

Autonomous Agent Navigation Based on Textural Analysis

Rand C. Chandler, A. A. Arroyo, M. Nechyba, E.M. Schwartz

Machine Intelligence Laboratory

Dept. of Electrical Engineering

University of Florida, USA

Tel. (352) 392-6605

Email: rand@mil.ufl.edu arroyo@mil.ufl.edu nechyba@mil.ufl.edu ems@mil.ufl.edu

URL: <http://www.mil.ufl.edu>

2002 Florida Conference on Recent Advances in Robotics
May 23-24, 2002, Florida International University

ABSTRACT

Navigating an autonomous agent outdoors can be a challenging task. As a basis for this research, the autonomous agent in question is that of an autonomous robotic lawnmower. Mowing a lawn can be a difficult, tedious, and sometimes hazardous chore for the human operator. The goal of this paper is to report on ongoing research of using the visual property of texture to design a system which gives an autonomous lawnmower robot the ability to mow a lawn similar to how a human would (in some type of pattern). Previous autonomous lawnmowers that have been designed, the LawnNibbler and LawnShark for example, rely on the principle of randomness to mow in a confined area. While this method will work, it will take an extraordinary long amount of time. This paper first presents an introduction into what texture is and how it can be used to perform navigation. A system for analyzing textures is then presented. Finally, the current state and future research of this project are then presented.

INTRODUCTION

The goal of this project is to design an autonomous lawnmower that safely and effectively mows an area typical of a homeowner's yard. The previous versions of autonomous robot lawnmowers that were designed relied on the principle of randomness to mow an area. Thus they had no realization where they had mowed or didn't mow. The premise was as follows: given a long period of time, eventually the vast majority of the mowing area would be cut. Obviously this process would take considerably longer than if a human had mowed the same area, but the point was (and still is) that the human did not have to perform the task.

The main effort of this research is to design a robotic lawnmower that mows similar in fashion to how a human would mow. This could be mowing in a plow type fashion (going back and forth) or starting at the outside of the perimeter of the mowing area and then working towards the center of the mowing area. In order for the mower to determine where the mower has cut and not cut, computer vision with image processing techniques will be used. Instead of attempting to identify specific objects in the mowing area (trees, hedges, sidewalks, etc.) the image will be segmented according to the different textures present in the image. Once the image has been segmented, each one of the textures present in the image can then be classified. The mower would then be able to track the boundary between the cut and the uncut lawn surface, thus giving it the ability to mow in a pattern.

To begin the process of texture analysis, the wavelet transform will be used. Features will then be extracted from the resulting wavelet data. These "features" will then be clustered, resulting in a segmented image. Finally, the clusters will then be classified in order to determine what they are.

METRIC OF TEXTURE

Texture is an important quality to consider when examining the contents of an image. Practically every object (natural or man made) contains some texture on the macroscopic level. Thus, the use of texture information would be a practical means of segmenting objects in an image.

It would also be highly infeasible and impractical to recognize every object in the mowing area. These objects could be such things as trees, shrubs, sidewalks, flowers, etc. Being able to

correctly identify just one object could possibly take seconds to compute, let alone several objects. Also, it is not important to be able to recognize objects by being able to identify every object distinctly, but so long as the robot is capable of knowing what is grass and what isn't is enough to enable it to accomplish the goal of mowing as a human would.

Thus, the property of texture was chosen to accomplish this task. Figure 1 shows some examples of different textures. They are all from the Brodatz texture database. Figure 2 shows a sample result of texture segmentation. This is what an "ideal" segmentation would look like.

Once the objects in an image are segmented based on their texture, every pixel in the image can be assigned a label indicating which region it belongs. Furthermore, each texture in the image can then be compared against a database in order to recognize what type of texture it is. This can be thought of as "texture classification." As stated earlier, the main texture of interest for this research is that for the grass in the mowing area. If we can correctly classify the cut and uncut textures, then we can track the boundary between the two regions and thus mow in a pattern.

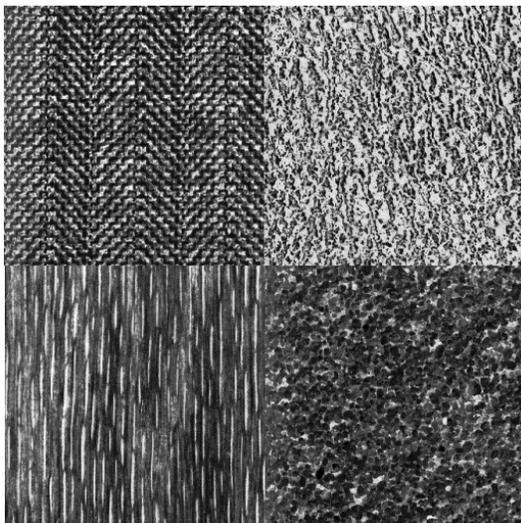


Figure 1: Some sample Brodatz textures. From upper left going clockwise, D17: Herringbone weave, D24: Pressed calf leather, D68: Wood grain, D29: Beach sand.

So far, this section has focused on segmenting an image based on texture information alone. However, how does one represent texture? Texture can be represented by using a statistical approach or by using a spatial/frequency

approach [1]. For this project, texture will be represented using a spatial/frequency approach by use of the wavelet transform.

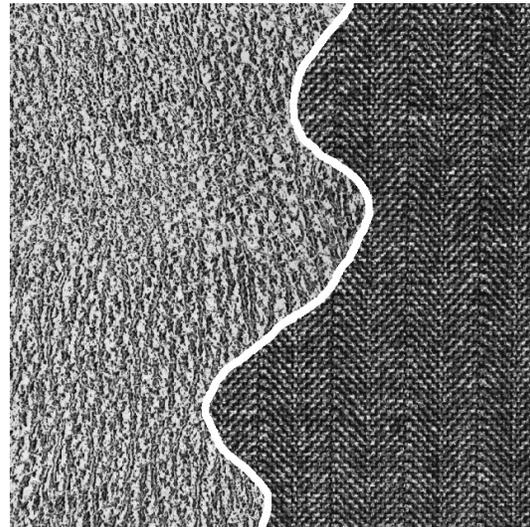


Figure 2: An example of texture segmentation.

TEXTURE ANALYSIS

As stated in the previous section, we will use the metric of texture to segment the input image. Figure 3 below shows a graphical breakdown of the proposed system.

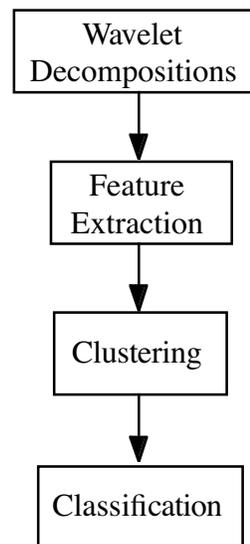


Figure 3: Graphical representation of the steps involved in the proposed research.

The first step involved in the process of textural analysis is the use of the wavelet transform. The wavelet transform breaks the input image into various wavelet sub-bands. Features are then

extracted from these sub-bands by means of the feature extraction stage. Thirdly, these features are then clustered to produce a segmented image. After clustering, the individual textures are then classified to determine what class the texture belongs (i.e. grass, concrete, etc.). What follows is an in-depth discussion about each of these steps in process of texture analysis.

Wavelet Transforms

Wavelet transforms can be thought of as a multi-resolution decomposition (multiple scale signal view) of a signal into a set of independent spatially oriented frequency channels [2][3]. The theory behind wavelets is quite complex. However, the wavelet can be simply thought of as a band-pass filter. To begin a wavelet analysis, one starts with a *prototype* wavelet, it is the “mother wavelet” from which all other wavelets are constructed. High frequency (contracted) versions of the prototype wavelet are used for fine temporal analysis while low frequency (dilated) versions are used for fine frequency analysis [3].

For this research, the discrete wavelet transform (DWT) will be used. To compute the (one-dimensional) DWT, a input vector x (length $2n$) is first convolved with some discrete time low-pass filter (LPF) of some given length and then it is convolved with some discrete time high-pass filter (HPF) of some given length. Figure 4 shows the first step involved in computing the DWT: low-pass filtering on the input vector x .

As can be seen, the filter is moved in increments of two along the length of the input vector, x . This is equivalent to down-sampling by a factor of two. The resultant vector produced is half as long as the original input signal x . Next, as shown in Figure 5, the input vector x is then convolved with a discrete time HPF in the same manner as the LPF.

However, the result of these convolutions are placed in the second half of the w vector. This is considered as a first level wavelet decomposition. This process can be continued again and again, each time working with the previous generated vector as the input to the next level. This process is illustrated in Figure 6. From the analysis point of view, the lowest frequencies will be isolated in the first four coefficients of the third level DWT. Moderate frequencies will be isolated in the mid-portion of

the transform (namely coefficients 5 through 8). High frequencies are then isolated in the right half of the transform. Because each level decreases the length of the workable vector by a factor of two, this process cannot be continued forever.

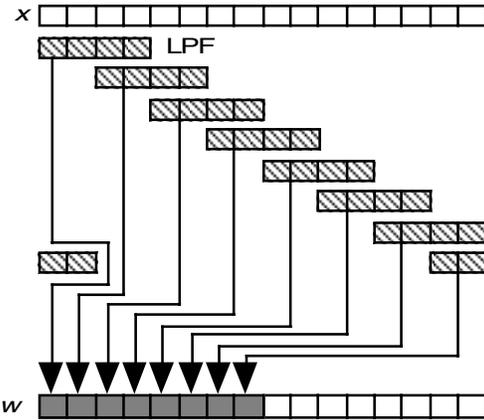


Figure 4: First step in computing the 1-d wavelet transform: low pass filtering (in increments of two) on an input signal x .

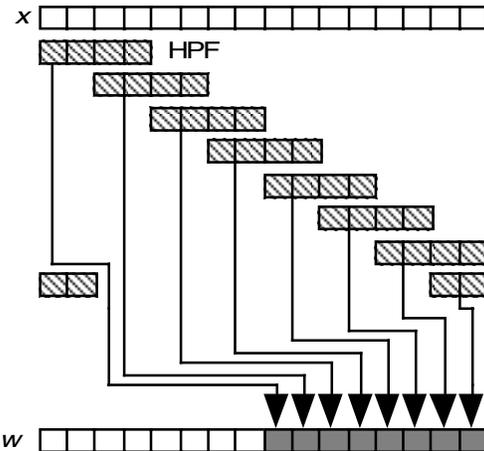


Figure 5: Second step in computing the 1-d wavelet transform: high pass filtering (in increments of two) on an input signal x .

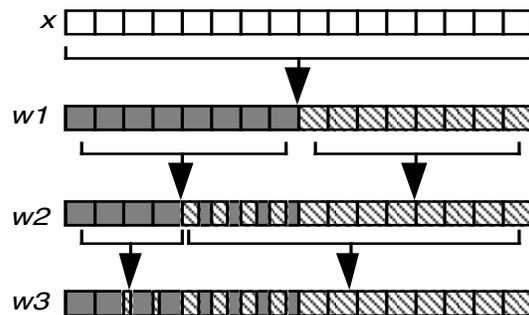


Figure 6: Three level wavelet transform

Another way to convolve the input vector with the discrete time filters is to not sub-sample the output at every level. This is known as an over-complete wavelet decomposition [4]. This has tremendous advantages when wavelet transforms are used for texture analysis. When performing texture analysis, an algorithm that is invariant due to translations in the input image would be highly desirable. Shifting of the input signal (as is done with the method stated earlier) will result in a modification of the wavelet transform [4]. By performing an over-complete wavelet transform, this should eliminate this problem [1].

All of the material on wavelet transforms up to this point has been for the one dimensional case. However, when dealing with images (or higher dimensions) a tensor product extension can be used [4]. A tensor product extension can be thought of as a cross-product. Thus, four distinct types of 2-d filters (basis functions) result from the cross-products of the high and low pass filters: LL, LH, HL, HH (where LL = low-pass low-pass, LH = low-pass high-pass, etc.). As shown in Figure 7 below, a 2-d wavelet decomposition can be carried out by means of successive one-dimensional processing along the rows and columns of the image. For example, if one wanted to compute the LH basis function, one would process the rows of the image with the low-pass filter and then process the columns of the result with the high-pass filter.

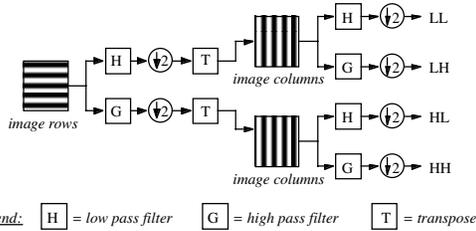


Figure 7: Computing the 2-D wavelet transform

Feature Extraction

Feature extraction is an essential part of the texture analysis process. This is due to the fact that the data (wavelet coefficients) that is generated as a result of the wavelet transform is unsuitable for use in clustering algorithms [1][5][6][7]. Thus, the feature extraction process can be viewed as an interface layer between the wavelet transform and clustering algorithm. Whichever method we choose to perform the feature extraction process, it should produce

“features” that are acceptable to the chosen classifier (in the classification stage) but not at the expense of decreasing the information content of the features themselves [6]. In addition, we would expect the feature extraction algorithm to be consistent among the pixels within a class but possess a high degree of discernability between classes.

Typically the feature extraction process usually involves the following steps: (1) filtering, (2) application of a nonlinear operator, and (3) application of a smoothing operator [8][9]. To illustrate the process of feature extraction, we present the following example. For the purposes of this example, a “synthetic” texture was generated as can be seen in Figure 8(a). This texture was created by appending two sinusoids of different frequency. Specifically, the sinusoid on the left is of low frequency and the one on the right is of high frequency. Figure 8(b) shows a horizontal slice through the image showing the two sinusoids.

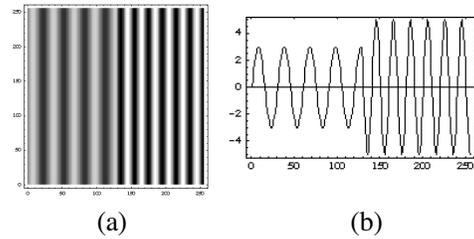


Figure 8: (a) Synthesized texture generated by appending two sinusoids. (b) horizontal slice of the synthesized texture in (a).

As stated above, the first step in the feature extraction process is filtering. Figure 9 below shows the result of applying a high pass filter to the synthesized texture shown in Figure 8. As can be seen, this result is still unsuitable for clustering.

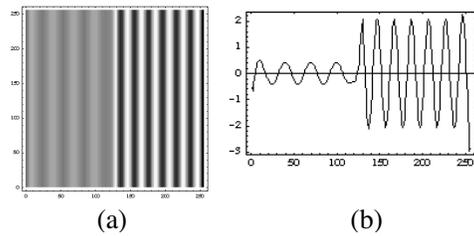


Figure 9: (a) Filtered result of the synthesized texture shown in Figure 8(a). (b) horizontal slice of the filtered result shown in (a).

The next step in the feature extraction process is to apply a non-linear operator to the resulting

filtered data. This usually means a squaring or taking the magnitude of the data. The problem with this approach is that this process usually does not produce results that are acceptable for clustering. To overcome this, we employ the use of an envelope (or peak) detector. Figure 10 below shows the results of applying an envelope detector to one horizontal line of the filtered results presented in Figure 9.

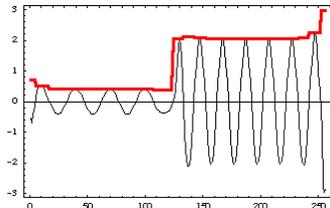


Figure 10: Result (solid line above wave peaks) of applying an envelope detector to the data presented in Figure 9.

When the envelope detector is applied to every row in the image, the following result is obtained as shown in Figure 11. As can be seen, this data would be suitable for data clustering. Obviously, the result one gets by the application of an envelope detector depends on the orientation that the envelope was applied. In the example just presented, the envelope detector was applied row-wise to the filtered data.

The envelope detector would be applied differently depending on the “features” present in the data. This is especially true of the results obtained by means of the wavelet transform where the different sub-bands isolate different features in the image based on their spatial orientation [10]. To illustrate this point, Figure 12(a) shows an image of two different textures merged together. Figure 12(b) shows the result of taking the level 1, LH wavelet sub-band of the image shown in Figure 12(a). This is the “filtering” step. Finally, Figure 12(c) shows the result of applying a column-wise envelope detector to the LH wavelet sub-band. As can be seen in Figure 12(c), this result is more suitable for clustering.

Clustering and Classification

In this sub-section, we discuss the stages of clustering and classification together since they are highly related to one another. Clustering is the process of finding natural groupings within a set of data. The data contained within a particular group (or cluster) should possess a high degree of similarity to one another [11]. For instance,

when talking about an image, one might want to cluster the data based on the different colors of the objects present in the image. With respect to this research, our goal at this stage is to cluster the data generated by the feature extraction phase. Clustering the data from the feature extraction phase will result in a segmented image based on the different textures present in the image.

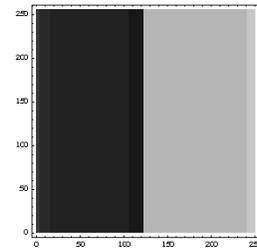


Figure 11: Result after the application of the envelope detector to all of the rows of the filtered data presented in Figure 8.

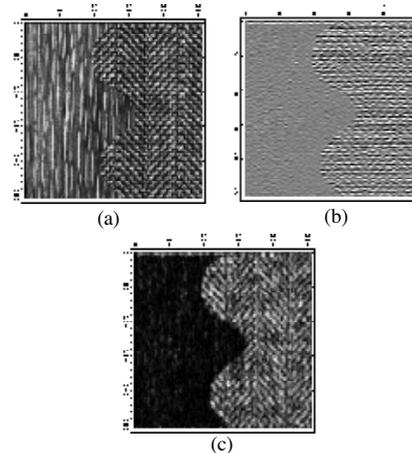


Figure 12: (a) Two textures merged together. (b) Level 1, LH wavelet sub-band of (a). (c) result after applying the envelope detector column-wise to (b).

To perform the clustering phase of this research, three methods are currently being investigated: (1) Vector Quantization (VQ), (2) Representing the data via a single Gaussian, and (3) Using a mixture of Gaussians to model the data. All of these methods are methods used in statistical modeling. In statistical modeling, our goal is to “learn” the statistical properties of a distribution of data so that the input data can be mapped to probabilities [12]. These methods also have the effect of reducing the size of the data set we will ultimately be using for classification purposes. For example, instead of a cluster of data being represented by the actual data contained within the cluster, the entire cluster can be represented

via the mean and co-variance matrix associated with fitting the data to a single Gaussian.

The process of classification (or pattern classification) is one of assigning input data to one of a finite number of categories [11]. In order to classify an object, one needs to obtain training data in order to have something to compare the unknown data against. Once one has collected training data, features are then extracted and then clustered from this training data in the same manner as the unknown data. To determine the class to which a particular cluster belongs, we evaluate the probability of the unknown data (cluster) belonging to each of the known clusters generated during the training phase. The unknown data is then assigned to the class having the highest probability given the statistical model of the training data.

CURRENT WORK

As of this writing, work has focused on implementing the feature extraction and clustering phases of this research project. The vector quantization method is currently being implemented in software. The other two methods mentioned in the previous section will also be implemented in order to choose the method that works most effectively.

FUTURE WORK

After the clustering stage of this research is complete, work on the classification stage will then begin. When all of the various stages are complete, they will all be integrated into a complete system. The systems' main goal will be to track the boundary between the cut and uncut lawn surface (although, it could easily be adapted to perform such tasks as path following, e.g. following a concrete sidewalk). Due to the hazardous nature and other issues involved with operating an autonomous lawnmower with a metal blade, a push mower mounted with a camera will be used for the collection of test data. This data will be fed into a computer via a frame-grabber card where the complete algorithm will be applied to the data. After these initial experiments, it will then be possible to conduct trials on a real platform.

REFERENCES

[1] A. Laine, J. Fan. "An Adaptive Approach for Texture Segmentation by

Multi-channel Wavelet Frames", University of Florida, Center for Computer Vision and Visualization, Aug. 1993.

[2] K.W. Abyoto, S.J. Wirdjosoedirdjo, T. Wantanabe. "Unsupervised Texture Segmentation Using Multiresolution Analysis for Feature Extraction"

東京情報大学研究論集

Vol. 2, No. 1, 1998, 49-61.

[3] M. Vetterli. "Wavelets and Filter Banks: Theory and Design" IEEE Trans. on Signal Processing, Vol. 40, No. 9, Sept. 1992, 2207-2232.

[4] M. Unser. "Texture Classification and Segmentation Using Wavelet Frames" IEEE Trans. On Image Processing, Vol. 4, No. 11, Nov. 1995, 1549-1560.

[5] A. Laine, J. Fan. "Frame Representation for Texture Separation" Center for Vision and Visualization, University of Florida.

[6] B. W. Ng and A. Bouzerdoum "Supervised Texture Segmentation using DWT and a Modified K-NN Classifier," IEEE 15th International Conference on Pattern Recognition, Volume 2, pp. 545-548, 2000.

[7] B. Wang, Y. Motomura, and A. Ono "Texture Segmentation Algorithm Using Multichannel Wavelet Frame," IEEE 1997, pp. 2527-2532.

[8] T. Randen and J. H. Husoy "Filtering for Texture Classification: A Comparative Study," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 21, No. 4, pp. 291-310, Apr. 1999.

[9] T. Randen and J. H. Husoy "Texture Segmentation Using Filters with Optimized Energy Separation," IEEE Transactions on Image Processing, Vol. 8, No. 4, pp. 571-582, Apr. 1999.

[10] M. Nechyba. "Brief Introduction to Wavelets" Class notes, EEL6668, Fall 2000, University of Florida.

[11] R.O. Duda and P.E. Hart, "Pattern Classification and Scene Analysis," J. Wiley and Sons, Inc., 1973.

[12] M. Nechyba. "Introduction to Statistical Modeling" Class notes, EEL6668, Fall 2000, University of Florida.