Australian Government
Department of Defence
Defence Science and
Technology Organisation

Sep 2004

DSTO

Implementation of the Hierarchical Discrete Radon Transform with Application to Image Registration

Ronald Jones, Nicholas J. Redding
and Timothy M. Payne

DSTO-TR-1599

Australian Government
Department of Defence
Defence Science and
Technology Organisation

Implementation of the Hierarchical Discrete Radon Transform with Application to Image Registration

Ronald Jones, Nicholas J. Redding and Timothy M. Payne

Intelligence, Surveillance and Reconnaissance Division
Information Sciences Laboratory

DSTO–TR–1599

ABSTRACT

This report discusses the background theory and implementation details of two new modules in the Analyst's Detection Support System (ADSS) environment. The first is the Hierarchical Discrete Radon Transform (HDRT), which provides a hierarchy of Radon transforms, from the Radon transform of single pixels in the image, right up to the Radon transform of the entire image. The second new module is an application of the HDRT to image registration. A coarse-to-fine strategy is implemented in order to handle local variations caused by terrain elevations and errors in global parameters. The use of the HDRT is fundamental to this multiresolution strategy.

Published by

DSTO Information Sciences Laboratory
PO Box 1500
Edinburgh, South Australia, Australia 5111

Telephone: (08) 8259 5555
Facsimile: (08) 8259 6567

AR No. 013-146
September, 2004

APPROVED FOR PUBLIC RELEASE

Implementation of the Hierarchical Discrete Radon Transform with Application to Image Registration

EXECUTIVE SUMMARY

In this report we discuss the implementation of two new modules in the Analyst's Detection Support System (ADSS) environment. The first is a Hierarchical Discrete Radon Transform, or HDRT. The Radon transform may be used to identify linear features in an image and has proven to be a useful tool for extracting roads and faint trails in Synethetic Aperture Radar (SAR) imagery. In particular, it is robust to the large amount of background clutter and speckle noise associated with such images. For little additional computational cost, the HDRT provides a hierarchy of Radon transforms, from the Radon transform of single pixel right up to that of the entire image. The resulting structure forms a multiresolution pyramid that would find application to any linear feature detection problem requiring a multiresolution strategy. In particular, the HDRT has application to image registration, as it is able to handle the local variations caused by terrain elevations and the errors in global parameters. The second new module for ADSS that we report on is a registration module based on HDRTs, that provides a multiresolution image registration algorithm based on the detection of linear image features in the Radon domain. The algorithm is able to capitalise on a fast cross-correlation algorithm in the Radon domain.

Authors

Ronald Jones
Intelligence, Surveillance and Reconnaissance Division

Ronald Jones received a B.Sc. and Ph.D. in Physics from Victoria University of Wellington and Monash University, Melbourne, in 1990 and 1994, respectively. Since completing his PhD, he has worked as a Research Scientist at the CSIRO, Division of Mathematical and Information Sciences, and Proteome Systems Ltd, a BioTech company based in Sydney. He joined DSTO as a Senior Research Scientist in 2004. His research interests currently include linear and nonlinear image processing techniques.

Nicholas J. Redding
Intelligence, Surveillance and Reconnaissance Division

Nicholas Redding received a B.E. and Ph.D. in electrical engineering all from the University of Queensland, Brisbane, in 1986 and 1991, respectively. In 1988 he received a Research Scientist Fellowship from the Australian Defence Science and Technology Organisation (DSTO) and then joined DSTO in Adelaide as a Research Scientist after completing his Ph.D. in artificial neural networks in 1991. In 1996 he was appointed as a Senior Research Scientist in the Microwave Radar Division (now Intelligence, Surveillance and Reconnaissance Division) of DSTO. Since joining DSTO he has applied image processing techniques to the automatic classification of ionospheric data, and more recently has researched target detection (both human and algorithmic) in synthetic aperture radar (SAR) imagery. He has recently returned from a one and a half year posting to the UK's Defence Evaluation and Research Agency where he continued the development of a suite of target detection algorithms for SAR imagery and researched new algorithms in SAR image forming using the circular Radon transform.

Timothy M. Payne
Intelligence, Surveillance and Reconnaissance Division

Timothy Payne received his B. Eng with Honours in Electronic Engineering in 1989 and a B. Sc. in Mathematics with Distinction in 1990 from Royal Melbourne Institute of Technology. He received a Ph. D. in Electrical Engineering from the University of Adelaide in 1994. In 1990 he commenced work as an electronic engineer at DSTO in the Surveillance Research Laboratory where he worked on control systems, image processing, laser power supplies and interfaces. He is currently the Head of the Image Analysis and Exploitation Group of the Intelligence, Surveillance and Reconnaissance Division.

Contents

1 Introduction

In this report we discuss the implementation of two new modules in the Analyst's Detection Support System (ADSS) environment. The first is a Hierarchical Discrete Radon Transform, or HDRT. A Radon transform [1] has found application in diverse areas such as tomographic medical imaging, seismic analysis and linear feature detection in images. It has been used previously in the ADSS environment to extract roads and faint trails in Synthetic Aperture Radar (SAR) imagery [3, 2]. The Radon transform is robust to background clutter and speckle noise, making it well suited to SAR image processing. The HDRT provides a hierarchy of Radon transforms, from the Radon transform of single pixels in the image, right up to the Radon transform of the entire image. The resulting structure forms a multiresolution pyramid that would be useful in any application requiring a multiresolution approach to the identification of linear image features. Our implementation of the HDRT is based on work presented in [4], and accompanying code. As detailed in this report, a number of modifications were necessary in order to convert the algorithm to the ADSS environment.

The second new module we discuss is an application of the HDRT to image registration of SAR images, and is based on work presented in [9] and further developed in [7, 8]. The algorithm relies on cross-correlation in the Radon domain, chosen because of the prevalence of linear features in SAR images. These linear features, which manifest as peaks in the Radon transform, are the actual basis for the matching and registration of images. Importantly, a coarse-to-fine strategy is implemented in order to handle local variations caused by terrain elevations and errors in global parameters. As discussed in this report, the use of the HDRT is fundamental to this multiresolution strategy.

This report will proceed as follows. In the following section we give a brief background theory of the HDRT, based on the exposition supplied in [5], and then discuss in detail our implementation within the ADSS environment. In Section 3 we give the theory and describe our implementation of image registration using HDRTs. Several modifications to the algorithm presented in [9] were used in our ADSS implementation. Finally, we conclude with some remarks in Section 4.

2 The Hierarchical Discrete Radon Transform

2.1 Background Theory

The following background theory and discussion follows that provided in [5]. Given an $N \times N$ discrete image $\tilde{f}(\mathbf{m})$ defined on a pixel grid M_0 and a continuous-domain 2D kernel $\eta(\mathbf{x})$ satisfying

$$\eta(\mathbf{m}) = \begin{cases} 1 & \mathbf{m} = \mathbf{0} \\ 0 & \mathbf{m} \neq \mathbf{0}, \mathbf{m} \in \mathcal{Z}^2 \end{cases} \tag{1}$$

we can form a continuous-domain function

$$f(\mathbf{x}) = \sum_{\mathbf{m} \in M_0} \tilde{f}(\mathbf{m}) \eta(\mathbf{x} - (\mathbf{m} - \mathbf{e}_0))$$

$$\text{where} \quad \mathbf{e_0} = \left[\frac{1}{2} \quad \frac{1}{2}\right]^T$$
$$\text{and} \quad M_0 = \left[-\frac{N}{2}+1, \frac{N}{2}\right] \times \left[-\frac{N}{2}+1, \frac{N}{2}\right],$$

such that $f(\mathbf{x})$ matches the values of $\tilde{f}(\mathbf{m})$ at the sample points:

$$f(\mathbf{x})|_{\mathbf{x}=\mathbf{m}-\mathbf{e_0}} = \tilde{f}(\mathbf{m}).$$

Using the translation rule for Radon transforms [6], we may write the continuous Radon transform (CRT) $g(\rho,\theta)$ of $f(\mathbf{x})$ as

$$g(\rho,\theta) = \sum_{\mathbf{m}\in M_0} \tilde{f}(\mathbf{m})\,(\mathcal{R}\eta)\,(\rho - (\mathbf{m}-\mathbf{e_0})^T\mathbf{n}(\theta),\theta) \tag{2}$$

where $\mathcal{R}\eta$ is the CRT of the kernel. If we approximate this function using a P by Q basis function expansion

$$(\mathcal{R}\eta)(\rho,\theta) \approx \sum_{p=-P/2+1}^{P/2} \sum_{q=1}^{Q} \gamma_{pq}^0\,\phi_{pq}^0(\rho,\theta) \tag{3}$$

we can write a similar approximation for $g(\rho,\theta)$:

$$g(\rho,\theta) \approx \sum_{p=-NP/2+1}^{NP/2} \sum_{q=1}^{NQ} \gamma_{pq}^K\,\phi_{pq}^K(\rho,\theta) \tag{4}$$
$$\text{where} \quad K \doteq \log_2 N. \tag{5}$$

To derive the discrete Radon transform directly, given expressions for the ϕ_{pq}^K, we use (2) to write down an expression for deriving the γ_{pq}^K from the γ_{pq}^0 and the sample values $\tilde{f}(\mathbf{m})$. This expression leads to a sparse matrix equation relating the columnised matrix of Radon coefficients to the columnised discrete image:

$$\left[\gamma_{pq}^K\right] = R\left[\tilde{f}(\mathbf{m})\right].$$

Here R is a $N^2PQ \times N^2$ matrix with entries that are combinations of the coefficients γ_{pq}^0. The form of R depends on the form of the basis functions ϕ_{pq}^K. In [9], R was computed explicitly for two models: piecewise constant (PWC), in which the ϕ_{pq}^K are shifted versions of the 2D indicator or box function, and piecewise bilinear (PWB), in which the ϕ_{pq}^K are shifted versions of the tensor product of the "tent" function with itself. In both these models the complexity of the discrete Radon transform, determined by the number of non-zero elements in R, is $O(N^3PQ)$ floating point operations (flops). We have implemented both the PWC and the PWB model in the ADSS module.

2.1.1 The HDRT

In the HDRT, instead of computing the γ_{pq}^K in one step, we work hierarchically from the pixel level upward. The computation is based on grouping adjacent samples by fours

to form *tiles* (where the notation $k:\mathbf{m}$ denotes level k of the hierarchy, pixel coordinate $\mathbf{m}$):

$$f^{0:\mathbf{m}}(\mathbf{x}) = \tilde{f}(\mathbf{m})\eta(\mathbf{x}), \quad \mathbf{m} \in M_0 \tag{6}$$

$$f^{k:\mathbf{m}}(\mathbf{x}) = \sum_{\mathbf{i}\in I} f^{k-1:2\mathbf{m}-\mathbf{i}}(\mathbf{x} - 2^{k-1}\mathbf{e_i}), \quad \mathbf{m} \in M_k \tag{7}$$

$$\text{where} \quad M_k = \left[-\frac{N}{2^{k+1}}+1, \frac{N}{2^{k+1}}\right] \times \left[-\frac{N}{2^{k+1}}+1, \frac{N}{2^{k+1}}\right] \tag{8}$$

$$I = [0,1]\times[0,1] \tag{9}$$

$$\text{and} \quad \mathbf{e_i} = -\mathbf{i} + \begin{bmatrix}\frac{1}{2} & \frac{1}{2}\end{bmatrix}^T \tag{10}$$

Transforming (6) and (7) to the Radon domain, and using the translation rule once again, we can write

$$g^{0:\mathbf{m}}(\rho,\theta) = \tilde{f}(\mathbf{m})\,(\mathcal{R}\eta)\,(\rho,\theta),\ \mathbf{m} \in M_0$$

$$g^{k:\mathbf{m}}(\rho,\theta) = \sum_{\mathbf{i}\in I} g^{k-1:2\mathbf{m}-\mathbf{i}}(\rho - 2^{k-1}\mathbf{e_i}^T\mathbf{n}(\theta),\theta), \quad \mathbf{m} \in M_k$$

Writing

$$g^{k:\mathbf{m}}(\rho,\theta) \approx \sum_{p=-P_k/2}^{Pk/2}\sum_{q=1}^{Qk} \gamma_{pq}^{k:\mathbf{m}}\phi_{pq}^{k}(\rho,\theta)$$

$$\text{where} \quad P_k = 2^k P$$

$$\text{and} \quad Q_k = 2^k Q$$

as the approximation to $g^{k:\mathbf{m}}(\rho,\theta)$, we can derive an expression relating the $\gamma_{pq}^{k:\mathbf{m}}$ to the $\gamma_{pq}^{k-1:\mathbf{m}}$. In sparse matrix form, this may be written as

$$\left[\gamma_{pq}^{k:\mathbf{m}}\right] = R_k\left[\gamma_{pq}^{k-1:\mathbf{m}}\right]$$

Noting that

$$\gamma_{pq}^{0:\mathbf{m}} = \gamma_{pq}^{0}\tilde{f}(\mathbf{m}),$$

the global Radon transform $g^{K:0}(\rho,\theta)$ may be written as

$$\left[\gamma_{pq}^{K}\right] = R_K R_{K-1}\ldots R_0\left[\tilde{f}(\mathbf{m})\right]$$

Because each matrix R_k has only $O(N^2PQ)$ entries, the number of operations required by the HDRT has been reduced to $O(N^2PQ\log_2 N)$.

2.1.2 HDRT Algorithm

The following provides a summary of the algorithm used to compute the HDRT, as detailed in [4]. Note that the algorithm assumes an input image that is square and with

dimensions of a power of two. This restriction is relaxed in our implementation of the HDRT in ADSS, as detailed in Section 2.2.1. Only the last two steps of this algorithm are image-dependent; the preceding steps need to be performed only once and can be precomputed for efficiency.

Input: An $N \times N$ matrix of image sample values $f(\mathbf{m})$, where $N = 2^{L-1}$.

1. *Select an interpolation kernel $\eta(\mathbf{x})$ with compact support, satisfying Eq.(1), for example bicubic spline, Gaussian with zero mean or the Haar kernel.*

2. *Select P and Q, the resolutions of the sample-level in the approximation of $(\mathcal{R}\eta)(\rho\theta)$, in Eq.(3).*

3. *Find the sample-level coefficients γ_{pq}^{1} for the chosen approximation (PWC or PWB) of $(\mathcal{R}\eta)(\rho\theta)$.*

4. *Define the matrix R_1 which performs the sample-level transformations $\gamma_{pq}^{1:\mathbf{m}} = f(\mathbf{m})\gamma_{pq}^{1}$.*

5. *For $l = 2 : \log_2(N) + 1$, form the matrix R_l which transforms $\gamma_{pq}^{l-1:\mathbf{m}}$ to $\gamma_{pq}^{l:\mathbf{m}}$.*

6. *Form the product $R_L R_{L-1} \ldots R_2 R_1[f]$ from right to left to obtain the coefficients γ_{pq}^{L}.*

7. *Form the global Radon transform $g(\rho, \theta)$ using Eq.(4).*

In simple terms then, the HDRT is a hierarchical algorithm which uses the set of Radon transforms at a given level l to compute the set of Radon transforms at the next level $l+1$. For any level l, the image is divided evenly into non-overlapping tiles of size 2^{l-1} by 2^{l-1}, from which the Radon transforms are computed. For example, at the bottom level $l = 1$, the image is divided into 1 by 1 sized regions, or individual pixels. There are then $N \times N$ resulting Radon transforms, where N is the width and height of the image (if we assume a square image). At level $l = 2$, the image is divided into 2 by 2 sized regions, resulting in $N/4$ Radon transforms, and so on. At the other extreme, at $L = \log_2(N) + 1$, a single Radon transform is generated from the entire N by N image.

2.2 ADSS Implementation of the HDRT

The ADSS implementation of the module `HDRT` was based on C-code by Alvin Goh, following [4]. In the next section we describe the technical issues that arose from implementing the HDRT in the ADSS environment, in particular with respect to streaming input data. In Section 2.2.2 we discuss issues around decimating the HDRT in order to produce a true multiresolution Radon transform. The use of double oversampling, which is necessary in order to improve the performance of downstream modules such as image registration, is discussed in Section 2.2.3. What the user can expect to see as output from the `HDRT` module is discussed in Section 2.2.4 and 2.2.5, while some extra pre- and post-processing options are discussed in Section 2.2.6. Finally, we summarise the parameters for the `HDRT` module in ADSS in Section 2.2.7.

2.2.1 Processing Rectangular Input Data

In the ADSS image processing environment, we generally cannot assume that the entire input image is available all at once as incoming data typically becomes available in

sequential blocks of image rows. Moreover, we would prefer to make no assumption that the final input size be square, or have a width or height that is a power of two, as this might be inconvenient in many cases. As the HDRT algorithm described in Section 2.1, and Goh's accompanying code, is based on dyadic partitioning of a square image into square blocks, we need to employ a new strategy that allows for computing the HDRT for rectangular regions of arbitrary size.

Figure 1 illustrates the approach that we use. In Fig. 1a is shown the rectangular domain of a given level of the HDRT that has been computed so far. In Fig. 1b is shown the next stage of the HDRT where the domain would be partitioned into four disjoint tiles. The dotted line shows the limit of the available information and we see that there is not enough data for the Radon transform of the lower two tiles to be computed. The strategy we use is simply to compute the Radon transforms that we are able to and leave the others to be processed if and when data becomes available. The result we obtain is shown in Fig. 1c, where only the top two Radon transforms are computed and the lower image area (in black) is left blank (zero valued). Fig. 1d shows the next (and final) stage of the HDRT, where the image would be partitioned into a single tile and the dotted line shows the limit of the information available from the previous HDRT level. Again, there is not enough information to correctly compute the Radon transform, so the whole area is left blank for this level, as shown in Fig. 1e.

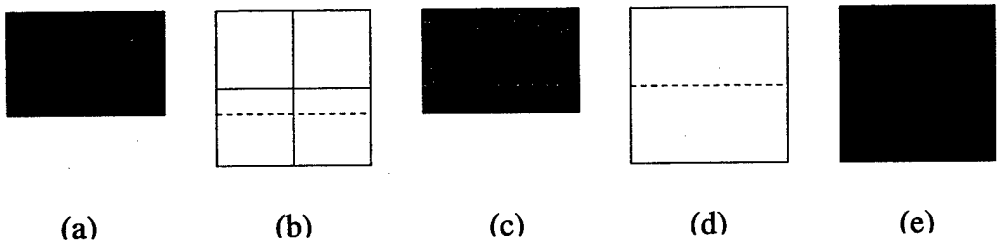

Figure 1: Processing rectangular regions of the HDRT. (a) Rectangular domain of a given level of the HDRT. (b) Stage $L-1$ of the HDRT where the domain would be partitioned into four disjoint tiles. (c) Result obtained at level $L-1$. (d) Final stage L of the HDRT. (e) Result obtained at level L.

If this is the extent of the available information that has come from the input image, then the HDRT would be finalised as shown: the top level is blank and the level below is only partially completed. However, if more information becomes available, then it percolates up through the levels of the HDRT as each is updated accordingly. For example, the level in Fig. 1a may be updated to provide enough information to complete the level shown in Fig. 1c, and this in turn would provide enough information to complete the top level in Fig. 1e.

When processing of the HDRT commences, the only size information for the input image that we can assume is its (fixed) width. In order to start processing the incoming data, the number of levels of the HDRT and other related characteristics are therefore based on the image width. We set the number of levels L in the HDRT to the floor value of $\log_2(\text{IMAGE\_WIDTH}) + 1$. For an image of width in the range $[256, 512)$, this would result in an HDRT with $L = 9$ levels. The actual size of the tile at the top level of the HDRT is given by 2^{L-1}, or 256 for $L = 9$. For example, in Fig. 2a is shown the rectangular

domain of a given level of the HDRT that has been computed so far, where the width is 300 and the height is 550. Shown in Fig. 2b is the final stage of the HDRT, where the image would be partitioned into a tiles of size 256×256. As shown by the result in Fig. 2c, areas that lie outside the tiled region are left blank. It should be noted that this particular area will not be blank for other levels of the HDRT. For example, the level 6 result of the HDRT is shown in Fig. 2d. Here the tile size is $2^{6-1} = 32$ and it is possible to fit and compute Radon transforms for these tiles within the blank area of Fig. 2c. In this case however, the results would not be percolated to higher levels of the HDRT.

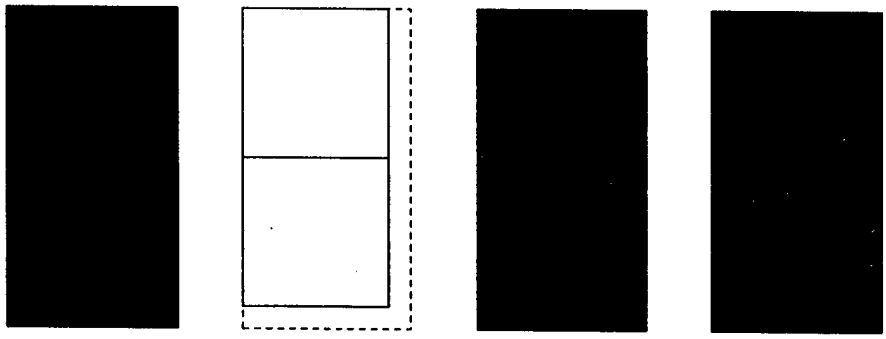

Figure 2: Processing rectangular regions of the HDRT. (a) Rectangular domain of width 300 and height 550. (b) Final stage L of the HDRT, where domain is partitioned into a tiles of size 256×256. *(c) Result for top level L. (d) Result for level* $l = 6$, *with tile size* 32×32.

In order to process the full extent of the image domain, the user also has the option to pad the input image with zeroes in the horizontal and/or vertical direction in order to increase the image size to the nearest power of two. For example, for an image of width $w \in (256, 512]$, padding in the horizontal direction will set the image width to 512, where $512 - w$ zero valued columns are added to the right hand side of the image. Similarly, an appropriate number of zero valued rows are added to the bottom of the image, if vertical padding is specified.

2.2.2 Decimation of the HDRT

Between one level and the next in the HDRT, the Radon transform doubles in size in both the ρ and the θ axes, *i.e.* a four-fold increase in total size. At the same time, the total number of Radon transforms decreases by a factor of four. This is a direct result of the bottom-to-top hierarchical processing approach, where adjacent quadruples of tiles at level l are grouped together to form a larger tile at level $l+1$ that spans the same area. It is also what we might predict, as the minimum range of ρ values required for a tile of size $N \times N$ is given by $\sqrt{2}N$, *i.e.* it is directly proportional to N. However, the burgeoning size of the Radon transform presents problems such as excessive memory and disk use and significantly longer implementation times. Moreover, a true multiresolution pyramid contains decreasing amounts of detail as the level increases, whereas the top level of an

undecimated HDRT contains all the available information for the whole image, and lower levels contain the same resolution of information only derived from subsets of the image.

As recommended in [9], a decimation approach is employed to address these issues. At a user-specified level d, the HDRT is decimated by 2 with respect to ρ and θ. The actual process of decimation is very simple: each 4×4 block of pixels in the image is replaced by its average value. The decimated Radon transforms at a given level are then used to compute the next level of the HDRT, which in turn is also decimated by 2. The process continues until a second user-specified level $d2$ (which defaults to the top level value L), after which the HDRT continues without decimation. The size of the Radon transforms at the levels $d, d+1 \ldots d2$ will all be the same, *i.e.* 2^{d-1} squared. Note that while in the interests of speed and efficiency it is desirable to set d as low as possible, if d is too small the resolution of the resulting Radon transforms would be too low for practical use.

2.2.3 Oversampling

In order to cater for downstream modules such as image registration, it is important to provide for double oversampling of tiles in the HDRT. By so doing we allow for a denser and more accurate matching of tiles when registering one HDRT with another. Illustrated in Fig. 3a is the HDRT at a given level l, where individual tiles at this level have been labeled. The critically sampled structure, *i.e.* one with no overlapping tiles and no redundancy, is shown in Fig. 3b. Here, level $l+1$ is constructed using disjoint quadruples, so that each tile at level l contributes to exactly one tile at level $l+1$, as labeled in the figure. In Fig. 3c is shown three new results for the level $l+1$ that are produced in addition to Fig. 3b, when using double oversampling. Each tile at level l (except those around the edges of the images) now contributes to four tiles at level $l+1$, with contributing tiles as indicated in the figure. Further increases in overlap are possible, at a geometrically increasing cost to computation and memory requirements, but double oversampling has been recommended [9] as an acceptable compromise between complexity and robustness.

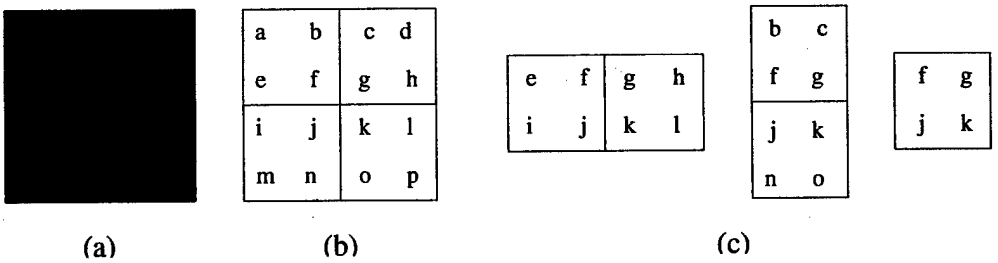

Figure 3: Critical sampling and double oversampling in the HDRT. (a) Domain of a given level of the HDRT at level l. (b) Critical sampling of the next level $l+1$ of the HDRT with contributing tiles as indicated. (c) The three additional components of level $l+1$ of the HDRT when double oversampling is used, with contributing tiles as indicated.

We see then that four components are produced for each level of the HDRT when using double oversampling. The three additional level components are not required to compute the next higher level of the HDRT, only the first (critically sampled) component. The user may also specify at what level of the HDRT double oversampling is to commence for the

requirements of downstream processing; up to the level specified only critical sampling is used. Finally, for the top level L of the HDRT, only the first two of the four components are produced, as illustrated in Fig. 4. As can be seen from the figure, it is only possible to double oversample in the vertical direction at the top level of the HDRT, and not in the horizontal direction.

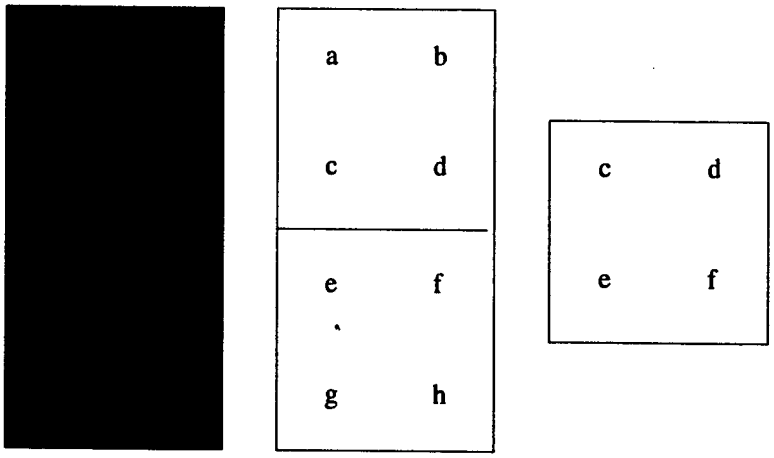

Figure 4: Double oversampling at the top level L of the HDRT; only two components are produced. (a) Example HDRT at level $L-1$. (b) First component of the output at the top level L, with contributing tiles as indicated. (c) Second component of the output at level L, with contributing tiles as indicated.

2.2.4 Output of the HDRT

Figure 5 shows an example result produced by the HDRT, illustrating the coordinate system used. Here we consider some $N \times N$ tile in the image containing a single point, as shown to the left of Figure 5. The coordinates of the point are parameterised by (ρ, θ) as shown, where θ is the angle of the perpendicular to the direction of the point, measured anticlockwise from horizontal. This accords with standard implementations for the Radon transform, for example the ADSS module `radon` [3].

The HDRT produces a Radon transform as illustrated to the right of the figure, where we see the expected sinusoidal curve. The angle θ, in the vertical direction of the image, has a range $[0, 180)$, where 0 is at the top of the image. The actual number of image rows in the vertical direction is given by $\frac{QN}{2}$, where Q is the number of basis functions in θ, as per Eq.(3). The value of ρ, in the horizontal direction, has a range $[-\frac{\alpha}{2}N, \frac{\alpha}{2}N]$, where α is determined by the model used for the interpolation kernel in Eq. (1), as described in Section 2.1. It has a value of 4, for the bicubic and Gaussian models, and $\sqrt{2}$, for the Haar kernel. We note however that the maximum required range in ρ for an $N \times N$ tile is only $[-\frac{\sqrt{2}}{2}N, \frac{\sqrt{2}}{2}N]$. Therefore, using a bicubic or Gaussian model will produce a certain amount of unnecessary space in the ρ dimension of the HDRT. We use the Haar kernel as the default model as the corresponding value $\alpha = \sqrt{2}$ produces no wasted space.

For the PWC interpolation model, the number of image columns is given by PN,

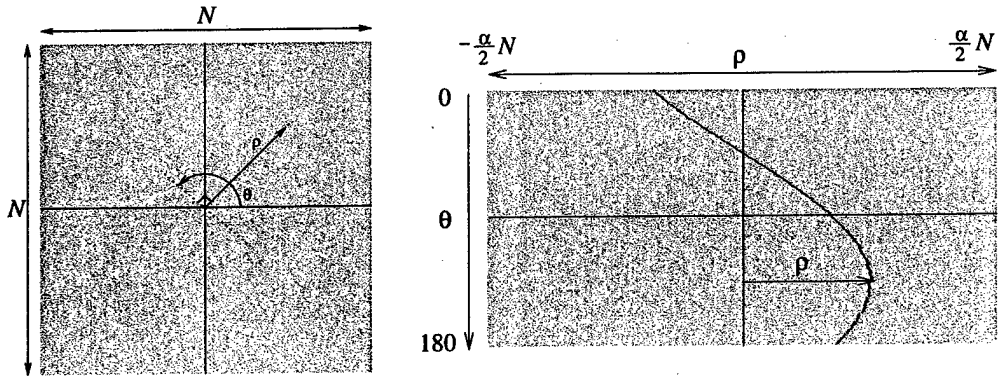

Figure 5: Example output produced by the HDRT. Left: Input tile of size $N \times N$ containing a single point. Right: Output Radon transform produced by the HDRT, with range in the ρ and θ axes as shown.

where P is the number of basis functions in ρ. A point (x, y) in the HDRT image then corresponds to a line parameterised by,

$$(\rho_x, \theta_y) = \Big((x - \frac{PN}{2})\frac{\alpha}{P},\ 360\frac{y}{QN}\Big). \tag{11}$$

For the PWB interpolation model, the number of image columns is given by $PN + 1$ and a point (x, y) in the HDRT image corresponds to a line,

$$(\rho_x, \theta_y) = \Big((x - \frac{PN+1}{2})\frac{\alpha}{P},\ 360\frac{y}{QN}\Big). \tag{12}$$

We note here that the use of the PWB interpolation model produces a more accurate approximation to the Radon transform, as it is a higher order approximation. However, this does come at the expense of an increase in processing time. Shown in Fig. 6b is the result from the HDRT using a PWC interpolation model, for the input tile shown in Fig. 6a. The result exhibits some structural noise, shown in close up in Fig. 6c. The result produced by the PWB model for the same close up region shows signficantly less structural noise, as illustrated in Fig. 6d.

2.2.5 Layered Output Images

The full output image that is produced by the HDRT is a layered image, where each level of the HDRT is stored in a separate layer and saved as a separate file, with a file name appended by a number starting from one, *e.g.* "`hdrt.img[1]`". The first layer to be output is the bottom layer of the HDRT, or layer 1, and the last layer is the top of the HDRT, or layer L. This is consistent with the order in which layers of the HDRT are computed and are made available for further processing (both within the HDRT and for downstream modules). The use of layers is necessary because the HDRT output image can have an arbitrary number of levels and each may have a different size (when applying decimation for example). As the first levels of the HDRT may generally not be useful for downstream processing, the first level of the HDRT required to be output can be specified by the user. Additionally, the user may specify the last level to be output if the upper

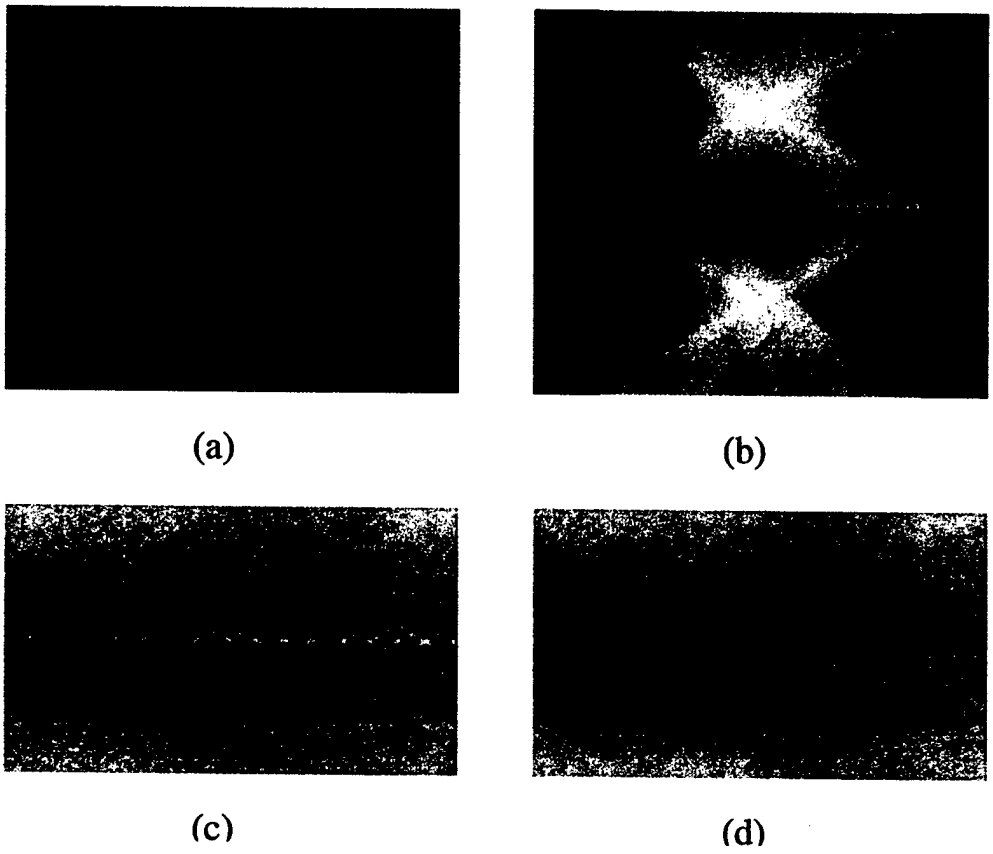

Figure 6: Use of different interpolation models. (a) Input tile. (b) Result for PWC interpolation model. (c) Close up of structural noise. (d) Same region using the PWB model.

levels of the HDRT are not required. The first output file is appended with the number one. If a level has been double oversampled, the three additional components for the level are output as image layers directly after the first component, in the sampling order that was illustrated in Fig. 3c. Once a portion of a layer l is written to an output file, we free as much of the memory as possible that is used by level l. However, we can only free memory up to the row where construction of the next level $l+1$ of the HDRT would begin, if and when more input data becomes available.

An example output for the HDRT is shown in Fig. 7. For an input image of size 256×256 pixels, as shown at the top of the figure, there are 9 levels in the HDRT. The figure shows in order from left to right, top to bottom, the six output layers that are produced from level 8 onwards. For this example, we have applied a background subtraction to the input image before computing the HDRT. As discussed in Section 2.2.6, this reduces finite domain edge effects and avoids dominating the HDRT with background pixel values.

Currently, it is not possible to set a different size for different layers in the output image; the one size specified applies to all layers. The image must be set to a size then that is large enough to contain the information in the first of the output levels output. If a layer is from decimated data and/or from one of the three oversampled components, there will be a significant amount of blank space in the output image. Note however that

for this first level at least we can (and do) trim the output image to fit to a smaller size for when decimation has been applied. The data is always written to the top left hand corner of the image layer. The first four layers in Fig. 7 are the component layers for level 8; the ability to resize would remove any blank regions. The final two are the components for level 9. The second component shown is blank as there is not enough image information to compute a double overlapping tile in the vertical direction. Such an image could be resized as well (perhaps to the size 1×1). As mentioned above, for any given level, the set of Radon transforms are stored side by side, with ρ in the horizontal direction and θ in the vertical direction.

2.2.6 Pre and Post Processing

Several pre- and post-processing options are available for the HDRT. For the purposes of image registration, it is important to apply a background subtraction to the input image before computing the HDRT. This reduces finite domain edge effects and avoids dominating the HDRT with background pixel values (which would tend to return zero translation estimates in registration) [9]. The approach recommended is to apply an adaptive high-pass filter: A median filter over a finite window is applied to the image to estimate the background. This background image is then subtracted from the input image. In the ADSS environment, this may be accomplished by piping the input image through the median filter module before the image data goes into the HDRT. Alternatively, the user may specify background subtraction in the `HDRT` module, although the implementation is different. The mean and range is calculated over the band of input data that is currently available for processing. The mean is then subtracted from the input data and the result divided by half the range. The input data then lies within the normalised range $[-1, 1]$.

As a post-processing option, the radial derivative of the HDRT transform has been successfully used to find edges of roads and trails in noisy SAR images [2]. A simple pointwise difference along the ρ axis is used to generate positive and negative edges at locations corresponding to line edges in the input image. The absolute value is taken in order to flip negative valued lines to positive values, and allow a single threshold to extract peak locations. A weighted mean may be optionally applied before and after the radial derivative, in order to smooth the affects of noise. The mean is implemented over a line of width three in the direction of ρ with weights $[\frac{1}{4}, \frac{1}{2}, \frac{1}{4}]$.

Finally, as a pre-processing option, the logarithmic value of the intensity may be applied to the input image, in order to prevent bright spots in the input image dominating the Radon transform [2]. Input values equal to zero are ignored.

2.2.7 HDRT Configuration Parameters

In Table 1 is shown a summary of the parameters for the `HDRT` module in ADSS. Note that the parameter L is given by $\log_2(\text{IMAGE\_WIDTH}) + 1$, and is determined by the module at run time.

(`image-tag` *ref-tag image-tag*)

Selects the `image` message with the *ref-tag* reference coordinate system name, and the

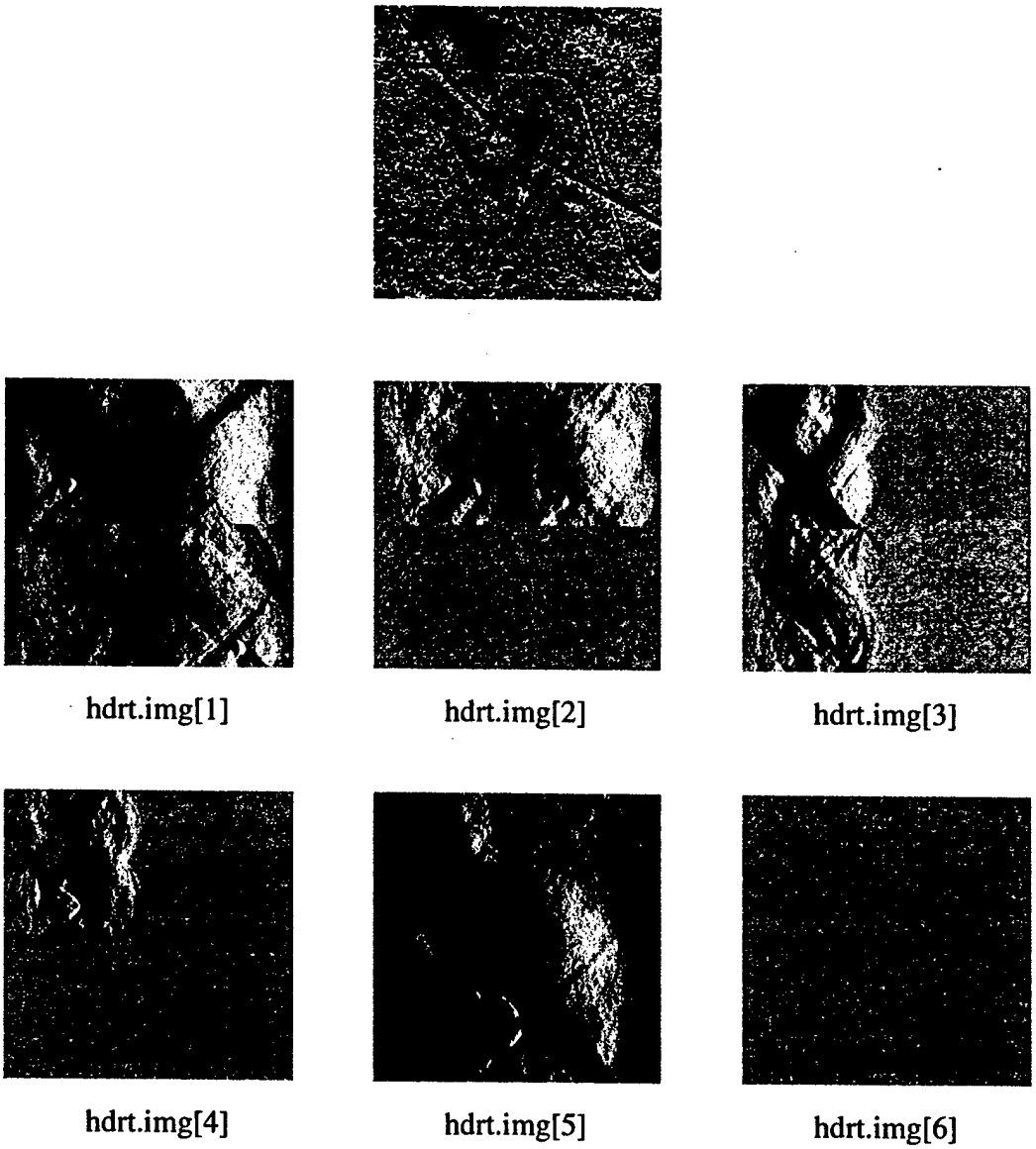

Figure 7: Example of output layers. Top: Input image of size 256×256. *Left to right, top to bottom: the four output layers for level* 8 *followed by the two output layers for level* 9 *of the HDRT.*

Config	Arg(s)	Type	Valid range	Default	Req
`image-tag`	*ref-tag*	symbol		`system`	N
	image-tag	symbol		`main`	
`P-resolution`	P	integer	even > 0	4	N
`Q-resolution`	Q	integer	even > 0	4	N
`first-level`	f	integer	$>= 1$	1	N
`last-level`	l	integer	>= f	L	N
`overlap-level`	o	integer	$>= 2$	L + 1	N
`first-decimation-level`	d	integer	$>= 1$	L + 1	N
`last-decimation-level`	$d2$	integer	>= d	L + 1	N
`pixel-type`	*type*	symbol	`bicubic`	`haar`	N
			`gaussian`		
			`haar`		
`use-log-intensity`	*uselog*	boolean		`false`	N
`take-derivative`	*deriv*	boolean		`false`	N
`apply-smoothing`	*smooth*	boolean		`false`	N
`normalise-range`	*norm*	boolean		`false`	N
`constant-interpolation`	*interp*	boolean		`true`	N
`vertical-padding`	*vp*	boolean		`false`	N
`horizontal-padding`	*hp*	boolean		`false`	N
`output-image`	*tag*	symbol		`hdrt`	N
	type	symbol		`adss`	
	name	string		`hdrt.img`	
`log-details`	*log*	boolean		`false`	N

Table 1: HDRT Configuration Parameters

designated *image-tag* as the image to be processed.

(`P-resolution` P)

Specifies the number of basis functions in the ρ (horizontal) direction.

(`Q-resolution` Q)

Specifies the number of basis functions in the θ (vertical) direction.

(`first-level` f)

Specifies the first level of the HDRT to output to the file system.

(`first-level` l)

Specifies the last level of the HDRT to compute and output to the file system.

(`overlap-level` *o*)

Specifies at what level of the HDRT to start producing double oversampled layers for output to the file system.

(`first-decimation-level` *d*)

Specifies at what level of the HDRT to start decimating (by 2) the HDRT.

(`last-decimation-level` *d2*)

Specifies at what level of the HDRT to stop decimating (by 2) the HDRT.

(`pixel-type` *type*)

Specifies interpolation model to use, as per Eq. (1), Section 2.1. Options include bicubic spline, gaussian with zero mean, or the Haar kernel. The corresponding value of α, which controls the extent of the Radon transform in the ρ direction, is then given by 4, 4 and $\sqrt{2}$ respectively.

(`use-log-intensity` *uselog*)

Specifies to take the logarithm of the intensity of the input image in order to suppress the effects of bright points in the input which tend to dominate the Radon transform.

(`take-derivative` *deriv*)

Specifies computation of the radial derivative of the HDRT, which has the effect of highlighting peaks in the HDRT. Implementation is by simple pointwise difference in the horizontal direction (ρ axis) of each level of the HDRT.

(`apply-smoothing` *smooth*)

Used in conjunction with the `take-derivative` parameter, specifies application of a simple one by three weighted mean in the horizontal direction (ρ axis) of each Radon transform in order to suppress noise.

(`normalise-range` *norm*)

Specifies to normalise the input data to the range $[-1, 1]$ before computing the HDRT.

(`constant-interpolation` *interp*)

Specifies whether to use a piecewise constant (PWC) or piecewise bilinear (PWB) interpolation model for the HDRT approximation scheme.

(`vertical-padding` *vp*)

Specifies whether to pad the image in the vertical direction with zero values, in order to increase the vertical size to the nearest power of 2.

(`horizontal-padding` *hp*)

Specifies whether to pad the image in the horizontal direction with zero values, in order to increase the horizontal size to the nearest power of 2.

(`output-image` *tag type name*)

Specifies the output image which contains the HDRT levels in layers. The module will issue an appropriate `image` message for this image.

(`log-details` *log*)

Specifies that status information should be written to `stderr` for debugging purposes.

3 Registration using HDRTs

The registration procedure we present here is an implementation of an approach presented in [9] based on HDRTs and designed for synthetic aperture radar (SAR) images. The use of Radon transforms is well suited because SAR images exhibit a prevalence of linear features, that are independent of "look angle", and a comparatively high volume of noise. Moreover, 2D cross correlation may be efficiently implemented in the Radon transform domain as a 1D correlation, followed by back-projection. In the following section, we provided a brief summary of the theory of image registration based on Radon transforms; the reader is referred to [9] for full details. We then describe those implementation details that are specific to our ADSS implementation in Section 3.2.

3.1 Global Registration of Radon Transforms

Suppose two images f_1 and f_2 are related by a global translation $\mathbf{t}$ such that

$$(\mathcal{T}_{\mathbf{t}} f_1)(x) = f_2(x),$$

where the translation operator

$$(\mathcal{T}_{\mathbf{t}} f)(\mathbf{x}) \triangleq f(\mathbf{x} - \mathbf{t}).$$

The aim of registration is to estimate $\mathbf{t}$ given f_1 and f_2. We will refer to f_2 as the *reference* image and f_1 as the *target* image. A robust estimate $\hat{\mathbf{t}}$ of $\mathbf{t}$ may be formulated using cross-correlation in the Radon domain:

$$\hat{\mathbf{t}} \triangleq \arg\max_{\mathbf{t}} T(\mathbf{t})$$

where

$$T(\mathbf{t}) \triangleq \int_0^{\pi} \int_{\Re} R_1(\rho - \mathbf{t}^T \mathbf{n}(\theta), \theta)\, R_2(\rho, \theta)\, d\rho d\theta.$$

Here, R_1 and R_2 are the Radon transforms of the target f_1 and the reference f_2 respectively, and $\mathbf{n}(\theta) \triangleq (\cos\theta, \sin\theta)^T$.

By defining M as the 1D convolution of R_1 and R_2 with respect to ρ, $M = R_1 *_\rho R_2$, it may then be shown that:

$$T(\mathbf{t}) = (\mathcal{R}^{\#} M)(\mathbf{t}),$$

where $\mathcal{R}^{\#}$ is the back-projection operator. Thus what is effectively a full 2D correlation in the spatial domain may be implemented using 1D convolution in the Radon domain, followed by back projection, a technique referred to in [9] as *fast correlation.*

3.1.1 Locally Adaptive Registration

In general, we require locally adaptive registration estimates rather than just a single global measure between the two given images as given above. To accomplish this, we implement a coarse-to-fine strategy of registration utilising the information provided by the HDRT. The registration process starts at the top level L of the HDRT, where we have two Radon tiles that subtend the domain of the reference and target images. Fast correlation is carried out between the two tiles, as defined above, generating a 2D translation map T. The location of the maximum in T provides the estimated global translation $\hat{\mathbf{t}}$ between the reference image (f_2) and the target image (f_1). This measure represents a weighted average of all translations taking place over the correlated tiles. We note in general though that there may be several maxima in T, corresponding to genuinely different translations taking place within the correlated tiles. Section 3.2.3 outlines our strategy for dealing with multiple peaks in the translation map.

At a given level l, we have a translation estimate $\hat{\mathbf{t}}_{l,i}$ for each tile i at level l in the reference image. These are used as the initial translation vectors for the corresponding tiles (subtended) in level $l-1$. As each tile at level $l-1$ may lie beneath one to four tiles at level l, we use the average of the one to four estimates $\hat{\mathbf{t}}_{l,i}$. Having an initial estimate of $\hat{\mathbf{t}}_{l-1,j}$ for each tile j at level $l-1$ in the reference image, we obtain the tile in the target image that is nearest to the location indicated. The two tiles are then fast correlated to obtain the translation map T. As described in Section 3.2.3, this initial estimate is used to guide selection of the new translation estimate for this tile. The cycle of propagating the translation estimates is continued down the levels through of the HDRT until the lowest desired level is reached.

3.1.2 Global Rotation

Allowing for a known global rotation between the reference and the target images can be conveniently handled in the Radon domain, where rotation is realised as a shift in the Radon transformation on the ρ axis. To account for a specified rotation of the reference image, the HDRT of the reference image is shifted by the required amount before fast correlation is performed.

3.2 ADSS Implementation of Registration using HDRTs

The ADSS implementation of registration using HDRTs is a C implementation based on the work in [9], as summarised above. In this section we describe some of the details particular to our implementation. In the following section we discuss the implications of streaming input data, before describing our implementation of back-projection using non-equidistant FFTs in Section 3.2.2. We then describe the algorithm for peak detection in the translation map in Section 3.2.3 and describe the output that is produced by the module in Section 3.2.4. Finally, we summarise the input parameters for the module in ADSS in Section 3.2.5.

3.2.1 Processing Rectangular Input Data

The ADSS implementation of the image registration expects as input an HDRT for both the reference and the target images, which have been prepared by the HDRT module using image background subtraction and double oversampling at all output levels. We note that not all L levels of the HDRTs need to be output; the user can select the first level l to be output by HDRT and this will be used as the lowest desired level of image registration by the module.

The process of image registration using the HDRT is in essence a top-to-bottom (or coarse-to-fine) process: An initial estimate of the translation between the two images is first obtained from the top level L of the two corresponding HDRTs. This estimate is then iteratively refined through to the lower levels of the HDRT until the process reaches the lowest supplied level. As we know, in the ADSS image processing environment, incoming data typically becomes available in sequential blocks of image rows. In order to implement the registration process then in ADSS, we must wait until enough information in the top level L becomes available for both HDRTs before commencing registration.

Depending on the shape of the input data, it will often be necessary to implement separate top-to-bottom registration processes on sequential square blocks of input data. Figure 8 shows an example of how this occurs. In Fig. 8a is an input image of dimensions $N \times 2N$, as shown. In Figs. 8b and c are shown the first and second components of the top level L of the HDRT, respectively. Each component is of size $NP \times NQ$ and we have vertically offset the second component in Fig. 8c for clarity. We note here that the top level of the HDRT then contains three separate Radon transforms, labeled A, B and C, corresponding to the three possible placements of an $N \times N$ tile in the $N \times 2N$ input image. For each of these top level L Radon transforms, it is clear we must implement a separate top-to-bottom registration process. The first of these commences when all of A becomes available in both incoming HDRTs, the second occurs when all of B becomes available and the last when all of C becomes available. Registration points will be output for all of these processes, and there will be overlap between the results output for A and B, and between the results for B and C.

3.2.2 Back-Projection of the Correlation Function

As discussed in Section 3.1, the translation map T is generated by back-projecting the result of 1D convolution in the Radon domain. The projection slice theorem states [1] that the 1D transform of any projection $p_\theta(\rho) = (\mathcal{R}f)(\rho, \theta)$ is equal to the 2D FFT of the image $f(x, y)$ with respect to polar coordinates, *i.e.*

$$(\mathcal{R}f)(\rho, \theta) = \mathcal{F}_\omega^{-1}\left[(\mathcal{F}f)(\omega, \theta)\right]. \tag{13}$$

Here $\mathcal{R}f$ denotes the Radon transform of f, $\mathcal{F}_\omega^{-1}$ denotes the 1D inverse FFT with respect to the variable ω (the frequency counterpart to ρ) and $(\mathcal{F}f)(\omega, \theta)$ denotes the 2D FFT of the image f at the (non-uniformly spaced) coordinates (ω, θ). Non-equidistant FFTs, which are leading edge, may be used to compute $(\mathcal{F}f)(\omega, \theta)$ quickly and accurately [10].

We make use of relation (13) to implement the back-projection of the correlation function, using code from an existing algorithm for the computation of the Radon transform [3].

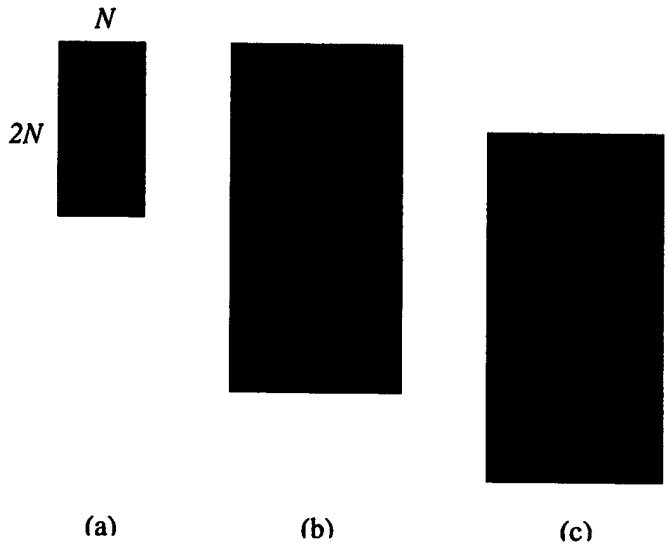

Figure 8: Registering rectangular images. (a) Input image of size $N \times 2N$. (b) First component of top level L of HDRT, of size $2 \times NP \times NQ$. (c) Second component of top level of HDRT, of size $2 \times NP \times NQ$.

The process is illustrated in Fig. 9. An example output from the Radon correlation process is shown in Fig. 9a. Here we see a characteristic sinusoid in the Radon domain that represents the translation vector we seek to find in the spatial domain. Each row of the correlation image is passed through a standard 1D FFT $\mathcal{F}_\omega$ to yield the function $(\mathcal{F}f)(\omega,\theta)$,

$$(\mathcal{F}f)(\omega,\theta) = \mathcal{F}_\omega[(\mathcal{R}f)(\rho,\theta)].$$

It may now be passed through a non-equidistant domain (NED) FFT, where the domain is given by (ω,θ), in order to reconstruct the image f. The result of this process is shown in Fig. 9b. Unfortunately we see that, although the translation vector is clearly evident as a peak, the result shows a pronounced background "bulge" and blurring; this can produce unsatisfactory results for the translation estimate. We have found that is necessary to apply a ramp filter $R(\omega)$, as illustrated in Fig. 9c, to each row of the correlation image directly after the 1D FFT $\mathcal{F}_\omega$ is applied, so that

$$(\mathcal{F}f)(\omega,\theta) \simeq R(\mathcal{F}_\omega[(\mathcal{R}f)(\rho,\theta)]).$$

As shown in Fig. 9d, the result is much sharper and will produce superior translation detection results.

3.2.3 Detection of Peaks in the Translation Map

Once we have the translation map produced by back-projection, the estimate of the translation that registers the image pairs can be extracted by finding the best candidate for the peak value in the map. In a typical scenario, the translation map will contain a certain amount of noise and probably several candidates for peaks. We employ the following strategy for selecting a peak from a translation map:

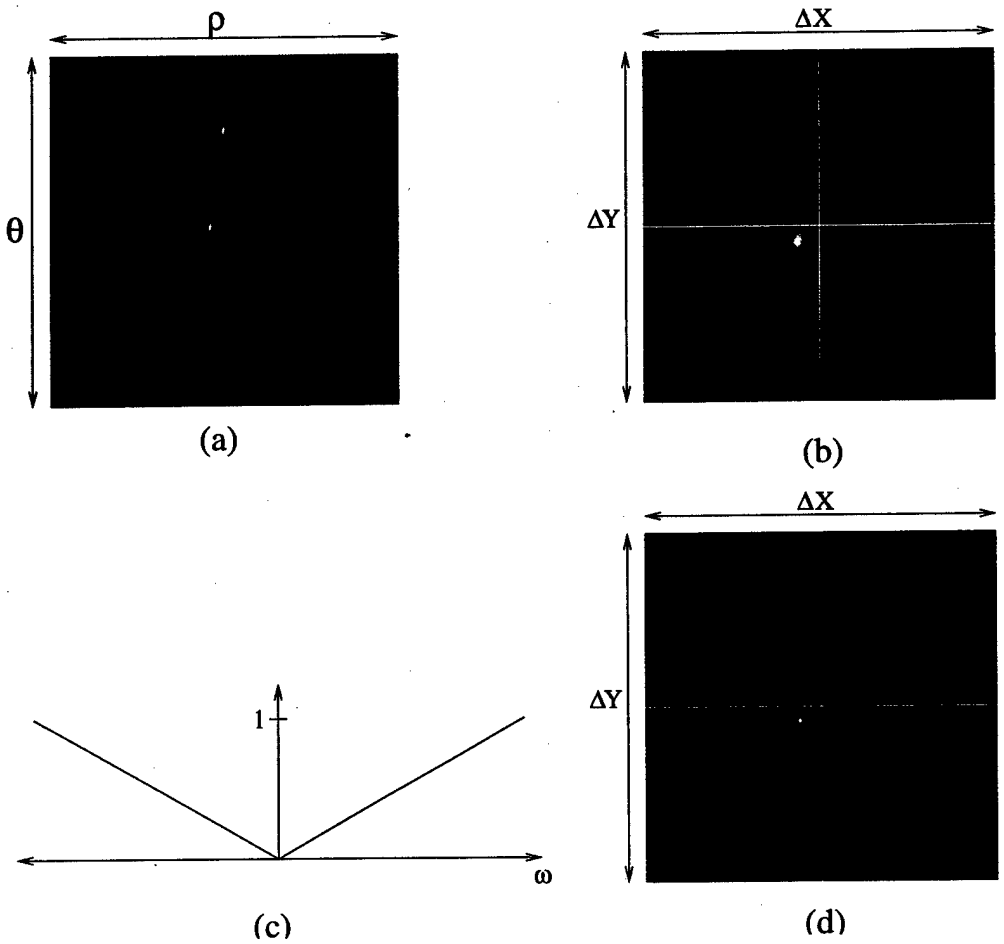

Figure 9: Back-Projection of the correlation Function. (a) Example correlation function. (b) Translation map as a result of NED. (c) Ramp filter that is applied. (d) Result when the ramp filter is applied during back-projection.

- Candidates for peaks are regional maxima: their (eight connected) immediate neighbours are strictly lower in value;
- Candidates for peaks are chosen that satisfy a certain threshold of significance. This value is specified by the user and is scaled by the size of the translation map and the level of decimation used;
- In the event that more than one candidate is found, the two highest candidates are considered and the one that is closest to the initial translation vector is selected;
- In the event that no candidates are found, the initial translation vector is selected.

Here, the initial translation vector is that handed down to the current level of the registration process from the level directly above.

3.2.4 Output Produced by the Registration Module

The output produced by the `registration` module is an ADSS CDL message of the form:

`(data registration (tie-point` x y T_x T_y`))`

The (x, y) coordinates are locations in the reference image and the vector (T_x, T_y) is that necessary to translate (x, y) to the matching point in the target image. This output format is consistent with that for existing ADSS image registration modules.

A registration tie-point is output for every tile in the first supplied level l of the given HDRT pair. For example, if the HDRT has 9 levels in total (based on an image size of 256×256) and the first level of the HDRT pair given is level $l = 6$, then there are a total of 15×15 Radon tiles in the HDRT pair at this level. Each of these tiles will yield a registration tie-point. For scenarios where more than one top-to-bottom registration process occurs, as discussed in Section 3.2.1, sets of registration tie-points are output and these results will overlap.

3.2.5 Registration Configuration Parameters

In Table 2 is shown a summary of the parameters for the `registration` module in ADSS. The parameters are as follows:

(`image-tag` *ref-tag image-tag*)

Selects the `image` message with the *ref-tag* reference coordinate system name, and the *image-tag* as the image to be processed, as the target HDRT.

(`reference-tag` *ref-tag image-tag*)

Selects the `image` message with the *ref-tag* reference coordinate system name, and the *image-tag* as the image to be processed, as the reference HDRT.

(`hdrt-levels` h)

Specifies L, the total number of HDRT levels, in the incoming HDRT transforms. For example, an input image of size 256×256 yields an HDRT with a total of $L = 9$ levels. The actual number of incoming layers that have been output from the HDRT may be different, and does not need to be specified.

(`first-decimation-level` d)

Specifies at what level decimation has commenced in the incoming HDRTs. The level of decimation is used to scale incoming `available` messages from the upstream `HDRT` module and to set the actual size of each radon transform.

(`last-decimation-level` $d2$)

Specifies at what level decimation has finished in the incoming HDRTs.

(`rotation-angle` r)

Specifies the angle of rotation between the reference and the target image. The angle applies to the reference image and is in an anti-clockwise direction.

Config	Arg(s)	Type	Valid range	Default	Req
`image-tag`	*ref-tag*	symbol		`system`	N
	image-tag	symbol		`main`	
`reference-tag`	*ref-tag*	symbol		`system`	N
	image-tag	symbol		`reference`	
`hdrt-levels`	*h*	integer	>= 1		Y
`first-decimation-level`	*d*	integer	>= 1	L + 1	N
`last-decimation-level`	*d2*	integer	>= d	L + 1	N
`rotation-angle`	*r*	real		0	N
`significance`	*s*	real		0	N
`P-resolution`	*P*	integer	even > 0	4	N
`Q-resolution`	*Q*	integer	even > 0	4	N
`pixel-type`	*type*	symbol	`bicubic`	`haar`	N
			`gaussian`		
			`haar`		
`constant-interpolation`	*interp*	boolean		`true`	N
`log-details`	*log*	boolean		`false`	N

Table 2: Registration Configuration Parameters

(`significance` *s*)

Specifies the significance threshold to apply to the translation map in order to select estimations of translation.

(`P-resolution` *P*)

Specifies the number of basis functions in the ρ (horizontal) direction. This must be the same as specified in the upstream module `hdrt`.

(`Q-resolution` *Q*)

Specifies the number of basis functions in the θ (vertical) direction. This must be the same as specified in the upstream module `hdrt`.

(`pixel-type` *type*)

Specifies interpolation model, as per Eq. (1), Section 2.1, that was used to compute the HDRT. This must be the same as specified in the module `hdrt`.

(`constant-interpolation` *interp*)

Specifies whether a piecewise constant (PWC) or piecewise bilinear (PWB) interpolation model was used for the HDRT approximation scheme. This must be the same as specified in the module `hdrt`.

(`log-details` *log*)

Specifies that status information should be written to `stderr` for debugging purposes.

4 Concluding Remarks

In this report we have given background theory and discussed implementation details for two new modules in ADSS: `hdrt` and `registration`. Although, the modules are designed to work together to perform image registration, the HDRT at least should find more general application. For example, it can be applied to multiresolution curvilinear feature detection to reliably detect long faint trails in SAR images [2]. An existing module, `curvilinear`, currently performs this task but uses tiled Radon transforms of a fixed user-specified size. The use of the HDRT would provide a full multiresolution approach to this problem. In terms of future work, effort could be directed at the problems of an unknown (as opposed to known) global rotation and/or an unknown image scaling between the target and reference images. Both are outstanding issues requiring a solution and should prove useful in SAR image data and other imaging modalities. Preliminary discussions indicate that there are possible solutions to the problem that could be investigated.

Acknowledgment

We would like to acknowledge the help of Dr. Garry Newsam, who provided the original inspiration and development of the Radon transform work.

References

1. Bracewell R. N. *The Fourier Transform and its Applications, Third Edition.* McGraw Hill, 2000.

2. Cooke T. A Radon Transform Derivative Method for Faint Trail Detection in SAR Imagery. Proceedings DICTA'99 Digital Image Computing: Techniques and Applications, pp. 31–34, 7-8 December 1999, Perth, Australia.

3. Jones R., Cooke T. and Redding N. J. Implementation of the Radon Transform using Non-equispaced Discrete Fourier Transforms. DSTO Technical Report DSTO-TR-1576, 2004.

4. Margarey J. F. A., Newsam G. N. and Payne T. M. A Fast Hierarchical Algorithm for Computing the Radon Transform of a Discrete Image. Proceedings of 2001 Workshop on Defence Applications of Signal Processing, pp. 269–273, 16-21 September, 2001, Barossa Valley Resort, Australia.

5. Margarey J. F. A. and Newsam G. N. The Hierarchical Discrete Radon Transform, Part II - Error Analysis and Numerical Results. Joint DSTO and CSSIP technical note (unpublished), January 1999.

6. Natterer F. *The Mathematics of Computerised Tomography* John Wiley, Chichester, 1986.

7. Payne T. M. Application of Radon Based Registration. DSTO Technical Report DSTO-TR-0981 (Classified), 2000.

8. Payne T. M. Impact of the Hierarchy on Radon Based Registration. DSTO Technical Report DSTO-TR-0982 (Classified), 2000.

9. Payne T. M., Margarey J. F. A. and Newsam G. N. Registration of Synthetic Aperture Radar Images Using a Multiresolution Radon Transform. Proceedings SPIE Vol. 3653, pp. 394–405, Visual Communications and Image Processing, 1999.

10. Potts D., Steidle G. and Tasche M. Fast Fourier Transforms for Nonequispaced Data: A Tutorial In *Modern Sampling Theory: Mathematics and Applications, J. J. Benedetto and P. Ferreira (Eds.)*, pp. 249–274, 2000.

DISTRIBUTION LIST

Implementation of the Hierarchical Discrete Radon Transform with Application to Image Registration

Ronald Jones, Nicholas J. Redding
and Timothy M. Payne

	Number of Copies
DEFENCE ORGANISATION	
Task Sponsor	
DGISREW	1
S&T Program	
Chief Defence Scientist FAS Science Policy AS Science Corporate Management Director General Science Policy Development	1
Counsellor, Defence Science, London	Doc Data Sheet
Counsellor, Defence Science, Washington	Doc Data Sheet
Scientific Adviser Joint	1
Navy Scientific Adviser	Doc Data Sheet
Scientific Adviser, Army	1
Air Force Scientific Adviser	1
Scientific Adviser to the DMO-ELL	Doc Data Sheet
Platform Sciences Laboratory	
Director, Platform Sciences Laboratory	1
Systems Sciences Laboratory	
Director, Systems Sciences Laboratory	1
Chief, Intelligence, Surveillance and Reconnaissance Division	1
Research Leader, Imagery Systems	1
Head, Image Analysis & Exploitation	1
Head, Signals Analysis	1
Guy Blucher	1
Dr David Booth	1
Dr David Crisp	1
Dr Ronald Jones	1
David I. Kettler	1
Dr Nicholas J. Redding	4
Rodney Smith	1
Merrilyn Fiebig	1

DEFENCE SCIENCE AND TECHNOLOGY ORGANISATION DOCUMENT CONTROL DATA				1. CAVEAT/PRIVACY MARKING
2. TITLE Implementation of the Hierarchical Discrete Radon Transform with Application to Image Registration			3. SECURITY CLASSIFICATION Document (U) Title (U) Abstract (U)	
4. AUTHORS Ronald Jones, Nicholas J. Redding and Timothy M. Payne			5. CORPORATE AUTHOR Information Sciences Laboratory PO Box 1500 Edinburgh, South Australia, Australia 5111	
6a. DSTO NUMBER DSTO–TR–1599	6b. AR NUMBER 013-146		6c. TYPE OF REPORT Technical Report	7. DOCUMENT DATE September, 2004
8. FILE NUMBER 2004/1065597	9. TASK NUMBER JTW04/202	10. SPONSOR DGICD	11. No OF PAGES 23	12. No OF REFS 10
13. URL OF ELECTRONIC VERSION http://www.dsto.defence.gov.au/corporate/reports/DSTO–TR–1599.pdf			14. RELEASE AUTHORITY Chief, Intelligence, Surveillance and Reconnaissance Division	
15. SECONDARY RELEASE STATEMENT OF THIS DOCUMENT *Approved For Public Release* OVERSEAS ENQUIRIES OUTSIDE STATED LIMITATIONS SHOULD BE REFERRED THROUGH DOCUMENT EXCHANGE, PO BOX 1500, EDINBURGH, SOUTH AUSTRALIA 5111				
16. DELIBERATE ANNOUNCEMENT No Limitations				
17. CITATION IN OTHER DOCUMENTS No Limitations				
18. DEFTEST DESCRIPTORS Radon transforms Radar images Image registration Synthetic aperture radar				

19. ABSTRACT

This report discusses the background theory and implementation details of two new modules in the Analyst's Detection Support System (ADSS) environment. The first is the Hierarchical Discrete Radon Transform (HDRT), which provides a hierarchy of Radon transforms, from the Radon transform of single pixels in the image, right up to the Radon transform of the entire image. The second new module is an application of the HDRT to image registration. A coarse-to-fine strategy is implemented in order to handle local variations caused by terrain elevations and errors in global parameters. The use of the HDRT is fundamental to this multiresolution strategy.