

FEATURE MATCHING AND TARGET RECOGNITION IN SYNTHETIC APERTURE RADAR IMAGERY

Reuven Meth & Rama Chellappa

Center for Automation Research and
Department of Electrical Engineering
University of Maryland at College Park
College Park, MD 20742, USA

ABSTRACT

An approach for target matching in Synthetic aperture radar (SAR) imagery is presented. The method is feature based where feature points in a target candidate are matched against those from an exemplar database. Matching is formulated as a non-linear optimization problem that encourages matches while minimizing the distance between the matched features. The formulation allows for missing, spurious and shifted feature points. A non-linear function is used to convexify the search space to enhance the search for the minimum objective cost. Extensions are presented for the use of two different feature types in the matching. Registration of the images is computed during the matching process in an iterative manner. Matching results are presented for simulated XPATCH and real MSTAR SAR target imagery.

1. INTRODUCTION

Synthetic aperture radar imagery has become a highly desirable sensor in the automatic target recognition (ATR) community. Its active imaging paradigm and high resolution independent of range distinguish SAR from other sensors. On the other hand, the non-literal appearance of SAR imagery, the phenomena of foreshortening, layover, and speckle, lack of amplitude calibration, high dependency on target aspect, large pose space including articulation and target variants and lack of clear edge maps, necessitate specialized processing techniques for SAR images. Significant research has been performed on model based feature extraction for use in target recognition [1, 8, 5]. This work builds on the feature extraction methods by providing a method for comparing feature maps to decide on an appropriate match. We describe a new approach to the SAR ATR matching problem where the matching problem is posed as a nonlinear optimization problem. Methods from statistical physics are applied using a recent matching approach called Softassign [3]. These methods are extended to incorporate multiple features. Experimental results of this matching scheme are presented demonstrating the utility

of this matching approach in registering the images and computing an appropriate target match.

2. FEATURE CORRESPONDENCE

Feature maps in SAR target imagery pose significant challenges. At nearby aspects, features may shift, disappear or appear. We desire to find a correspondence between two sets of features where some features may be missing, added, or displaced relative to each other, in the presence of a possible global displacement of all of the features. We are not concerned with rotation or scaling of the features since SAR features are not stable over significant changes in aspect [5], and SAR images have a fixed known scale. Further, we desire a one-to-one mapping between feature points since at the same aspect, there should be a one-to one correspondence. This is an assignment problem in which we desire a two way matching constraint to be satisfied. We will follow the development of Gold et al. [3]. We extend this method to incorporate multiple features and apply it to SAR target recognition.

2.1. Objective Function Formulation

Consider two 2D point sets $\{X_j\}$ and $\{Y_k\}$ which may be under correspondence with some translation t . We can represent the correspondence between points in the two sets via a match matrix $M = \{m_{jk}\}$ where

$$m_{jk} = \begin{cases} 1 & \text{if point } X_j \text{ corresponds to point } Y_k \\ 0 & \text{otherwise} \end{cases}$$

Consider the following objective function [3]:

$$\begin{aligned} \hat{E}_{2D}(m, t) = & \sum_{j=1}^J \sum_{k=1}^K m_{jk} \|X_j - t - Y_k\|^2 - \alpha \sum_{j=1}^J \sum_{k=1}^K m_{jk} \\ & + \sum_{j=1}^J \mu_j \left(\sum_{k=1}^{K+1} m_{jk} - 1 \right) + \sum_{k=1}^K \nu_k \left(\sum_{j=1}^{J+1} m_{jk} - 1 \right) \\ & + \frac{1}{\beta} \sum_{j=1}^{J+1} \sum_{k=1}^{K+1} m_{jk} (\log m_{jk} - 1) + \lambda \|t\|^2 \end{aligned} \quad (1)$$

Supported in part by the Army Research Laboratory Federated Laboratory Advanced Sensors Consortium under contract DAAL-01-96-2-0001 and the DARPA/AASERT contract DAAH-04-96-1-0219 monitored by the Army Research Office.

The first term in the cost function is the square Euclidean distance of between the matched points after they have been registered by the translation t . Since this term could be minimized trivially by setting all of the m_{jk} to zero, the second term is added to encourage matches. The distance tolerated between matched features is controlled by the parameter α . Since a crude registration (for the translation parameter) may be known a priori, a penalty may be placed on the allowed translation through the parameter λ . The third and fourth terms contain Lagrange multipliers μ and ν for the row and column sums. These restrictions impose the two-way constraints that a feature in one set may be matched to only one feature in the other set. A slack row and column are added to the match matrix to allow for cases when no match is present. While the match matrix elements must be zero or one (ie. M is a permutation matrix - aside from the slack row and column which may sum to larger than one,) this condition is relaxed to allow the elements to take on any value on the interval $[0,1]$. This transformation from a discrete to a continuous domain simplifies the minimization of the above function. The $x \log x$ term ensures that the elements of M are positive. This is similar to a barrier function [4] which is used to transform a constrained optimization problem into an unconstrained optimization problem, but it differs in that it does not favor points in the interior of the feasible set over the boundary [2]. This smoothing function pushes the minimum of the objective away from the discrete points by making the objective more convex, with the parameter β controlling the degree of convexity.

2.2. Minimizing the Objective Function

Similar to the method of minimization using barrier functions, the minimum of the objective function may be found by choosing a sequence $\{\beta_k\}$ such that $\beta_k \geq 0$, $\beta_{k+1} > \beta_k$ and $\beta_k \rightarrow \infty$, and minimizing the objective at each β_k [4]. This method of stepping through values of β and minimizing at each value is referred to as deterministic annealing.

At each "temperature" β , the function may be minimized by taking the partials with respect to (wrt) each of the parameters and setting the result to zero. We want to minimize with respect to the matrix elements m_{jk} and Lagrange multipliers μ_j and ν_k . Taking the partials and setting them to zero, we get,

$$\begin{aligned} m_{jk} &= \exp[-\beta(\|X_j - t - Y_k\|^2 - \alpha + \mu_j + \nu_k)] \\ &\stackrel{\text{def}}{=} -Q_{jk}, \quad j \in \{1, \dots, J\}, \quad k \in \{1, \dots, K\} \end{aligned} \quad (2)$$

$$\sum_{k=1}^{K+1} m_{jk} = 1, \quad j \in \{1, \dots, J\} \quad (3)$$

A similar column sum condition holds for the minimization wrt ν . We can solve for m_{jk} by performing a coordinate ascent on the Lagrange parameters. Since the optimization of these parameters gives conditions on the m_{jk} , we may apply these conditions, in an alternative fashion, to converge to the minimum m_{jk} . Gold et al. [3] explain that the structure of the fixed point solution for m allows a Lagrange parameter scheme where all the μ are updated followed by

the ν . Considering such an alternative updating scheme, we have

$$m_{jk}^{(2n+1)} = \exp[\beta(Q_{jk} - \mu_j^{(n+1)} - \nu_k^{(n)})] \quad (4)$$

$$m_{jk}^{(2n)} = \exp[\beta(Q_{jk} - \mu_j^{(n)} - \nu_k^{(n)})] \quad (5)$$

Taking the ratio of (4) and (5)

$$\frac{m_{jk}^{(2n+1)}}{m_{jk}^{(2n)}} = \exp[-\beta(\mu_j^{(n)} - \mu_j^{(n+1)})] \quad (6)$$

The minimization with respect to μ in (3) and (4) imply

$$\exp[\beta\mu_j^{(n+1)}] = \exp[\beta(Q_{jk} - \nu_k^{(n)})] \quad (7)$$

From (5), (6) and (7), we have

$$\begin{aligned} m_{jk}^{(2n+1)} &\stackrel{(6),(7)}{=} \frac{m_{jk}^{(2n)}}{\exp[\beta(Q_{jk} - \nu_k^{(n)} - \mu_j^{(n)})]} \\ &\stackrel{(5)}{=} \frac{m_{jk}^{(2n)}}{\sum_k m_{jk}^{(2n)}}, \quad j \in \{1, \dots, J\} \end{aligned} \quad (8)$$

Similarly, the update for minimizing ν is a column normalization. In other words, by alternatively performing row normalizations (aside from the slack row) and column normalizations (aside from the slack column), we can converge to the m_{jk} that minimizes the objective function. Rangarajan et al. [7] have proven the convergence of this algorithm.

2.3. Solving for the Translation

The translation between the two images may be solved by minimizing the objective function (1) with respect to the vector t . Taking the partial and setting to zero yields

$$t = \frac{\sum_{j,k} m_{jk}(X_j - Y_k)}{\sum_{j,k} m_{jk} + \lambda} \quad (9)$$

This is updated once for each temperature β after the optimum m_{jk} are found.

2.4. Extension to Multiple Features

The above formulation is for a single set of feature points. There are times when multiple feature sets are used for classification. For example, one may desire to use ridge and ravine features in the classification. These feature sets are disjoint and matching of features should be limited to the appropriate feature set. Following a similar procedure as above while accounting for the extra set of features in the objective function yields

$$m_{jk} = \begin{cases} \exp[-\beta(\|X_j - t - Y_k\|^2 - \alpha + \mu_j + \nu_k)] & (j, k) \in (\{1, \dots, J_1\} \times \{1, \dots, K_1\}) \cup (\{J_1 + 1, \dots, J_1 + J_2\} \times \{K_1 + 1, \dots, K_1 + K_2\}) \\ 0 & \text{otherwise} \end{cases} \quad (10)$$

The updates are again row and column normalizations at each temperature β , and the translation component is solved for at each temperature β after the m_{jk} converge. t is the

same as in Equation (9) except that the domain for (i, j) is larger in that it is for both point sets. In essence, each of the feature matches are solved for each of the sets independently. The augmented matrix could be separated into two distinct matrices where each feature correspondence is solved for separately. However, they are combined in the translation update. Note the form of the match matrix

$$M = \left[\begin{array}{c|c|c} M_1 & 0 & s_{r_1} \\ \hline 0 & M_2 & s_{r_2} \\ \hline s_{c_1} & s_{c_2} & s \end{array} \right]$$

where the M_i are the match matrices for feature i and the s_{c_i} (s_{r_i}) are the column (row) slacks for feature i .

3. EXPERIMENTAL SETUP AND RESULTS

We have applied the above algorithm to features extracted from SAR imagery. Matching results are shown for Topographical Primal Sketch (TPS) ridge features as well as for the combination of ridge and ravine features [5]. A description of these tests and results for use in SAR classification follows.

3.1. Parameter Initialization and Updates

In order to perform the annealing process that results from the introduction of the barrier like function in the above process, it is necessary to decide on the initial annealing parameter β_0 , an updating scheme for β and terminating condition. At each temperature β , practical terminating conditions must be set for the row and column updates which exhibit the search along the Lagrange multipliers.

Convergence of the translation updates t and the match matrix M are evaluated based on the sum of the absolute values

$$\begin{aligned} \|t_{\text{previous}} - t_{\text{update}}\|_1 &< \epsilon_1 \\ \|M_{\text{previous}} - M_{\text{update}}\|_1 &< \epsilon_2 \end{aligned}$$

We also incorporate limits on the maximum number of iterations for different steps in the algorithm to avoid spending too much time on any temperature at which there is very slow convergence. Table 1 lists the parameter values used with a brief explanation for each. Some of these values are based on those used in [2]. These have been modified to account for the specifics of our application such as a different tolerance allowed for displaced features (α), no penalty for translation (λ), and a quicker annealing schedule (β_r) to increase speed (since we have more feature points). With respect to the annealing parameter β , we incorporated a linear update. However, we found that the initial low values of β were useful in finding the proper translation, while the larger values effected the convergence of the match matrix. Thus, the search can be reduced by annealing at a few “low” temperatures and jumping to “high” temperatures for the remainder of the annealing process.

3.2. Classification Method

Candidate targets are classified as a specific target type by comparing the extracted features to those in a stored (offline) database. Only those aspects near the candidate’s

	Value	Purpose
α	.6	Encourage matches
λ	0	Translational penalty
β_0	.00091	Initial temperature
β_r	5	Rate for temp update
β_f	40	Terminating Temperature
ϵ_0	.01	Used in initialization of M
ϵ_1	.05	Convergence distance for t
ϵ_2	.05	Convergence distance for M
I_0	4	Max # of iterations at same β to get t
I_1	30	Max # of row/col normalizations

Table 1: Parameter initializations for optimization algorithm.

aspect are tested since the aspect may be estimated [6] and this greatly reduces the search space. The recognition is performed by applying the above matching technique to these features. Since the cost function does not account for the difference in the number of features in each image, all of the computed costs were scaled by the ratio of the number of features between the two images (or its inverse - so that it is forced to be below one). Letting n_i represent the number of feature pixels in image i , the matching cost C was computed as

$$C = r_n \hat{E}_{2D}(m, t)$$

where

$$r_n = \min\left(\frac{n_1}{n_2}, \frac{n_2}{n_1}\right).$$

When two features were used, the cost due to each feature was scaled by the fraction of feature pixels in the test image with that feature label. Letting M_j represent the number of pixels in the test image with feature j and $\hat{E}_{2D_j}(m, t)$ be the cost due to feature j , the two feature matching cost $C_{2feature}$ was computed as

$$C_{2feature} = \rho \hat{E}_{2D_1}(m, t) + (1 - \rho) \hat{E}_{2D_2}(m, t)$$

where

$$\rho = \frac{M_1}{M_1 + M_2}$$

If the target with the lowest matching cost is the correct target, a match is declared. The matching algorithm presented above was tested on a simulated XPATCH dataset as well as the MSTAR dataset of real SAR imagery.

3.2.1. XPATCH Dataset

For the XPATCH dataset, each target was tested against an exemplar of that target from a single neighboring viewpoint (1° away), as well as exemplars from 15 other targets at two nearby aspects (the same aspect as the candidate as well as 1° away). Table 2 shows the correct matching percentages using TPS ridges as the feature set in the above matching scheme. (Due to space limitations, correct matching percentages are shown in instead of confusion matrices.) One average, 86% of the targets were recognized properly. Multiple features were not used simultaneously for this dataset

since this would have significantly increased the number of features and matching time. They were incorporated for the MSTAR dataset which had less feature pixels per image.

Vehicle Number	1	2	3	4	5	6	7	8
	9	10	11	12	13	14	15	16
% Correct	89	87	82	100	95	84	66	95
Recognition	79	82	87	92	79	87	89	89

Table 2: Recognition for XPATCH dataset.

3.2.2. MSTAR Dataset

The MSTAR dataset contains three targets; two with three variants each, and one with a single variant. Classification was performed with respect to each target type and was not variant specific. Initial tests showed that a single exemplar for matching was not sufficient to get high matching confidence. We therefore used multiple exemplars from each target to perform the matching. These include multiple aspects of each target around each estimated aspect as well as from each variant. Since multiple exemplars are available for each target, it is less likely that variations in the test target instance will cause an incorrect match. A match is correct if the same target type at a nearby aspect has the highest matching score. The confusion matrix for this dataset is shown in Table 3. The matching algorithm was then applied using the combination of TPS ridges and ravines. The confusion matrix using both features is shown in Table 4. The overall percent recognition rose from 92.3% to 94.5% when two features were used simultaneously.

	bmp2	btr70	t72
bmp2	137	2	8
btr70	6	40	3
t72	4	3	138

Table 3: Confusion Matrix using TPS ridge features (MSTAR dataset).

	bmp2	btr70	t72
bmp2	142	1	4
btr70	5	42	3
t72	4	2	140

Table 4: Confusion Matrix using TPS ridge and ravine features (MSTAR dataset).

4. CONCLUSION

We have presented an approach for SAR feature matching using a nonlinear optimization technique. The method has

significant advantages in that it allows for feature migration (where the features need not all migrate in the same direction) and institutes a two way matching constraint. By introducing a barrier like function, the search space was made more convex and the minimum could be found using deterministic annealing. By alternating minimizations with respect to the match matrix and the translation parameters, both may be found simultaneously. The versatility of this method makes it applicable to many feature sets. We have implemented the algorithm using TPS features extracted from SAR imagery with good performance. To obtain more significant confidence, multiple exemplars were used in the matching process. We have extended the matching approach to incorporate two features which improved matching performance for the MSTAR dataset.

5. ACKNOWLEDGMENTS

The authors would like to thank Professor Anand Rangarajan for insightful discussions on the Softassign optimization method.

6. REFERENCES

- [1] B. Bahnu, G. Jones, J. Ahn, M. Li, and J. Yi, "Recognition of Articulated Objects in SAR Images," in *Proceedings of the ARPA Image Understanding Workshop*, (San Francisco, CA), pp. 1237–1250, Morgan Kaufman Publishers, February 1996.
- [2] S. Gold and A. Rangarajan, "A Graduated Assignment Algorithm for Graph Matching," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 18, pp. 377–388, April 1996.
- [3] S. Gold, A. Rangarajan, C. Lu, S. Pappu, and E. Mjolsnes, "New Algorithms for 2D and 3D Point Matching: Pose Estimation and Correspondence," *in press, Pattern Recognition*, 1998.
- [4] D. G. Luenberger, *Linear and Nonlinear Programming*, Addison-Wesley, 1984.
- [5] R. Meth and R. Chellappa, "Stability and Sensitivity of Topographic Features for SAR Target Characterization," *Proceedings, 2nd Annual ARL Federated Laboratory Technology Symposium*, pp. 188–192, Feb 1998.
- [6] R. Meth, R. Chellappa, and S. Kuttikkad, "Target Aspect Estimation from Single and Multi-pass SAR Images," *Proceedings, International Conference on Acoustics, Speech and Signal Processing*, pp. 2745–2748, May 1998.
- [7] A. Rangarajan, A. Yuille, S. Gold, and E. Mjolsness, "A convergence proof for the softassign quadratic assignment algorithm," in *Advances in Neural Information Processing Systems 9* (M. Mozer, M. Jordan, and T. Petsche, eds.), pp. 620–626, MIT Press, 1997.
- [8] T. W. Ryan and B. Egaas, "SAR Target Indexing with Hierarchical Distance Transforms," *Algorithms for Synthetic Aperture Radar Imagery III*, Edmund G. Zelnio, Robert J. Douglass, Editors, *Proc. SPIE*, Vol. 2757, pp. 243–252, April 1996.