Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI 2015)

Quiet: Faster Belief Propagation for Images and Related Applications

Yasuhiro Fujiwara† **and Dennis Shasha**‡
†NTT Software Innovation Center, 3-9-11 Midori-cho Musashino-shi, Tokyo, 180-8585, Japan
‡Department of Computer Science, New York University, 251 Mercer Street, New York, NY 10012, USA
fujiwara.yasuhiro@lab.ntt.co.jp, shasha@cs.nyu.edu

Abstract

Belief propagation over Markov random fields has been successfully used in many AI applications since it yields accurate inference results by iteratively updating messages between nodes. However, its high computation costs are a barrier to practical use. This paper presents an efficient approach to belief propagation. Our approach, *Quiet*, dynamically detects converged messages to skip unnecessary updates in each iteration while it theoretically guarantees to output the same results as the standard approach used to implement belief propagation. Experiments show that our approach is significantly faster than existing approaches without sacrificing inference quality.

1 Introduction

Markov random fields is one of the popular graphical models in AI and machine learning [Nakatsuji and Fujiwara, 2014; Fujiwara *et al.*, 2014; Shiokawa *et al.*, 2013; Zhu *et al.*, 2008]. Belief propagation is known to be very effective in performing inference on Markov random fields [Choi and Darwiche, 2008]. It was first applied to infer the label of each node by solving the energy minimization problem [Sun *et al.*, 2003; Weiss and Freeman, 2001]. The label of a node represents some property in the real world such as the same object or the disparity in given images, where nodes correspond to pixels. Belief propagation performs inference by iteratively updating messages between nodes for each label. While belief propagation was originally intended for graphical models without loops, for which it is guaranteed to provide the globally optimal solution, it is empirically effective on a number of loopy graphical models including Markov random fields [Cozman and Polastro, 2008].

Due to its effectiveness and solid theoretical foundation, belief propagation is used by many applications. For example, stereo matching is one of the most popular applications [Ogawara, 2010]. It is exploited to extract 3D information from digital images, and is highly important in the fields of telecommunications and robotics since depth information allows for a system to separate occluding image components [Pérez and Sánchez, 2011]. Another popular application is the computation of optical flow [Lipski *et al.*, 2010]; optical flow is the apparent motion of the brightness pattern between images [Szeliski, 2010]. It can be used in image morphing to change an image into another through a seamless transition in the field of computer art [Lipski *et al.*, 2010]. Other than above, belief propagation is used in a variety of applications such as image restoration [Felzenszwalb and Huttenlocher, 2006], computer-assisted colorization [Noma *et al.*, 2009], and image segmentation [Zhao *et al.*, 2014]. We omit detail descriptions due to the space limitations.

Although belief propagation is effective in various applications, one of the most important research concerns is its speed since a naive approach of belief propagation incurs quadratic computation cost in the number of labels [Sun *et al.*, 2003; Weiss and Freeman, 2001]. If N, K, and T are the number of nodes, labels, and iterations, respectively, it needs $O(NK^2T)$ time. The proposal by Felzenszwalb *et al.* can reduce the computation cost from $O(NK^2T)$ to $O(NKT)$ [Felzenszwalb and Huttenlocher, 2006; 2004]; it is linear in the number of labels. Their approach is now regarded as the standard approach in implementing belief propagation [Pérez and Sánchez, 2011]. However, current belief propagation applications involve many nodes (pixels); this indicates that N can have large a value. In the early 2000's, when belief propagation was first applied to the inference problem [Sun *et al.*, 2003; Weiss and Freeman, 2001], image size was around 640×480 or so. Recently, images of 1280×720 pixels must be processed in real-time to realize 3D telepresence systems [Pérez and Sánchez, 2011]. Moreover, recent optical flow-based image morphing systems must process 1920×1080 images [Lipski *et al.*, 2010]. To increase the processing speed, Ogawara proposed to enhance the standard approach by averaging the outgoing messages from the nodes [Ogawara, 2010]. Although it is more efficient than the standard approach by reducing the number of updated messages, the computation cost of their approach is $O(NKT)$; the same as the standard approach. Furthermore, its inference results are different from those of the standard approach. Sacrificing the inference results makes it difficult to realize truly effective applications [Fujiwara *et al.*, 2013].

This paper proposes *Quiet* a novel, highly-efficient algorithm for belief propagation that provably guarantees to output the same inference results as the standard approach. In order to reduce the computation cost, we dynamically skip unnecessary updates by identifying converged messages in each iteration. This approach reduces the computation cost from $O(NKT)$ to $O(MKT)$, where M is the average number of message updates in the given image and we have $M \ll N$. Experiments demonstrate that our approach cuts the processing time by over 80% from the standard approach.

The remainder of this paper is organized as follows. Section 2 briefly reviews the existing approaches. Section 3 details our approach. Section 4 reviews the results of our experiments. Section 5 provides our conclusion.

2 Preliminary

This section briefly reviews the standard approach by Felzenszwalb *et al.* and the approximate variant by Ogawara. This paper focuses on the pairwise model since the higher-order models are not general to use in applications compared to the pairwise model as described in [Zikic *et al.*, 2010]. However, our approach can be easily applied for other models [Singla *et al.*, 2014; Sato, 2007].

Let $\mathbf{P}$, $\mathbf{L}$, and f be the set of pixels in an image, a set of labels, and a function that assigns label $l_i \in \mathbf{L}$ to each pixel $p \in \mathbf{P}$, respectively. Inference quality of a labeling is given by energy function $E(f)$:

$$E(f) = \textstyle\sum_{p \in \mathbf{P}} D_p(l_i) + \sum_{(p,q) \in \mathbf{N}} V_{pq}(l_i, l_j), \qquad (1)$$

where $\mathbf{N}$ are the neighbor edges in the four-connected image grid. $D_p(l_i)$ is the matching cost of assigning label l_i to pixel p which is set to suit the application. For example, in stereo matching, the matching cost is typically set as follows:

$$D_p(l_i) = \min\{||I_l(x,y) - I_r(x - l_i, y)||, \tau\}. \qquad (2)$$

In this equation, label l_i is a non-negative integer, and $I_l(x,y)$ are the values of pixel $p = (x,y)$ of the left image, and $I_r(x,y)$ the values of pixel $p = (x,y)$ of the right image [Ogawara, 2010]. τ is a positive constant to suppress the effect of outliers. In addition, $V_{pq}(l_i, l_j)$ is the smoothness cost of assigning label l_i and l_j to neighboring pixels p and q, respectively. The most widely used smoothness cost is Potts model, which can prevent edge oversmoothing [Chen *et al.*, 2012]. Potts model takes the following form:

$$V_{pq}(l_i, l_j) = \begin{cases} 0 & (l_i = l_j) \\ d & (l_i \neq l_j). \end{cases} \qquad (3)$$

In this model, the cost is zero for equal labels and a positive constant, d, for different labels (*i.e.*, $d > 0$). Finding a labeling with minimum energy corresponds to the inference problem for the graphical model of the given image as described in the previous papers [Felzenszwalb and Huttenlocher, 2006; 2004]. In order to optimize the labeling, belief propagation iteratively passes messages around the graph according to the connectivity given by the edges the same as other approaches in machine learning [Nakatsuji *et al.*, 2014; Fujiwara and Irie, 2014; Farahmand *et al.*, 2010; Chen *et al.*, 2009]. Each message is a vector whose dimension is given by the number of labels, K. In particular, the max-product algorithm is used to find an approximate minimum cost labeling of the energy function [Fujiwara *et al.*, 2011]. If $m^t_{pq}(l_j)$ is the message of node p to node q for label l_j in the t-th iteration, the message is iteratively computed as follows:

$$m^t_{pq}(l_j) = \min_{l_i}\{V_{pq}(l_i, l_j) + D_p(l_i) + \textstyle\sum_{s \in \mathbf{N}(p) \backslash q} \overline{m}^{t-1}_{sp}(l_i)\}, \qquad (4)$$

where $\mathbf{N}(p) \setminus q$ are the neighbors of node p other than q. In addition, $\overline{m}^t_{pq}(l_j)$ is the normalized message of $m^t_{pq}(l_j)$ which is computed to prevent overflow/underflow [Martin *et al.*, 2011]. Note that $m^t_{pq}(l_j)$ and $\overline{m}^t_{pq}(l_j)$ are initialized to zero. After T iterations, label l^*_q at node q is selected by using the following equation:

$$l^*_q = \mathrm{argmin}_{l_j}\{D_q(l_j) + \textstyle\sum_{p \in \mathbf{N}(q)} \overline{m}^T_{pq}(l_j)\}. \qquad (5)$$

The naive implementation of belief propagation requires $O(NK^2T)$ time; it is quadratic in the number of labels. Therefore, reducing the computation cost of belief propagation is crucial to enhancing its usefulness.

In order to reduce the computation cost, the proposal of Felzenszwalb *et al.* exploits three techniques [Felzenszwalb and Huttenlocher, 2006; 2004]. The first technique computes message updates efficiently. They express Equation (4) as follows with the Potts model of Equation (3):

$$m^t_{pq}(l_j) = \min\{h^{t-1}_p(l_j), \min_{l_i}\{h^{t-1}_p(l_i)\} + d\}, \qquad (6)$$

where $h^t_p(l_i)$ is computed as follows:

$$h^t_p(l_i) = D_p(l_i) + \textstyle\sum_{s \in \mathbf{N}(p) \backslash q} \overline{m}^t_{sp}(l_i). \qquad (7)$$

In Equation (6), the minimization over label l_i is performed once independently of the value of l_j; we can update the message in linear time w.r.t. the number of labels. Their second technique updates messages to every other pixel on even iterations and vice versa on odd iterations. This technique eliminates the need to store messages from the previous iteration when updating current messages. Their multi-level technique runs the belief propagation algorithm in a coarse-to-fine manner. It efficiently computes messages over long-range distances by progressively increasing the resolutions to get estimates for the messages at the next finer level; the b-th level corresponds to a labeling problem where blocks of $2^{b-1} \times 2^{b-1}$ pixels are grouped together ($b \geq 1$). By using the three techniques, their approach reduces the computation cost from $O(NK^2T)$ to $O(NKT)$, and is now regarded as the standard approach to implementing belief propagation [Pérez and Sánchez, 2011].

Ogawara proposed to use approximate values of messages to enhance the processing speed of the standard approach [Ogawara, 2010]. Specifically, their approach updates messages as follows:

$$m^t_{pq}(l_j) = \min_{l_i}\{V(l_i, l_j) + D_p(l_i) + \tfrac{|\mathbf{N}(p)|-1}{|\mathbf{N}(p)|} \textstyle\sum_{s \in \mathbf{N}(p)} \overline{m}^{t-1}_{sp}(l_i)\}, \qquad (8)$$

where $|\mathbf{N}(p)|$ is the number of nodes included in $\mathbf{N}(p)$. In the original algorithm, messages have different values even if they are sent from the same node as shown in Equation (4). However, in the approximate approach, all messages from a node are made identical to each other. Therefore, the number of message updates for a node is reduced to 1 from the $|\mathbf{N}(p)|$ of the original algorithm. However, while their approach is more efficient than the standard approach, it does not guarantee the same inference results as the standard approach. In addition, their approach requires $O(NKT)$ time which is the same as the standard approach.

3 Proposed Approach

This section introduces Quiet, our efficient approach for belief propagation; it gives the same inference results as the standard approach. We first give an overview and then a full description. We also give theoretical analyses and extensions of our approach in this section.

3.1 Adaptive Message Update

The second technique of the standard approach updates messages to every other pixel in even iterations and vice versa in odd iterations as described in the previous section. It splits the nodes into two sets by coloring the pixels of the image in a checkerboard pattern. As a result, messages can be divided into two sets based on the nodes that send the messages. Let $\mathbf{M}$, $\mathbf{M}_e$, and $\mathbf{M}_o$ be the set of all the messages, the set of messages updated at even iterations, and the set of messages updated at odd iterations, respectively; we have $\mathbf{M}_e \cup \mathbf{M}_o = \mathbf{M}$ and $\mathbf{M}_e \cap \mathbf{M}_e = \emptyset$. In Equation (4), messages in $\mathbf{M}_e$ can be computed only by the messages in $\mathbf{M}_o$ and vice versa. The standard approach exploits this property to reduce memory cost. However, the standard approach still incurs high computation cost, especially for images of high-resolution since they entail large numbers of nodes and messages.

We reduce the computational cost of the standard approach by skipping unnecessary message updates. This technique is based on the observation that the exactly same values of messages are repeatedly sent for most pairs in the iterations; we call these messages convergence. By exploiting this property, we dynamically update the messages only if values of the messages can be changed by the updates in each iterations. Since we can skip most messages, we can reduce the computational cost of the standard approach.

This section first defines the set of messages whose values are updated in the current iteration. It next shows three properties of converged messages. Finally, it introduces the property of the set of updated messages by utilizing the three properties.

Let l'_p be the label that gives the minimum value of $h_p^{t-2}(l_i)$, *i.e.*, $l'_p = \mathrm{argmin}_{l_i \in \mathbf{L}}\{h_p^{t-2}(l_i)\}$, and $\mathbf{m}_{pq}^t$ be the set of messages that are sent from node p to node q in the t-th iteration, *i.e.*, $\mathbf{m}_{pq}^t = \{m_{pq}^t(l_j) : l_j \in \mathbf{L}\}$. The set of messages whose values will be updated in the t-th iteration ($t \geq 1$), $\mathbf{U}_t$, is given as follows:

Definition 1 *When $t \neq 1$, message set $\mathbf{m}_{pq}^{t+1}$ is included in $\mathbf{U}_{t+1}$ if at least one of the following conditions holds for message $\overline{m}_{sp}^t(l_i)$;*

$$
\begin{aligned}
&(1)\, \overline{m}_{sp}^t(l_i) \neq \overline{m}_{sp}^{t-2}(l_i) \text{ and } l_i = l'_p,\\
&(2)\, l_i \neq l'_p \text{ and } h_p^t(l_i) < h_p^{t-2}(l'_p) + d, \qquad (9)\\
&(3)\, \overline{m}_{sp}^t(l_i) \neq \overline{m}_{sp}^{t-2}(l_i),\ l_i \neq l'_p, \text{ and } h_p^{t-2}(l_i) < h_p^{t-2}(l'_p) + d,
\end{aligned}
$$

where $\overline{m}_{sp}^t(l_i) \in \mathbf{U}_t$ and $s \in \mathbf{N}(p) \setminus q$. If $t = 1$, we have $\mathbf{U}_t = \mathbf{M}_o$.

In order to describe the property that defines the set of updated messages $\mathbf{U}_t$, we introduce the three cases of converged messages for label $l_i \in \mathbf{L}$ where (1) $l_i = l'_p$, (2) $l_i \neq l'_p$ and $h_p^{t-2}(l_i) \geq h_p^{t-2}(l'_p) + d$, and (3) $l_i \neq l'_p$ and $h_p^{t-2}(l_i) < h_p^{t-2}(l'_p) + d$. Note that these three case are mutually exclusive. The three properties on the converged messages are as follows (proofs are shown in Appendix);

Lemma 1 *For label l'_p, $m_{pq}^{t+1}(l'_p) = m_{pq}^{t-1}(l'_p)$ holds if we have (1) $\overline{m}_{sp}^t(l'_p) = \overline{m}_{sp}^{t-2}(l'_p)$ for all node s such that $s \in \mathbf{N}(p) \setminus q$ and (2) $h_p^t(l_i) \geq h_p^{t-2}(l'_p)$ for all label l_i such that $l_i \neq l'_p$.*

Lemma 2 *For label l_i such that $l_i \neq l'_p$ and $h_p^{t-2}(l_i) \geq h_p^{t-2}(l'_p) + d$, we have $m_{pq}^{t+1}(l_i) = m_{pq}^{t-1}(l_i)$ if (1) $\overline{m}_{sp}^t(l'_p) = \overline{m}_{sp}^{t-2}(l'_p)$ holds for all node s such that $s \in \mathbf{N}(p) \setminus q$ and (2) $h_p^t(l_i) \geq h_p^{t-2}(l'_p) + d$ holds for all label l_i such that $l_i \neq l'_p$.*

Lemma 3 *We have $m_{pq}^{t+1}(l_i) = m_{pq}^{t-1}(l_i)$ for label l_i such that $l_i \neq l'_p$ and $h_p^{t-2}(l_i) < h_p^{t-2}(l'_p) + d$ if we have $\overline{m}_{sp}^t(l_i) = \overline{m}_{sp}^{t-2}(l_i)$ for all node s such that $s \in \mathbf{N}(p) \setminus q$ where (1) $l_i = l'_p$ or (2) $l_i \neq l'_p$ and $h_p^{t-2}(l_i) < h_p^{t-2}(l'_p) + d$.*

Lemma 1, 2, and 3 correspond to the three cases in Definition 1. By exploiting these lemmas, we introduce the property of set of updated messages $\mathbf{U}_t$ as follows:

Lemma 4 *If message $\mathbf{m}_{pq}^{t+1}$ is not included in the set of updated messages $\mathbf{U}_{t+1}$, $m_{pq}^{t+1}(l_j) = m_{pq}^{t-1}(l_j)$ holds for all messages such that $m_{pq}^{t+1}(l_j) \in \mathbf{m}_{pq}^{t+1}$.*

Proof We consider three cases of label $l_i \in \mathbf{L}$ for message $m_{pq}^{t+1}(l_i)$; (1) $l_i = l'_p$, (2) $l_i \neq l'_p$ and $h_p^{t-2}(l_i) \geq h_p^{t-2}(l'_p) + d$, and (3) $l_i \neq l'_p$ and $h_p^{t-2}(l_i) < h_p^{t-2}(l'_p) + d$ which correspond to the three lemmas.

In the first case, from Lemma 1, we have $m_{pq}^{t+1}(l'_p) \neq m_{pq}^{t-1}(l'_p)$ if (1) $\overline{m}_{sp}^t(l'_p) \neq \overline{m}_{sp}^{t-2}(l'_p)$ holds where $s \in \mathbf{N}(p) \setminus q$ or (2) $h_p^t(l_i) < h_p^{t-2}(l'_p)$ holds where $l_i \neq l'_p$. If $h_p^t(l_i) < h_p^{t-2}(l'_p)$ holds, $h_p^t(l_i) < h_p^{t-2}(l'_p) + d$ must hold for label l_i since $d > 0$. Therefore, from the first and second conditions of Definition 1, $\mathbf{m}_{pq}^{t+1}$ must be included in $\mathbf{U}_{t+1}$ if at least one message included in $\mathbf{m}_{pq}^{t+1}$ is not converged in the $(t+1)$-th iteration.

In the second case, we have $m_{pq}^{t+1}(l_i) \neq m_{pq}^{t-1}(l_i)$ if (1) $\overline{m}_{sp}^t(l'_p) \neq \overline{m}_{sp}^{t-2}(l'_p)$ where $s \in \mathbf{N}(p) \setminus q$ or (2) $h_p^t(l_i) < h_p^{t-2}(l'_p) + d$ holds from Lemma 2. Therefore, if at least one message included in $\mathbf{m}_{pq}^{t+1}$ is not converged in the $(t+1)$-th iteration, $\mathbf{m}_{pq}^{t+1}$ must be included in $\mathbf{U}_{t+1}$ from the first and second conditions of Definition 1.

In the third case, $m_{pq}^{t+1}(l_i) \neq m_{pq}^{t-1}(l_i)$ holds if we have $\overline{m}_{sp}^t(l_i) \neq \overline{m}_{sp}^{t-2}(l_i)$ for node s such that $s \in \mathbf{N}(p) \setminus q$ where (1) $l_i = l'_p$ or (2) $l_i \neq l'_p$ and $h_p^{t-2}(l_i) < h_p^{t-2}(l'_p) + d$ from Lemma 3. As a result, from the first and third conditions of Definition 1, $\mathbf{m}_{pq}^{t+1}$ must be included in $\mathbf{U}_{t+1}$ if at least one message included in $\mathbf{m}_{pq}^{t+1}$ is not converged in the $(t+1)$-th iteration. □

Lemma 4 indicates that, a message can change its value in the $(t+1)$-th iteration only if the message is included in the set of updated messages $\mathbf{U}_{t+1}$. As shown in Definition 1, we can compute $\mathbf{U}_{t+1}$ without messages in the $(t+1)$-th iteration. Therefore, we can dynamically identify messages whose values are to be updated in the $(t+1)$-th iteration from the messages of the t-th iteration which makes it possible to improve the efficiency.

Algorithm 1 Quiet

1: **for** $b = B$ to 1 **do**
2:   **for** each message **do**
3:     **if** $b = B$ **then**
4:       initialize the message to zero;
5:     **else**
6:       initialize the message by the results of the previous level;
7:   $\mathbf{U}_1 = \mathbf{M}_o$;
8:   **for** $t = 1$ to T **do**
9:     **for** each $m^t_{sp}(l_i) \in \mathbf{U}_t$ **do**
10:       compute $m^t_{sp}(l_i)$ from Equation (4);
11:       compute $\overline{m}^t_{sp}(l_i)$ for $m^t_{sp}(l_i)$;
12:     $\mathbf{U}_{t+1} = \emptyset$;
13:     **for** each $\overline{m}^t_{sp}(l_i)$ such that $m^t_{sp}(l_i) \in \mathbf{U}_t$ **do**
14:       **if** $\overline{m}^t_{sp}(l_i) \neq \overline{m}^{t-2}_{sp}(l_i)$ and $l_i = l'_p$ **then**
15:         add $\mathbf{m}^{t+1}_{pq}$ to $\mathbf{U}_{t+1}$ where $q \in \mathbf{N}(p) \setminus s$;
16:       **if** $l_i \neq l'_p$ and $h^t_p(l_i) < h^{t-2}_p(l'_p) + d$ **then**
17:         add $\mathbf{m}^{t+1}_{pq}$ to $\mathbf{U}_{t+1}$ where $q \in \mathbf{N}(p) \setminus s$;
18:       **if** $\overline{m}^t_{sp}(l_i) \neq \overline{m}^{t-2}_{sp}(l_i)$, $l_i \neq l'_p$, and $h^{t-2}_p(l_i) < h^{t-2}_p(l'_p) + d$ **then**
19:         add $\mathbf{m}^{t+1}_{pq}$ to $\mathbf{U}_{t+1}$ where $q \in \mathbf{N}(p) \setminus s$;
20: **for** each node **do**
21:   determine the label of the node from Equation (5);

3.2 Algorithm

Algorithm 1 gives the full description of our approach, Quiet. Similar to the previous approximate approach [Ogawara, 2010], it is based on the standard approach [Felzenszwalb and Huttenlocher, 2006; 2004] which utilizes the multi-level technique. In Algorithm 1, B is the number of levels. Note that, in the b-th level, blocks of $2^{b-1} \times 2^{b-1}$ are grouped together as described in Section 2. Algorithm 1 first initializes the messages to zero if $b = B$ (lines 3-4). Otherwise, it initializes the messages by the results of the previous level (lines 5-6). The details of these processes are shown in the previous papers [Felzenszwalb and Huttenlocher, 2006; 2004]. In the iterations, it updates messages that are included in $\mathbf{U}_t$ from Equation (4), and it normalizes messages (lines 9-11). By using the normalized message, it determines the set of updated messages $\mathbf{U}_{t+1}$. If $\overline{m}^t_{sp}(l_i) \neq \overline{m}^{t-2}_{sp}(l_i)$ holds where $l_i = l'_p$, message set $\mathbf{m}^{t+1}_{pq}$ is included in the set of updated messages $\mathbf{U}_{t+1}$ from the first condition of Definition 1 (lines 14-15). In addition, if $h^t_p(l_i) < h^{t-2}_p(l'_p) + d$ holds where $l_i \neq l'_p$, $\mathbf{m}^{t+1}_{pq}$ is added to $\mathbf{U}_{t+1}$ from the second condition of Definition 1 (lines 16-17). Similarly, if we have $\overline{m}^t_{sp}(l_i) \neq \overline{m}^{t-2}_{sp}(l_i)$, $l_i \neq l'_p$, and $h^{t-2}_p(l_i) < h^{t-2}_p(l'_p) + d$, $\mathbf{m}^{t+1}_{pq}$ is added to $\mathbf{U}_{t+1}$ from the third condition of Definition 1 (lines 18-19). After the iterations, it determines the label of each node (lines 20-21).

3.3 Theoretical Analyses

We introduce the properties of our approach in this section. The first property is the inference results of our approach:

Theorem 1 *Algorithm 1 outputs exactly the same results as the standard algorithm of belief propagation.*

Proof Prior to proving Theorem 1, we first prove that we have $\mathbf{U}_t \subseteq \mathbf{M}_o$ in the odd iterations and $\mathbf{U}_t \subseteq \mathbf{M}_e$ in the even iterations by mathematical induction.
Initial step: We have $\mathbf{U}_1 = \mathbf{M}_o$ from Definition 1. As shown in Definition 1 and Lemma 4, a message is included in $\mathbf{U}_{t+1}$ only if the message can change its value by update in the t-th iteration. In addition, messages included in $\mathbf{M}_e$ are updated from messages included in $\mathbf{M}_o$ by using Equation (4). As a result, since $\mathbf{U}_1 = \mathbf{M}_o$, it is clear that $\mathbf{U}_2 \subseteq \mathbf{M}_e$.
Inductive step: From Lemma 4, if a message is not included in $\mathbf{U}_t$, the message cannot change its value in the t-th iteration. Therefore, we can identify unconverged messages in the $(t+1)$-th iteration from $\mathbf{U}_t$. In addition, messages in $\mathbf{M}_o$ can be computed only by the messages in $\mathbf{M}_e$ and vice versa. Thus, it is clear that we have $\mathbf{U}_{t+1} \subseteq \mathbf{M}_o$ and $\mathbf{U}_{t+1} \subseteq \mathbf{M}_e$ in the odd and even iterations, respectively, if $\mathbf{U}_t \subseteq \mathbf{M}_o$ and $\mathbf{U}_t \subseteq \mathbf{M}_e$ hold in the odd and even iterations, respectively.

Therefore, from the result of the mathematical induction and from Lemma 4, we can compute messages exactly the same as the standard approach which computes messages included in $\mathbf{M}_o$ and $\mathbf{M}_e$ in even and odd iterations. Consequently, our approach provably guarantees the same inference results as the standard approach. □

As described in Section 2, the standard approach needs $O(NKT)$ time. We have the following property for the computation cost of our approach:

Theorem 2 *If M is the average number of updated messages in the given image, Algorithm 1 requires $O(MKT)$ time.*

Proof As shown in Algorithm 1, our approach iteratively updates and normalizes messages; these processes need $O(MKT)$ time. It then determines the set of updated messages $\mathbf{U}_{t+1}$. As shown in Algorithm 1, we can compute $\mathbf{U}_{t+1}$ from the messages included in $\mathbf{U}_t$. Thus, this process also requires $O(MKT)$ time. As a result, our approach needs $O(MKT)$ time. □

Theorem 1 and 2 indicate that we can obtain the inference results more efficiently than the standard approach without sacrificing the inference results.

3.4 Extension of Quiet

The previous sections assumed the use of the Potts model and the max-product algorithm for belief propagation. This section briefly describes the extension of our techniques through the use of other models and the sum-product algorithm.

Even though the Potts model is the most popular approach in many applications to the smoothness cost used in Markov random fields [Chen *et al.*, 2012], we can also use the truncated linear model and the truncated quadratic model, both are very well known and are given as follows [Felzenszwalb and Huttenlocher, 2006; 2004]:

$$V_{pq}(l_i, l_j) = \min\{c||l_i - l_j||, d\}, \tag{10}$$

$$V_{pq}(l_i, l_j) = \min\{c||l_i - l_j||^2, d\}, \tag{11}$$

where c is the rate of increase in the cost, and d determines when the increase stops. As shown in Equation (10), the truncated linear model increases the cost linearly based on the amount of difference between label l_i and l_j while the truncated quadratic model quadratically increases the cost as shown in Equation (11). For these models, our approach determines the set of updated messages as follows:

$$\mathbf{U}_{t+1} = \{\mathbf{m}^{t+1}_{pq} | \exists \overline{m}^t_{sp}(l_i) \text{ s.t. } s \in \mathbf{N}(p) \setminus q \wedge \overline{m}^t_{sp}(l_i) \neq \overline{m}^{t-2}_{sp}(l_i)\}, \tag{12}$$

where $l_i \in \mathbf{L}$. By using this technique, we can obtain the same inference results as the standard approach in $O(MKT)$ time although we omit the proof due to space limitations.

While the max-product algorithm based belief propagation is utilized in many applications [Ogawara, 2010; Noma *et al.*, 2009], the sum-product algorithm can be used for belief propagation. The sum-product based belief propagation computes the posterior probability by selecting the most probable label for each pixel. It updates the messages as follows:

$$m^t_{pq}(l_j) = \sum_{l_i}\{\Psi_{pq}(l_i, l_j)\Phi_p(l_i)\prod_{s\in\mathbf{N}(p)\backslash q}\overline{m}^{t-1}_{sp}(l_i)\}, \quad (13)$$

where $\Psi_{pq}(l_i,l_j) = e^{-V_{pq}(l_i,l_j)}$ and $\Phi_p(l_i) = e^{-D_p(l_i)}$. The standard approach updates the message at $O(NKT)$ time in the following manner by using the box sum method as described in the previous paper [Wells, 1986]:

$$m^t_{pq}(l_j) = \sum_{l_i}\{\Psi_{pq}(l_i, l_j)g^{t-1}_p(l_i)\}, \quad (14)$$

where $g^{t-1}_p(l_i) = \Phi_p(l_i)\prod_{s\in\mathbf{N}(p)\backslash q}\overline{m}^{t-1}_{sp}(l_i)$. Equation (13) and (14) indicate that the sum-product algorithm updates messages by computing the sums of messages instead of taking the minima of messages as done by the max-product algorithm. Therefore, it is clear that our approach can be directly exploited to enhance the processing speed for sum-product algorithm based belief propagation.

4 Experimental Evaluation

We perform stereo matching to compare our approach to the standard approach [Felzenszwalb and Huttenlocher, 2006; 2004] and the approximate approach [Ogawara, 2010]. We used Art, Moebius, Shopvac, Flowers, and Umbrella images obtained from the Middlebury Stereo Datasets[1]; their sizes are 1390×1110, 1390×1110, 2356×1996, 2772×1980, 2880×1980, and 2960×2016, respectively. The six images are shown in Figure 1. In this section, "Quiet", "Standard", and "Approximate" represent the results of our approach, the standard approach, and the approximate approaches, respectively. In the experiments, we set the number of labels, $K = 100$, the number of iterations in each level, $T = 50$, the number of levels, $B = 4$, and the parameter of the Potts model, $d = 20$, by following the previous paper [Felzenszwalb and Huttenlocher, 2004]. All experiments were conducted on a Linux 2.70 GHz Intel Xeon server.

4.1 Efficiency

We evaluated the processing time of each approach. Figure 2 shows the results. In addition, Figure 3 shows the numbers updated messages in the iterations for Art dataset and Table 1 details the numbers of messages updated by each approach where $b = 1$. Since we dynamically skip unnecessary message updates in the iterations, Table 1 shows the average numbers of updated messages for our approach.

Figure 2 shows that our approach is much faster than the previous approaches; our approach cuts the processing time by up to 82% and 75% from the standard and the approximate approaches, respectively. As described in Section 2,

[1] http://vision.middlebury.edu/stereo/data/


(a) Art

(b) Moebius

(c) Shopvac

(d) Storage

(e) Flowers

(f) Umbrella

Figure 1: Dataset.

the standard approach updates all messages in each iteration. In addition, the approximate approach reduces the number of updated messages by using approximate values for messages to raise inference efficiency. However, the approximate approach incurs the same computation cost as described in Section 2. On the other hand, since most messages reach convergence in a first few iterations, our approach effectively limits the updated messages whereas previous approaches update constant numbers of messages in the iterations as shown in Figure 3. As a result, our approach reduces the number of updated messages in the iteration by detecting the converged messages as shown in Table 1. Thus, our approach has better processing speed than the previous approaches.

4.2 Effectiveness

One major advantage of our approach is that it outputs the same results as the standard approach of belief propagation while the approximate approach yields different results. In this section, we evaluated the precision of the inference results by the proposed approach and the approximate approach against the standard approach. Precision is the fraction of inference results of an approach that match the inference results of the standard approach. Precision takes a value between 0 and 1; precision is 1 if the inference results are identical to those of the standard approach. We show precision of each approach in Figure 4.

Figure 4 shows, as expected, the precision of our approach is 1 under all conditions examined. This is because our approach has the theoretical property that the inference results are same as the standard approach, see Theorem 1. In contrast, the approximate approach has precision under 1; the approximate approach and the standard approach output different inference results. This is because, in the approximate approach, all the messages from the same node are made identical to each other, even though the messages actually have different values in the standard approach.

Figure 4, along with Figure 2, indicates that the approximate approach enhances the processing speed by sacrificing the quality of inference results. On the other hand, our approach achieves higher processing speed than the previous approaches while its inference results replicate those of the standard approach. This indicates that our approach is an attractive option for the research community in performing belief propagation.

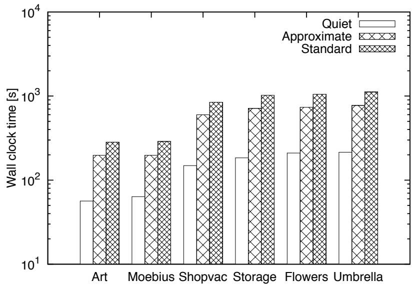

Figure 2: Processing time of each approach.

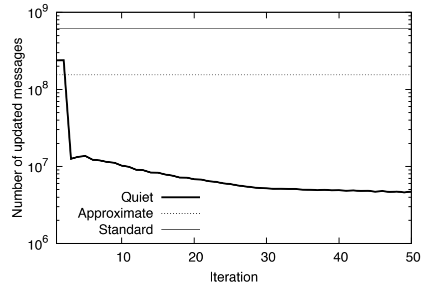

Figure 3: Numbers of updated messages in the iterations.

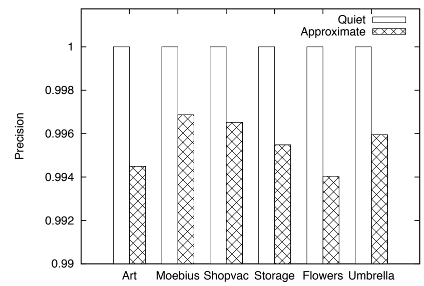

Figure 4: Precision of the inference results against the standard approach.

Table 1: Average numbers of updated messages ($b = 1$).

Dataset	Approach		
	Quiet	Approximate	Standard
Art	1.46×10^7	1.54×10^8	6.17×10^8
Moebius	1.58×10^7	1.54×10^8	6.17×10^8
Shopvac	3.99×10^7	4.70×10^8	1.88×10^9
Storage	4.70×10^7	5.49×10^8	2.20×10^9
Flowers	5.25×10^7	5.70×10^8	2.28×10^9
Umbrella	5.32×10^7	5.97×10^8	2.39×10^9

5 Conclusions

In this paper, we proposed an efficient algorithm that gives the same inference results as the standard approach to belief propagation. The proposed approach skips unnecessary updates by detecting the converged messages in each iteration to enhance the processing speed. Experiments showed that the proposed approach achieves high efficiency without sacrificing inference quality; our approach outputs the same results as the standard approach unlike the approximate approach. The proposed approach will allow many applications to be processed more efficiently, and help to improve the effectiveness of future applications.

Appendix

In this section, we show proofs of Lemma 1, 2, and 3.

Lemma 1

Proof From Equation (6), we have

$$m_{pq}^{t-1}(l'_p)=\min\{h_p^{t-2}(l'_p),\min_{l_i}\{h_p^{t-2}(l_i)\}+d\}=h_p^{t-2}(l'_p), \quad (15)$$

since we have $\min_{l_i}\{h_p^{t-2}(l_i)\} = h_p^{t-2}(l'_p)$ from the definition of l'_p.

In addition, from Equation (7), if we have $\overline{m}_{sp}^{t}(l'_p) = \overline{m}_{sp}^{t-2}(l'_p)$ for all s such that $s \in \mathbf{N}(p)\backslash q$,

$$\begin{aligned} h_p^t(l'_p) &= D_p(l'_p) + \textstyle\sum_{s\in\mathbf{N}(p)\backslash q}\overline{m}_{sp}^{t}(l'_p) \\ &= D_p(l'_p) + \textstyle\sum_{s\in\mathbf{N}(p)\backslash q}\overline{m}_{sp}^{t-2}(l'_p) = h_p^{t-2}(l'_p). \end{aligned} \quad (16)$$

Therefore, from Equation (6), we have

$$\begin{aligned} m_{pq}^{t+1}(l'_p) &= \min\{h_p^t(l'_p),\min_{l_i}\{h_p^t(l_i)\}+d\} \\ &= \min\{h_p^{t-2}(l'_p),\min_{l_i\neq l'_p}\{h_p^t(l_i)\}+d\}. \end{aligned} \quad (17)$$

If $h_p^t(l_i) \geq h_p^{t-2}(l'_p)$ holds for all label l_i such that $l_i \neq l'_p$,

$$\min_{l_i\neq l'_p}\{h_p^t(l_i)\}+d \geq h_p^{t-2}(l'_p)+d. \quad (18)$$

As a result, $m_{pq}^{t+1}(l'_p) = h_p^{t-2}(l'_p)$ from Equation (17) and (18). Therefore, from Equation (15), we have $m_{pq}^{t+1}(l'_p) = h_p^{t-2}(l'_p) = m_{pq}^{t-1}(l'_p)$. □

Lemma 2

Proof From Equation (6) and the definition of l'_p, we have

$$m_{pq}^{t-1}(l_i)=\min\{h_p^{t-2}(l_i),\min_{l_k}\{h_p^{t-2}(l_k)\}+d\}=h_p^{t-2}(l'_p)+d, \quad (19)$$

for label l_i such that $l_i\neq l'_p$ and $h_p^{t-2}(l_i)\geq h_p^{t-2}(l'_p)+d$.

In addition, for all such label l_i, if $h_p^t(l_i)\geq h_p^{t-2}(l'_p)+d$,

$$\min_{l_i\neq l'_p}\{h_p^t(l_i)\}\geq h_p^{t-2}(l'_p)+d > h_p^{t-2}(l'_p), \quad (20)$$

since $d > 0$ holds. As a result, we have

$$\begin{aligned} &\min_{l_k}\{h_p^t(l_k)\}+d = \\ &\min\{h_p^{t-2}(l'_p),\min_{l_i\neq l'_p}\{h_p^t(l_i)\}\}+d = h_p^{t-2}(l'_p)+d, \end{aligned} \quad (21)$$

since $h_p^t(l'_p) = h_p^{t-2}(l'_p)$ holds if $\overline{m}_{sp}^{t}(l'_p) = \overline{m}_{sp}^{t-2}(l'_p)$ for all node s such that $s \in \mathbf{N}(p)\backslash q$ from Equation (7). Therefore, if we have $h_p^t(l_i) \geq h_p^{t-2}(l'_p)+d$,

$$\begin{aligned} m_{pq}^{t+1}(l_i) &= \min\{h_p^t(l_i),\min_{l_k}\{h_p^t(l_k)\}+d\} \\ &= \min\{h_p^t(l_i), h_p^{t-2}(l'_p)+d\} = h_p^{t-2}(l'_p)+d, \end{aligned} \quad (22)$$

from Equation (6). As a result, from Equation (19) and (22), $m_{pq}^{t+1}(l'_p) = h_p^{t-2}(l'_p)+d = m_{pq}^{t-1}(l'_p)$ holds. □

Lemma 3

Proof If the condition of Lemma 3 holds, for all s such that $s \in \mathbf{N}(p)\backslash q$, we have

$$\begin{aligned} h_p^t(l_i) &= D_p(l_i) + \textstyle\sum_{s\in\mathbf{N}(p)\backslash q}\overline{m}_{sp}^{t}(l_i) \\ &= D_p(l_i) + \textstyle\sum_{s\in\mathbf{N}(p)\backslash q}\overline{m}_{sp}^{t-2}(l_i) = h_p^{t-2}(l_i), \end{aligned} \quad (23)$$

where (1) $l_i = l'_p$ or (2) $l_i \neq l'_p$ and $h_p^{t-2}(l_i) < h_p^{t-2}(l'_p)+d$ from Equation (7). Therefore, from Equation (6),

$$\begin{aligned} m_{pq}^{t+1}(l_i) &= \min\{h_p^t(l_i),\min_{l_k}\{h_p^t(l_k)\}+d\} \\ &= \min\{h_p^{t-2}(l_i), h_p^{t-2}(l'_p)+d\} = h_p^{t-2}(l_i), \end{aligned} \quad (24)$$

since $h_p^{t-2}(l_i) < h_p^{t-2}(l'_p)+d$. In addition, from Equation (6),

$$m_{pq}^{t-1}(l_i)=\min\{h_p^{t-2}(l_i),\min_{l_k}\{h_p^{t-2}(l_k)\}+d\}=h_p^{t-2}(l_i), \quad (25)$$

if $h_p^{t-2}(l_i) < h_p^{t-2}(l'_p)+d$. Consequently, from Equation (24) and (25), we have $m_{pq}^{t+1}(l_i) = h_p^{t-2}(l_i) = m_{pq}^{t-1}(l_i)$. □

References

[Chen *et al.*, 2009] Huanhuan Chen, Peter Tiño, and Xin Yao. Predictive ensemble pruning by expectation propagation. *IEEE Trans. Knowl. Data Eng.*, 21(7):999–1013, 2009.

[Chen *et al.*, 2012] S. Y. Chen, Hanyang Tong, and Carlo Cattani. Markov models for image labeling. *Mathematical Problems in Engineering*, 2012.

[Choi and Darwiche, 2008] Arthur Choi and Adnan Darwiche. Focusing generalizations of belief propagation on targeted queries. In *AAAI*, pages 1024–1030, 2008.

[Cozman and Polastro, 2008] Fabio Gagliardi Cozman and Rodrigo Bellizia Polastro. Loopy propagation in a probabilistic description logic. In *SUM*, pages 120–133, 2008.

[Farahmand *et al.*, 2010] Amir Massoud Farahmand, Rémi Munos, and Csaba Szepesvári. Error propagation for approximate policy and value iteration. In *NIPS*, pages 568–576, 2010.

[Felzenszwalb and Huttenlocher, 2004] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Efficient belief propagation for early vision. In *CVPR*, pages 261–268, 2004.

[Felzenszwalb and Huttenlocher, 2006] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Efficient belief propagation for early vision. *International Journal of Computer Vision*, 70(1):41–54, 2006.

[Fujiwara and Irie, 2014] Yasuhiro Fujiwara and Go Irie. Efficient label propagation. In *ICML*, pages 784–792, 2014.

[Fujiwara *et al.*, 2011] Yasuhiro Fujiwara, Go Irie, and Tomoe Kitahara. Fast algorithm for affinity propagation. In *IJCAI*, pages 2238–2243, 2011.

[Fujiwara *et al.*, 2013] Yasuhiro Fujiwara, Makoto Nakatsuji, Hiroaki Shiokawa, Takeshi Mishima, and Makoto Onizuka. Fast and exact top-k algorithm for pagerank. In *AAAI*, 2013.

[Fujiwara *et al.*, 2014] Yasuhiro Fujiwara, Go Irie, Shari Kuroyama, and Makoto Onizuka. Scaling manifold ranking based image retrieval. *PVLDB*, 8(4):341–352, 2014.

[Lipski *et al.*, 2010] Christian Lipski, Christian Linz, and Marcus A. Magnor. Belief propagation optical flow for high-resolution image morphing. In *SIGGRAPH*, 2010.

[Martin *et al.*, 2011] Victorin Martin, Jean-Marc Lasgouttes, and Cyril Furtlehner. The role of normalization in the belief propagation algorithm. *CoRR*, abs/1101.4170, 2011.

[Nakatsuji and Fujiwara, 2014] Makoto Nakatsuji and Yasuhiro Fujiwara. Linked taxonomies to capture users' subjective assessments of items to facilitate accurate collaborative filtering. *Artif. Intell.*, 207:52–68, 2014.

[Nakatsuji *et al.*, 2014] Makoto Nakatsuji, Yasuhiro Fujiwara, Hiroyuki Toda, Hiroshi Sawada, Jin Zheng, and James Alexander Hendler. Semantic data representation for improving tensor factorization. In *AAAI*, pages 2004–2012, 2014.

[Noma *et al.*, 2009] Alexandre Noma, Luiz Velho, and Roberto M. Cesar. A computer-assisted colorization approach based on efficient belief propagation and graph matching. In *CIARP*, pages 345–352, 2009.

[Ogawara, 2010] Koichi Ogawara. Approximate belief propagation by hierarchical averaging of outgoing messages. In *ICPR*, pages 1368–1372, 2010.

[Pérez and Sánchez, 2011] Jesús M. Pérez and Pablo Sánchez. Real-time stereo matching using memory-efficient belief propagation for high-definition 3d telepresence systems. *Pattern Recognition Letters*, 32(16):2250–2253, 2011.

[Sato, 2007] Taisuke Sato. Inside-outside probability computation for belief propagation. In *IJCAI*, pages 2605–2610, 2007.

[Shiokawa *et al.*, 2013] Hiroaki Shiokawa, Yasuhiro Fujiwara, and Makoto Onizuka. Fast algorithm for modularity-based graph clustering. In *AAAI*, 2013.

[Singla *et al.*, 2014] Parag Singla, Aniruddh Nath, and Pedro Domingos. Approximate lifting techniques for belief propagation. In *AAAI*, pages 2497–2504, 2014.

[Sun *et al.*, 2003] Jian Sun, Nanning Zheng, and Heung-Yeung Shum. Stereo matching using belief propagation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(7):787–800, 2003.

[Szeliski, 2010] Richard Szeliski. *Computer Vision: Algorithms and Applications*. Springer, 2010.

[Weiss and Freeman, 2001] Yair Weiss and William T. Freeman. On the optimality of solutions of the max-product belief-propagation algorithm in arbitrary graphs. *IEEE Transactions on Information Theory*, 47(2):736–744, 2001.

[Wells, 1986] William M. Wells. Efficient synthesis of gaussian filters by cascaded uniform filters. *IEEE Trans. Pattern Anal. Mach. Intell.*, 8(2):234–239, 1986.

[Zhao *et al.*, 2014] Liang Zhao, Sheng-Jun Xu, Jun Lu, and Deng-Feng Chen. Local minimum energy of lbp algorithm concrete ct image segmentation. *International Journal of Signal Processing, Image Processing and Pattern Recognition*, 7(4):177–186, 2014.

[Zhu *et al.*, 2008] Jun Zhu, Zaiqing Nie, Bo Zhang, and Ji-Rong Wen. Dynamic hierarchical markov random fields for integrated web data extraction. *Journal of Machine Learning Research*, 9:1583–1614, 2008.

[Zikic *et al.*, 2010] Darko Zikic, Ben Glocker, Oliver Kutter, Martin Groher, Nikos Komodakis, Ali Kamen, Nikos Paragios, and Nassir Navab. Linear intensity-based image registration by markov random fields and discrete optimization. *Medical Image Analysis*, 14(4):550–562, 2010.