The adaptive chirplet: an adaptive generalized wavelet–like transform

Steve Mann and Simon Haykin

McMaster University, Communications Research Laboratory
Hamilton, Ontario, L8S 4K1; manns@McMaster.CA

ABSTRACT

We propose a new distance metric for a Radial Basis Functions (RBF) neural network. We consider a two dimensional space of time and frequency. In the usual context of RBF, a two dimensional space would imply a two dimensional feature vector. In our paradigm, however, the input feature vector may be of any length, and is typically a time series (say, 512 samples).

We also propose a rule for positioning the *centers* in Time–Frequency (TF) space, which is based on the well–known Expectation Maximization (EM) algorithm. Our algorithm, for which we have coined the term "Logon Expectation Maximization" (LEM) adapts a number of *centers* in TF space in such a way as to fit the input *distribution*. We propose two variants, LEM1 which works in one dimension at a time, and LEM2, which works in both dimensions simultaneously.

We allow these "circles" (somewhat circular TF contours) to move around in the two dimensional space, but we also allow them to dilate into ellipses of arbitrary aspect ratio. We then have a generalization which embodies both the Weyl–Hiesenberg (eg. sliding window FFT) and affine (eg. wavelet) spaces as special cases.

Later we allow the "ellipses" to adaptively "tilt". (In other words we allow the time series associated with each *center* to chirp, hence the name "chirplet transform".)

It is possible to view the process in a different space, for which we have coined the term "bowtie" space. In that space, the adaptivity appears as a number of ⋈ shaped *centers* which also move about to fit the input distribution in this new space.

We use our chirplet transform for Time–Frequency analysis of Doppler radar signals. The chirplet essentially embodies a constant acceleration physical model. This model almost perfectly matches the physics of constant force, constant mass objects (such as cars, with fixed throttle, starting off at a stoplight).

Our transform resolves general targets (those undergoing non–constant acceleration) better than the classical Fourier Doppler periodogram.[1,2] Since it embodies the constant velocity (Doppler periodogram) as a special case, its extra degrees of freedom better capture the physics of moving objects than does classical Fourier processing.

By making the transform adaptive, we may better represent the signal with fewer transform coefficients.

1 TIME–FREQUENCY SPACE

In 1945, Gabor proposed an expansion of a signal onto a set of bases which he illustrated by way of "logons", which are portions of a time–frequency (TF) space[100]. He showed that an arbitrary signal can be decomposed onto a set of functions which are all just modulated versions of a single Gaussian envelope. This notion is known formally in physics literature as the Weyl Hiesenberg group. In electrical engineering literature, what we refer to as the *sliding window Fourier transform* belongs to the Weyl Hiesenberg group since we can think of the bases as being modulated versions of one *parent window*. Gabor emphasized the use of a Gaussian window, since it minimizes the uncertainty product $\Delta t \Delta f$ (provided that these support measures are quantified in terms of the root mean square deviations from their mean epochs[4]). The time–frequency logon diagrams, depicted in figure 1 show how the *Gabor function* bases cover the time–frequency space, and how we can trade frequency resolution for improved temporal resolution. Gabor originally used rectangles to designate each of these *elementary signals*. If each of these rectangles were a pixel, and its brightness

[100] Although the logon representation is not a "frame",[3] and is therefore not numerically stable, we proceed in the development of the line of thought, and will later address the issue of numerical stability, by *over–representing* the expansion.

0-8194-0693-7/91/$4.00

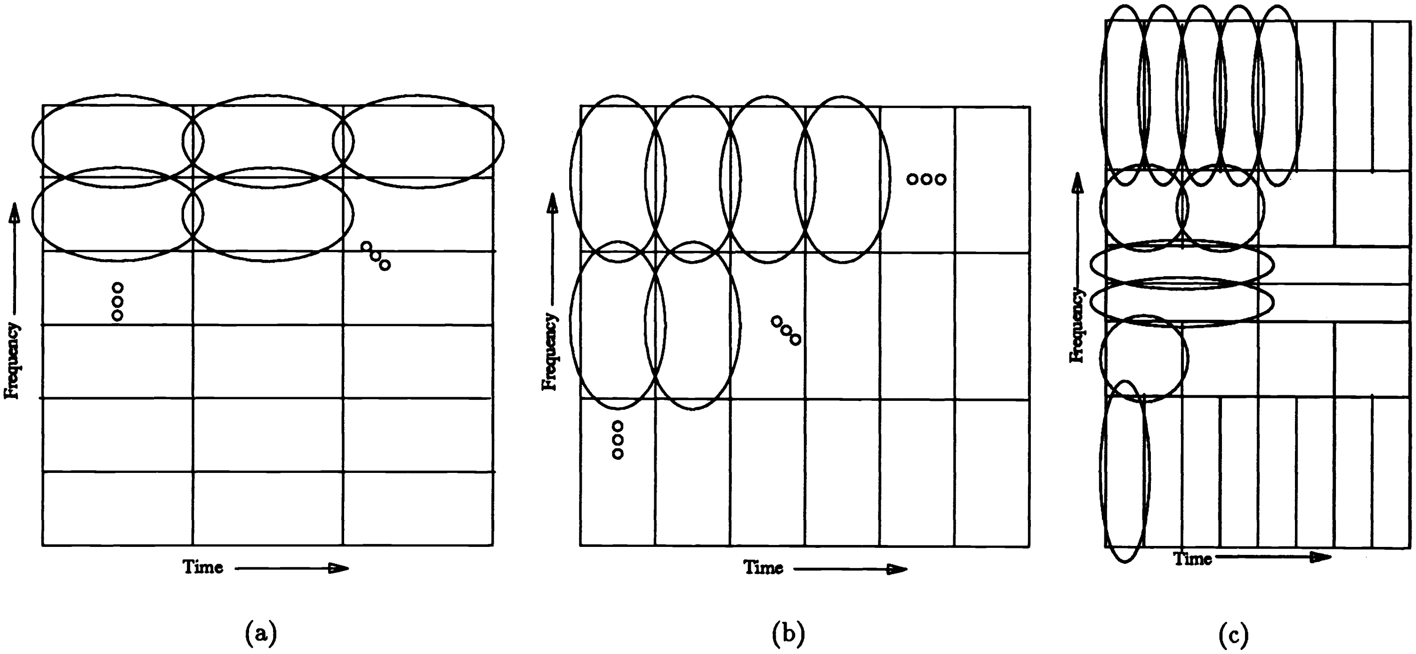


Figure 1: Comparison between Gabor's fixed scale, expansion onto a set of *elementary signals*, and the wavelet transform, in terms of TF spaces. (a) "Complete" set of Gabor bases, each of which is a tone packet of relatively long duration and narrow bandwidth. (b) A different set of bases where we have traded off good frequency resolution to get good temporal resolution.

(c) Logon representation corresponding to a wavelet transform. The affine bases manifest themselves as *constant Q* filters in TF space. The aspect ratio is governed by the distance from the $f = 0$ line running across the center of the space.

was set in accordance with the appropriate coefficient in the signal expansion (for example by replacing the rectangle with a dither pattern), the logon diagram would be a density plot (image) of the TF distribution.

Within the logon paradigm, the wavelet transform may be thought of as breaking up the signal into a set of affine–shaped bases, as illustrated in figure 1 (c). In all our TF figures we show frequency on a linear scale. Thus we can readily see that the aspect ratios of the bases are proportional to the distances from the $f = 0$ axis. The aspect ratios are still constrained, but the constraint is different. Instead of the constant size constraint of the Weyl–Hiesenberg type expansion, we have a constant "Q" (ratio of bandwidth to center frequency) constraint.

1.1 Distinction between "wavelets" and wavelets

In our development of a generalized Logon transform, we first removed the constraints of having either a constant aspect ratio or constant Q. We expanded arbitrarily onto a basis of "wavelets" of varying shape, with the only constraint being that each "wavelet" be a particular class of functions, for example, a Gabor function.

In the past, the term "wavelet" often denoted any arbitrary basis function which acted as a bandpass filter (when used as a convolutional kernel). In more recent literature, however, the term *wavelet* has been used to denote a basis function of constant shape (from an affine group of translates and dilates of one *mother wavelet*). Indeed, in the earlier papers, the term "wavelets of constant shape"[5] was often used to denote the affine basis explicitly. Whenever the word "wavelet" appears in quotes, we mean "wavelet" in the wide sense: any ephemeral burst of energy, finite in physical support (which includes bases in both the Weyl-Hiesenberg and the affine spaces).

2 LEM: EXPECTATION MAXIMIZATION (EM) IN TIME–FREQUENCY SPACE

We propose an adaptive generalized wavelet–like transform, a signal dependent Time–Frequency method for which we have coined the term "Logon Expectation Maximization" (LEM). It is based on Expectation Maximization. (The well known EM algorithm essentially fits a mixture distribution by a sum of Gaussian (or other) random processes.)

We fit the time–frequency distribution by a number of translates and dilates of some scaling function in time frequency space. For each *center* of the scaling function in this joint space there is a corresponding "wavelet" in the physical (time) domain. Thus we have abstracted each adaptive "wavelet" by looking at it as a scaling function in the time–frequency ***distribution*** space.

Using this adaptive "wavelet" paradigm, we were able to represent radar signals in a very compact form.

The use of our LEM paradigm leads to a novel variant of the Radial Basis Function (RBF) neural network classifier. Rather than classifying based on Mahalanobis distance from the *centers* as in conventional RBF networks, we classify based on the receptive field output in time–frequency space. Thus we treat each input time series, not as a feature vector which lies somewhere in a Probability Density Function (PDF), but rather we treat the time series by how it lies in the time–frequency space[101]. Using another interpretation, we may think of our paradigm as a new distance metric, which is inherently two dimensional regardless of the "dimension" of the time series. The time series is always a vector (an array of "index dimension" one), and is typically of very high "vector dimension" (say 512 or 1024).

The RBF network then becomes an interpolator in TF space.

In our case we are not interested in fitting a PDF, but rather we would like to approximate a TFD. Initially, suppose we wish to use only one *center* to fit the TF distribution of an arbitrary time series s. A simple selection of the center location (where a bivariate Gaussian TFD is implicitly being assumed) is as follows:

$$(t_c, f_c) = \left(\frac{< s|t|s >}{< s|s >}, \frac{< S|t|S >}{< S|S >}\right) \tag{1}$$

Where t_c and f_c are the coordinates of the center in time and frequency, and S is the Fourier transform of s, given by $S(f) =< exp(+j2\pi ft)|s(t) >$.

Note that by Plancherel's theorem (conservation of energy in the transform domain) the denominators in both coordinates of equation 1 are both equal to the L^2 norm (energy) of the signal g.

When more than one center is to be used to fit the distribution, we propose a variant of the EM algorithm. Duda and Hart,[7] chapter 6 is a good standard reference for this algorithm, although they do not explicitly refer to it as "Expectation Maximization". Another standard reference is Dempster *et al.*[8] Our variant of EM follows directly from Hinton, page 5.8,[9] from the section on Radial Basis Functions (RBF) neural networks, where he outlines three steps to fitting a distribution by a sum of Gaussians:

- For each data point, d, compute the probability density $p(d|i)$ for each unit, i, using the current mean and variance for that unit.
- Normalize these probability densities to get the probability that each Gaussian gave rise to each data point (the "blame" assigned to that Gaussian):
$$p(d \leftarrow i) = \frac{p(d|i)}{\sum_j p(d|j)} \tag{2}$$
- Using these normalized probabilities as weighting factors, compute a new mean and variance for each Gaussian. (ie find the maximum likelihood fit to the weighted data points.)

In our case, since we do not have discrete points, we use a batch process, on the whole distribution. We use the Gaussian TFD, because it is smooth in both time and frequency, and also has some other nice properties.

We let the centers be designated by the functions c_k, where $k = 1..K$. These centers may be thought of as functions of time, of frequency, or as somewhat elliptical blobs in TF space.

[101] We know that a probability function of time and frequency is disallowed,[6] but nevertheless we proceed, pretending we were not aware of this limitation (we consider ways to *approximate* PDF behavior in a time–frequency distribution).

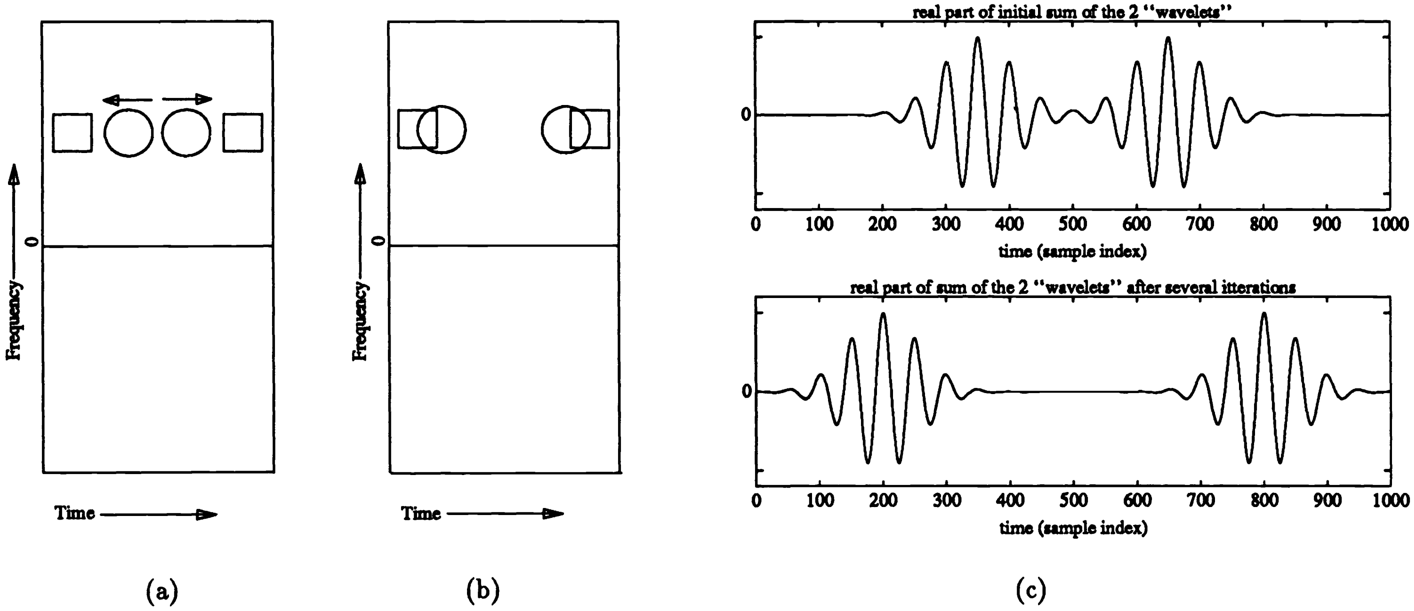


Figure 2: The energy distribution of the "wavelets" iteratively adapt to the energy distribution of the signal. In this simple example, we are trying to fit a signal which is a sum of two Slepian functions, by two Gabor "wavelets". (Recall that zero frequency is in the center, since we are using wavelets which are in Hardy space.) (a) View in TF space: initially the two *centers* are arbitrarily positioned. (b) View in TF space after the two "centers" are pulled onto temporal maxima (iteratively). (c) The same process, displayed in the time domain rather than the TF domain. Top subplot indicates initial guess corresponding to TF distribution shown in (a), lower subplot is after running LEM (corresponding to (b)).

We update the location of center k, iteratively, according to:

$$(t_k, f_k) \leftarrow \left(\frac{< sc_k|t|sc_k >}{< sc_k|sc_k >}, \frac{< SC_k|f|SC_k >}{< SC_k|SC_k >}\right) \tag{3}$$

Where, this time the denominators in each of the two coordinates are no longer equal; they represent the overlap in the energy content of the signal and the center in each of the two domains (time and frequency).

Figure 2 is a somewhat idealized illustration, showing how the centers migrate along the time axis only, as seen in both the Time–Frequency (TF) domain, and the time domain.

Likewise Figure 3 is a hypothetical illustration of how the centers move along the frequency axis only, to match the distribution in that direction.

By repeatedly applying equation 3, the centers would move to a local maximum in any TFD, moving around in the two dimensional space. If the signal, s has a distribution with distinct peaks, then the algorithm works quite well. If, however, there are two or more peaks very close together, then the convergence is very slow. There must also be some reasonable amount of disjointness in the starting values of the centers in both time and frequency.

In figure 4 we show an example, illustrating how the time and frequency update rules are applied together. The four subplots on the left are the real parts of the time domain signals corresponding to each of the four TFD contour plots on the right. The upper subplot pair is of the signal, two *RF pulses* (portions of sinusoids). The next subplot pair shows the initial starting guess of the *centers* (two Gabor functions). We can see that the *centers* move toward the peaks in the distribution, while the time domain signals look more and more like the desired function leading to as good a fit as can be expected using Gabor functions to approximate RF pulses. (Note that a perfect fit is impossible, given that we are trying to fit rectangular envelopes with Gaussian envelopes; the idea of LEM is simply to locate a "blob" of energy in the rough vicinity of the energy of the signal.)

If the initial values of the centers are chosen poorly, the convergence will be extremely slow. For example, if the centers ever lose their disjointness in either domain (time or frequency) they will become "locked" together in that

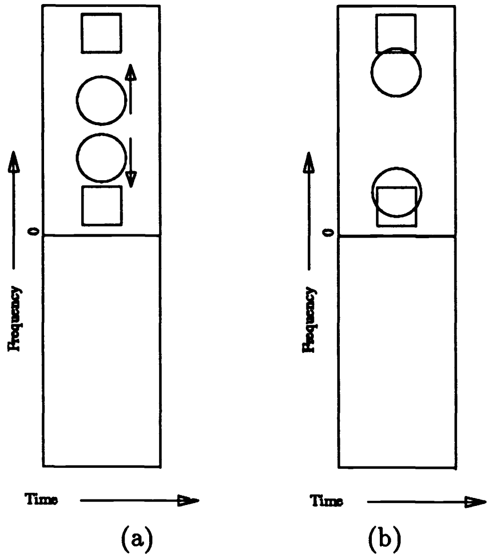


Figure 3: Even thought the signal contains two components "on top of one another" in time, they can be resolved by the adaptive "wavelets" since they are separated in frequency (disjoint in Fourier space). (a) Initially. (b) Eventually the two "centers" are pulled onto frequency maxima.

domain. Figure 5 illustrates what happens when the centers become "frequency–locked".

In the example of figure 4, if we had chosen the starting values so that the left *center* was lower in frequency than the right one, the *centers* would have become "locked" in at least one of the two domains. Therefore, we propose another algorithm, LEM2, which operates in the two dimensions simultaneously, rather than alternately. The coordinates for each center are updated from values computed from each "overlap center" defined by:

$$(\underline{C}_{kt}, \underline{C}_{kf}) = \frac{<\underline{S} \circ \underline{C}_k|(|1><t|, |f><1|)|\underline{S} \circ \underline{C}_k>}{<\underline{S} \circ \underline{C}_k|\underline{S} \circ \underline{C}_k>} \tag{4}$$

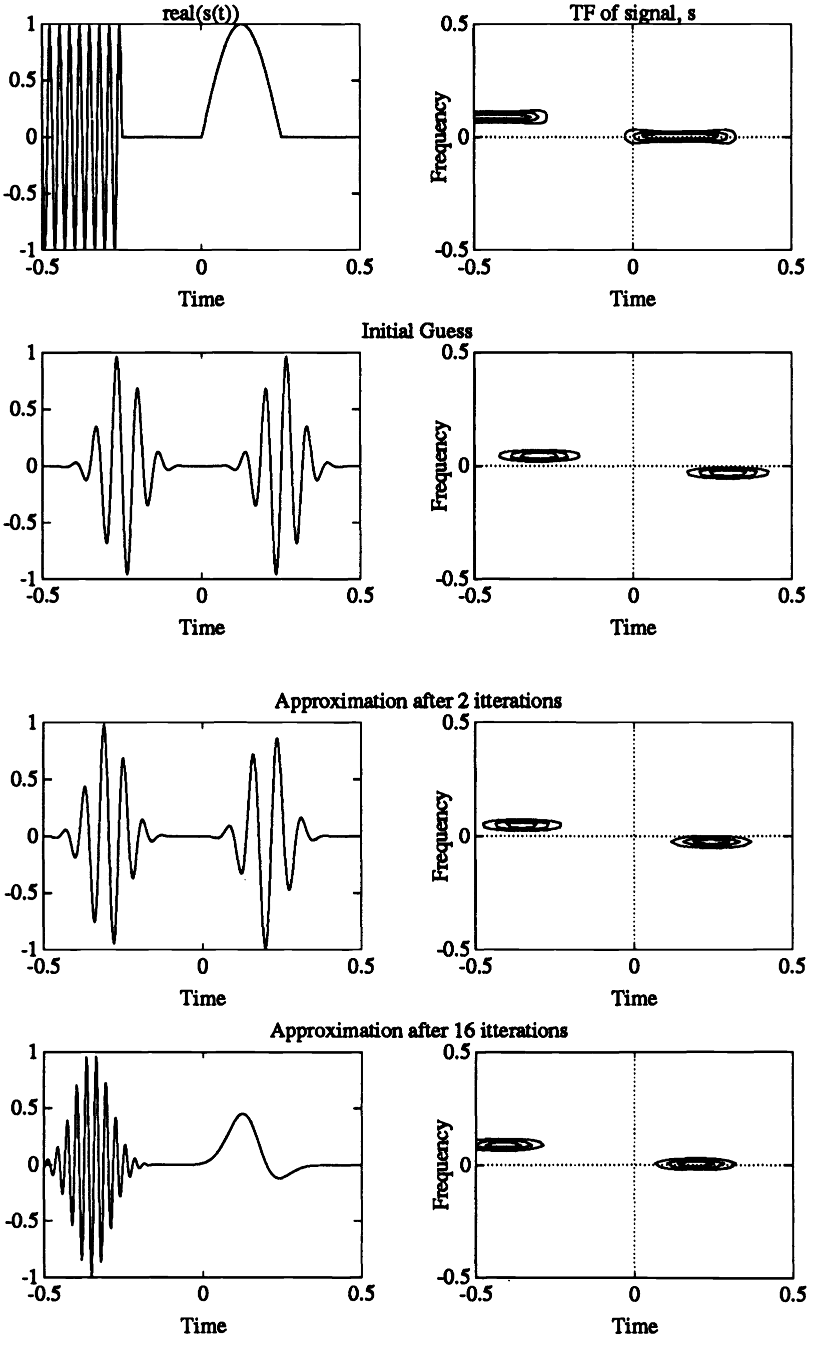
where underlined upper case characters indicate Time–Frequency distributions, which, for computational purposes, are matrices containing a discrete sampling of the actual TFDs, although, the theory holds for both continuous and discrete cases. $(|1><t|, |f><1|) = (T, F)$ represents the continuous or discrete outer product spaces of time and frequency. In the discrete case, the 1 is taken as a vector of ones, with the same length as t and f. In the continuous case, 1 is just a function of time, which has a constant value of one. In the discrete case, the symbol $\circ$ indicates the Hadamard product (also known as the Schur product, or simply element by element (component–wise) multiplication, as given by the symbols .* in matlab). In the continuous case, $\circ$ simply reduces to multiplication of functions of two variables.

Equation 1 is applied to TF distribution $\underline{C}_k$. The first moments of its energy distribution give the location of t_k and f_k.

In figure 6 we show how use of LEM2 breaks "locked" centers. Disjointness between *centers* is now only required in one domain.

The drawback of LEM2 is that it requires a number of Time–Frequency evaluations, but this drawback can be overcome by using the following strategy:

- select initial guess: position the centers at some starting points
- apply LEM2 with once with very coarse TF resolution
- if necessary apply LEM2 again with finer resolution in TF space ... (apply in coarse to fine manner if necessary)
- apply LEM1 at full resolution.



real(s(t))

TF of signal, s

Initial Guess

Approximation after 2 itterations

Approximation after 16 itterations

Figure 4: Using LEM to approximate two *RF pulses* with two Gabor functions

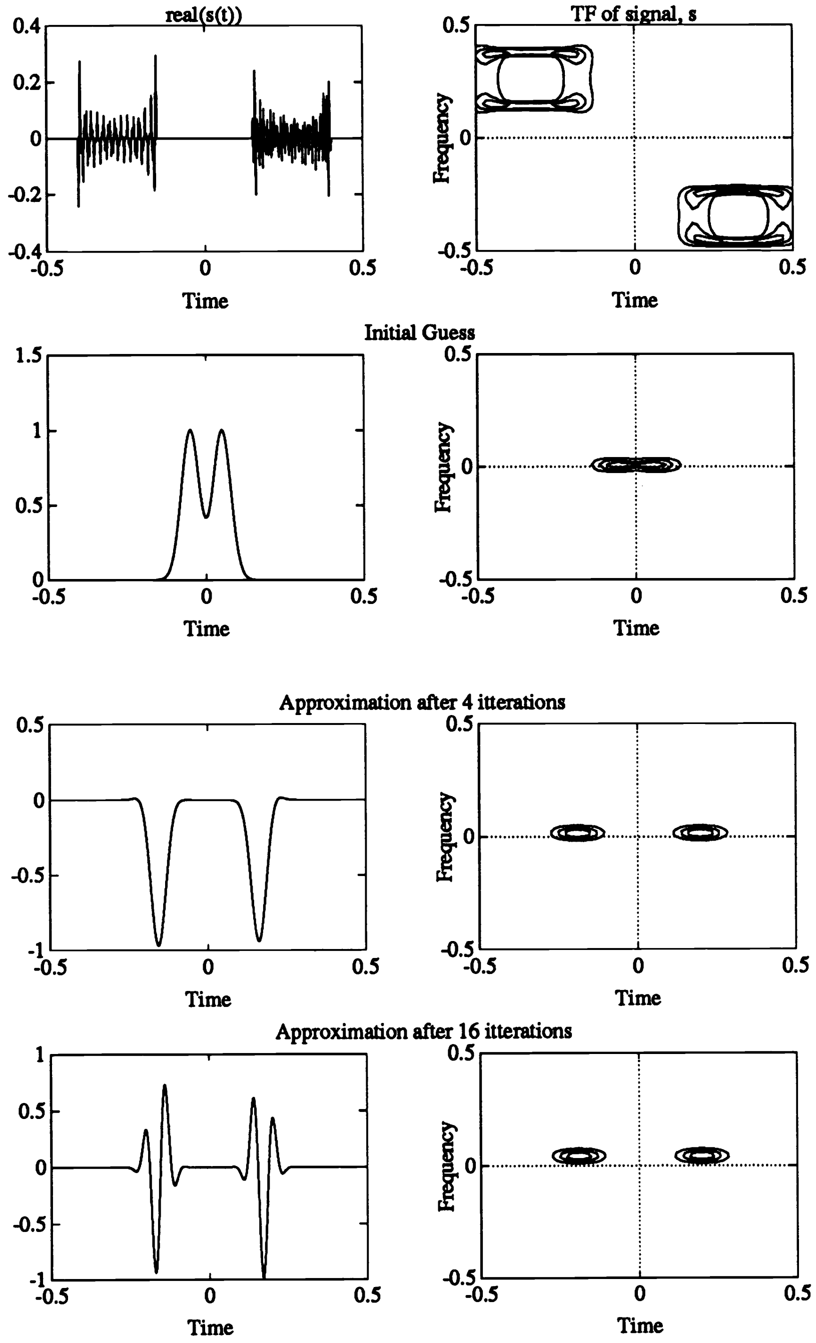


Figure 5: Here we illustrate "frequency–locking", a failure mode of LEM, by deliberately selecting starting values not disjoint in Fourier space.

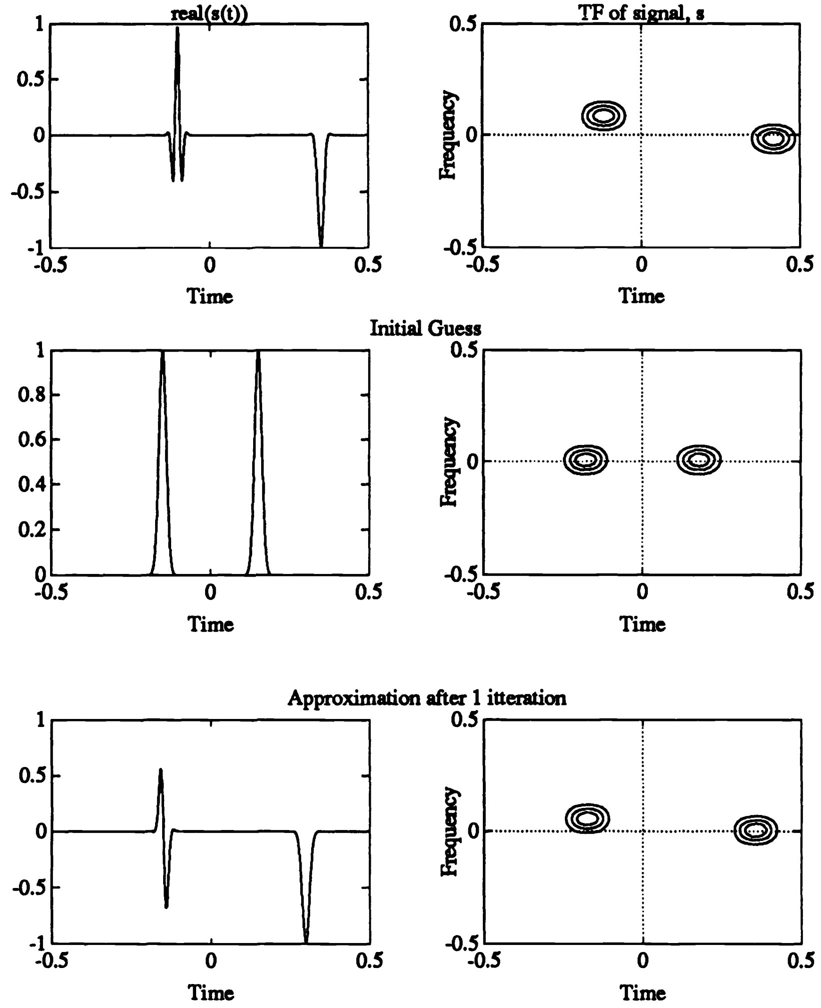


Figure 6: LEM2 should be used for the first one or two iterations to prevent loss of disjointness. Here we show how it can specifically break "lockup", allowing the *centers* to pass over each other in one domain if necessary.

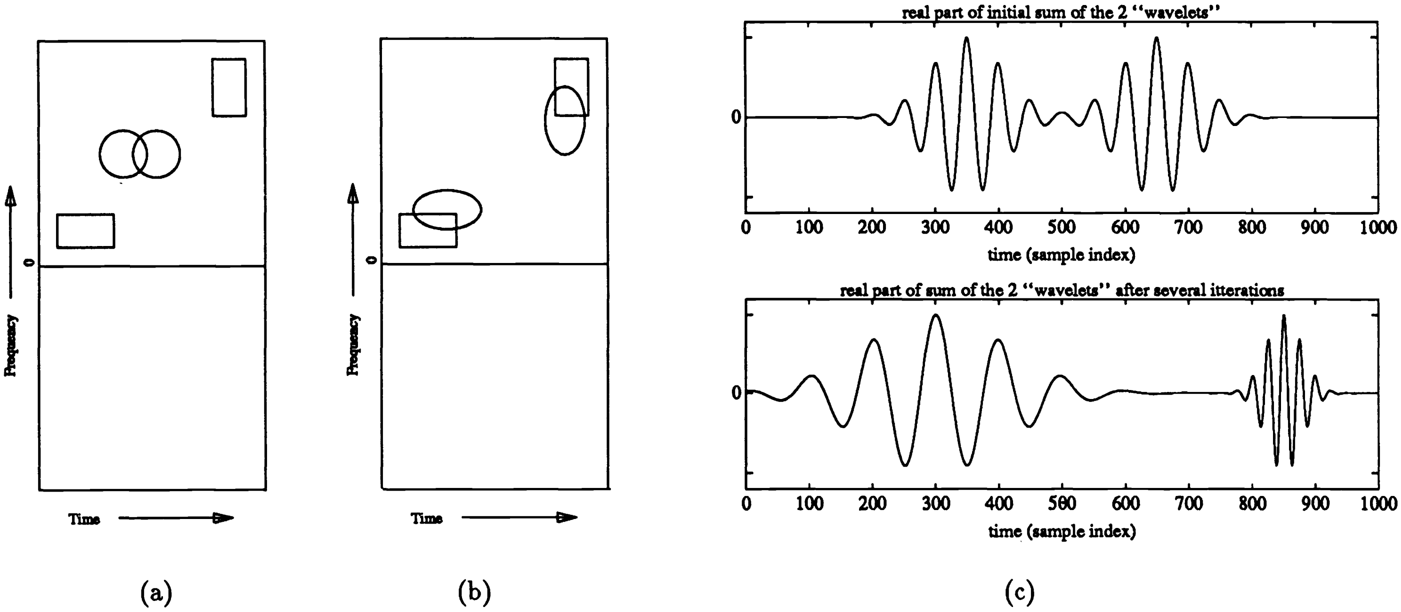


Figure 7: Here we have a mixed Weyl–Hiesenberg and affine space, where translation, dilation, and modulation are free to vary independently to match the input distribution. (a) Initial TF distribution (b) TF distribution after a few iterations: the *centers* translate and dilate (while preserving constant area) to match the input distribution. (c) Corresponding time domain representation: Top subplot shows time series of initial guess corresponding to the TF distribution in (a). Bottom subplot (corresponding to TF distribution in (b)) shows time series after running LEM.

Typically a single coarse run of LEM2 will be sufficient, although a coarse to fine strategy will also help to refine the initial estimate. For example, a 512 point time series may be started with LEM2 at 32 by 32 point resolution, followed by a few iterations of LEM1.

We may also wish to adapt the variances of the distributions. Unlike the general probability distributions, we desire to maintain a constant area logon. Thus, rather than changing the diameter of the circle, as is done in approximating probability density functions (Hinton[9]), we allow the aspect ratio to vary. Such a hypothetical example is illustrated, in TF space, in Figure 7, along with the corresponding time domain representation. These functions are neither affine wavelets (wavelets of constant shape) nor are they modulated versions of a single function as in the spectrogram basis. They are free to vary in both dilation, temporal center, and modulation. In the usual EM algorithm, the variances of the *centers*, in each of the two dimensions vary independently, but in LEM, the variances are governed by the "uncertainty relation". Thus in the full implementation of EM, each center has 4 degrees of freedom, while in LEM, each center only has three degrees of freedom.

The above results enable us to "learn" a time–frequency distribution. The signal s need not be fixed, but, rather, the system may be presented with a series of signals $s_1, s_2, ...s_I$. It will then "learn" the average trends. For example, the above system may be presented with radar data, which has a slowly varying TF distribution. Some of the centers will track the clutter, and some will track the targets. Computing the inner product of each of the centers with the current radar signal will then provide a feature vector which may be passed on to a classifier. We propose the use of a neural network similar to the Radial Basis Function neural network. We suggest simply modifying the input layer to become these inner products, and leaving the other layers as they normally are.

3 CHIRPLETS

We next propose an expansion of an arbitrary function onto a basis of multi–scale chirps (swept frequency wave packets). Generally we window the chirp so that it will be finite in extent. We may use the Gaussian window[102]

$$env = \sqrt{\frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{1}{2}(\frac{t}{\sigma})^2}} \tag{5}$$

$$= \frac{1}{\sqrt{\sqrt{\pi c}}} e^{-\frac{1}{2}(\frac{t}{c})^2} \tag{6}$$

Where we put $c = \sqrt{2}\sigma$ to obtain unit $L2$ norm, rather than unit $L1$ norm.

Now we use the bases

$$\psi_{abc} = env.e^{j2\pi(at+b)t} \tag{7}$$

or

$$\psi_{abc} = env.e^{j2\pi(a(t/c)+b)(t/c)} \tag{8}$$

If we set $a = 0$ in Equation 7, our transform reduces to the sliding window Fourier transform with the frequency given by b and the time, τ, given by the location of the window, $env(t - \tau)$. If we set $a = 0$ and $b = 1$ in Equation 8, our transform reduces to the affine wavelet transform, with scale parameter c (which also affects the envelope size), and time parameter τ as above.

We then have a chirp "wavelet" for which we have coined the term "chirplet". Figure 8 shows that the relationship of a chirplet to a chirp is analogous to that of a wavelet to a wave.

For a particular value of the dilation parameter c, we define a "bowtie" space:

$$\bowtie (a, b) = < \psi_{ab} | x(t) > \tag{9}$$

Where $x(t)$ is an arbitrary time series, and the two dimensions in the transform domain are the slope of the frequency rise a and the center frequency b.

The fact that the $\bowtie$ transform of a chirplet itself is not a perfectly sharp spike is a consequence of the overlap of the bases and the fundamental resolution limit inherent in any transform.

3.1 The Nyquist boundary problem

Ideally we would like our "bowtie" space to have nice "Manhattan" rectangular boundaries for convenient viewing on a video display.

It turns out that the Nyquist boundary in the $\bowtie (a, b)$ space is diamond–shaped, and so we tilt the space 45°.

We define the starting frequency as the frequency of the wave at the beginning of the window, and the ending frequency as the frequency at the end of the window. (When we use Gaussian windows, we generally define the window boundaries in terms of the 3σ points.)

The diagonal line, $f_{beg} = f_{end}$, is just the Fourier transform of the original time domain signal. Points along this line give a measure of how strong the component at that fixed frequency is. The signal may be entirely re–constructed from only the diagonal of the complex $\bowtie$ transform.

3.2 A simple example with a single component

In this first example, we simply allow an object to fall onto a small radar unit[103]. We compute the chirplet transform, which is illustrated in $\bowtie$ space. Its density plot appears as an image in Figure 9(a). The response has a very high peak, as evidenced from Figure 9(b).

[102] We have greatly simplified our analysis here: what we really want is an *approximation* to some scaling function, such as a bivariate Gaussian in TF space; the bases are actually found by optimization techniques, based on certain constraints.

[103] We positioned the radar horn facing upward and held a volleyball two meters above the horn, and released the ball, while sampling only the inphase component. The sampling rate was 8kHz.

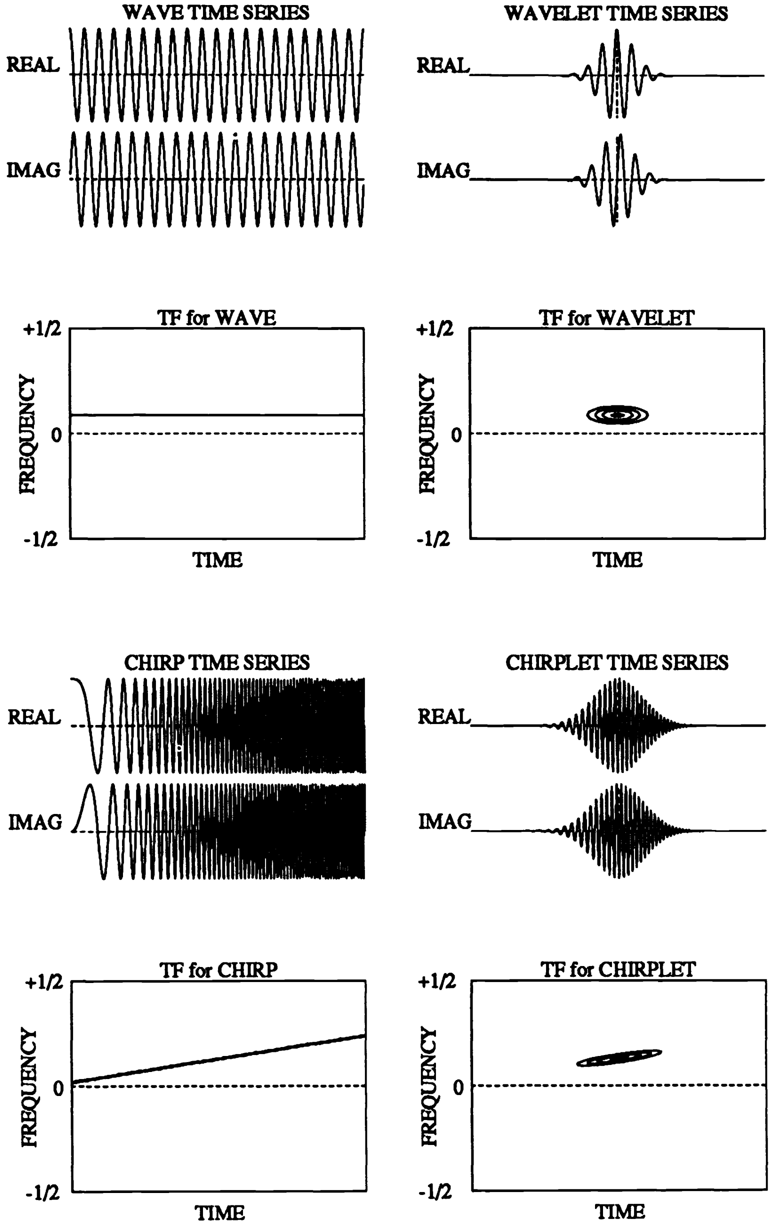
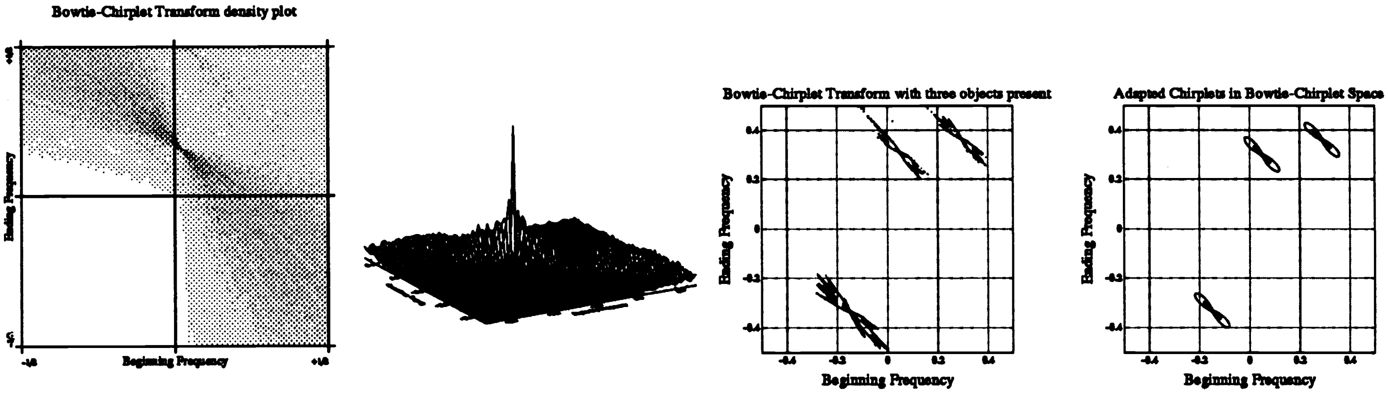


Figure 8: Relationship between wave, "wavelet", chirp and "chirplet", in terms of time series and magnitude Time–Frequency (TF) distributions.



(a) (b) (c) (d)

Figure 9: Bowtie space: a single scale snapshot of the chirplet transform. (a) "bowtie transform" of radar data from falling object. Note the single chirplet component indicating a strong "chirpyness" in the data. (b) Mesh plot showing the bowtie component's strong peak. (c) Bowtie transform of radar return from scene with three objects each undergoing constant acceleration. (d) Result of running LEM on the signal, viewed in our bowtie–space.

4 LEM WITHIN THE BOWTIE SPACE

Our new chirplet space provides an alternate means of visualizing LEM. In particular, the constant scale chirplet snapshot (bowtie transform) gives us a means of viewing these LEM *centers* as they move around in the bowtie space. Rather than having ellipses, we have bowties which move about to match the distribution. In Figure 9(c) we have a bowtie–chirplet transform of a time series from a Doppler radar pointed at three objects which were simultaneously accelerating with different accelerations and different speeds. In Figure 9(d) we show the result of running LEM on the time series, but this time we show our result in our new ⋈–space.

5 REFERENCES

[1] E.F. Greneker and M.A. Corbin. Speed timing radar – new methods to quantify accuracies achievable under various target conditions. In *Carnahan Conference on Security Technology*, pages 253–258, University of Kentucky, Lexington, Kentucky, May 12–14 1982.

[2] B.W. Currie, S. Haykin, and C Krasnor. Time-varying spectra for dual polarized radar returns from targets in an ocean environment. *IEEE Conference proceedings RADAR90, Washington D.C.*, May 1990.

[3] Ingrid Daubechies. *The Wavelet Transform: a method for time-frequency localization.* Appears as a chapter in Advances in Spectrum Analysis and Array Processing: Simon Haykin, Editor, 1991.

[4] D. Gabor. Theory of communication. *J. Inst. Elec. Eng.*, 93:429–456, 1946.

[5] A. Grossman and J. Morlet. Decomposition of functions into wavelets of constant shape. *Mathematics and Physics 2*, December 1984. Report No. 11.

[6] D Lowe. Joint representations in quantum mechanics and signal processing theory: why a probability function of time and frequency is disallowed. *Royal Signals and Radar Establishment*, 1986. report 4017.

[7] Richard Duda and Peter Hart. *Pattern Recognition and Scene Analysis.* John Wiley and Sons, 1973.

[8] A.P. Dempster, N.M. Laird, and D.B. Rubin. Maximum liklihood from incomplete data via the EM algorithm. *Read before the Royal Statistical Society at a meeting organized by the Research Section*, December 8th 1976.

[9] Geoffrey E. Hinton. Neural networks for industry. December 1989.