Honda Research Institute EU
Honda Research Institute Europe GmbH
https://www.honda-ri.de/

Belief Propagation in Spatiotemporal Graph Topologies for the Analysis of Image Sequences

Volker Willert, Julian Eggert

2010

Preprint:

This is an accepted article published in Proceedings of the VISAPP. The final authenticated version is available online at: https://doi.org/[DOI not available]

BELIEF PROPAGATION IN SPATIOTEMPORAL GRAPH TOPOLOGIES FOR THE ANALYSIS OF IMAGE SEQUENCES

Volker Willert
Control Theory and Robotics Lab, TU Darmstadt, Landgraf-Georg-Str. 4, 64283 Darmstadt, Germany
vwillert@rtr.tu-darmstadt.de

Julian Eggert
Cognitive Control Group, Honda Research Institute, Carl-Legien-Str. 30, 63073 Offenbach, Germany
julian.eggert@honda-ri.de

Keywords: Early Vision, Dynamical Vision, Belief Propagation, MRF, DBN, Denoising

Abstract: Belief Propagation (BP) is an efficient approximate inference technique both for Markov Random Fields (MRF) and Dynamic Bayesian Networks (DBN). 2D MRFs provide a unified framework for early vision problems that are based on static image observations. 3D MRFs are suggested to cope with dynamic image data. To the contrary, DBNs are far less used for dynamic low level vision problems even though they represent sequences of state variables and hence are suitable to process image sequences with temporally changing visual information. In this paper, we propose a 3D DBN topology for dynamic visual processing with a product of potentials as transition probabilities. We derive an efficient update rule for this 3D DBN topology that unrolls loopy BP for a 2D MRF over time and compare it to update rules for conventional 3D MRF topologies. The advantages of the 3D DBN are discussed in terms of memory consumptions, costs, convergence and online applicability. To evaluate the performance of infering visual information from dynamic visual observations, we show examples for image sequence denoising that achieve MRF-like accuracy on real world data.

1 INTRODUCTION

Probabilistic graphical models like 2D grid-based Markov Random Fields (Szeliski et al., 2008) combined with approximate inference techniques like Belief Propagation (Pearl, 1988) are successfully used for several kinds of early vision tasks like image denoising, stereo vision and optical flow computation. The main ability of a MRF is to robustly infer hidden states of a visual scene like pixel depth or movement based on local noisy image measurements. The core of a 2D MRF for low level visual processing is the resolution of estimation ambiguities via incorporation of models describing the spatial context between neighboring pixels. At the moment the vision community investigates in extending these models to improve robustness and estimation accuracy (Roth, 2007; Ramalingam et al., 2008; Komodakis and Paragios, 2009) and develops fast approximate inference techniques to head for realtime applications (Szeliski et al., 2008; Felzenszwalb and Huttenlocher, 2006; Petersen et al., 2008). Another important aspect concerning online visual knowledge extraction is the dynamics of visual scenes. Hence, temporal changes in the observed visual data and the fact that future observations cannot be accessed have to be considered. To keep all the properties of a 2D MRF but also model a whole sequence of images, the 3D MRF was introduced by (Williams et al., 2005) and has further been investigated by several researchers (Yin and Collins, 2007; Larsen et al., 2007; Chen and Tang, 2007; Huang et al., 2008). Here, 2D MRFs like in figure 1 a) that model the spatial relations within a single time slice are stacked into a 3D spatiotemporal MRF as shown in figure 1 b). To impose spatiotemporal constraints usually each node of the 2D MRF in the current frame is connected to its four image neighbors and additionally to the nodes with the same grid coordinates in the previous and next frames. This results in a graph of cardinality six on which the basic BP algorithm can be used without modifications (Williams et al., 2005).

Whenever visual observations with moving objects or taken from a moving camera system are processed, then the temporal correspondences between image coordinates of consecutive time slices are changing (Chen and Tang, 2007; Huang et al.,

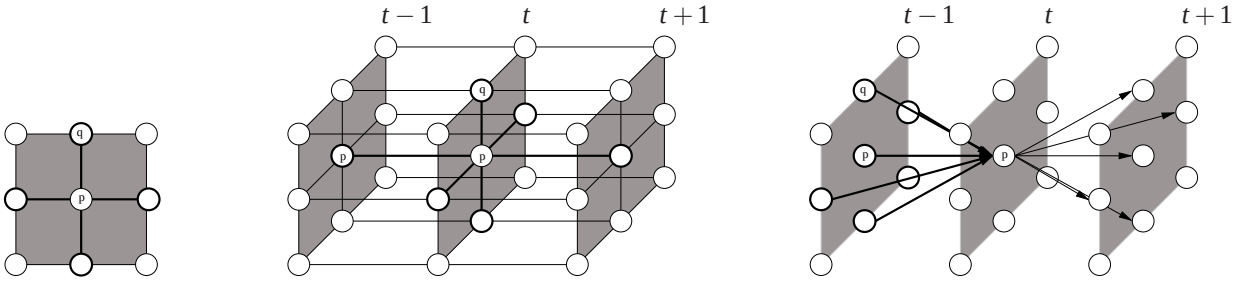


Figure 1: a) A 2D grid-like pairwise MRF with four neighboring nodes (fat black lines) in the Markov blanket of a hidden node p (white circles). b) A 3D grid-like pairwise MRF with 6 nodes in the Markov blanket. c) A 3D DBN with directed links between spatiotemporal neighboring nodes (5 node Markov blanket) indicated by black arrows (only one connection pattern is shown for clarity). All observable nodes are omitted.

2008). Hence, knowledge about *on which coordinates is projected the same scene content between two time slices* has to be incorporated into the temporal transitions. As a *correspondence field* the optical flow can be used. This leads to a spatially and temporally adaptive neighborhood connection between nodes of consecutive time slices. Since in most cases the optical flow cannot be estimated unambiguously for every pixel in an image sequence special care has to be taken for the integration of such an uncertain correspondence information.

Instead of extending MRFs from a 2D grid to a 3D grid structure, another class of probabilistic graphical models could be used to model *spatiotemporal* dependencies, called Dynamic Bayesian Networks. They generalize coupled hidden Markov models (HMM) (Brand, 1997) by representing the hidden state in terms of state variables with complex interdependencies and are suitable for *online filtering and prediction* (Murphy and Weiss, 2001).

The main aim of this paper is to answer the question: What kind of connectivity is the most efficient one if both spatial and temporal information should be fed into every node of a 3D grid-like graph? The question is motivated by finding an efficient solution to early vision problems with visual observations being *dynamic*. To do so, we compare belief propagation for different MRF graph topologies with DBN graph topologies. We propose to use a special connectivity in a 3D DBN topology that allows to come as close as possible to the basic BP algorithm for MRFs. Hence, for the 3D DBN we assume the same 3D grid-like structure as for 3D MRFs but different node connections.

The main differences between the conventional 3D MRF topology and the proposed 3D DBN topology are: (I) The conventional 3D MRF has several undirected links between neighboring nodes within a time slice but usually only two temporal neighbors (see figure 1 b)) for an example with 4 spatial neighbors). (II) The proposed 3D DBN has no undirected links between neighboring nodes within a time slice but directed links between one node at current time and several neighboring nodes of an arbitrary number from the past time slice (see figure 1 c)) for an example with 4 spatiotemporal neighbors).[1] Introducing a factorised transition probability for DBNs, we are able to compare BP in 2D MRFs and 3D MRFs to BP in 3D DBNs. We show that inference in a 3D DBN with the proposed topology and connectivity is less memory consuming, less computationally expensive, and better suited for online filtering than their MRF counterpart.

To cope with a temporally changing image content because of moving objects and therefore with temporally adaptive correspondences between image pixels of consecutive frames, we use optical flow as a correspondence field for adaptation of the node connections of the 3D DBN as it was already proposed in (Chen and Tang, 2007) and (Huang et al., 2008) but for 3D MRFs.

2 MRF AND DBN DEFINITIONS

For early vision problems usually a pairwise MRF is used that is based on a 2D grid-like graph structure as shown in figure 1 a). Following the notation in (Felzenszwalb and Huttenlocher, 2006), let $\mathcal{P}$ be the set of pixels $p \in \mathcal{P}$ in an image. Each pixel is assigned two nodes - one for a hidden random variable $x_p \in \mathcal{X}$ and one for an observed random variable $y_p \in \mathcal{Y}$. $\mathcal{X}$ denotes a finite set of variable values and $\mathcal{Y}$ a finite

[1] In principle, the connections and the number of neighboring nodes can vary but we restrict ourselves to equal connection patterns for every node on the image grid.

set of observations[2]. Further on, let $(p,q) \in \mathcal{N}$ be an edge between the hidden random variables x_p and x_q from the set $\mathcal{N}$ of all edges in the graph. A hidden state X consists of an amount of state variables $\{x_p\}_{p\in\mathcal{P}}$ for each pixel p and the observation Y comprises all pixel observations $\{y_p\}_{p\in\mathcal{P}}$. The joint distribution of a pairwise 2D MRF is given by

$$P(X,Y) = \frac{1}{Z}\prod_{p\in\mathcal{P}}\phi(x_p,y_p)\prod_{(p,q)\in\mathcal{N}}\psi(x_p,x_q), \quad (1)$$

where $\phi(x_p,y_p)$ denotes the *node potential* that defines the similarity between the observed pixel measurement and the hidden labels and $\psi(x_p,x_q)$ corresponds to the *edge potentials* that encode the similarities of spatially neighboring labels. The quantity Z is a normalization constant, called the *partition function*.

Extending the 2D MRF to a 3D MRF leads to an increase in the number of state variables with an additional time index $t \in \mathcal{T}$ for a sequence of $\mathcal{T} = (1,2,\ldots,T)$ time slices. The hidden and observed states, $X^t = \{x_p^t\}_{p\in\mathcal{P}}^{t\in\mathcal{T}}$ and $Y^t = \{y_p^t\}_{p\in\mathcal{P}}^{t\in\mathcal{T}}$, are represented in terms of hidden and observed state variables, $x_p^t \in \mathcal{X}$ and $y_p^t \in \mathcal{Y}$. This allows for temporally changing observations. Thus, the joint distribution of a pairwise 3D MRF for a sequence of image observations $Y^{1:T}$ is given by

$$\begin{aligned} P(X^{0:T},Y^{1:T}) &= \frac{1}{Z}\prod_{(p,q)\in\mathcal{N}}\psi(x_p^0,x_q^0)\times \\ &\prod_{t\in\mathcal{T}}\prod_{p\in\mathcal{P}}\phi(x_p^t,y_p^t)\psi(x_p^t,x_p^{t-1})\times \\ &\prod_{(p,q)\in\mathcal{N}}\psi(x_p^t,x_q^t). \end{aligned} \quad (2)$$

Here, the edge potentials $\psi(x_p^0,x_q^0)$ at time $t=0$ introduce some initial state conditions.

To transfer the spatiotemporal properties of a 3D MRF to a 3D DBN, we restrict ourselves to *regular* [3]DBNs (Murphy and Weiss, 2001) with a special topology like depicted in figure 1 c). The generative model of a DBN is precisely defined by the specification of the *observation likelihood* $P(Y^t|X^t) = \prod_p P(y_p^t|x_p^t)$ and the *transition probability*[4] $P(X^t|X^{t-1}) = \prod_p P(x_p^t|\{x_q^{t-1}\})$ with $q \in \mathcal{N}(p)$ and $\mathcal{N}(p)$ being a neighborhood of grid position p including a set of neighboring hidden nodes $\{x_q^{t-1}\}$ at grid positions q in the past time slice. The joint likelihood of a DBN up to time T is given by

$$\begin{aligned} P(X^{0:T},Y^{1:T}) &= \prod_{p\in\mathcal{P}}P(x_p^0)P(x_p^1|\{x_q^0\})\times \\ &\prod_{t\in\mathcal{T}}\prod_{p\in\mathcal{P}}P(y_p^t|x_p^t)P(x_p^t|\{x_q^{t-1}\}), \end{aligned} \quad (3)$$

where $P(x_p^0)$ are the priors of the state variables that serve as initialisation.

3 BELIEF PROPAGATION

Now, we focus on comparing inference in MRFs and in DBNs for online applications. The strategy is to select an approximate inference technique for MRFs that is closely related to forward filtering in DBNs and to choose the transition of a DBN such that the differences between inference on the MRF and the DBN get as small as possible. This allows to discuss the advantages of the remaining differences in the inference algorithms.

3.1 Belief Propagation for a 2D MRF

Using loopy BP in a 2D MRF at each iteration step n, an approximation for the marginal probability $P(x_p,Y)$ called the belief for each node p can be computed

$$b^n(x_p) \propto \phi(x_p,y_p)\prod_{q\in\mathcal{N}(p)}m_{q\to p}^n(x_p). \quad (4)$$

Here, $m_{q\to p}^n(x_p)$ are the incoming messages to node p within the spatial neighborhood $\mathcal{N}(p)$ where the proportionality $\propto$ considers $\sum_{x_p} m_{q\to p}^n = 1$. Applying the sum-product algorithm (Bishop, 2006) the message update visualised in figure 2 a) is given by

$$\begin{aligned} m_{q\to p}^n(x_p) &\propto \sum_{x_q\in\mathcal{X}}\psi(x_p,x_q)\phi(x_q,y_q)\prod_{s\in\mathcal{N}(q)\backslash p}m_{s\to q}^{n-1}(x_q) \\ &\propto \sum_{x_q\in\mathcal{X}}\psi(x_p,x_q)\frac{b^{n-1}(x_q)}{m_{p\to q}^{n-1}(x_q)}, \end{aligned} \quad (5)$$

where $m_{s\to q}^{n-1}(x_q)$ are the incoming messages to node q other than p: $\mathcal{N}(q)\backslash p$. For initialisation all messages are set to be uniform $m_{q\to p}^0 = 1, \forall(p,q)$.

3.2 Belief Propagation for a 3D MRF

For an online application with temporally changing observations we can apply a forward BP on a 3D MRF

[2]Without loss of generality, the observations can be discrete or continuous.

[3]Within one time slice the hidden and observable nodes are arranged like the 2D pixel-grid of an image. Directed connections are only allowed from hidden nodes x_p^t to observables y_p^t at the same time or to neighboring hidden nodes x_q^{t-1} from the past time slice. Intra-slice connections between neighboring hidden nodes x_p^t and x_q^t at the same time are forbidden.

[4]Sometimes also called conditional probability tables.

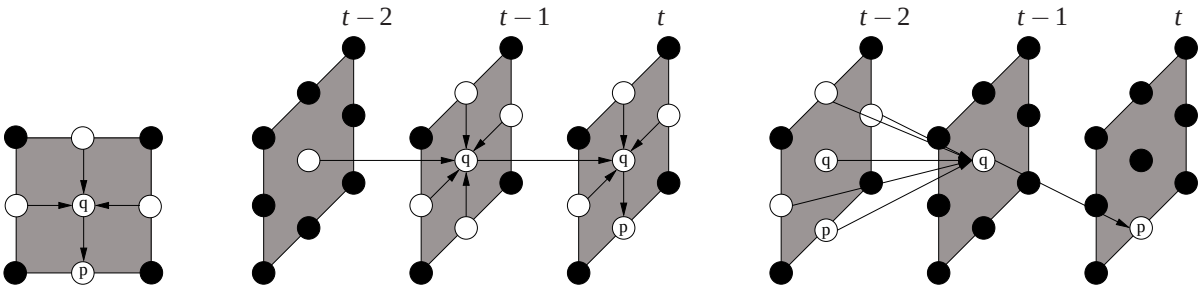


Figure 2: Message passing schedules for a) a pairwise 2D Markov random field b) a pairwise 3D Markov random field and c) a 3D DBN. For the 3D DBN the spatiotemporal information flow is not restricted to the five neigbors from past time slice but can be an arbitrary number of node connections. Observation nodes are omitted for clarity. Hidden nodes being operated on for a message update are shown white with the message flow indicated by arrows; other hidden nodes are displayed black.

with a message update schedule that does not access future observations (Yin and Collins, 2007). Now the forward belief for each node p for each timestep t is defined as

$$b^n(x_p^t) \propto \phi(x_p^t, y_p^t)\, m_{p\to p}(x_p^t) \prod_{q\in\mathcal{N}(p)} m_{q\to p}^n(x_p^t). \quad (6)$$

The messages have to be updated according to the following defined order if we do not want to go backwards in time. First, the messages for the temporal transition (see figure 2 b)) are calculated as given by

$$\begin{aligned} m_{q\to q}(x_q^t) &\propto \sum_{x_q^{t-1}\in\mathcal{X}} \psi(x_q^t, x_q^{t-1})\phi(x_q^{t-1}, y_q^{t-1}) \times \\ &\quad m_{q\to q}(x_q^{t-1}) \prod_{s\in\mathcal{N}(q)} m_{s\to q}^n(x_q^{t-1}) \\ &\propto \sum_{x_q^{t-1}\in\mathcal{X}} \psi(x_q^t, x_q^{t-1}) b^n(x_q^{t-1}), \end{aligned} \quad (7)$$

which is recurrent over time t in $m_{q\to q}(x_q^t) \leftarrow f(m_{q\to q}(x_q^{t-1}))$[5]. Second, the spatial messages within a time slice are calculated (see figure 2 b)) as follows

$$\begin{aligned} m_{q\to p}^n(x_p^t) &\propto \sum_{x_q^t\in\mathcal{X}} \psi(x_p^t, x_q^t)\phi(x_q^t, y_q^t) \times \\ &\quad m_{q\to q}(x_q^t) \prod_{s\in\mathcal{N}(q)\setminus p} m_{s\to q}^{n-1}(x_q^t) \\ &\propto \sum_{x_q^t\in\mathcal{X}} \psi(x_p^t, x_q^t) \frac{b^{n-1}(x_q^t)}{m_{p\to q}^{n-1}(x_q^t)}, \end{aligned} \quad (8)$$

which are dependent on the temporal transitions $m_{q\to q}(x_q^t)$ and the spatial transitions $m_{s\to q}^{n-1}(x_q^t)$ that can be iteratively refined along $n = 1...N$ iterations anew for each time slice t. Here, all the speed-ups proposed for message passing in 2D MRFs like in (Felzenszwalb and Huttenlocher, 2006; Tappen and Freeman, 2003) can be applied.

[5] f denotes the function that transfers $m_{q\to q}(x_q^{t-1})$ to the next time t.

3.3 Belief Propagation for a 3D DBN

For the forward BP on a 3D DBN we use an efficient approximate inference algorithm called factorised frontier algorithm (Murphy and Weiss, 2001). It assumes that the posterior probability $P(X^t|Y^{1:t}) := \prod_p P(x_p^t|Y^{1:t})$ is approximated as a product of marginals which is equivalent to loopy BP assuming that the messages coming into a node are independent. For inference, we define the observation likelihood as the product of the node potentials used for the 3D MRF

$$P(Y^t|X^t) = \prod_{q\in\mathcal{P}} P(y_p^t|x_p^t) = \prod_{p\in\mathcal{P}} \phi(x_p^t, y_p^t), \quad (9)$$

and the transition as the normalised product of the edge potentials

$$P(x_p^t|\{x_q^{t-1}\}) \propto \prod_{q\in\mathcal{N}(p)} \psi(x_p^t, x_q^{t-1}), \quad (10)$$

where $\propto$ denotes that $\sum_{x_p^t} P(x_p^t|\{x_q^{t-1}\}) = 1$. It is further assumed that every $\psi(x_p^t, x_q^{t-1})$ is equivalent to $\psi(x_p^t, x_p^{t-1})$, $\forall q \in \mathcal{N}(p)$ of the 3D MRF. With the beforehand mentioned assumption of a fully factorised posterior each factor of the posterior $P(x_p^t|Y^{1:t})$ is equivalent to the belief $b(x_p^t)$ when applying forward BP on the 3D DBN

$$P(x_p^t|Y^{1:t}) = b(x_p^t) \propto \phi(x_p^t, y_p^t) \prod_{q\in\mathcal{N}(p)} m_{q\to p}(x_p^t). \quad (11)$$

The corresponding forward message update rule (see figure 2 c)) reads

$$\begin{aligned} m_{q\to p}(x_p^t) &\propto \sum_{x_q^{t-1}\in\mathcal{X}} \psi(x_p^t, x_q^{t-1})\phi(x_q^{t-1}, y_q^{t-1}) \times \\ &\quad \prod_{s\in\mathcal{N}(q)} m_{s\to q}(x_q^{t-1}) \\ &\propto \sum_{x_q^{t-1}\in\mathcal{X}} \psi(x_p^t, x_q^{t-1}) b(x_q^{t-1}). \end{aligned} \quad (12)$$

Inserting the message update (12) into (11) we arrive at a temporal recurrent belief update

$$b(x_p^t) \propto \phi(x_p^t, y_p^t) \prod_{q \in \mathcal{N}(p)} \sum_{x_q^{t-1}} \psi(x_p^t, x_q^{t-1}) b(x_q^{t-1}), \tag{13}$$

which is recurrent over time t *only* in the beliefs $b(x_p^t) \leftarrow f(\{b(x_q^{t-1})\})$. Here, f denotes the function that transfers past beliefs at time $t-1$ to beliefs $b(x_p^t)$ at current time t. This is a nice advantage compared to the belief update in the 3D MRF according to (6) which is not recurrent over time in the beliefs only.

4 QUALITATIVE COMPARISON

Now the advantages of the recurrent belief update of the 3D DBN (13) compared to the belief updates of a 2D MRF (4) and a 3D MRF (6) along with the message updates are discussed.

Inference The forward BP in a 3D DBN as proposed in (13) is equivalent to the α-expansion of the forward-backward-algorithm for Bayesian networks (Bishop, 2006) as can be seen by the following conversion

$$\begin{aligned}
&b(x_p^t) = P(x_p^t | Y^{1:t}) \\
&\propto P(y_p^t | x_p^t) \sum_{X^{t-1}} P(x_p^t | X^{t-1}) P(X^{t-1} | Y^{1:t-1}) \\
&\propto P(y_p^t | x_p^t) \sum_{X^{t-1}} \prod_{q \in \mathcal{N}(p)} \psi(x_p^t, x_q^{t-1}) \prod_{z \in \mathcal{P}} P(x_z^{t-1} | Y^{1:t-1}) \\
&\propto \underbrace{P(y_p^t | x_p^t)}_{\phi(x_p^t, y_p^t)} \prod_{q \in \mathcal{N}(p)} \sum_{x_q^{t-1}} \psi(x_p^t, x_q^{t-1}) \underbrace{P(x_q^{t-1} | Y^{1:t-1})}_{b(x_p^{t-1})} \times \\
&\quad \underbrace{\sum_{\substack{X^{t-1} \\ \setminus x_q^{t-1}}} \prod_{z \neq q} P(x_z^{t-1} | Y^{1:t-1})}_{=1},
\end{aligned} \tag{14}$$

assuming the transition probability to be a product of potentials as defined in (10) and the approximation of a fully factorised marginal distribution (11) within a time slice t. Of course, all the derivations also hold for the max-product algorithm by simply replacing the marginal sums with the max-operator (Bishop, 2006) to estimate the most probable sequence of states. Obviously, the DBN update rule (12) looks pretty much the same as the 2D MRF update rule (5). The only three differences are that

1. index n is now an iteration through time t and for each timestep a new state variable x_p^t is defined,
2. hence the past node x_s^{t-1} at position $s = p$ is also allowed to send a message $m_{s=p \to q}(x_q^{t-1})$ to contribute to the update of message $m_{q \to p}(x_p^t)$ (which means the belief $b(x_q^{t-1})$ does not have to be divided by the message $m_{p \to q}(x_q^{t-1})$)
3. and all observations y_q^{t-1} are allowed to change over time.

As a result, the proposed forward filter for the 3D DBN *unrolls* the standard loopy BP for a 2D MRF that iterates n-times within a time slice along the physical time t and thus can deal with temporally changing observations. Vice versa, this relation can be used to define a more approximative but less memory consuming BP for 2D MRFs simply by replacing physical time t in (13) with iteration index n which reads

$$b^n(x_p) \propto \phi(x_p, y_p) \prod_{q \in \mathcal{N}(p)} \sum_{x_q} \psi(x_p, x_q) b^{n-1}(x_q), \tag{15}$$

where the beliefs are initialised equally distributed for $n = 0$. To compare it with the standard belief update, we insert (5) into (4). This leads to

$$b^n(x_p) \propto \phi(x_p, y_p) \prod_{q \in \mathcal{N}(p)} \sum_{x_q \in \mathcal{X}} \psi(x_p, x_q) \frac{b^{n-1}(x_q)}{m_{p \to q}^{n-1}(x_q)}, \tag{16}$$

which is dependent on the past beliefs $\{b^{n-1}(x_q)\}$ *and* the past messages $\{m_{p \to q}^{n-1}(x_q)\}$. The simpler update rule (15) is much more efficient because it is recursive in the belief only.

Comparing the belief update of a 3D DBN (13) with the update of a 3D MRF (6) we have seen in section 3.2 that (6) is not recurrent in the beliefs only but is (13). Thus, for the 3D MRF all the messages have to be stored which is not the case for the belief update of the 3D DBN.

Memory Loopy BP methods that do not explicitly compute and store messages but directly update the belief recurrently as in (13) and (15) require less amount of memory than state-of-the-art message passing update rules as in (5) and (7,8). The *space complexity* for storing beliefs in (13) and (15) is $\mathcal{P} \times \mathcal{X}$. Whereas for storing messages using (5) or (7,8) the space complexity is $\mathcal{P} \times \mathcal{X} \times \mathcal{N}(p)$ or $\mathcal{P} \times \mathcal{X} \times (\mathcal{N}(p) + 1)$, respectively.

5 QUANTITATIVE COMPARISON

Costs The *time complexity* for one message update step in MRFs increases quadratically in the number

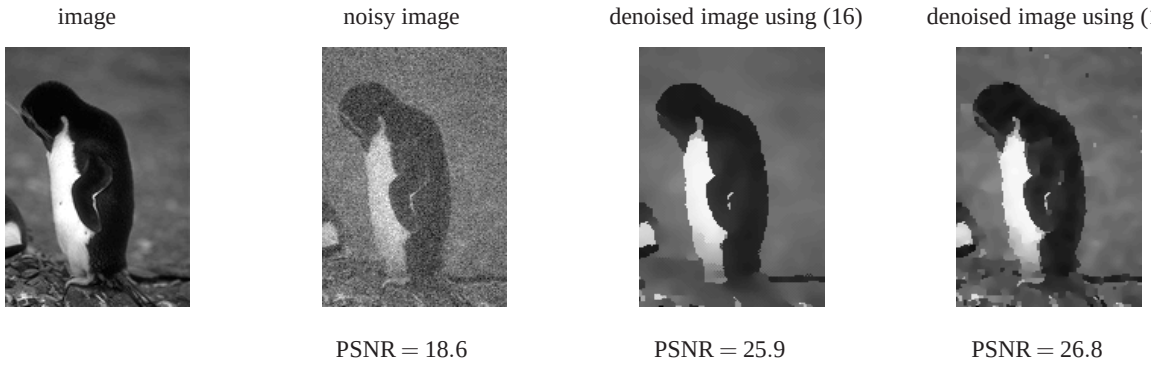


Figure 3: Denoising results for applying (16) and (15) to the penguin benchmark.

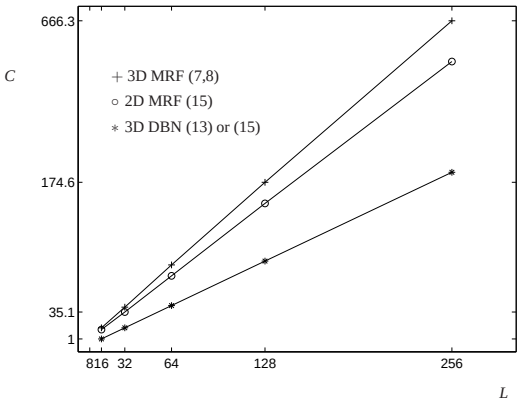


Figure 4: Costs C for different update rules and label numbers L with $C = 1$ equals a processing time of $0.27\frac{s}{u}$ seconds per update. The y-axis is square-root-scaled because the time complexity increases quadratically with the number of labels $\mathcal{X}$.

of state values and linearly in the number of pixels and messages $\mathcal{O}(\mathcal{P} \times \mathcal{X}^2 \times \mathcal{N})$ whereas for the DBN update the order reduces to $\mathcal{O}(\mathcal{P} \times \mathcal{X}^2)$. A quantitative comparison that confirms the time complexity is given in figure 4. Here, the times needed to compute one update of all beliefs or messages for an image with 120×180 pixels for different numbers of labels and different BP methods are visualized[6]. It confirms the quadratic increase of the costs with linear increase in labels and the computational advantage of (13) or (15) compared to (5) or (7,8).

Accuracy To compare the accuracy of the different BP methods, we start by applying (16) in comparison to (15) on the penguin denoising example (Szeliski et al., 2008). For the denoising test we add *static* Gaussian noise with a variance of 30 to the image with a size of 120×180 pixels. The state values x_p and observables y_p are intensities with $\mathcal{X} = (1,\ldots,256)$. For the node potentials we use quadratic costs $\phi(x_p,y_p) = exp(-l_\phi(x_p - y_p)^2)$ and for the edge potentials we use truncated absolute costs $\psi(x_p,x_q) = exp(-l_\psi min(t_\psi,|x_p - x_q|))$[7]. Figure 3 shows the denoising results after $n = 30$ iterations for applying (16) and (15) to the penguin benchmark. The Peak-Signal-To-Noise-Ratio (PSNR) which quantifies the denoising quality is slightly better for (15) than for (16) although the computational effort for (15) is much less as already shown in Fig. 4.

6 DYNAMIC DENOISING

Temporal correspondence If the visual scene moves, then the temporal pixel correspondences c_p^t between image coordinates of consecutive time slices are changing. This requires a spatially adaptive neighborhood $\mathcal{N}(q + c_q^t)$ for the temporal message passing schedule already proposed in (Chen and Tang, 2007; Huang et al., 2008). Now, in (7) the temporal messages $m_{(q+c_q^t)\to q}(x_q^t)$ adapt dependent on the correspondences c_q^t and in (13) the belief update gets adaptive in the product over spatiotemporal neighboring beliefs $q \in \mathcal{N}(p + c_p^t)$.

To demonstrate the real world applicability of the proposed BP in 3D DBNs (13) we apply them to solve an image sequence denoising task and compare the result to BP in 3D MRFs (7,8). For the 3D MRF, we restrict the iterations within a time slice (8) to $n = 1$ to get a fair comparison to the 3D DBN results. If the n-iterations are increased the convergence

[6]The algorithms are implemented in matlab and have been run on an Intel Core2 2.4GHz with 2GB RAM.

[7]The parameters are fixed to $-l_\phi = 0.01$, $-l_\psi = 0.1$, $t_\psi = 20$.

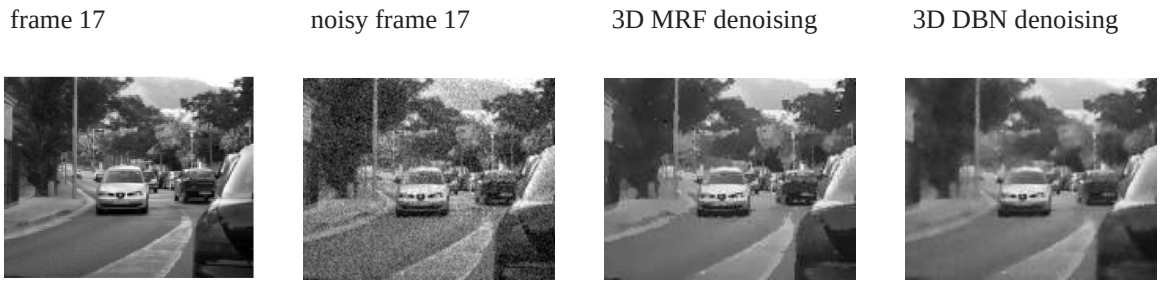
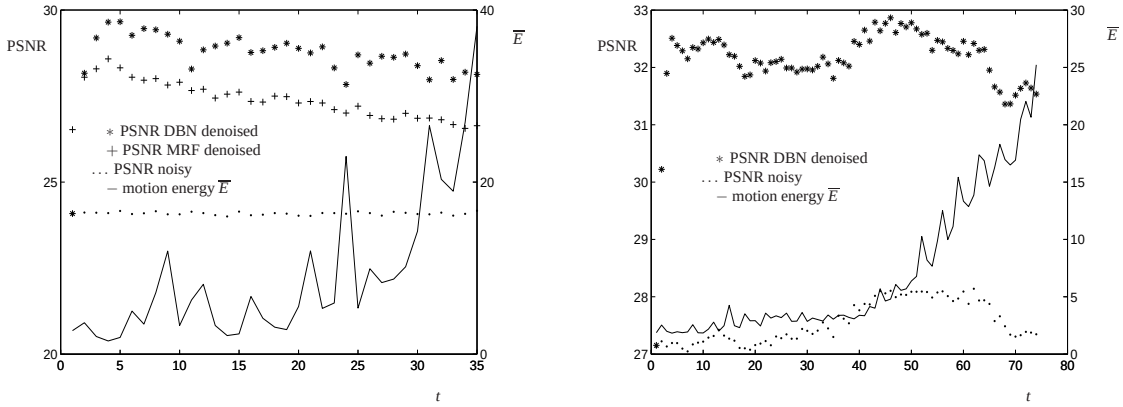


Figure 5: Denoising results for the car sequence.



(a) Car sequence (b) Fish sequence

Figure 6: PSNR and Mean Motion Energy $\overline{E}$.

gets faster per temporal t-iteration but also the computation time increases n-times. Figure 5 shows the denoising result for BP in a 3D DBN compared to a 3D MRF applied to the *car* sequence and using optical flow for estimating the pixel correspondences c_p^t [8]. We used the same node and edge potentials as for the penguin example but now the noise is a *dynamic* additive Gaussian noise with a variance of 16. The corresponding course of the PSNRs is shown in Fig. 6a. The DBN ($*$) always outperforms the denoising quality of the MRF ($+$) (the PSNR of the noisy sequence is shown with the dotted line). To judge the dependency of the denoising quality on the amount of movements in the sequence also the mean motion energy $\overline{E}^t = 1/\mathcal{P} \sum_p ||c_p^t||^2$ is plotted. With increasing motion energy the denoising quality decreases. Figure 7 shows another denoising example with a detail for better visual inspection of the results. Although, the variance of the noise varies between $14-16$ and the motion energy increases, the denoising quality follows quite stable.

[8] All sequences and optical flows are taken from http://people.csail.mit.edu/celiu/motionAnnotation/index.html.

We would like to mention that if the number of intra-time iterations $n \geq 1$ is increased it is likely that the MRF- result could surpass the accuracy of the DBN but with the disadvantage of increasing the computing time n-times. Further on, more temporal neighbors could be used in the MRF, a choice that is also likely to improve the quality of the MRF-result but again leads to additional computing time.

7 SUMMARY AND CONCLUSION

We introduce a special 3D DBN topology with an efficient class of transition probabilities as a basic framework for low level vision applications suited for active vision systems. It provides promising results in terms of memory amount, computational costs, and robustness. Applications for image denoising show that for static scenes with static noise the proposed approximate BP achieves similar or better accuracy for denoising than standard BP in 2D MRFs. For dynamic scenes an efficient spatiotemporal node connection for a DBN topology is introduced that allows

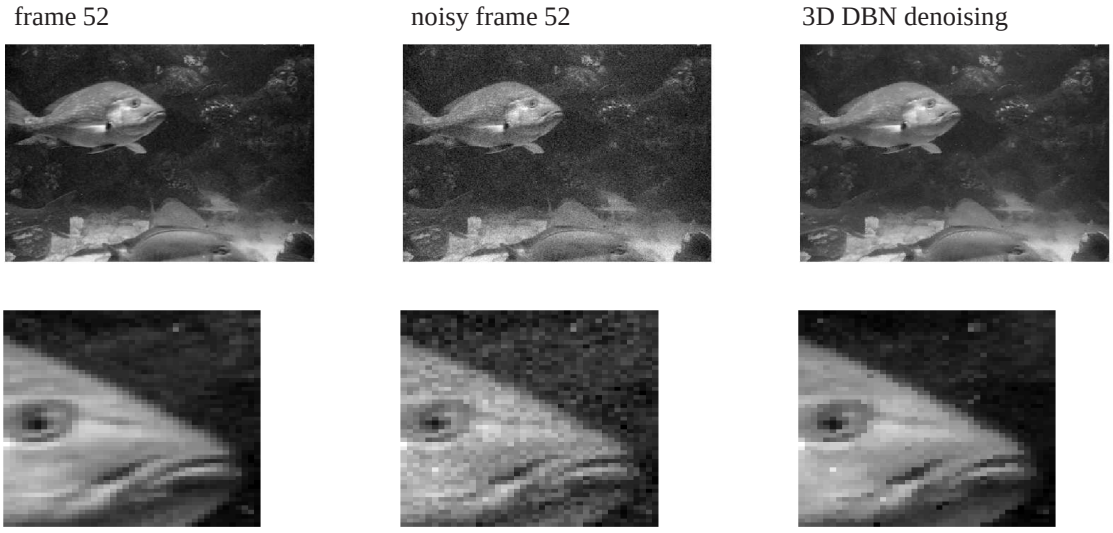


Figure 7: Denoising results for the fish sequence.

for fast BP with less memory load than standard 3D MRF approaches and more accurate denoising results on noisy real world image sequences.

REFERENCES

Bishop, C. (2006). *Pattern Recognition and Machine Learning*. Springer Science+Business Media.

Brand, M. (1997). Coupled hidden markov models for complex action recognition. In *Proc. of IEEE Conf. on CVPR*.

Chen, J. and Tang, C. (2007). Spatio-temporal markov random field for video denoising. In *Proc. of IEEE Conf. on CVPR*.

Felzenszwalb, P. and Huttenlocher, D. (2006). Efficient belief propagation for early vision. *Int. J. Comput. Vision*, 70:41–54.

Huang, R., Pavlovic, V., and Metaxas, D. (2008). A new spatio-temporal mrf framework for video-based object segmentation. In *Proc. of 1st Int. Workshop on Machine Learning for Vision-based Motion Analysis*.

Komodakis, N. and Paragios, N. (2009). Pairwise energies: Efficient optimization for higher-order mrfs. In *Proc. of IEEE Conf. on CVPR*.

Larsen, E., Mordohai, P., Pollefeys, M., and Fuchs, H. (2007). Temporally consistent reconstruction from multiple video streams using enhanced belief propagation. In *Proc. of IEEE Conf. on CVPR*.

Murphy, K. and Weiss, Y. (2001). The factored frontier algorithm for approximate inference in dbns. In *Proc. of 17th Conf. on Uncertainty in Artificial Intelligence*.

Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann.

Petersen, K., Fehr, J., and Burkhardt, H. (2008). Fast generalized belief propagation for map estimation on 2d and 3d grid-like markov random fields. In *Proc. of 30th Conf. of the German Association for Pattern Recognition*.

Ramalingam, S., Kohli, P., Alahari, K., and Torr, P. (2008). Exact inference in multi-label crfs with higher order cliques. In *Proc. of IEEE Conf. on CVPR*.

Roth, S. (2007). *High-Order Markov Random Fields for Low-Level Vision*. PhD thesis, Brown University.

Szeliski, R., Zabih, R., Scharstein, D., Veksler, O., Kolmogorov, V., Agarwala, A., Tappen, M., and Rother, C. (2008). A comparative study of energy minimization methods for markov random fields with smoothness-based priors. *IEEE Trans. on PAMI*, 30:1068–1080.

Tappen, M. and Freeman, W. (2003). Comparison of graph cuts with belief propagation for stereo, using identical mrf parameters. In *Proc. of IEEE ICCV*.

Williams, O., Isard, M., and MacCormick, J. (2005). Estimating disparity and occlusions in stereo video sequences. In *Proc. of IEEE Conf. on CVPR*.

Yin, Z. and Collins, R. (2007). Belief propagation in a 3d spatio-temporal mrf for moving object detection. In *Proc. of IEEE Conf. on CVPR*.