Adaptive Signal Processing Laboratory (ASPL)
Electrical and Computer Engineering Department
University of Florida

Overlapping tree approach for multi-scale modeling and estimation

ASPL Report No. Rep_2004-12-005

Hojin Jhee and K. Clint Slatton

December 12, 2004

Point of Contact:
Prof. K. Clint Slatton
University of Florida; PO Box 116130; Gainesville, FL 32611
Tel: 352.392.0634, Fax: 352.392.0044, E-mail: slatton@ece.ufl.edu

Problem statement:

In summer 2004, we implemented Fieguth's 'overlapping' tree to test its ability to reduce blockiness in standard MKS estimates. It does indeed smooth, but does not eliminate the bloky artifacts that are a consequence of the quadtree data struction and the rigid Markovian communication from parent to exactly 4 children. Still, it is worth saving the details in a report.

Overlapping tree approach for multi-scale modeling and estimation

Sweungwon Cheung

Hojin Jhee

1. Introduction

In this study we introduce the motivation of overlapping multi-scale framework. This framework is developed with specific interest of producing smooth estimates without blocky artifact.

In spite of the computational efficiency, mean-square estimation error and providing error covariance information, the multi-scale approach which we have studied so far has certain performance limit. Specifically, estimate based on this “normal” multi-scale framework suffers visually distracting blockiness which is a serious limit for smooth estimation problem [1]. Although estimate blockiness can be mitigated by the following two methods, these methods have their own disadvantages.

By Postprocessing (by applying low pass filter). [1]

Disadvantage : The postprocessing can render less clear the proper interpretation of error covariance information provided by the estimation algorithm, and it also limits the resolution of fine-scale detail in the postprocessed estimate, since the added smoothness is achieved by spatial blurring.

By Accurate representations of broad classes of random fields. [1]

Disadvantage : To achieve a high level of smoothness, these methods require the use of multi-scale processes of high dimension, thereby leading to a reduction in the significant computational advantages that the multi-scale modeling framework offer.

The computational complexity of multi-scale estimation is

$$O(N\lambda^3)$$

where N is the number of nodes and λ is dimension of state vector. Therefore, if we have high dimensional states (λ), the computational complexity could be increased significantly. So, there exists tradeoff between increase of smoothness and complexity.

Before we move our step to details about overlapping tree structure, we need to figure out the source of blockiness. A most critical property in multi-scale models is that they are Markov. The Markoviniaty states that if *x(s)* is the value of the state at node *s*, then conditioned on the value of *x(s),* the states at any node in the *q* children of *s* are uncorrelated. This decorrelation property of multi-scale model provides us both efficient algorithm and the source of blockiness [2].

In our discussion, we discard the assumption that distinct nodes at a given level of a tree correspond to disjoint portions of the image domain and allowing the tree nodes to correspond to overlapping regions. [1]. By this modification, we can remove hard boundaries between image domain pixels which are separated far in the sense of tree distance.

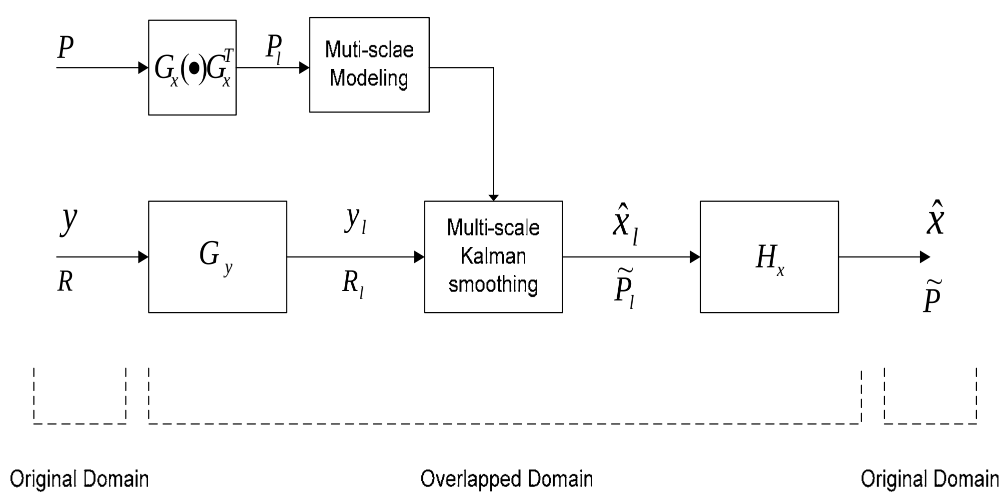

Figure.1 Overview of overlapping approach based on multi-scale modeling

Figure.1 shows the overview of overlapping tree algorithm which optimally estimates the random field *x*, given noisy observation *y*. In this structure, we assume that

1) *We have knowledge of correlation structure P of some random field x*
   - We can control the overlap parameter *O* to build the proper overlapping structure.
2) *The prior model is internal model.*
   -We can get MKS parameters easily.
3) *The dimension of state vector is reduced by canonical correlation method.*
   - The computational complexity is decreased.

In overlapping structure, we can guess that as the number of pixels to be overlapped increases the resulting simulated or estimated fields become smoother but the computational complexity of estimation algorithms increases. Through our discussion, we are interested in the finest-scale estimate of random field.

.

2. Specification of operators (G_x, G_y, H_x)

1. G_x

-It lifts actual random field x into "lifted" domain as follows:

$$x_l = G_x x$$

where x_l is the lifted-domain field and x is real image domain field

2. G_y

-It lifts actual observation y of the random field x into lifted domain. (whose role is same as G_x.)

$$\begin{aligned} y_l &= G_y y \\ &= G_y(Cx+v) \\ &= G_y Cx + G_y v \\ &= C_l G_x x + G_y v \\ &= C_l x_l + v_l \end{aligned}$$

where y_l is the observation in lifted domain and y is the observation in real domain.

-These observations are processed by MKS algorithm that we have used to produce lifted version estimate $\hat{x}_l$.

3. H_x

-It maps lifted domain estimate $\hat{x}_l$ to that of original domain $\hat{x}$.

$$\hat{x} = H_x \hat{x}_l$$

To illustrate how we can specify the operators G_x and H_x required for overlapping processes, one simple example will be shown.

Example (Size of original domain: 3x3 ↔ Size of lifted domain: 4x4)

1) Specification of G_x

Let's assume that we have random field x to be lifted and the number of pixels to be overlapped is one.

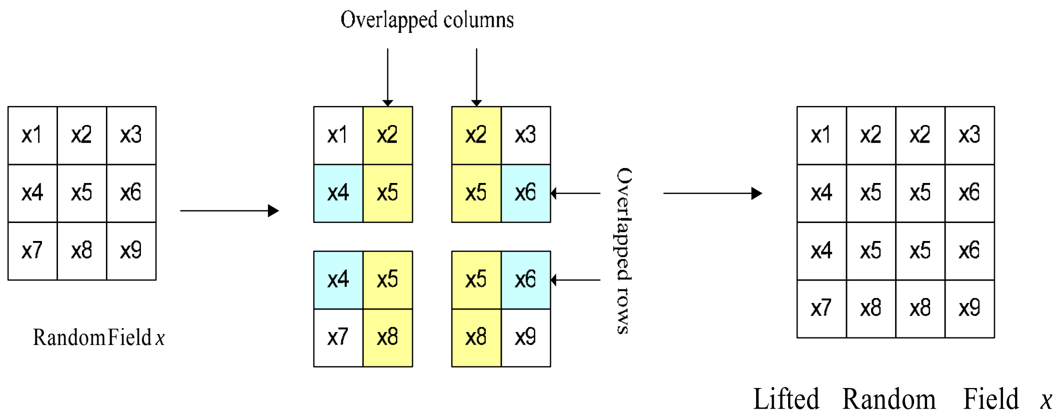

Figure 2. Illustration of tree overlapping. The pixels within overlapped region are copied to construct the lifted random field.

Figure.2 illustrates how we can produce lifted version of random field from original filed. The operator G_x projects the statistics of x into the lifted domain. The constraints of constructing G_x are [1].

1 It contains entirely of zeros and ones.
2. Each column has at least one nonzero entry
3. Each row has exactly one non zero entry.

The constraints shown guarantee that every pixel point in original domain must be positioned at any point in lifted domain. The specification of G_x completely depends on the overlapping structure of the tree. From our example, we can get

$$x_l = G_x x$$

$$vec(x_l) = \underbrace{\left[\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \right]}_{G_X} vec(x)$$

where *Vec(A)* is vectorization of matrix *A* and indicates Kronecker product between two matrices.

2) Specification of G_y and R_l

As shown in Figure.1, we need to have lifted version of measurement for MKS algorithm in lifted domain. For this purpose, let's define lifting operator G_y to produce lifted measurement such that $y_l = G_y y$. We simply copy the actual measurement value at any pixel in original domain to all finest-scale nodes of lifted domain which are associated with that pixel. For example, let's consider again previous 3 by 3 field case. Assume that the pixels *x2*, ,*x7* and *x8* in real image domain are not measured. Missed pixels are shown as black entries in Figure.3. We can get the operator G_y from G_x by just discarding the columns which correspond to missed measurement points.
See figure 3. In this example, measurement points for *x2*,*x7* and *x8* are missed, so we simply discard third, forth and sixth columns to get of G_y from G_x (third, forth and sixth elements of vectorized measurement correspond to the measurement points of the corresponding vectorized field *x*).

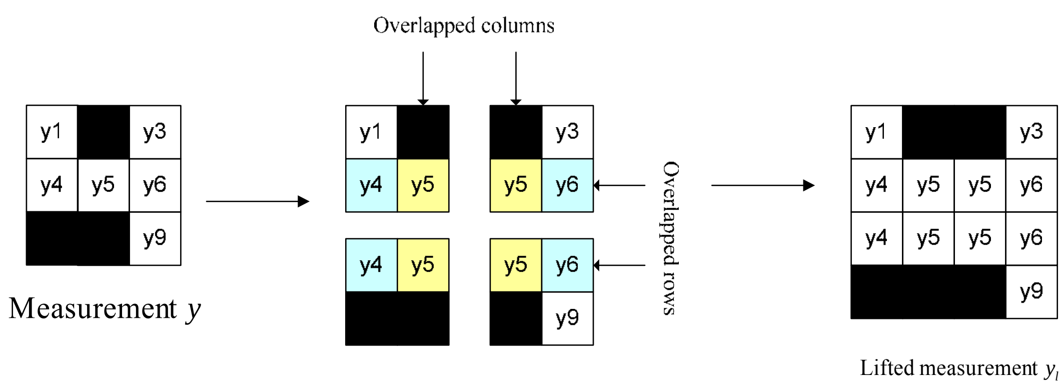

Figure 3. Illustration of measurements overlapping.

$$vec(x)=\begin{bmatrix}x_1\\x_4\\x_7\\x_2\\x_5\\x_8\\x_3\\x_6\\x_9\end{bmatrix}\quad\Rightarrow\quad vec(y)=\begin{bmatrix}y_1\\y_4\\\times\\\times\\y_5\\\times\\y_3\\y_6\\y_9\end{bmatrix}\quad \text{where "}\times\text{" represents no data available}$$

One problem we can have in this situation is that by modeling y_l by this way we have more information than we actually have. To compensate this, we count the number of measurement points being replicated and multiply this number to the corresponding measurement noise variance in our model.

For example, if y_2 is expanded to y_2 and y_2,

$$y_2\rightarrow\begin{bmatrix}y_2\\y_2\end{bmatrix}$$

, then v_2 is expanded as follows:

$$v_2\rightarrow\begin{bmatrix}\sqrt{2}v_2\\\sqrt{2}v_2\end{bmatrix}$$

The relationship between R and R_l can be described as:

$$G_y^T R_l^{-1} G_y = R^{-1}$$

3) Specification of H_x

Once we specify the operators G_x ,G_y and R_l , we can obtain the finest scale estimate $\hat{x}_l$ and $\widetilde{P}_l$ by multi-scale Kalman smoothing algorithm in lifted domain. The final step of our procedure is to define H_x so that we can obtain final smooth version of estimate $\hat{x}$ and error covariance $\widetilde{P}$ in original image domain. We can summarize the role of H_x for smoothness such that the relative contributions of a node should taper towards zero as one approach an overlapped end of the interval associated with the node. This is illustrated in Figure.4.

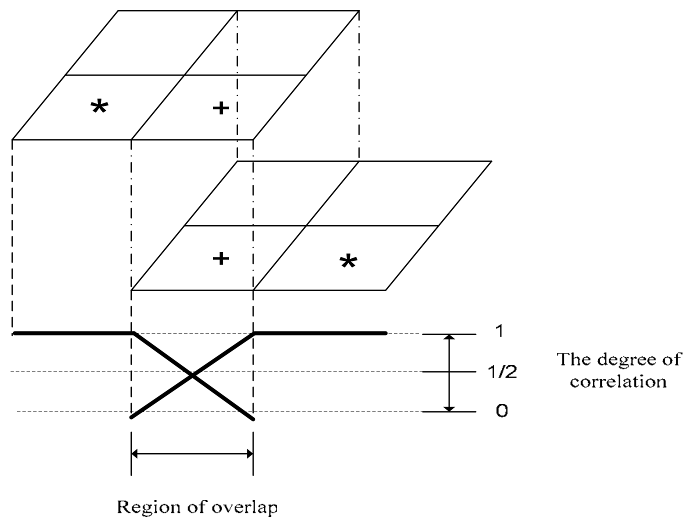

Figure 4. The relative contribution (or degree of correlation between the corner end pixel (*) and
end of overlapped pixel (+)) to finest-scale pixel must sum to one (Only one directional (horizontal)
overlapping is described). So, in our example, H_x can be obtained as

$$\hat{x} = H_x \hat{x}_l$$

$$vec(\hat{x}) = \left[\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \right] vec(\hat{x}_l)$$

where *Vec(A)* is vectorization of matrix *A* and indicates Kronecker product between two matrices.

3. Matlab result

33x 33 (real image domain) ↔ 64x 64 (lifted image domain)

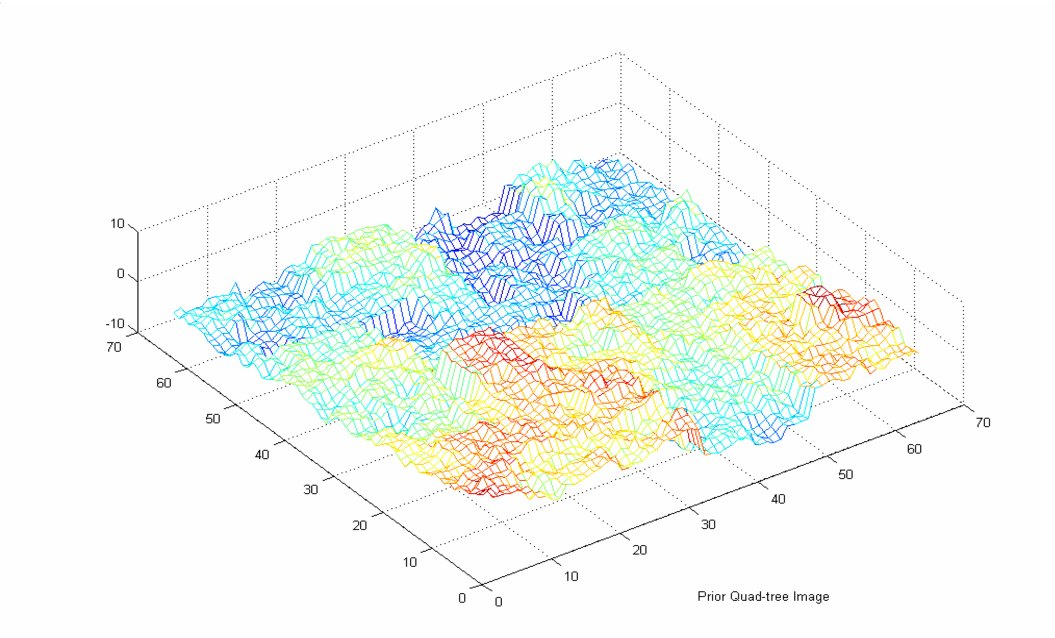

Figure 5. Prior model in lifted domain

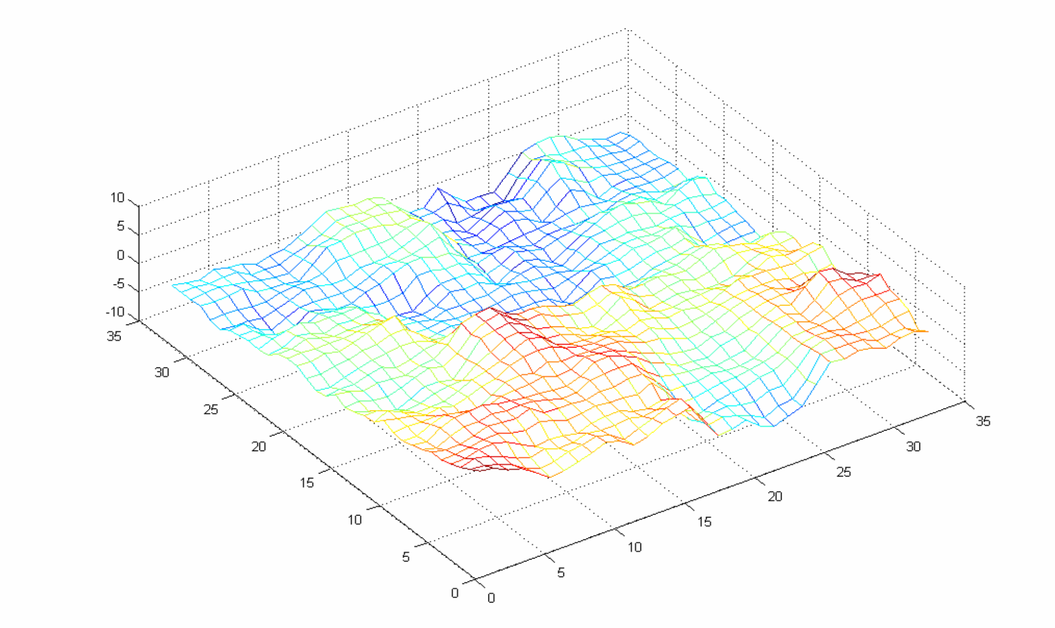

Figure 6. Prior model in real domain

We only consider the prior model. As we know if the prior model has no blockiness, we can get smooth

estimate. If we increase the Mbase (size of image domain), we can not remove big blockiness, because
we assume that each pixel is correlated only to the nearest neighbors.

References

[1]W. Irving, P. Fieguth, and A. Willsky, “An overlapping tree approach to multiscale stochastic modeling and estimation,” *IEEE Trans. Image Processing*, vol. 6, pp. 1517–1529, Nov. 1997.

[2] P. Fieguth, “Application of multiscale estimation to large scale multidimensional imaging and remote sensing problems,” Ph.D. dissertation, Dept. Electr. Eng. Comput. Sci., Mass. Inst. Technol., Cambridge, 1995.

[3]Peyman Milanfar, Robert R. Tenney, Robert B. Washburn, and Alan S. Willsky, “ Modeling and estimation for a class of multiresolution random field, IEEE 1994

[4]M. Luettgen, W. Karl, A. Willsky, and R. Tenney, “Multiscale representations of Markov random fields,” *IEEE Trans. Signal Processing*, vol. 41, pp. 3377–3396, 1993.