DISCRETE MULTI-DIMENSIONAL LINEAR TRANSFORMS OVER ARBITRARILY SHAPED SUPPORTS

Krishna Ratakonda and Narendra Ahuja

Department of Electrical and Computer Engineering
Beckman Institute, University of Illinois, Urbana, IL 61801.
krishna@stereo.ai.uiuc.edu

ABSTRACT

Inorder to apply a multi-dimensional linear transform over an arbitrarily shaped support, the usual practice is to fill out the support to a hypercube by zero padding. This does not however yield a satisfactory definition for transforms in two or more dimensions. The problem that we tackle is: how do we redefine the transform over an arbitrary shaped region suited to a given application? We present a novel iterative approach to define any multi-dimensional linear transform over an arbitrary shape given that we know its definition over a hyper-cube. The proposed solution is (1) extensible to all possible shapes of support (whether connected or unconnected) (2) adaptable to the needs of a particular application. We also present results for the Fourier Transform, for a specific adaptation of the general definition of the transform which is suitable for compression or segmentation algorithms.

1 INTRODUCTION

Discrete linear transforms in two (or more) dimensions are in most cases defined over a rectangular (hypercubic) support[1]. The usual practice when we want to apply the transform over an arbitrarily shaped support is to fill out the rest of the support with zeros to make up the rectangle (hypercube) and then use the natural definition of the transform over a rectangle (hypercube). This is an extension of the one dimensional case where we fill out an arbitrary length data set with zeros to form a data set of length 2^n either to increase the computational speed (through FFTs for Fourier Transforms) or to satisfy the definition of the transform (in the case of dyadic Wavelets). This however does not lead to a satisfactory definition of the

---

[1] we will consider only the usual uniform sampling on a square grid in this paper.

linear transform in two or more dimensions for many applications. An example can be used to illustrate this point. The fourier transform of a function that is constant on a circular support in 2-D is a Jinc (see figure 1). As can be seen from the figure the magnitude of the Fourier coefficients do not have any relation to the smoothness of the function, which is a *constant* within its support. Heuristically we can explain this by saying that the coefficients of the transform represent in some sense the shape of the support apart from the variation within the support's interior.

The above discussion leads us to the following question: what should be the values attributed to the sample points which lie within the rectangle but not within the support of the function? The answer is evidently not unique and depends upon the application. With each possible choice of the values for the pixels which lie outside the support but within the rectangular (hypercubic) region, we can associate a possible function-transform pair. The aim of this paper is to algorithmically constrain the choice of the possible function-transform pairs in such a way as to lead to the optimal choice of the function-transform pair for the particular application under consideration. We will consider through out this paper transforms which have a definition over rectangular supports; thus our work is aimed at redefining these transforms over an arbitrarily shaped support in an application oriented fashion.

The applications that we consider in this paper assume that we have a smooth 2-D function defined on an arbitrarily shaped, connected support; we would like to define the free pixels (the pixels within the rectangular region but outside the support) so as to minimize the high frequency content in the Fourier domain. This kind of objective is suitable for image coding with segmentation information [5, 6] or region merging algorithms [7] (which need to estimate the

frequency content within a particular region). Other criteria might also be formulated in the framework that we develop, but will not be dealt with in this paper.

Previous work in this area [1, 3] has been limited to particular transforms and particular kinds of supports. Barnard [1] implemented a method which performs heirarchical zero padding to define the dyadic wavelet transform in one dimension over a connected interval of arbitrary length. The optimality of such a method is questionable. Further more one needs to resort to adhoc fixes when the number of dimensions is more than one. Chen's work [3] is more in the spirit of the present paper. He uses an iterative procedure to obtain optimal DCT coefficients (from a compression point of view) over an arbitrary connected support. However his work is limited to the DCT and connected supports. Also, he does not provide a general framework in which other solutions, suited to particular applications, may be developed.

Section II provides a general definition of the redefined linear transform over an arbitrarily shaped support. In *Section III* we will make specific the general definition of the transform, for the applications to be considered. Illustrative results are presented in the case of the Fourier transform in *Section IV*.

2 REDEFINED LINEAR TRANSFORM

Since most transforms of interest are unitary let us consider such a transform acting over a uniform hyperlattice $\mathcal{Z}^n$ of sample points ($\mathcal{L}$). Let the value of the function which is to be transformed be known on a support $\mathcal{S}$ defined by[2]:

$$\mathcal{S} = \{x : (x \text{ is a lattice point}) \wedge$$
$$(x \text{ is within the support of } \mathcal{S})\}$$

Thus the function to be transformed[3] is $f : \mathcal{S} \to \mathcal{R}$. It may be noted that the lattice points which belong to $\mathcal{L} \cap \mathcal{S}'$ ($\mathcal{S}'$ is the complement of $\mathcal{S}$) are free and can be assigned any real value. For each such possible assignment we obtain a different function-transform pair (since we consider the transform to be unitary it is also injective) as described already. One approach to defining the general transform that can be obtained is as an element of the set of all possible transforms. But such a definition is not constructive in the sense that it does not tell us how to implement the transform. We will give a constraint based definition to avoid this

---

[2] the mathematical symbols $\ni$ and $\wedge$ stand for 'such that' and 'and' respectively.

[3] it might be noted that this function f is nothing but a multi-dimensional sequence (in 2D an image).

pitfall. Thus the transform over an arbitrarily shaped support may be viewed as searching for a function-transform pair that lies at the intersection of various constraint sets.

Definition: A generalised multi-dimensional discrete linear transform of a multi-dimensional sequence $f(\mathcal{S})$ defined over $\mathcal{S}$ is another multi-dimensional sequence defined over the support $\mathcal{L}$. The transformed sequence lies in the space of the constrained multi-dimensional sequences $\mathcal{C}$ which is a subset of all possible multi-dimensional sequences over $\mathcal{L}$ (which we term $\mathcal{G}$). The function space $\mathcal{C}$ is given by:

$$\mathcal{C} = \mathcal{C}_1 \cap \ldots \cap \mathcal{C}_n, \quad n \in \mathcal{N}$$

where $\mathcal{C}_1 = \{g(x) : (g : \mathcal{L} \to \mathcal{R}) \wedge (x \in \mathcal{S} \Rightarrow g(x) = f(x)\}$ and $\mathcal{C}_2, \ldots, \mathcal{C}_n$ are subspaces of the function space $\mathcal{G}$ acting as constraints to restrict possible $g \in \mathcal{C}$ (they are to be defined according to the application; in *Section III* we will be defining $\mathcal{C}_2$ to make the definition of the transform specific to the case of smooth functions). Also the cardinality of $\mathcal{C}$ should be one to ensure that each function $f(\mathcal{S})$ has a unique transform. In most cases the algorithmic implementation implicitly defines a unique member of $\mathcal{C}$, although the cardinality of $\mathcal{C}$ is not one (as in *Section III*). The transform is assumed to bear its usual definition over the hypercube($\mathcal{L}$) of samples.

3 APPLICATION MOTIVATED DEFINITION

In the parlance of the previous section, we need to define the sets $\mathcal{C}_2, \ldots$ based on the application in question. $\mathcal{C}_1$, as defined in *Section II*, is a convex set. It would be advantageous from an implementational point of view to define the rest of the constraints as convex sets too; this would enable us to treat the transformation as a search for a solution that lies at the intersection of various convex sets (such a search has nice convergence properties [2]).

From now on we will consider only two dimensional functions (images) and the Fourier Transform. As stated before the principal objective is the minimization of the high frequency content in the transform domain. This can be approached in different ways as the definition of the term "high frequency" is imprecise. For this purpose we define ζ as the side of a square containing the low frequency components in the transform domain (see figure 1(b)) and define the constraint set $\mathcal{C}_2$ as:

$$\mathcal{C}_2 = \{g : (g : \mathcal{L} \to \mathcal{R}) \wedge$$
$$(g \in \mathcal{C}_1) \wedge (\|g - f\| \text{ is minimized}, f \in \mathcal{E})\}$$
$$\mathcal{E} = \{f : (f : \mathcal{L} \to \mathcal{R}) \wedge$$

$(\mathcal{T}f(x) = 0, \forall x \in \{|\text{frequency}| > \zeta\},$
$\mathcal{T}f : \mathcal{L} \to \mathcal{R}$ is the transform of f over $\mathcal{L})\}$

In plain words $\mathcal{C}_2$ consists of those functions, $f \in \mathcal{C}_1$, which are closest to the set $\mathcal{E}$ (in the least squares sense); $\mathcal{E}$ in turn consists of all those functions whose high frequency components greater than ζ are zero (ζ is a constant to be defined; see figure 1(b)). Note carefully that this is an indirect definition of the constraint set $\mathcal{C}_2$; $\mathcal{E}$ "attracts" those solutions in $\mathcal{C}_1$ of possible interest to us. The advantage of this definition is evident; as ζ increases we "allow" more and more variation in the free pixels. Evidently $\zeta = 0$ allows no variation in the free pixels and we get the usual definition of the transform which restricts all free pixels to be zero. However if ζ is put to too high a value we have too much variation in the free pixels which jeopardizes the smoothness requirement. Thus ζ is to be chosen as some intermediate value which would remove the unsmoothness caused by the shape of the support and yet not introduce too much adhoc unsmoothness in the free pixels (see *Section IV* for examples).

The search is formulated in terms of the often used Projection on Convex Sets formalism [4] and we include a few more constraints (for example bounded variation of the free pixels) to speed up the search. The search procedure involves projecting on to $\mathcal{C}_1$, $\mathcal{C}_2$ etc. (the convex constraint sets) alternately. The projection operators in our case are very simple and computationally efficient. The projection operator on to $\mathcal{C}_1$ involves setting those values which lie within the support of the function to their actual values in the spatial domain. The projection operator on to $\mathcal{C}_2$ involves putting all the function values which lie outside the shaded regions (figure 1(b)) to zero. The initial value for the pels outside the functions support have been chose to be zero.

4 SIMULATION RESULTS

Results for the Fourier transform as ζ is varied are presented in figure 2. The example function is a smooth region taken out of a natural image (Lena). As can be seen, an increase of ζ allows more variation in the free pixels in the spatial domain thus destroying the adhoc shape information (the pixels within the support are unchanged in all images). Similar results maybe obtained if we replace the Fourier transform with either DCT or Wavelets. The formulation would be very similar except for trivial modifications of the consraint set $\mathcal{C}_2$.

5 REMARKS

In this paper we have presented a general framework to modify the definition of a linear transform de-

fined over a hyper-cube to an arbitrary support. Moreover, the redefinition of the transform embeds within itself flexibility to adapt to a given application. The advent of such a technique would enable the merging of non-linear image processing methods like segmentation and linear methods (eg., wavelets) in a seamless fashion.

Image and video compression is a field where the proposed technique would find a large number of applications. Segmentation has been used by many researchers [5, 6] for image compression. However, the results, although promising, were limited by inefficient moment based techniques which are normally used to represent the image intensity variation within a segmented region. By using the technique presented in this paper, one can potentially replace the moment based representation by any linear transform of choice. Interpolation of Fourier data, which is available only at select arbitrary points on a cartesian grid, is a classical problem. One solution to this problem may be formulated using the technique outlined here. These examples form only a subset of the problems which might benifit from the proposed approach. We envisage using the redefined version of the transform in specific applications in forthcoming papers.

References

[1] H. J. Barnard. Image and video coding using a wavelet decomposition. *Ph. D. Thesis, Technische Universiteit Delft.*

[2] D. P. Bertsekas. *Non linear programming.* Athena Scientific, 1995.

[3] H. H. Chen, M. R. Civanlar, and Haskell B. G. A block transform coder for arbitrarily shaped image segments. *IEEE International Conference on Image Processing,* 1:85–89, 1994.

[4] B. Javidi et al. *Real time optical information processing.* Academic Press, 1994.

[5] M. Kunt, A. Ikonomopoulos, et al. Second generation image coding techniques. *Proc. IEEE,* 73(4):549–574, 1985.

[6] O. J. Kwon and R. Chellappa. Region based subband image coding scheme. *Proc. IEEE IP,* 2:859–863, 1994.

[7] D. K. Panjwani et al. MRF models for unsupervised segmentation of textured color images. *IEEE Trans. PAMI,* 17(10):939–54, 1995.
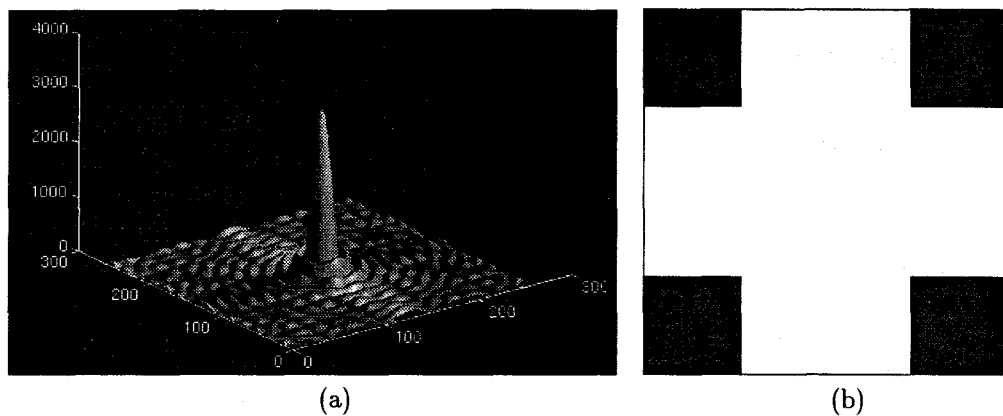
(a)   (b)

Figure 1: (a) *Section I:* Fourier Transform of a Circle in 2-D (sampled on a 256x256 grid). Absolute values of the Fourier coefficients are represented by height in the z-dimension. (b) *Section III:* The non-zero values of the Fourier Transform of a function in $\mathcal{E}$ are shaded. The four shaded squares together represent the low frequency components in the DFT domain. ζ is the length of a side of any of the shaded squares.

(a)   (b)   (c)
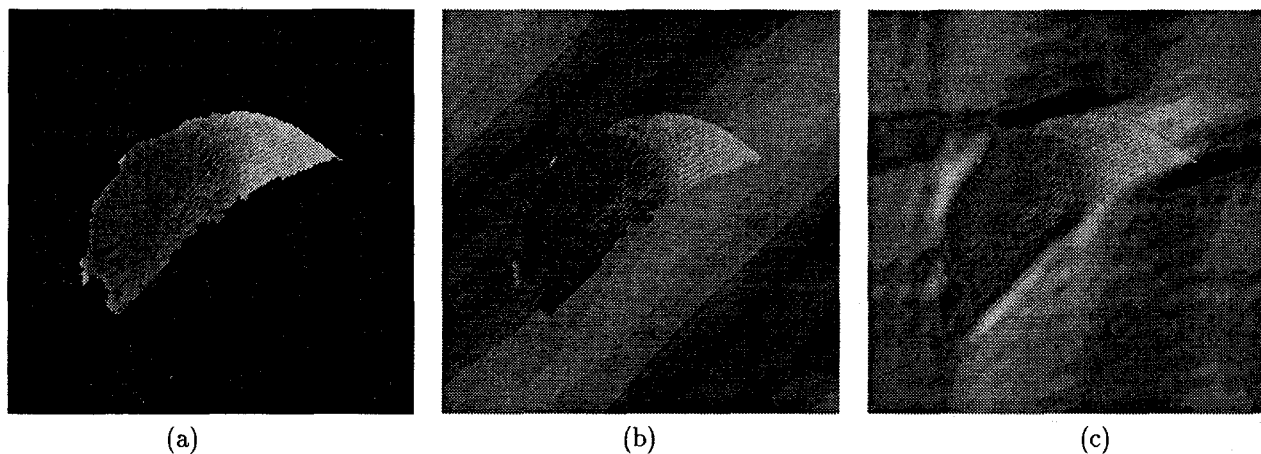
Figure 2: Redefined Fourier Transform in 2-D: The images are for a rectangular region in the spatial domain, in which only the pixels within the support of the function have definite values. The free pixels are varied so that we achieve smoothness in the Fourier domain. (a) Actual function (256x256, all free pixels set to zero) in the spatial domain. (b) $\zeta = 2$ (2x2 box of fourier components are allowed to be non-zero for functions in $\mathcal{E}$ as in figure 1(b)) (c) $\zeta = 16$.