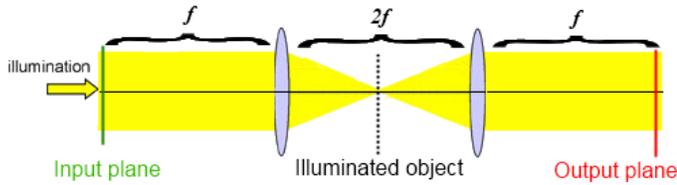


Fig. 2: Gaussian Telescope (4f) Optical System.

B. Object Characterisation

Given that the FFT is a discrete operation, it depends on the correct choice of sampling window (D), number of points (n) and sample spacing (d) to produce an accurate result. For accuracy the value of D should be bigger than the size of the object (a). A non-optimal plot is shown in fig. 3, which combined with fig. 4 illustrate the effect of source plane sampling on the corresponding FT plane. Furthermore the shape of the distribution of the samples also has an effect on the amplitude pattern, thus the object's shape may be recognized via its FT intensity pattern (i.e. output plane) where Intensity = $|A|^2$ and A is the amplitude of the incoming signal.

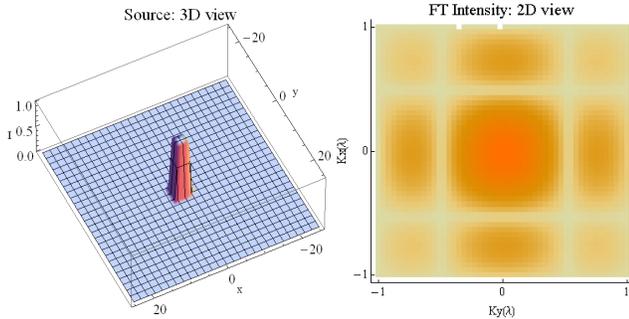


Fig. 3: Source centred at (0,0), $n = 50$, $D = 50\lambda$ and output plane.

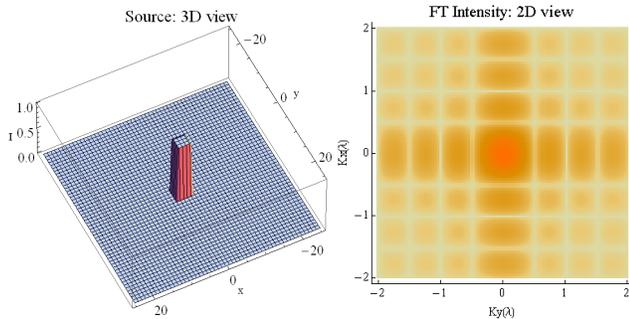


Fig. 4: Source centred at (0,0), $n = 100$, $D = 50\lambda$ and output plane.

While the size of the sampling extent and the number of sample points do have an impact on the Fourier intensity pattern, position of the input source within the sampling extent does not, as the transform only takes into account the spacing between the sample points rather than their spatial coordinates as illustrated in figs 5 and 6 where the changes in position do not have an impact on the spacing between samples (i.e. the pattern remains the same).

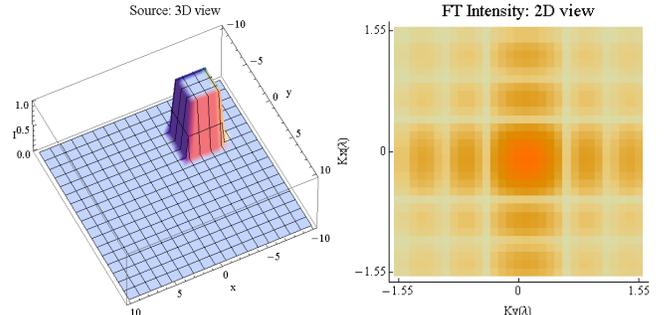


Fig. 5: Source centred at (-5,-5), $n = 31$, $D = 20$ and resultant output image.

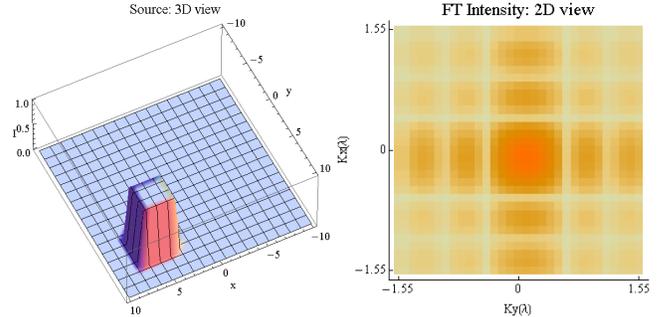


Fig. 6: Source centred at (5,5), $n = 31$, $D = 20\lambda$ and resultant output image.

C. Object position

Thus far only intensity information has been considered, however the complex Fourier data also contains phase information. This information can be used to infer characteristics of the source that amplitude alone cannot. Phase information can be derived out of the complex Fourier data as

$$\theta_{FT} = \arctan\{im[FT(x)]/re[FT(x)]\} \quad (1)$$

Considering the $4f$ system discussed and applying (1), it can be shown that object displacement has an impact on the FT phase pattern. However, it is difficult to infer a common pattern with phase information without performing certain processing. This is a result of phase 'wrapping' (i.e. it is unbounded), thus it requires to be bound over the -2π to 2π range [10] [11]. Phase unwrapping can therefore be useful to detect common patterns for object displacement. As can be seen from fig. 7, phase information now presents a definitive slope, which can be analysed. To do so a line fitting algorithm has been applied that allows us to determine the slope of a line that best fits all data points. This was performed on a data set that kept the extent and the object size fixed but changed the object position. The number of samples and the wavelength were also kept constant. Fig. 8 shows the change on phase when the object is moved to another position. Note that the slope of the line fitted phase data has changed according to the quadrant the object is located on. This shows that there is a direct relationship between phase slope and object position, meaning that we can limit our scanning area to a specific quadrant. Having determined that the FT presents specific patterns according to object shape, size & position, it is now necessary to investigate a method to automatically recognise such patterns.

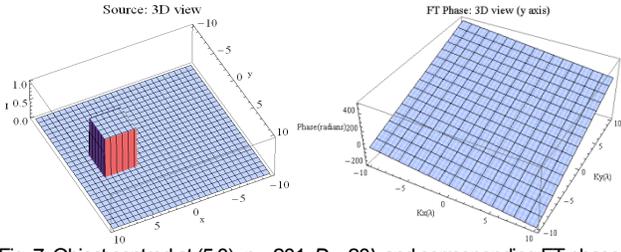


Fig. 7: Object centred at (5,0), $n = 201$, $D = 20\lambda$ and corresponding FT phase.

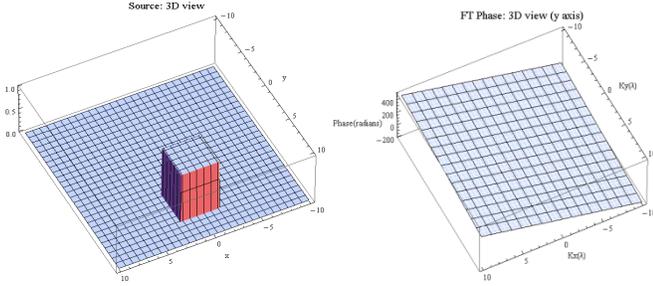


Fig. 8: Object centred at (0,5), $n = 201$, $D = 20\lambda$ and corresponding FT phase.

III. PATTERN RECOGNITION

A. Edge detection

An edge is considered as the section whose gradient (difference between a set of pixels and the neighbouring set) changes above a certain threshold, however these pixels may not necessarily be linked together into a meaningful shape. Given that edge detectors are generally based on contrast between pixels one usually pre-processes the image by removing colour information. This can be achieved by grey scaling the image.

The Sobel edge detector works by convoluting the image with what is called a Sobel mask (or kernel). This mask is applied to one square set of pixels in the image at a time and slid across until all parts of the image have been analysed. A Sobel mask is a 3x3 matrix with zero values along the centre column (for horizontal edges in the picture) or row (for vertical edges in the picture). Fig. 9 shows an example of the mask for each case.

$$M_x \begin{Bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{Bmatrix}, M_y \begin{Bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{Bmatrix}$$

Fig. 9: Horizontal (left) and vertical (right) masks for the Sobel filter.

Note that the sum of all values on either matrix is equal to zero. The image to be edge filtered is firstly convoluted with the M_x mask (result referred to as G_x) and its result convoluted with M_y (result referred to as G_y). This is used to compute the gradient magnitude in either direction. The total magnitude of the gradient can then be calculated as $|M| = \sqrt{G_x^2 + G_y^2}$ and the direction of the gradient (θ) as $\theta = \tan^{-1}(G_y/G_x)$.

Both masks are required in order to detect both cases of edges [12]. The horizontal mask will highlight rapid changes in the vertical direction while the vertical mask will do the same in the horizontal direction. However, this filter falters when an edge is not covered by either case (e.g. an edge angled at 45°). The Sobel edge detection is most efficient when used in images with

a small amount of noise. It is an inadequate approach when the noise levels in the image are too high as the small differences between noise pixels and the rest of the image will be amplified and interpreted incorrectly as an edge [13]. This effect can be seen in figs. 10 and 11 (both images have been filtered with the two kernels described above).

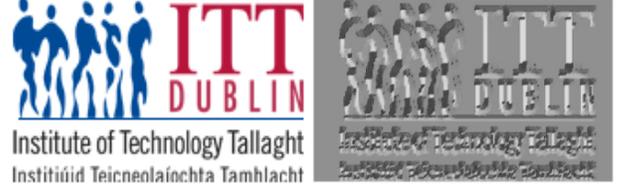


Fig. 10: Sobel filter applied to a low noise source.



Fig. 11: Sobel filter applied to a noisy source.

While edge detectors can isolate regions of sharp change and define boundaries, they cannot indicate whether these boundaries are associated to the same object or not. For example, a set of edges detected in an image may or may not be related to the same object. It is therefore necessary to determine a method for detecting the occurrence of a specific pattern.

B. Hough Transform

The Hough Transform is a tool for detection of patterns within an image. It was patented by Paul Hough in 1962 while working for IBM and was originally intended for the detection of straight lines within an image [14]. It was extended later on to detect circles [15] and put into a generalized form known as the Generalised Hough Transform (GHT) for detecting other arbitrary shapes by various authors [16]. This method works by letting each feature on an edge (i.e. a point on the shape sought), vote for all possible models (e.g. a line) that might contain it. Each of these votes is added to an accumulator. The model that has the most votes is the one that fits all features and therefore the sought shape.

There may be cases where the sought shape cannot be defined parametrically. In such cases it becomes necessary to derive a new definition for the shape. This is done by means of a reference image that contains solely the exact sought shape, i.e. the shape in the reference is of the same size and rotation as the sought shape. This is important because the method described herein does not include orientation and scaling invariance, therefore only exact matches will be detected. As with the previous versions both the target image and the reference image must be passed through an edge filter. From the reference shape, for each edge point, the direction of the edge (θ) is extracted. The distance between a reference point and the edge point (r) is calculated as well as the angle defined between the line going through both points and the x axis (α). This last angle is calculated as $\alpha = \tan^{-1}(m)$ where m is the slope of the line going through the reference point and an edge point. It is important to preserve the quadrant information from this operation as it will be seen later on in the section. Fig. 12 shows these elements in reference to a simple shape. The information is stored as sets of (r, α) pairs indexed by θ . This data structure is what is known as the r-table and is what

defines the sought shape. Table I shows the typical structure of the r-table.

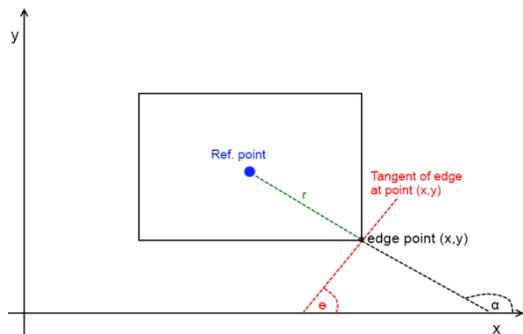


Fig. 12: Definition of a reference shape

TABLE I. TYPICAL STRUCTURE OF R-TABLE

θ_0	$(r_1, \alpha_1)_0, (r_2, \alpha_2)_0, (r_3, \alpha_3)_0, (r_4, \alpha_4)_0, \dots, (r_m, \alpha_m)_0$
...	...
θ_{359}	$(r_1, \alpha_1)_{359}, (r_2, \alpha_2)_{359}, (r_3, \alpha_3)_{359}, \dots, (r_n, \alpha_n)_{359}$

To provide an example for this step of the algorithm, consider the simple input and reference image shown in fig. 13(a). Performing edge filtering on both images returns the edge intensity and edge gradient shown in fig. 13(b)&(c) respectively for the input image and fig. 14 for the reference image. Fig. 15 shows the r-table for such reference image.

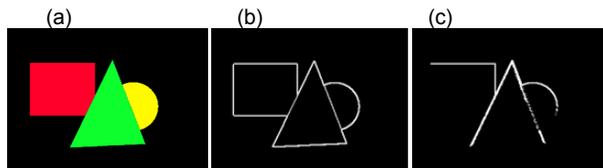


Fig. 13: (a) Input image, its (b) edge intensity and (c) edge gradient.

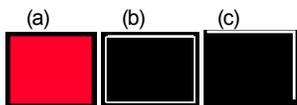


Fig. 14: (a) Reference image, its (b) edge intensity and (c) edge gradient.

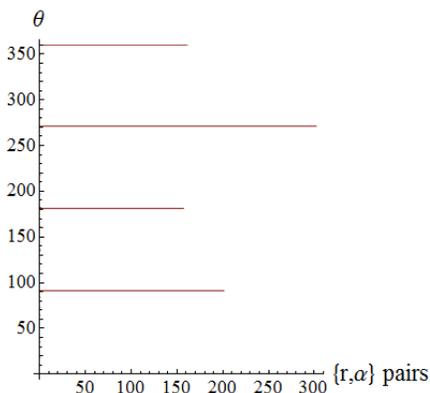


Fig. 15: R-table for square reference object.

The r-table generated in the previous step can be then used to detect the sought shape within the input image. In this step

the algorithm cycles through each edge pixel and uses the gradient of each pixel to retrieve the associated pairs from the r-table. The algorithm then cycles through each pair to calculate each possible position of the reference point that could form part of the sought shape. This is calculated as

$$x_c = x + r_\theta \cos(\alpha_\theta), y_c = y + r_\theta \sin(\alpha_\theta) \quad (4)$$

where x_c, y_c is the coordinate of a possible reference point and (r, α) is a pair indexed by θ . Each x_c, y_c is accumulated in a matrix of the same size as the input image, i.e. the reference point is bound to be within the input image. The coordinates that have the most occurrences is where the reference point that belongs to an instance of the sought shape is located. Expanding on the example from the previous section the corresponding Hough plane is shown in fig. 16. For clarity, the figure shows the accumulator overlaid over the original input image. This algorithm follows a brute force approach in that the reference shape is assumed to be present in the target image, thus to detect the presence of an object, it is necessary to threshold the resulting accumulator.

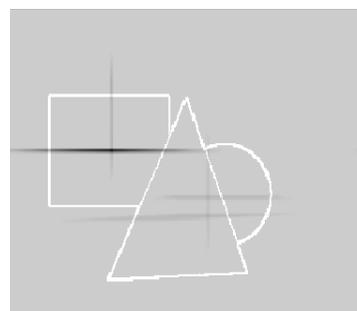


Fig. 16: GHT accumulator overlaid over original input.

The algorithm presented in the previous section is sensitive to scaling and rotation of the input image. That is, the supplied reference image must match the expected object exactly and any other version of the same object will not be detected. This is unfeasible for a real airport security scenario as a concealed weapon can be in any size/angle configuration, therefore is necessary to modify the GHT as to make it scaling and rotation invariant. To do so the accumulator must be modified in order to take into account the new information. That is for every new variable to be taken into account a new dimension must be added to the accumulator. Thus the accumulator is now a 4D array. The detection phase is very similar as before: The algorithm will brute-forcedly attempt every combination of rotation and scaling within specified parameters to detect the correct settings of the image supplied. The object definition step (i.e. r-table) remains unmodified. This section will explore the addition of each variable independently at first and then the use of both in conjunction.

Any difference in size of the image will invalidate the distance r , this must be taken into account for scaling. Thus the algorithm must be modified so that $r' = rs$ where r' is the new distance from the edge pixel to the centre of the object and s is a scaling factor. This must be done for every possible scale, from $s = 1$ to a maximum scale. This value can be any arbitrary number as long as it is bigger or equal to one. In other words, the GHT is able to detect object bigger or equally sized to the reference, but not smaller. Thus the accumulator now has three dimensions ($[x, y, s]$). To analyse the use of scaling information two different input images were used: object with size 50 x 50 pixels and object with size 100 x 100 pixels. As a reference, an

image of size 60x60 containing an object of size 50x50 was used for all 2 cases. The images in question are shown in fig. 17. For each case, the cross sections (x & y) of the accumulators for all three cases are shown for all levels, i.e. if the maximum scale is two then the accumulator is shown as $[x,y,1],[x,y,2]$. One must keep in mind that all graphs are colour coded and scaled according to their local maximum. The resulting accumulators can be seen in fig. 18. Note that for all cases the accumulator shows an absolute maximum at the position corresponding to the scaling factor i.e. if the input image contains an object three times the size of the one in the reference then the absolute maxima will be seen at position $[x,y,2]$ in the accumulator. This is represented by a peak that is significantly larger than any other peak at the remaining positions. From the same figure, one should note that the larger the shape in the input image, the longer it takes for the GHT to be processed. This is due to that larger objects have a larger number of edge pixels (i.e. pixels with a positive intensity). However the total size of the accumulator does not change, i.e. the maximum size of the accumulator is still $[width, height, maxS]$.

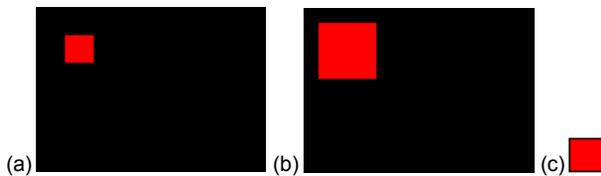


Fig. 17: Input images with objects of pixel grid size (a) 50x50 and (b) 100x100. (c) Reference image with object of size 50x50.

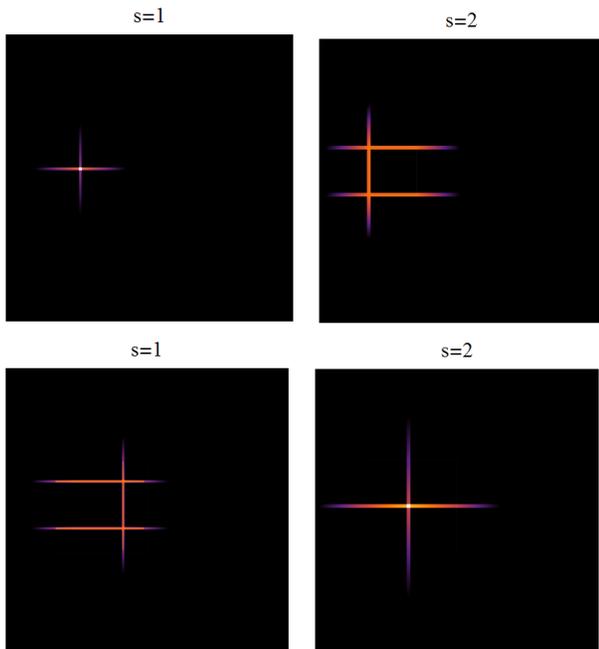


Fig. 18: GHT matching for scales $s=1$ and $s=2$, using object (above) same size and (below) twice the size of reference object.

Contrary to the case of scaling, rotating a shape by a given factor ϕ requires a change in the edge orientation (θ) and angle of displacement vector (α) by the same factor [17]. For any single point on an edge, one can calculate its new position as

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos \phi & -\sin \phi \\ \sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \quad (5)$$

where (x',y') are the new coordinates of the edge point. Note that any chosen reference point also is affected in the same manner and by consequence, angles θ and α are also affected. Thus, in this case, the new values of θ and α are 45° and 135° respectively. Observe that the length (r) remains unaffected.

Thus a potential reference point (i.e. centre) is calculated as

$$\begin{aligned} x_c &= x - (x' \cos(\phi) - y' \sin(\phi)) \\ y_c &= y - (x' \sin(\phi) - y' \cos(\phi)) \end{aligned} \quad (6)$$

where

$$x' = r \cos(\theta), y' = r \sin(\theta) \quad (7)$$

In terms of pseudo-code the sequence of steps is:

For every edge pixel $[x,y]$ (i.e. pixel with intensity >0)

For each possible rotation Φ to $\max \Phi$

- a. $\theta = \theta' - \Phi$
- b. Use θ to extract $\{r,\alpha\}$ from the r -Table
- c. Calculate x_c, y_c .
- d. Add occurrence of combination $[x_c, y_c, \Phi]$

To put this in practical terms consider the case where an input image of 100 x 100 pixels containing an object of 50 x 50 in size along with a reference image of the same parameters (see fig. 19) are used. Note that the size of the object has been kept constant given that the algorithm is only being tested from a rotation perspective (see previous section for details on scaling). Fig. 20 shows the resulting GHT as applied to the above image.



Fig. 19: Reference image with simple object at 0° (left) and input image with simple object at 45° (right).

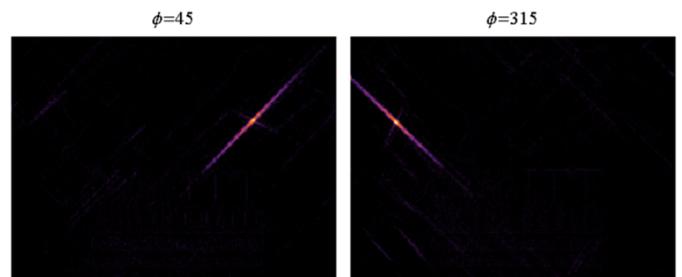


Fig. 20: Complex object - Accumulator top view at (left) $\Phi = 45$ (left) and (right) $\Phi = 315$ for $\max \Phi = 360$.

Again, a greater number of edges instigates a longer running time. This is of particular importance for complex

images as the running time can be significant as well as memory requirements. From the same example, limiting the rotation range to the first quadrant reduced the running time (from 1239.69 to 297.69 seconds) and the size on disk (from 70.5 MB to 17.7 MB) by a factor of four.

IV. EFFICIENCY

The GHT is a very accurate algorithm but it is also both time and resource consuming [18]. The algorithm works by taking every pixel in the input image as being the centre (or reference point) for a possible instance of the reference image. Every instance is stored and every repetition of a particular instance is considered a 'vote' towards that instance and stored as an accumulator. The pixel with the highest number of votes is the estimated centre of the detected object. The GHT in its basic form requires an extra dimension in the accumulator array for each property of the object to be taken into account (e.g. rotation, scaling, etc.). This makes the algorithm impractical in terms of complexity and memory requirements [19]. This section will explore possible improvements on the input for the proposed pattern recognition tool. Special attention will be paid to the Fast Fourier Transform (FFT) in terms of sampling and spatial frequency.

A. Sampling Efficiency

Consider the case where the sampling area of the scanning system is larger than that of the reference image. For example consider a reference image with extent of 150λ and 150λ sample points and an input image with extent of 170λ and 170 sample points. To process such input image would take a considerable amount of time. Given that in the image plane the stepping size between sample points is dictated by sampling extent (D) and the number of samples (n) as $d_{img} = D/n$, one cannot simply reduce the number of samples. Reducing the number of samples would change the resolution of the image and by extension change the object structure as shown in fig. 21. This would make it difficult for the GHT to match it to the reference, e.g. fig. 22. Note that while the GHT is still able to detect the target across all cases, the results become increasingly erratic as resolution decreases (i.e. more false results peaks on the GHT output).



Fig. 21: Images with (a) $d = 1\lambda$ (b) $d = 2.66\lambda$ and (c) $d = 4.59\lambda$.



Fig. 22: GHT overlaid over input image with resolution $d = 1\lambda$ (left), $d = 2.66\lambda$ (centre) and $d = 4.59\lambda$ (right).

The resolution in the Fourier plane (d_{FT}) is dictated by the sampling extent (D) and the number of samples (n) of the source (image) plane as such: $d_{FT} = 1/n d$. Similarly, the sampling extent in the Fourier plane (D_{FT}) is determined by $D_{FT} = n/D$. So increasing the number of sample points on the image plane increases the extent in the Fourier plane, whereas an increase in the sampling extent in the image plane translates as an decrease in the step size in the Fourier plane. This means that sampling a wider area around the target object will yield a

greater resolution in the FT plane. However the extent in the Fourier plane is also reduced. This is due to n remaining fixed while D increases. By doing so the object is effectively changed relatively to the extent, i.e. resolution on the object is reduced. Following Nyquist sampling theory, it is necessary to sample at $\lambda/2$ for an optimal resolution. This means that the resolution must be kept constant, as a result n must increase. Thus for a large value of D one would also have a large number of points. There may be cases where the input image contains a large area sampled around the object but contains no information.

Having determined the effects of a constant sampling rate in the FT plane, this section will now investigate its effect on the GHT. To test this, the GHT has been applied using as a reference the FT pattern of a source with $D = 50\lambda$, $n = 100$ and as input the FT pattern of a source with the same sampling rate but two times (i.e. $D = 100\lambda$, $n = 200$) and three times (i.e. $D = 150\lambda$, $n = 300$) larger. Figures 23 and 24 show the results of the corresponding GHT. The Hough Transform is a probabilistic algorithm and so the correct scaling is determined by the level of the accumulator with the highest votes (i.e. level the single brightest point). Note that the GHT does not take into account step size (i.e. it always assumes $d = 1$) making the number of samples the determinative of the size of the image, consequently it detects the object as having different scaling. This is represented by the position (or level) of the scale parameter in the accumulator.

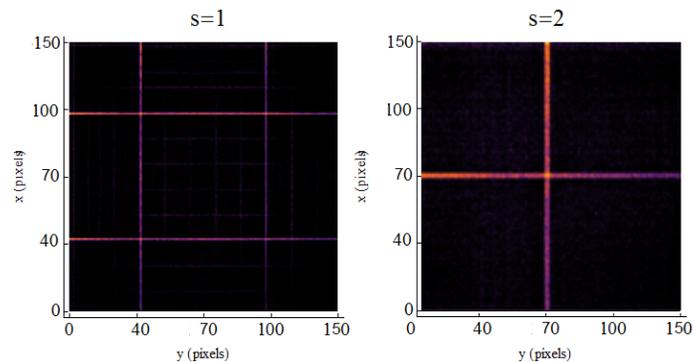


Fig. 23: 2D view of accumulator for $D = 100\lambda$, $n = 200$.

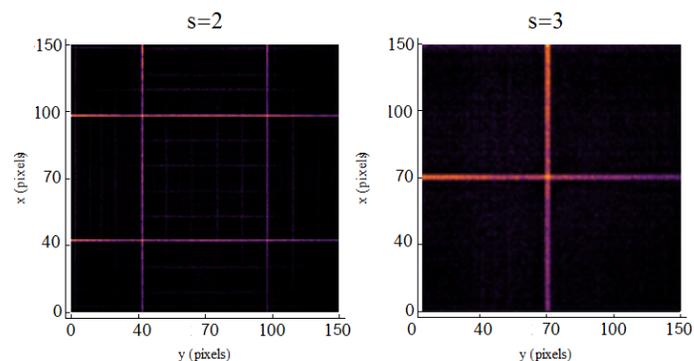


Fig. 24: 2D view of accumulator for $D = 150\lambda$, $n = 300$.

With the step size and dimensions of the object kept constant, it is possible to disregard a number of sample points in the FT plane without changing its resolution. The number of samples to disregard is given by the scaling difference in sampling extent (image plane) between the reference and the target. For example, if the area of the input image is four times bigger than that of the reference image, then, one in every four sample points (2 by 2 pixels) can be disregarded at the input

image. Using the patterns from the previous step as input for the GHT yields the output seen in figs. 25 and 26. Note that the GHT shows $s = 1$ as being the most likely scale for both cases, i.e. the GHT detects both cases as being of the same scale. This has two advantages:

- a) The number of pixels is reduced, thus a smaller amount of time is needed to process the image (e.g. 243.127 seconds for fig. 24 vs. 31.247 seconds for fig. 26, i.e. ~ 7 times faster).
- b) Given that the object has a constant scaling from the GHT perspective, the accumulator can be reduced to a 3D problem (i.e. no scaling dimension).

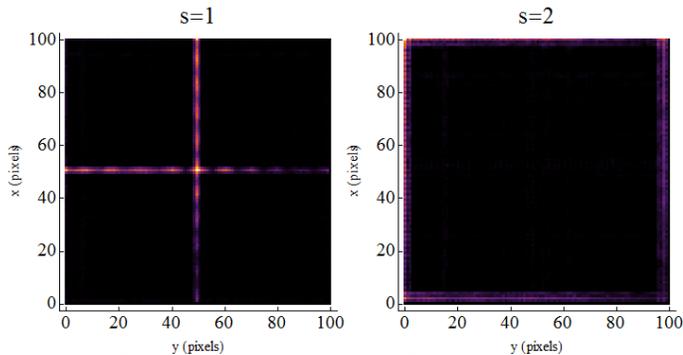


Fig. 25: 2D view of accumulator for $D = 100\lambda$, $n = 200$.

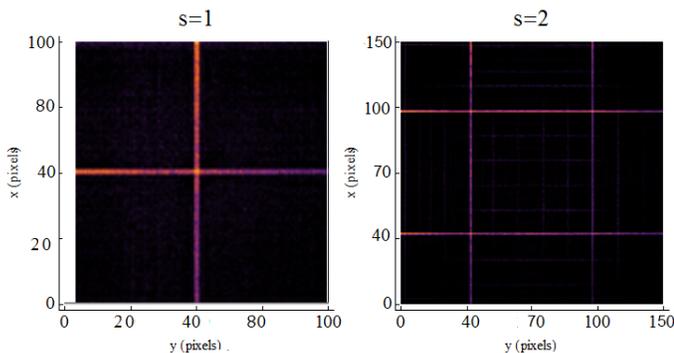


Fig. 26: 2D view of accumulator for $D = 150\lambda$, $n = 300$.

B. Detail Filtering

There may be cases whereby the reference image contains more features than the input image. However, these features may not be descriptive to the overall sought shape. Thus it becomes necessary to develop a way to filter out non-descriptive features in order to maintain matching possibilities with a more primitive shape. Consider the three cases shown in fig. 27 where a gun is reduced in detail to its minimum shape characteristics. The GHT is unable to determine a definitive match between the three cases. This section will analyse the possibility of using of the Fourier plane to circumvent this problem. Performing a FFT of the aforementioned images yields the intensity patterns shown in fig. 28. The figure shows a decibel scaled plot of the data to highlight differences between the patterns. Note both patterns present similar characteristics with the majority of the differences visible at the side lobes. This side lobes represent higher spatial frequencies, thus by filtering out this discrepancies it may be possible to facilitate a match and thus permit spare sampling.



Fig. 27: Left, fully detailed object. Right, sparsely sampled object representation.



Figure 28: GHT attempting to match sparsely detailed object to a fully detailed one.

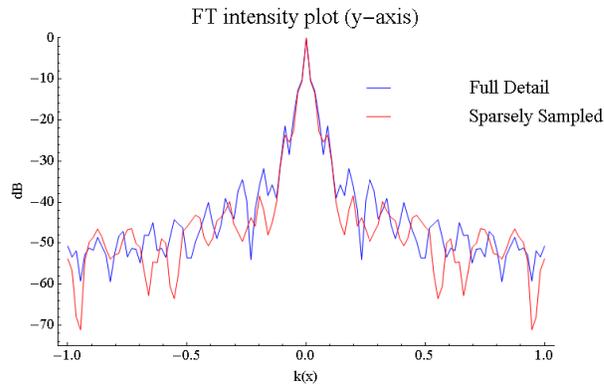
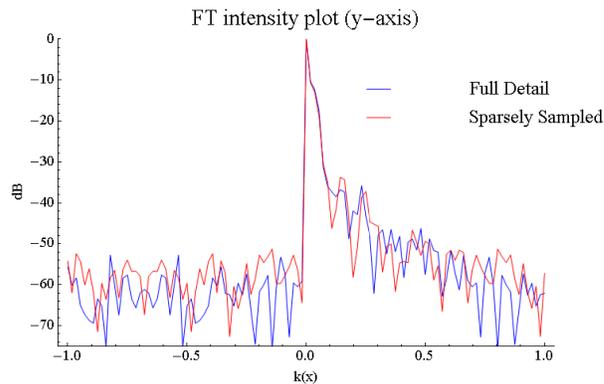


Fig. 29: FT patterns of fully-detail and sparsely sampled object compared on (above) x & (below) y axis.

A commonly used filter is the Bartlett filter dictated by the function seen in (12) and represented in fig. 30.

$$B(n, r) = \begin{cases} \frac{2n}{r}, & 0 \leq n \leq \frac{r}{2} \\ 2 - \frac{2n}{r}, & \frac{r}{2} \leq n \leq r \end{cases} \quad (12)$$

where r is the width of the filter and n is any given sample point value. The Bartlett filter effectively reduces the value of a sample point in a linear manner the further it is from the centre of the filter. All values outside the span of the filter are set to zero.

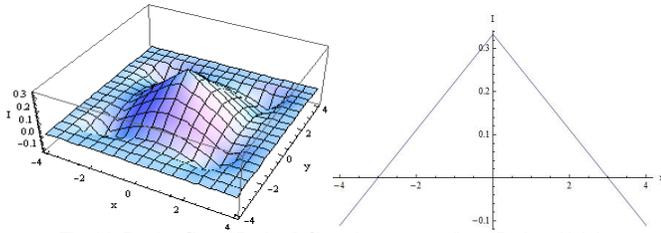


Fig. 30: Bartlett filter 3D plot (left) and corresponding 1D view (right).

The Bartlett filter effectively reduces the value of a sample point in a linear manner the further it is from the centre of the filter. All values outside the span of the filter are set to zero.

The main objective of this section is to apply pattern recognition directly on the filtered FFT pattern. For that purpose, the FFT pattern of a fully-detailed object will be filtered by means of a Bartlett kernel. The radius of the kernel was set to $r=D/4$ so that all data from the first side lobe outwards would be filtered. The resulting output will be used as input image for the GHT, see fig. 31(a). The unfiltered FFT pattern of a sparsely sampled object will be used as a reference image (see fig. 31(b)). Both the input images and the reference are 113x113 pixels in size. Given that there is no rotation or scaling involved, a simpler version of the GHT will be applied however, if the two additional parameters were to be added, the required processing time would scale up linearly for each parameter. Figure 32 shows the results of applying the GHT on these images. The GHT is able to identify the object after its non descriptive features have been reduced. A match is indicated by a single overall maximum. Note that the match occurs at the centre, as is always the case when processing FFT patterns. Furthermore, when compared against the result of applying the GHT directly on the image plane, it has been found that our methodology provides a speed-up of 47%.

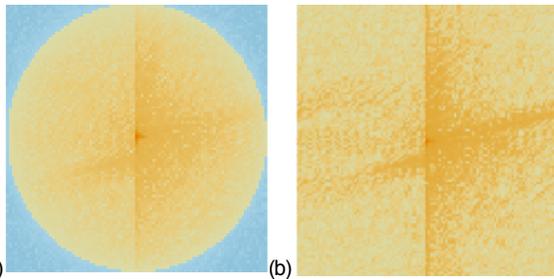


Fig. 31:(a) Filtered FT pattern of fully detailed object and (b) unfiltered pattern of sparsely sampled object.

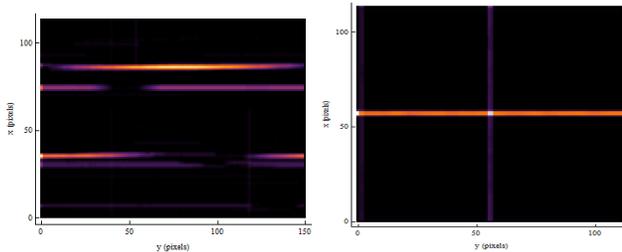


Fig. 32: (Left) GHT directly on image plane, (Right) GHT of FT plane.

V. CONCLUSIONS

THz security is an emerging area in need of efficient processing software. In this paper we presented the use of the Fourier Transform, a Gaussian telescope optical system in combination with pattern recognition algorithms (GHT), for automatic and efficient detection of concealed weapons (see fig. 33). This allows to address two key concerns within personnel scanning: Privacy issues and detection speed. The FT allows to address privacy issues as the FT does not resemble the input shape, i.e. no silhouettes are visible. Furthermore the FT allows to address the issue of detection speed in three ways:

- FT phase information allows to isolate a location quadrant
- FT sampling properties allow to reduce the scanning area
- The use of FT filters allows to match a low resolution input against a higher resolution reference

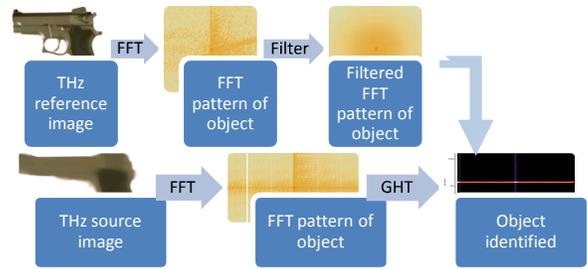


Fig. 33: Overall sequence of operations.

VI. ACKNOWLEDGMENT

The authors acknowledge the support of Science Foundation Ireland (SFI) under the Strand 1 Programme.

VII. REFERENCES

- [1] Peter H. Siegel, "Terahertz Technology," *IEEE Transactions on Microwave Theory and Techniques*, vol. 50, no. 3, March 2002.
- [2] J.M. Chamberlain, "Where optics meet electronics: Recent progress in decreasing the terahertz gap," pp. 199-213, December 2003.
- [3] E. N. Grossman, "Broadband and multispectral response of planar antennas for terahertz security screening," in *Antennas and Propagation (EUCAP), Proceedings of, Rome, 2011*, pp. 3171-3172.
- [4] V. Krozer et al., "Terahertz Imaging Systems With Aperture Synthesis Techniques," *IEEE Transactions on Microwave Theory and Techniques*, vol. 58, no. 7, pp. 2027-2039, 2010.
- [5] Christian am Weg et al., "Fast Active THz Cameras with Ranging Capabilities," *Infrared Milli Terahertz Waves*, no. 30, pp. 1281-1296, 2009.
- [6] Dr. Erich N. Grossman. (2008) Quantum information and Terahertz technology project. [Online]. http://www.boulder.nist.gov/div815/QITT_Project/Accomplishments.htm
- [7] Jurgen R. Meyer-Arendt, *Introduction to classical and modern*

- optics*, 4th ed., Kathleen M. Lafferty, Ed. New Jersey: Prentice Hall, 1995.
- [8] Raymond G. Wilson, *Fourier Series and Optical Transform Techniques in Contemporary Optics*. Toronto: Wiley-Interscience Publications, 1995.
- [9] Joseph W. Goodman, *Introduction to Fourier optics*, 3rd ed. Greenhill Village, Colorado: Roberts & Company, 2005.
- [10] Hamid Al-Nashi, "Phase unwrapping of digital signals," *Acoustics, speech and signal processing*, vol. 37, no. 11, pp. 1693-1702, 1989.
- [11] Richard McGowan and Kuc Roman, "A Direct relation between signal time series and its unwrapped phase," *Acoustics, speech and signal processing*, vol. ASSP-30, no. 5, pp. 719-726, 1982.
- [12] Mark S. Nixon and Alberto S. Aguado, *Feature Extraction and Image Processing*, 1st ed. Woburn: Newnes, 2002.
- [13] Maria Petrou and Panagiota Bosdogianni, *Image Processing: The fundamentals*. Chichester: Wiley, 1999.
- [14] P. V.C. Hough, "Method and means for recognizing complex patterns," software 3069654, 1962.
- [15] Richard O Duda and Peter E. Hart, "Use of the Hough Transform to detect lines and curves in pictures," *Communications of the ACM*, vol. 15, no. 1, pp. 11-15, 1971.
- [16] D. H. Ballard, "Generalizing the Hough Transform to detect arbitrary shapes," Rochester, 1979.
- [17] John McDonald and David Vernon, "A New Hough Transform for the Detection of Arbitrary 3-Dimensional Objects," in *Proceedings of the Optical Engineering Society of Ireland & Irish Machine Vision and Image Processing Joint Conference*, Maynooth, 1998, pp. 243-255.
- [18] Linda Shapiro and George Stockman, *Computer vision.*: Prentice-Hall, Inc, 2001.
- [19] Earl Gose, Richard Johnsonbaugh, and Steve Jost, *Pattern Recognition and Image Analysis*, 1st ed., Karen Gettman, Ed. New Jersey: Prentice Hall, 1996.

VIII. VITA

Demian Tinkiel graduated with a B.Sc. Honours in Computing from the Institute of Technology Tallaght Dublin in 2009. He is now a full-time research postgraduate in the Computing Department at the Institute of Technology Tallaght Dublin, funded by Science Foundation Ireland (SFI) working in the area of THz security applications.

David White is a full-time lecturer in the Computing Department at the Institute of Technology Tallaght Dublin and is pursuing research in the THz optics area. In 2005 Dr. White completed his PhD in the THz area in NUI Maynooth ('Development of Efficient CAD Software for Terahertz Optical Design and Analysis').

Neil Trappe graduated with a Ph.D. in long wavelength optical analysis techniques in the Experimental Physics Department at NUI Maynooth in January 2002, before continuing to work in this research area as a Post-doctoral Research Associate until June 2003. Currently he is employed as a Lecturer in the Experimental Physics Department. Dr. Trappe's research interests are in the field of far-infrared space optics, specifically working on the HIFI instrument for the Herschel Space Observatory (ESA cornerstone project, launched May 2009).