

IMAGE SEGMENTATION USING FACTOR GRAPHS

Robert J. Drost and Andrew C. Singer

Coordinated Science Laboratory
University of Illinois at Urbana-Champaign

ABSTRACT

Factor graphs were first studied in the context of error correction decoding and have since been shown to be a useful tool in a wide variety of applications. In this paper, we provide a brief introduction to factor graphs with an emphasis on their broad applicability, and then describe a new algorithm for segmenting binary images that have been blurred and corrupted by additive white Gaussian noise. Though the algorithm is developed for this particular class of images, generalizations are immediate. Simulation results detail the performance of the algorithm for images in three separate blurring conditions. The results suggest the potential for this approach, providing additional evidence of the usefulness of the factor graph framework.

1. INTRODUCTION

In recent years, factor graphs have been used to formulate algorithms for a wide variety of applications. Initially studied in the area of error correction decoding, factor graphs have been applied to such problems as equalization, phase unwrapping, network tomography, and Kalman filtering [1][2][3]. The wide applicability of factor graphs can be attributed in part to their fundamental nature: a factor graph expresses the factorization of a global function of many variables into the product of several local functions, each of a subset of the original variables. Whenever such a function arises, a factor graph can be formed and may comprise the basis for a divide-and-conquer approach to the problem at hand. Developing an algorithm on a graph for a particular application may require some deeper insight, but often the sum-product algorithm can be used and produces desirable results. In this paper, we discuss some basic concepts concerning factor graphs and apply this graphical model to the segmentation of a particular class of images.

Image segmentation is an important process in the areas of pattern analysis and machine vision. From the detection and classification of an object to the 3D reconstruction of a scene, image segmentation can play a vital role in a number of applications. Many techniques, including, for example, snakes and deformable templates, have been used to perform segmentation [4][5]. In this paper, we take a statistical approach by considering the probabilistic relationship between an observed image and the corresponding desired segmentation. Factor graphs and the sum-product algorithm yield an efficient algorithm for estimating the underlying segmented image. Simulation results are provided that compare the performance of this algorithm to several other methods and demonstrate the potential for this approach.

This work was supported in part by the Department of the Navy, Office of Naval Research, under grant N00014-01-1-0117 and by the National Science Foundation under grant CCR-0092598 (CAREER).

2. FACTOR GRAPHS

The following discussion is based largely on the material and notation of [1]. Let $X = \{x_1, x_2, \dots, x_n\}$ be a set of variables, and let f be a function of those variables that factors according to

$$f(X) = \prod_{k=1}^m g_k(X_k), \quad (1)$$

for some set of functions $\{g_1, g_2, \dots, g_m\}$, where $X_k \subset X$ for all k and the notation $f(X)$ denotes a function of the variables in set X . The function f is termed the global function, while g_1, g_2, \dots, g_m are termed local functions. A factor graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is then an undirected bipartite graph that expresses this factorization, where the vertex set

$$\mathcal{V} = X \cup \{g_1, g_2, \dots, g_m\}, \quad (2)$$

and the edge set

$$\mathcal{E} = \{(g_k, x_i) : x_i \in X_k\}. \quad (3)$$

Note that while f is often a probability density function, this is not necessarily the case. For example, factor graphs can be used to represent the DFT kernel or the indicator function testing whether or not a codeword belongs to an error correction code [1]. This is a significant difference between factor graphs and many other graphical models, such as Markov random fields (MRFs), which accounts for some of their general applicability. We also note that while the product operation is usually taken as standard pointwise multiplication, one can generalize this to other operations, assuming certain requirements are met, including summation and convolution [6].

While representing a factorization in a factor graph may be interesting, it is of course the algorithms that can be formulated from the graph that makes it a useful tool. Usually such algorithms are stated as message-passing schemes. For example, one is often interested in marginalizing the global function for some set of variables. The sum-product algorithm is a message-passing scheme that efficiently computes these marginals either exactly if the graph is a tree or approximately if the graph contains cycles. The efficiency of the sum-product and related algorithms derives from exploiting a divide-and-conquer strategy.

It was noted in [1] that the sum-product algorithm is one member of a larger class of algorithms that can be obtained by replacing the sum and product operations with any two operations that form a semi-ring, where the second operation depends on the type of factorization a given graph represents. For example, Kalman filtering is derived using an integral-product algorithm, while the Viterbi algorithm can be formulated as a max-sum algorithm.

3. IMAGE SEGMENTATION

Here we provide an application of factor graphs to image segmentation. In particular, we wish to segment an observed image that can be represented as a binary image that has been blurred by a 2D linear filter and corrupted by additive white Gaussian noise. Images obtained in a recent LIDAR experiment in which a submerged object was imaged from above water might be modeled in this way.

Let $Y = \{y_{i,j} : (i,j) \in I\}$ be the set of observed image pixels, and let $X = \{x_{i,j} : (i,j) \in I\}$ be the set of underlying binary pixels, where I is an indexing set. Also, let $\bar{X}_{i,j} = \{x_{i,j}, x_{i+1,j}, x_{i,j+1}, x_{i+1,j+1}\}$, and let $\hat{X}_{i,j}$ denote the smallest subset of X such that the observation $y_{i,j}$ is conditionally independent of $X \setminus \hat{X}_{i,j}$ given $\bar{X}_{i,j}$. This set $\hat{X}_{i,j}$ can be easily determined by locating the nonzero elements of the impulse response of the blurring filter.

We impose the following factorization of the conditional probability density function of X given Y :

$$\begin{aligned} f(X|Y) &= \frac{1}{f(Y)} f(X) f(Y|X) \\ &= \frac{1}{Z} \prod_{(i,j) \in I} \psi_{i,j}(\bar{X}_{i,j}) f(y_{i,j}|\hat{X}_{i,j}), \end{aligned} \quad (4)$$

where the function $\psi_{i,j}$ is formulated based on an a priori preference for connected images, the function $f(y_{i,j}|\hat{X}_{i,j})$ is the conditional probability distribution of $y_{i,j}$ given $\hat{X}_{i,j}$, and Z is a normalization constant. For example, Figure 1 illustrates the factor graph resulting from a blurring filter that has a 2×2 region of support. In actuality, we ignore the constant Z in the factor graph model since the value of Z is not easily determined and will not have an effect on the final result.

To determine an appropriate function for $\psi_{i,j}$, we first assume that the statistics of the segmented image are spatially invariant so that we may set $\psi_{i,j}(\bar{X}_{i,j}) = \psi(\bar{X}_{i,j})$ for all (i,j) . Then, assuming the segmentation region is connected and large relative to the size of a pixel, we note that certain configurations of the pixels in $\bar{X}_{i,j}$ would be most likely, namely those with four pixels either inside or outside the segmentation region. For these configurations, ψ should be the largest. Similarly, certain configurations should rarely, if ever, appear, namely those with $x_{i,j} = x_{i+1,j+1}$, $x_{i+1,j} = x_{i,j+1}$, and $x_{i,j} \neq x_{i+1,j}$. Hence, the value of ψ should be the smallest in these cases. The remaining configurations occur at the edge of the segmentation region and, therefore, should appear at a frequency in between the first two cases. Through empirical means, respective values of 1000, 1, and 10 were assigned to the function ψ for configurations in each of the three cases described. The algorithm does not appear sensitive to small changes in ψ , and the above values seemed adequate irrespective of the particular blurring or SNR used to create the observed image.

Now, for the class of images being considered, it is clear that $f(y_{i,j}|\hat{X}_{i,j})$ will be a conditional Gaussian distribution with a variance equal to the noise variance and a conditional mean that is a linear combination of the pixels in $\hat{X}_{i,j}$. Unfortunately if $\hat{X}_{i,j}$ contains more than a few pixels, then the sum-product algorithm rapidly becomes intractable. Consequently, it may be necessary to approximate the blurring filter with one that contains a few, possibly sparsely spaced, nonzero elements.

Given the factor graph induced by the above factorization, the sum-product algorithm provides, after normalization, estimates of

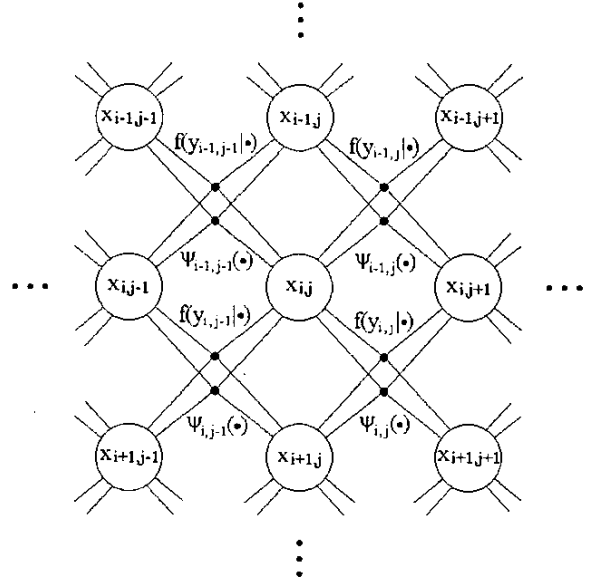


Fig. 1. Factor graph corresponding to a 2×2 blur

the marginal probability distributions $f(x_{i,j}|Y)$. Choosing the most likely value for each $x_{i,j}$ from this distribution is one method for obtaining an estimate for the desired segmentation.

We note that the factorization of $f(Y|X)$ expressed in Equation (4) may also arise when performing image segmentation using MRFs [7]. However, the MRF framework has led to different algorithms to perform the segmentation. For example, in [7], the segmentation is accomplished using a dynamic programming algorithm. When various forms of belief propagation have been used, they have been applied to different formulations of the image segmentation problem, e.g. [8].

4. SIMULATION RESULTS

Here we describe the simulation of the above algorithm on three different test cases. In each case, a binary image of a simply connected object was generated, blurred, and corrupted with additive white Gaussian noise of a range of variances. In the first simulation, the unit impulse function, $\delta(n_1, n_2)$, was used for the blurring filter, i.e. no blurring was introduced. The blurring filter for the second simulation was the FIR filter

$$\begin{aligned} h[n_1, n_2] &= a_1 \delta(n_1, n_2) + a_2 \delta(n_1 + 5, n_2 + 5) \\ &\quad + a_3 \delta(n_1 + 5, n_2 - 5) \\ &\quad + a_4 \delta(n_1 - 5, n_2 + 5) \\ &\quad + a_5 \delta(n_1 - 5, n_2 - 5), \end{aligned} \quad (5)$$

with $a_i = 0.2$ for all i . Since this filter contains only five nonzero taps, the exact factor graph model yields a tractable algorithm for segmentation. In the third simulation, however, a 2D Gaussian shaped pulse was used to blur the image. To obtain reasonable complexity in the factor graph model used to segment the image (not to filter the data), we approximated this impulse response by the function given in Equation (5) with $a_1 = 0.6$ and

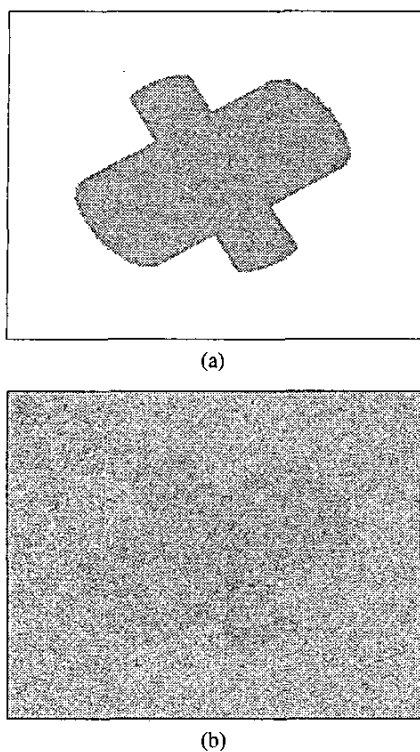


Fig. 2. An example of (a) a silhouette and (b) an observed image used in simulation at an SNR of -5dB.

$a_2 = a_3 = a_4 = a_5 = 0.1$. Figure 2 illustrates a representative example of a binary silhouette used in the simulation along with a corresponding observed image using the second filter with a signal-to-noise ratio (SNR) of -5 dB.

Figure 3 provides three plots that detail the performance of the factor graph algorithm (FG) on the simulated images. The plots give Pe , the ratio of mislabeled pixels to total number of pixels considered, as a function of SNR, where the signal power was computed from the blurred images. For comparison, the plots also detail the performance of image segmentation techniques involving heuristic measures (H), Wiener filtering (WF), and the use of the deterministic least square error 11×11 inverse filter (LS). Note that the last technique uses the actual binary image to compute the filter. Since this image is generally not available, the curve for the LS method should be interpreted as a bound on the performance of any similar technique.

The superiority of the factor graph technique in the first two cases is clear from Figures 3a and 3b. Figure 3c shows that in the Gaussian blur case, the factor graph algorithm again outperforms the other methods at low SNR, but suffers a significant degradation in performance at high SNR. This is probably the result of error induced by the approximation of the Gaussian shaped pulse by a simple five-tap filter. To mitigate the effect of this model inaccuracy, we also simulated the factor graph algorithm in the case of Gaussian blurring at high SNR using a factor graph model that

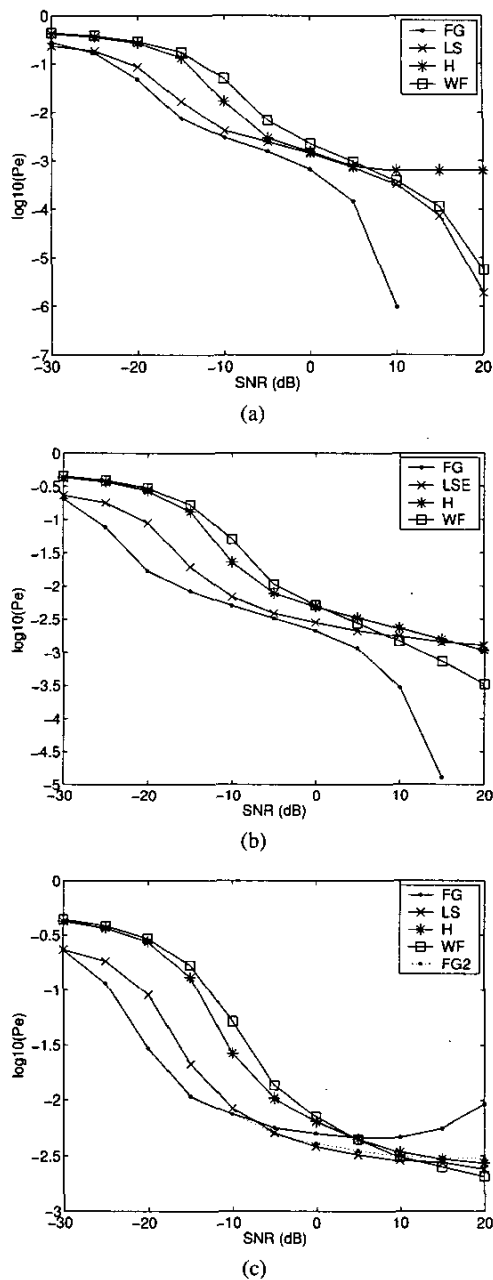


Fig. 3. Image segmentation error for (a) no blur, (b) an FIR blur and (c) a Gaussian pulse shaped blur.

assumes a constant SNR of -5 dB independent of the noise variance. Hence, we model the approximation error as an additional source of additive white Gaussian noise. The resulting plot is shown in Figure 3c and is labeled FG2. A significant improvement is evident.

We also note that most of the curves exhibit three performance regions that result from the difference in the difficulty of correctly classifying pixels that are near segmentation edges versus those that are well within a segmentation region. This is an important observation since the SNR region of interest can vary over a span of more than 20 dB depending on the application. For instance, when using segmented images of an object from multiple views to reconstruct the 3D shape of the object, the basic features of the object may be extracted without accurately classifying those pixels near the segmentation edges. Of course, to obtain highly detailed reconstructions, even those pixels must also be correctly classified.

5. CONCLUSION

In this paper, building on the wide applicability of factor graph models, we provided a new algorithm for image segmentation. The simulation results demonstrated the superiority of this algorithm as compared to several other techniques. Additional improvement in performance might, in some cases, be attained by the development of low complexity factor graph algorithms that tractably allow a more accurate approximation of the impulse response of the blurring filter. Finally, we also note that despite the development of our algorithm for a specific class of images, several generalizations are immediate. For example, it is a straightforward extension to consider the segmentation of an image into multiple regions or to consider a non-Gaussian additive noise model.

6. REFERENCES

- [1] Frank R. Kschischang, Brendan J. Frey, and Hans-Andrea Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Info. Theory*, vol. 47, no. 2, pp. 498-519, February 2001.
- [2] B. J. Frey, R. Koetter, and N. Petrovic, "Codes on images and iterative phase unwrapping," in *Proc. IEEE Information Theory Workshop*, September 2001, pp. 9-11.
- [3] M. Coates and R. Nowak, "Network for networks: Internet analysis using graphical statistical models," in *Proc. IEEE Workshop on Neural Networks for Signal Process.*, December 2000, vol. 2, pp. 755-764.
- [4] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *Int. J. Comput. Vision*, vol. 1, no. 4, pp. 321-331, 1988.
- [5] A. K. Jain, Y. Zhong, and S. Lakshmanan, "Object matching using deformable templates," *IEEE Trans. PAMI*, vol. 18, no. 3, pp. 267-278, March 1996.
- [6] Yongyi Mao and F.R. Kschischang, "On factor graphs and the fourier transform," in *Proc. IEEE Int. Symposium on Info. Theory*, June 2001, p. 224.
- [7] H. Elliot, H. Derin, R. Cristi, and D. Geman, "Application of the gibbs distribution to image segmentation," in *IEEE Int. Conf. on Acoustics, Speech, and Signal Process.*, March 1984, vol. 9, pp. 678-681.
- [8] A. Minagawa, K. Uda, and N. Tagawa, "Region extraction based on belief propagation for gaussian model," in *Proc. 16th Int. Conf. on Pattern Recognition*, August 2002, vol. 2, pp. 507-510.