

Feature Matching from SAR Arctic Data Using Neural Networks

P. E. Silveira, M. Van Dyne and C. Tsatsoulis.

Department of Electrical Engineering and Computer Science.
The University of Kansas, Lawrence, KS 66044.

ABSTRACT

Matching sea ice features consists of recognizing the same features on ice in two different images. Feature matching can be used as a tool in different applications, like ice motion tracking and assisting on feature classification by checking its consistency. The problem described consists on using neural networks to classify different features and decide how likely they are to be the same. A backpropagation neural network is trained to determine how closely different features match, considering possible deformations. The input to the network is the contour of each feature. Different input patterns are studied and a pixel grid representation of the features contour is shown to yield better results than a chain code representation. A database containing ice features from different images is used as an implementation tool for extracting the desired data. The efficiency of using such a database for extracting data from sea ice features is shown and some details are given about its implementation. The effectiveness of the neural network in matching features in different images over time is demonstrated and its applications on tracking ice motion and assisting on features classification is explained.

INTRODUCTION

Given the importance of the Arctic region to the northern hemisphere's climate and maritime navigation, synthetic aperture radar (SAR) satellites were put in orbit to monitor this region and collect data. As a

consequence, a demand for new applications that would utilize this data developed. The movement of ice floes was a phenomenon of special interest. This movement causes the exposure of large areas of sea water to the cold Arctic air and the heat transfer that occurs as a consequence of this exposure has great influence on global weather. Therefore, the development of a method for tracking the sea ice motion is of great significance. It is in an attempt to solve this problem that this approach, using neural networks, was created.

The previous existing methods for detecting ice motion consisted of either parametrizing the ice features shape and finding the correlation between two images [1]-[2], using probability distributions to track sea ice [3], or tracking the correlations between two images [4]. The advantage of the neural network method is that once the network is trained, feature matching is performed very fast. Another advantage of using a neural network compared to statistical methods is because of the network's flexible pattern recognition ability, which allows it to detect differences in shape which usually would not be detected by conventional numerical methods. After feature matching is performed the database can be used to provide precise information about the feature movement and distortion.

Finally, feature matching can be used not only as a tool to track ice motion, but also as an auxiliary tool on feature classification by verifying the consistency of the classification.

FEATURE MATCHING

After being collected and pre-processed,

the SAR satellite images from the Arctic region are analyzed by a feature classification program which detects significant features in the image and extracts them. These features are described using a chain code representation, which is used to trace the features contour and represent them by a series of numbers. The chain code representation consists of tracing the pixels which compose the image, starting at an initial pixel, and then using a number to describe the direction of the next point. The direction described by each number in the chain code is shown in figure 1.

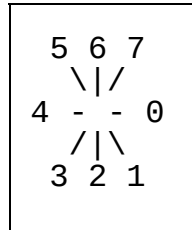


Figure 1:
Chain codes

Besides generating the feature chain code, the classification program also calculates several other parameters from each feature, including their centroid, orientation, area, perimeter, circularity and wiggleness (i.e., how wiggled it is). All this information is stored in a relational database created with the purpose of facilitating the organization of information about the sea ice images and assisting in its access. It was implemented with the use of "Ingres", a public domain database management system, and the EqueL language, a query language that was used imbedded into C programs.

A back-propagation neural network algorithm was used for feature matching, because of its well studied behavior and successful use in the past in other applications [5]. Several other parameters in the back propagation network had to be defined, starting by the format of the input. The first approach attempted was to feed the chain codes directly into the neural network. Unfortunately, this process proved not to be very efficient. Training was very difficult and the performance of the network was hard to predict. In order to solve this problem, a new format for the input data was sought. The

solution found was to use a pixel grid representation of the data. A pixel grid representation is simply a sequence of numbers representing the corresponding points in the image, where number 1 (one) represents a point from the features contour and number 0 (zero) represents no point (see figure 4 for an example).

This representation of the data was expected to yield better results due to the non-statistical nature of the neural networks. The chain code representation inhibits the network performance by translating shapes into numbers, what hides part of the information regarding the shape of the features and, therefore, makes the shapes harder to recognize. By transforming the numbers back into a shape, as is done in the pixel grid representation, the neural networks are expected to be able to function appropriately once more. The assumption regarding the performance of the pixel grid representation was correct. Using this representation, not only the behavior of the network became more stable, but the learning process also became much faster.

The features stored in the database were represented by their chain codes. Therefore, it was necessary to convert the chain codes to pixel grids. However, this represented a problem. The neural networks function better when a fixed matrix of points is used as inputs, but the sea ice features vary greatly in shape and size. It was necessary to create a process to standardize them, making it possible to represent them in a standard pixel grid. It was trying to solve this problem that the chain code reduction program was created. It consisted of a C program that read the desired chain code and then reduced it to an equivalent smaller chain code, but with a standard size. Then it converted this chain code to a standard pixel grid. This way all features would be represented by a chain code of the same length and, consequently, by a pixel grid of the same dimensions.

However, this approach proved to be unsatisfactory, since a lot of detail was lost in the chain code reduction. Therefore, it was decided not to execute the chain code reduction. Instead, a larger grid is used to represent features in their original size.

The input features are be up to 55 pixels wide and 55 pixels tall. This results in an input layer of 6050 nodes (55 pixels X 55 pixels X 2 features). The output layer contains only one node, with an output value that ranges from zero (no matching) to one (perfect matching). The size of the middle layer was defined empirically to be composed of 20 nodes. It was decided to choose the minimum number of nodes that would yield satisfactory results.

The network was trained with features from images from the same regions, taken 3 days apart. The training set consists of 15 pairs of features, some of which do match and some of which do not. After training was completed the performance of the network was tested, with satisfactory results. Figures 2 and 3 show two images of the same feature. Figure 4 show their grid representation. When these features are input in the network, the output

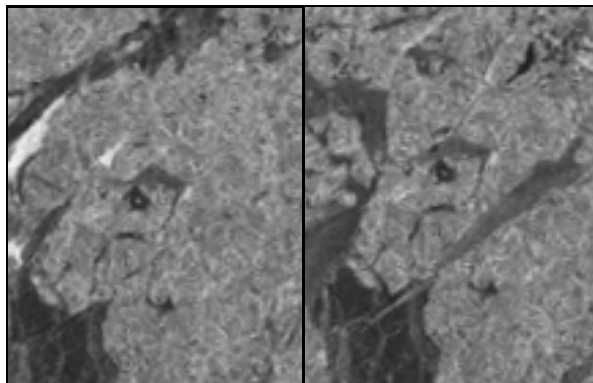


Figure 3: image of sample feature **Figure 2:** image of sample feature

produced is 0.981, indicating a close match, as expected.

APPLICATIONS

Notice that once the features have been matched, the motion detection problem is trivial. Using the information in the database about the centroid and orientation of the features it is possible to measure exactly how much the feature translated, in what direction, and also to measure its rotation. In order to do that the network has to be integrated with the database. A similar integration has already been done during training, so further integration would be a simple task.

Finally, feature matching can be used as an aid in classifying the different features in an image. The network can be used to match the classified features in two different images from the same region a few days apart. Then the classification of each feature pair is analyzed. If extreme discrepancies are found then it is clear that an error occurred in the classification process. For example, if a feature is classified as open water in one image and as multi-year ice in the next, an error definitely occurred. A new classification would be required to determine the correct result.

CONCLUSION

The best configuration of neural networks for feature matching was researched. It was determined that a back-propagation neural network with 6050 input nodes, 20 middle layer nodes and 1 output node was adequate for the purpose of feature matching. The format of the input data was also studied

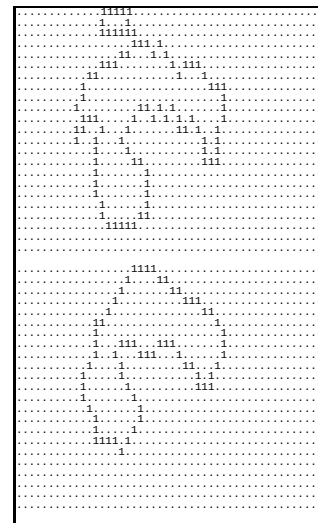


Figure 4: Grid pattern representation of features. Zeros were replaced with dots for better visualization.

and it was concluded that a pixel grid presentation exhibited a better network performance than simply a chain code representation. It was also decided that chain code reduction resulted in excessive loss of detail. It is better to use the complete feature as input, without reduction.

Finally, the network was shown to yield good results for two general input features. It was also described how the feature matching technique can be used in combination with the database to track ice motion and to aid on feature classification.

REFERENCES

- [1] R. McConnel, R. Kwok, J. C. Curlander, W. Kober, and S. S. Pang, " Ψ - $\mathcal{S}$ Correlation and Dynamic Time Warping: Two Methods for Tracking Ice Floes in SAR Images," IEEE Transactions on Geoscience and Remote Sensing, Vol. 29, No. 6, November 1991, pp. 1004-1012.
- [2] R. Kwok, J. Curlander, R. McConnel, and S. S. Pang, "An Ice-Motion Tracking System at the Alaska SAR Facility," IEEE Journal of Oceanic Engineering, Vol. 15, No. 1, January 1990.
- [3] J. Banfield, "Automated Tracking of Ice Floes: A Stochastic Approach," IEEE Transactions on Geoscience and Remote Sensing, Vol. 29, No. 6, Nov. 1991, pp. 905-911.
- [4] M. Fily and D. A. Rothrock, "Sea Ice Tracking by Nested Correlations," IEEE Transactions on Geoscience and Remote Sensing, Vol. GE-25, No. 5, September 1987, pp. 570-580.
- [5] J. Key, J. A. Maslanik, and A. J. Schweiger, "Classification of Merged AVHRR and SMRR Arctic Data with Neural Networks," Photogrammetric Engineering and Remote Sensing, Vol. 55, September 1989, pp. 1331-1338.