Unsupervised Progressive Parsing of Poisson Fields Using Minimum Description Length Criteria

Robert D. Nowak*

Dept. of Electr. and Comp. Eng.
Rice University
Houston, TX, **U.S.A.**

Mário A. T. Figueiredo†

Instituto de Telecomunicações, and
Instituto Superior Técnico
1049-001 Lisboa, **Portugal**

Abstract

This paper describes novel methods for estimating piecewise homogeneous Poisson fields based on minimum description length *(MDL) criteria. By adopting a coding-theoretic approach, our methods are able to adapt to the the observed field in an unsupervised manner. We present a parsing scheme based on fixed multiscale trees (binary, for 1D, quad, for 2D) and an adaptive recursive partioning algorithm, both guided by MDL criteria. Experiments show that the recursive scheme outperforms the fixed tree approaches.*

1 Introduction

Consider a realization from a spatial Poisson point process whose underlying intensity function is (or can be approximated as) piecewise constant (as exemplified in Figure 1(a)). This type of data arises in photon-limited imaging, particle and astronomical physics, computer traffic network analysis, and many other applications involving counting statistics. In particular, photon-limited images are formed by detecting and counting individual photon events (*e.g.,* in gamma-ray astronomy or nuclear medicine).

From an observed realization (see Figure 1(b)), our goal is to *parse*, or *segment*, the observation space into regions of (roughly) homogeneous intensity (Figure 1(c)); *i.e.,* regions of space in which the distribution of points is well modeled by a spatial Poisson distribution with constant intensity. In this paper, we propose two new methods for unsupervised progressive parsing of Poisson fields based on Rissanen's *minimum description length* (MDL) principle [1]. One of the interesting aspects of our development is that, since the Poisson data (counts) are integer-valued, we are able to derive MDL criteria without recourse to asymptotic approximations. In contrast, most applications of MDL involve Gaussian statistics (which are real-valued) and require asymptotic arguments. Hence, our application of MDL here is especially simple and well motivated. This work is a coding-theoretic alternative to related Bayesian estimation schemes [2, 3, 4].

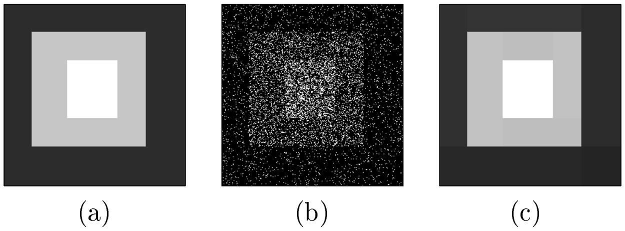


Figure 1: *Illustration of the problem addressed in this paper: (a) piecewise-constant intensity function (Poisson intensities 0.05, 0.2, and 0.4); (b) scatter-plot of observed photon events; (c) intensity field parsing by MDL-based recursive algorithm.*

2 Mininum Description Length

The MDL principle addresses the following question: given a set of generation models, which best explains the observed data? To get a handle on the notion of "best," Rissanen employed the following *gedankenexperiment.* Suppose that we wish to transmit the observed data x to a hypothetical receiver. Given a (probabilistic) generation model for the data, say $p(x|\theta)$, the Shannon-optimal code length is $-\log p(x|\theta)$. Of course, the receiver would also need to know the model parameters θ to decode the transmission; then, if θ is *a priori* unknown, we also need to estimate it, code it, and transmit it. Now, consider a set of K competing model classes $\{p_i(x|\theta_i)\}_{i=1}^K$. In each class i, the "best" model is the one that gives the minimum code length,

$$\widehat{\theta}_i = \arg\min_{\theta_i} \{-\log p_i(x|\theta_i)\} = \arg\max_{\theta_i} p_i(x|\theta_i);$$

[*] Supported by the National Science Foundation, grant no. MIP–9701692.

[†] Supported by the *Foundation for Science and Technology* (Portugal), grant Praxis-XXI T.I.T.-1580.

this is simply the *maximum likelihood* (ML) estimate within model class i. But if the class is *a priori* unknown, the "best" overall model is the one that leads to the minimum *description length*: the sum of $-\log p_i(x|\theta_i)$ with the length of the code for θ_i itself. The fundamental aspect of MDL is that it performs model selection (which the ML criterion alone does not) by penalizing more complex model classes (those requiring longer parameter code lengths). MDL criteria have been successfully used in several image analysis/processing problems (see references in [5]).

The delicate issue in applying MDL is in how to encode the parameter θ_i; appropriate parameter code lengths are usually based on asymptotic approximations; *e.g.*, the well known $(1/2)\log N$, where N is the amount of data, is an asymptotic code length [1]. In this paper, we are able to avoid asymptotic approximations and obtain exact code lengths.

3 MDL for Poisson Data

We introduce next two progressive (multiscale or multiresolution) approaches to coding Poisson data. These approaches use MDL model class selection criteria as the basic building block.

3.1 Binary Multiscale Tree

Assume that the observed data is a (1D) sequence of counts, $\{x_k\}_{k=0}^{N-1}$, whose length $N = 2^J$. Let $x_{J,k} \equiv x_k$, for $k = 0, ..., 2^J - 1$; for $j = J-1, \ldots, 0$, let us define the multiscale analysis of the data according to:

$$x_{j,k} \equiv x_{j+1,2k} + x_{j+1,2k+1}, \;\; k = 0, \ldots, 2^j - 1. \quad (1)$$

The $\{x_{j,k}\}$ are Haar scaling coefficients; for more details on multiscale analyses of Poisson data, see [2, 3].

Now suppose we wish to transmit the observed data $\{x_{J,k}\}$. Adopting a predictive coding approach, operating in scale, from coarse to fine, we naturally start by transmitting the total count $x_{0,0}$; this can be coded using Elias' technique for arbitrarily large integers [1]. We then progressively transmit the data (in a coarse-to-fine fashion) by next sending $x_{1,0}$, then $x_{2,0}, x_{2,2}$, and so on. Note that we only need to send the scaling coefficients with even k indices; the corresponding odd-indexed data are deduced from them and the previously transmitted coarser data (*e.g.*, $x_{1,1} = x_{0,0} - x_{1,0}$). At each stage, our progressive transmission scheme takes advantage of the coarser data already sent. Formally, we are interested in the conditional probability $p(x_{j+1,2k}|x_{j,k})$. It is well known (see [2, 3]) that this probability is binomial with parameters $x_{j,k}$ and $\rho_{j,k} \equiv \frac{\lambda_{j+1,2k}}{\lambda_{j,k}}$, where $\lambda_{j,k}$ and $\lambda_{j+1,2k}$ are the intensities underlying the Poisson counts $x_{j,k}$ and $x_{j+1,2k}$, respectively. So,

$$\begin{aligned} p(x_{j+1,2k}\,|\,x_{j,k},\rho_{j,k}) &= \mathcal{B}i(x_{j+1,2k}\,|\,x_{j,k},\rho_{j,k}) \\ &= \binom{x_{j,k}}{x_{j+1,2k}} \rho_{j,k}^{x_{j+1,2k}} (1-\rho_{j,k})^{x_{j+1,2k+1}}. \quad (2) \end{aligned}$$

Now consider two available model classes. Model Class 0 assumes a homogeneous Poisson process; then, $\lambda_{j+1,2k} = \lambda_{j+1,2k+1}$ and consequently, since $\lambda_{j,k} = \lambda_{j+1,2k} + \lambda_{j+1,2k+1}$, we have $\rho_{j,k} = \frac{1}{2}$. Alternatively, in Model Class 1, $\rho_{j,k}$ is a free parameter.

Hence, we have two possible description lengths (recall that $x_{j,k}$ is already known by the receiver).

Model Class 0: Since $\rho_{j,k} = 1/2$, it doesn't require encoding; the description length is then simply

$$\begin{aligned} L_0 &= -\log \mathcal{B}i(x_{j+1,2k}\,|\,x_{j,k}, 1/2) \quad (3) \\ &= -\log \binom{x_{j,k}}{x_{j+1,2k}} + x_{j,k} \log 2 \end{aligned}$$

Model Class 1: In this case, the first step consists in estimating, coding, and transmitting $\rho_{j,k}$. Its ML estimate is $\widehat{\rho}_{j,k} = \frac{x_{j+1,2k}}{x_{j,k}}$. Because $x_{j,k}$ was already transmitted, it suffices to encode and transmit $x_{j+1,2k}$; since $x_{j+1,2k} \in \{0, 1, ..., x_{j,k}\}$, this requires $\log(x_{j,k}+1)$ bits. Surprisingly, we find that while encoding the ML estimate of the parameter, we have encoded the data $x_{j+1,2k}$ itself, and so no additional coding is needed. The resulting description length is simply

$$L_1 = \log(x_{j,k} + 1) = -\log \frac{1}{x_{j,k}+1}. \quad (4)$$

We transmit the data according to Model Class 0 if $L_0 < L_1$, and using Model Class 1, otherwise (what we choose when $L_0 = L_1$ is irrelevant). We then define the optimal MDL parsing of the data according to the same criterion, applied at each scale and location. If $L_0 < L_1$, we put $\widehat{\rho}_{j,k} = 1/2$; otherwise, $\widehat{\rho}_{j,k} = \frac{x_{j+1,2k}}{x_{j,k}}$.

The underlying piece-wise constant intensity (scale J) is reconstructed/estimated from the total counts $x_{0,0}$ and the sequence of splitting proportion estimates $\{\widehat{\rho}_{j,k}\}$. Set $\widehat{\lambda}_{0,0} \equiv x_{0,0}$. Next, with $j = 1$, $\widehat{\lambda}_{1,0} = \widehat{\lambda}_{0,0}\widehat{\rho}_{0,0}$ and $\widehat{\lambda}_{1,1} = \widehat{\lambda}_{0,0}(1-\widehat{\rho}_{0,0})$. Repeat this refinement process for $j = 2, \ldots, J$, to obtain $\{\widehat{\lambda}_{J,k}\}$.

Finally, it is worth pointing out that the same criterion to choose between Model Classes 0 and 1 can be obtained under a Bayesian model selection perspective. Let y denote a sample of a binomial random variable (same as $x_{j+1,2k}$) with probability function $\mathcal{B}i(y\,|\,n,\rho)$ and consider the problem of deciding between two hypotheses: H_0: $\rho = 1/2$, or H_1:

$\rho \neq 1/2$ (otherwise totally unknown). Furthermore, with no *a priori* preference for H_0 or H_1 we use a prior $p(H_0) = p(H_1) = 1/2$. The models for ρ under the two hypotheses are

$$\begin{aligned} p(\rho|H_0) &= \delta(\rho - 1/2), & (5)\\ p(\rho|H_1) &= U(\rho\,|\,0,1), & (6) \end{aligned}$$

where $\delta(x-a)$ denotes a Dirac delta function (a point mass) at a and $U(\rho\,|\,a,b)$ stands for a uniform probability density function between a and b. Naturally, we decide for H_1 if $p(H_1|y) \geq p(H_0|y)$, which is equivalent to $p(y|H_1) \geq p(y|H_0)$ because $p(H_0) = p(H_1)$. The marginal likelihoods are particular cases of the *binomial-Beta* distribution (see [6], pp. 117)

$$\begin{aligned} p(y|H_0) &= \int_0^1 p(y|\rho)\, p(\rho|H_0)\, d\rho \;=\; \mathcal{B}i(y\,|\,n, 1/2)\\ p(y|H_1) &= \int_0^1 p(y|\rho)\, p(\rho|H_1)\, d\rho \;=\; \frac{1}{n+1}. \end{aligned}$$

Then, comparing $p(y|H_0)$ versus $p(y|H_1)$ is the same as comparing L_0 versus L_1, as given by (3) and (4).

3.2 Adaptive Recursive Parsing

One of the limitations of the multiscale approach above is that the parsing is restricted to a fixed binary tree. In general, the best locations for parsing may not coincide with the dyadic partition enforced by the multiscale analysis above. Hence, we now consider an adaptive recursive approach that allows for splits at arbitrary locations. Again, suppose we wish to transmit a length N sequence of counts $x = \{x_k\}_{k=0}^{N-1}$. Unlike in the fixed tree approach, N no longer needs to be a power of 2.

Under Model Class 0, we code and transmit the N counts assuming that they are Poisson samples with a common intensity λ. Alternatively, we split the data into two (connected) components with two different constant intensities; however, unlike in the fixed tree approach described in the previous subsection, we are allowed to look for the best location to split the counts. Accordingly, this alternative actually consists of $N-1$ model classes, one for each split location, which we will index by $i \in \{1, 2, ..., N-1\}$. We thus have a total of N candidate classes; if they are all *a priori* equiprobable, each index requires the same $\log N$ code length which can then be dropped from any comparisons.

Since this basic building block will be included in a recursive/predictive, coarse-to-fine, procedure, we assume that the total count $s_N = \sum_{k=0}^{N-1} x_k$ is known to the receiver and need not be encoded. The description lengths achieved are:

Model Class 0: With a constant intensity model, and given the total count s_N, the individual counts follow a multinomial distribution with all parameters equal to $1/N$, *i.e.*,

$$p(x_1, ..., x_N | s_N) = \left(\begin{array}{c} s_N \\ x_1 \ldots x_N \end{array} \right) \left(\frac{1}{N} \right)^{s_N} \mathbf{1}_{\sum_{k=0}^{N-1} x_k \,=\, s_N},$$

where $\mathbf{1}_C$ is the indicator function of condition C (equal to one if C is true, zero otherwise); the multimonial coefficients are given by

$$\left(\begin{array}{c} s_N \\ x_1 \ldots x_N \end{array} \right) = \frac{s_N!}{x_1!\; x_2!\; \cdots x_N!}.$$

In this case, there is no parameter to estimate and the resulting description length is simply

$$L_0 = -\log \left(\begin{array}{c} s_N \\ x_1 \ldots x_N \end{array} \right) + s_N \log N. \qquad (7)$$

Observe that (3) is a particular case of this expression, for $N = 2$, $s_N = x_{j,k}$, and $x_1 = x_{j+1,2k}$.

Model Classes $1, ..., N-1$: Model class i assumes that $\{x_k\}_{k=0}^{i-1}$ and $\{x_k\}_{k=i}^{N-1}$ are sets of Poisson samples of different intensities. Given s_N, the individual counts are still multinomially distributed. However, the first i parameters are now equal to, say, ρ/i, and the $N-i$ last ones equal to $(1-\rho)/(N-i)$. Note that with $\rho = i/N$, we recover Model Class 0. Then,

$$\begin{aligned} p(x_1, ..., x_N | s_N, \rho) = \left(\begin{array}{c} s_N \\ x_1 \ldots x_N \end{array} \right) \times \\ \left(\frac{\rho}{i} \right)^{s_i} \left(\frac{1-\rho}{N-i} \right)^{s_N - s_i} \mathbf{1}_{\sum_{k=0}^{N-1} x_k \,=\, s_N} \qquad (8) \end{aligned}$$

where $s_i \equiv \sum_{k=0}^{i-1} x_k$, $i = 1, \ldots, N$.

To use this model to encode the data, we first have to estimate ρ; its ML estimate is simply $\widehat{\rho} = \frac{s_i}{s_N}$. Since s_N is known, all that needs to be encoded is s_i which involves a code-length of $\log(1 + s_N)$ (since $s_i \in \{0, 1, ..., s_N\}$). But after transmitting s_i, we can build a better code for the data, because we know that $\sum_{k=0}^{i-1} x_k = s_i$ and $\sum_{k=i}^{N-1} x_k = s_N - s_i$. Specifically, each set of counts is itself multinomially distributed, leading to a total code length

$$\begin{aligned} L_i \;=\; \log(1 + s_N) - \log \left(\begin{array}{c} s_i \\ x_1 \ldots x_{i-1} \end{array} \right) + s_i \log i \\ - \log \left(\begin{array}{c} s_N - s_i \\ x_i \ldots x_{N-1} \end{array} \right) + (s_N - s_i) \log(N - i). \end{aligned}$$

Notice that (4) is a particular case of this expression for $N = 2$, $s_N = x_{j,k}$, $x_1 = x_{j+1,2k}$, $x_2 = x_{j+1,2k+1}$.

Our progressive/recursive parsing (or transmission) scheme, proceeds as follows. As above, we start by encoding the total count s_N by using, *e.g.*, Elias' technique for arbitrary integers [1]. Then, from the full data set, we compute all the L_i's. If $L_0 < \min\{L_1, ..., L_{N-1}\}$, our criterion states that the data is best encoded as a single piece, and the procedure stops. Otherwise, there is one best partition of the data, say $\{x_k\}_{k=0}^{i-1}$ and $\{x_k\}_{k=i}^{N-1}$. We then transmit i and s_i and apply the criterion to the two segments $\{x_k\}_{k=0}^{i-1}$ and $\{x_k\}_{k=i}^{N-1}$. The receiver can compute the second partial count from s_N (which it already received) and s_i as $s_N - s_i$; *i.e.*, when the procedure is applied to each of the subsegments, the respective lengths and totals were already transmitted. By recursively repeating this procedure independently to the resulting sub-blocks of data we obtain a very efficient recursive scheme of refinement. The process stops when no further splits are indicated by the criterion (*i.e.*, we keep splitting blocks until L_0 is selected for each sub-block). The underlying intensity field estimate is piece-wise constant, with the segments defined by the obtained parsing and the corresponding intensities as the ML estimates inside each segment. Of course, this is a suboptimal scheme, because at each level we are ignoring that each segment will be further subdivided into even smaller pieces, thus achieving an even shorter code length. An optimal scheme would be computationally extremely heavy.

We conclude this section with an illustrative example. As seen in Figure 2, the recursive parsing scheme clearly outperforms the binary tree approach when the underlying intensity is piece-wise constant.

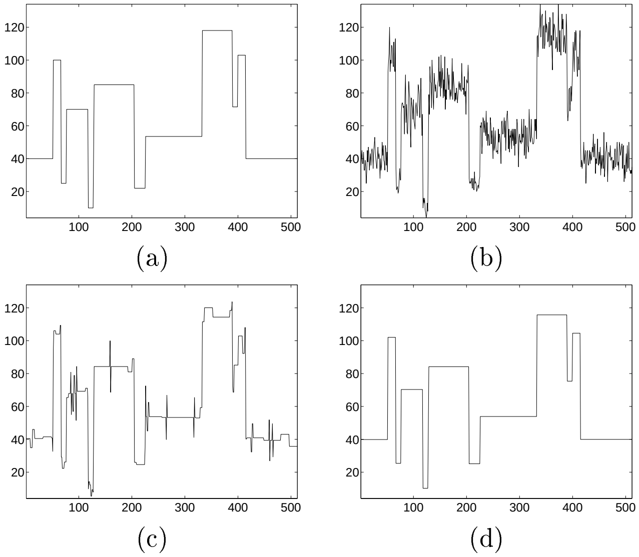
4 Parsing in Two or More Dimensions

The 1D strategies described above are easily extended to 2D (or higher) using rectangular parsing. To illustrate the minor modifications encountered in higher dimensions, we look at both methods in the 2D (image) setting.

4.1 Quadtree-based Image Parsing

In 2D, the Haar multiscale data analysis is as follows. We begin with Poisson data $\{x_{k,l}\}$, $k, l = 0, \ldots, 2^J - 1$, and define $x_{J,k,l} \equiv x_{k,l}$ and for $j = J-1, \ldots, 0$

$$\begin{aligned} x_{j,k,l} &= x_{j+1,2k,2l} + x_{j+1,2k+1,2l} + \\ &\quad x_{j+1,2k,2l+1} + x_{j+1,2k+1,2l+1}. \end{aligned}$$



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

Figure 2: *Example of estimating/parsing a piece-wise constant intensity function from observed counts. (a) Intensity. (b) Counts. (c) Estimate from multiscale binary tree algorithm. (d) Adaptive recursive estimate.*

Here, $j = J$ and $j = 0$ are the highest (finest) and lowest (coarsest) resolutions (scales), respectively. In a progressive coding scheme, at scale $j+1$ we transmit each triple $x_{j+1,2k,2l}$, $x_{j+1,2k+1,2l}$, $x_{j+1,2k,2l+1}$, since the receiver already has the corresponding total $x_{j,k,l}$. The conditional probability $p(x_{j+1,2k,2l}, x_{j+1,2k+1,2l}, x_{j+1,2k,2l+1} | x_{j,k,l})$ is multinomial (rather than binomial, as in the 1D case) with parameters $\rho_{j+1,2k,2l} = \frac{\lambda_{j+1,2k,2l}}{\lambda_{j,k,l}}$, $\rho_{j+1,2k+1,2l} = \frac{\lambda_{j+1,2k+1,2l}}{\lambda_{j,k,l}}$, and $\rho_{j+1,2k,2l+1} = \frac{\lambda_{j+1,2k,2l+1}}{\lambda_{j,k,l}}$, where the $\{\lambda_{j,k,l}\}$ are the intensities underlying the Poisson counts.

Analogous to the 1D case, we consider two alternative models.

Model Class 0 (no split): The ρ's are set to 1/4, requiring no encoding, and the description length is the $-$log of the multinomial probability.

Model Class 1 (split into four): The parameters are coded and transmitted (which, just as in the 1D case, encodes the data itself). The parameters can be transmitted progressively with $\log_2(x_{j,k,l} + 1)$, $\log_2(x_{j,k,l} - x_{j+1,2k,2l} + 1)$, and $\log_2(x_{j,k,l} - x_{j+1,2k,2l} - x_{j+1,2k+1,2l} + 1)$ bits, respectively. The sum is the description length.

This MDL rule can be shown to be equivalent to a Bayesian selection criterion (see Subsection 3.1).

4.2 Adaptive Recursive Image Parsing

In 2D we have more freedom in how we split the data. To maintain a manageable algorithm, we restrict the splitting to rectangular tesselations of the plane. In our recursive scheme, the MDL criterion is applied to rectangular blocks to select one of the following possibilities: **a)** no splitting (the rectangle is considered homogeneous); **b)** the rectangle is split into four subrectangles defined by a common vertex (the best possible such splitting is chosen); and, **c)** the rectangle is split horizontally or vertically into two sub-rectangles (the best possible such splitting is chosen). As in the 1D case, the code lengths for these options are derived from the multinomial probabilities.

We start by applying the criterion to the full image. Every time one rectangular block (the image itself, to start) is split (into 2 or 4 sub-rectangles), the criterion is again applied to the resulting sub-regions. The parsing process stops when no further splits are indicated by the MDL criterion. The final estimate of the intensity field is piece-wise flat, with the rectangular regions defined by the parsing; the corresponding intensities are the ML estimates based on the data inside each region.

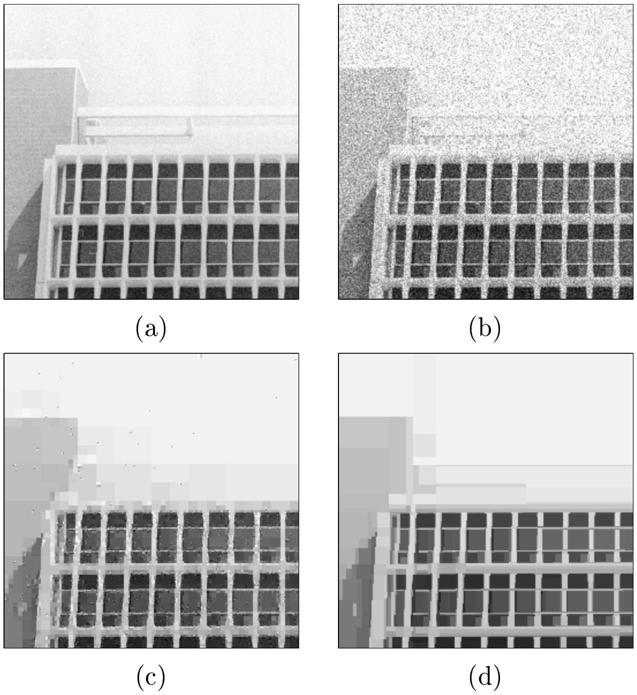
Application of the quadtree and recursive parsing schemes to a natural intensity image is shown below in Figure 3; see also Figure 1. The true intensity is near piece-wise constant, and because the recursive scheme is more adaptive in its selection of parsing regions, it does a better job in this case.

5 Conclusions and Future Work

Our MDL multiscale tree-based parsing scheme is an alternative to the Bayesian methods of [2, 3]. Recall that we have shown that our MDL criterion is, in fact, a special case of a Bayesian approach. The MDL approach, however, has no free parameters; it is fully data-driven. Due to the predictive (coarse-to-fine) nature of the encoding/estimation scheme, we were able to write exact (non-asymptotic) expressions for the parameter code-lengths.

Our adaptive recursive method is related to the "Bayesian Blocks" procedure developed in [4] (the recursive structure is similar); however, the Bayesian selection rule used in [4] differs considerably from the MDL criterion, and only 1D data is considered there.

The 2D methods described here are based on rectangular tesselations. We could use more general refinement schemes based on polygonal region splitting. For example, in the recursive scheme, at each step we could search for the optimal (in MDL sense) line(s) partitioning a given polygon into to smaller polygons.



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

Figure 3: *Parsing a natural image. (a) Intensity. (b) Counts, (normalized) MSE = 1.00. (c) Estimate from multiscale quadtree algorithm, MSE =0.83. (d) Adaptive recursive estimate, MSE = 0.54.*

References

[1] J. Rissanen, *Stochastic Complexity in Stastistical Inquiry.* Singapore: World Scientific, 1989.

[2] K. Timmermann and R. Nowak, "Multiscale modeling and estimation of Poisson processes with application to photon-limited imaging," *IEEE Trans. on Info. Theory*, vol. 45, pp. 846–862, 1999.

[3] E. Kolaczyk, "Bayesian multi-scale models for Poisson processes," *J. Amer. Statist. Assoc.*, Sept., 1999.

[4] J. Scargle, "Studies in astronomical time series analysis. V. Bayesian blocks, a new method to analyze structure in photon counting data," *Astrophysical Journal*, vol. 504, pp. 405–418, 1998.

[5] M. Figueiredo and J. Leitão, "Unsupervised image restoration and edge location using compound GMRFs and the MDL principle," *IEEE Trans. on Image Proc.*, vol. 6, pp. 1089–1102, 1997.

[6] J. Bernardo and A. Smith, *Bayesian Theory.* Chichester, UK: J. Wiley & Sons, 1994.