Supplementary Document for Towards Probabilistic Volumetric Reconstruction using Ray Potentials

Ali Osman Ulusoy Andreas Geiger Michael J. Black

Max Planck Institute for Intelligent Systems, Tübingen, Germany
{osman.ulusoy,andreas.geiger,black}@tuebingen.mpg.de

1 Sum-product Message Derivations

This section provides derivation of the ray factor to variable message equations for occupancy and appearance variables. The simplified form of the equations are presented in Eq. 13, 14, 15 in the submission.

In the sum-product algorithm, the general form of the messages are given by

$$\mu_{f\to x}(x) = \sum_{\mathcal{X}_f\setminus x} \phi_f(\mathcal{X}_f) \prod_{y\in\mathcal{X}_f\setminus x} \mu_{y\to f}(y) \tag{1}$$

$$\mu_{x\to f}(x) = \prod_{g\in\mathcal{F}_x\setminus f} \mu_{g\to x}(x) \tag{2}$$

where f is the factor, x is the variable, $\mathcal{X}_f$ denotes all variables associated with factor f and $\mathcal{F}_x$ is the set of factors to which variable x is connected [1].

Consider a single ray r associated with ray factor ψ_r. Since we're dealing with a single ray, we drop the index r for brevity. The ray factor ψ is connected to N occupancy and N appearance variables. The potential equation for the ray factor (see Eq. 4 in the submission) is as follows:

$$\psi(\mathbf{o},\mathbf{a}) = \sum_{i=1}^{N} o_i \prod_{j<i} (1-o_j)\, \nu(a_i). \tag{3}$$

where $\mathbf{o}$ and $\mathbf{a}$ denote the set of occupancy and appearance variables connected to the factor respectively. Here, $\nu(a)$ measures the probability of observing intensity/color a given that the actual pixel observation I. Assuming Gaussian noise, we model $\nu(a) = \mathcal{N}(a|I,\sigma)$.

The ray potential measures how well the occupancy and appearance of voxels along the ray explain the pixel observation I. Specifically, the potential measures the similarity of the appearance of the first occupied voxel along the ray and I. This can be better observed when Eq. 3 is expressed as follows:

$$\psi(\mathbf{o},\mathbf{a}) = \begin{cases} \nu(a_1) & \text{if } o_1 = 1 \\ \nu(a_2) & \text{if } o_1 = 0, o_2 = 1 \\ \nu(a_3) & \text{if } o_1 = 0, o_2 = 0, o_3 = 1 \\ \vdots & \\ \nu(a_N) & \text{if } o_1 = 0, ..., o_{N-1} = 0, o_N = 1 \end{cases} \tag{4}$$

It can be seen that $\psi(\mathbf{o},\mathbf{a}) = \nu(a_i)$ where i is the index of the first occupied voxel along the ray.

This equation exposes the special *sparse* structure of the ray potential. In particular, if i is the first occupied voxel along the ray, the occupancy state of all voxels after i can be safely ignored as they do not change the result of ψ. This behavior of the ray potential is intuitive since voxels after i are being occluded by voxel i. In this case, note also that ψ is a function of only a_i. This result is again intuitive since the camera records the appearance of only the visible occupied voxel. This special structure will play a key role in the message equation derivations below.

1.1 Factor to occupancy variable message

Positive case: We begin with the message equation to ith occupancy variable when it is occupied, $o_i = 1$. For this case, the general message equation (Eq. 1) becomes,

$$\mu_{\psi\to o_i}(o_i = 1) = \sum_{o_1} \cdots \sum_{o_{i-1}} \sum_{o_{i+1}} \cdots \sum_{o_N} \int_{a_1} \cdots \int_{a_N} \psi(\mathbf{o},\mathbf{a}) \prod_{\substack{j=1 \\ j\neq i}}^{N} \mu(o_j) \prod_{j=1}^{N} \mu(a_j) \tag{5}$$

As mentioned in the submission, naïvely evaluating this equation is not feasible since it involves summing over $N-1$ binary variables where N is typically on the order of hundreds. However, we show that the special structure of the ray potential allows computing the messages very efficiently.

Our strategy is to carry out the summations over the occupancy variables one by one. We begin with the summation over o_1 as follows:

$$\begin{aligned} \mu_{\psi\to o_i}(o_i = 1) = & \overbrace{\mu(o_1 = 1)\Big[\sum_{o_2} \cdots \sum_{o_{i-1}} \sum_{o_{i+1}} \cdots \sum_{o_N} \int_{a_1} \cdots \int_{a_N} \psi(o_1 = 1, o_2, ..., o_N, \mathbf{a}) \prod_{\substack{j=2 \\ j\neq i}}^{N} \mu(o_j) \prod_{j=1}^{N} \mu(a_j) \Big]}^{(\dagger)} + \\ & \underbrace{\mu(o_1 = 0)\Big[\sum_{o_2} \cdots \sum_{o_{i-1}} \sum_{o_{i+1}} \cdots \sum_{o_N} \int_{a_1} \cdots \int_{a_N} \psi(o_1 = 0, o_2, ..., o_N, \mathbf{a}) \prod_{\substack{j=2 \\ j\neq i}}^{N} \mu(o_j) \prod_{j=1}^{N} \mu(a_j) \Big]}_{(\ddagger)} \end{aligned} \tag{6}$$

For the top expression (†), the ray potential $\psi(o_1 = 1, ..., o_N, a_1, ..., a_N)$ evaluates to $\nu(a_1)$, see Eq. 4. Since $\nu(a_1)$ only depends on the integral over a_1, it can be brought out of the summation/integrals over the other occupancy and appearance variables as follows:

$$(\dagger) = \left[\int_{a_1} \nu(a_1)\mu(a_1)\,\mathrm{d}a_1\right] \underbrace{\sum_{o_2}\cdots\sum_{o_{i-1}}\sum_{o_{i+1}}\cdots\sum_{o_N}\int_{a_2}\cdots\int_{a_N}\prod_{\substack{j=2\\j\neq i}}^{N}\mu(o_j)\prod_{j=2}^{N}\mu(a_j)}_{\text{this term evaluates to 1.}} \tag{7}$$

Furthermore, assuming all incoming messages μ are normalized such that they sum/integrate to 1, the terms highlighted with the underbrace evaluate to 1. In our implementation, we make sure all messages are normalized to exploit this property, as well as for numerical stability.

For the expression (‡), the ray potential does not simplify. We then expand the summation over o_2.

$$\begin{aligned}(\ddagger) =& \mu(o_2=1)\overbrace{\left[\sum_{o_3}\cdots\sum_{o_{i-1}}\sum_{o_{i+1}}\cdots\sum_{o_N}\int_{a_1}\cdots\int_{a_N}\psi(o_1=0,o_2=1,o_3,...,o_N,\mathbf{a})\prod_{\substack{j=3\\j\neq i}}^{N}\mu(o_j)\prod_{j=1}^{N}\mu(a_j)\right]}^{(\square)}+\\ & \mu(o_2=0)\underbrace{\left[\sum_{o_3}\cdots\sum_{o_{i-1}}\sum_{o_{i+1}}\cdots\sum_{o_N}\int_{a_1}\cdots\int_{a_N}\psi(o_1=0,o_2=0,o_3,...,o_N,\mathbf{a})\prod_{\substack{j=3\\j\neq i}}^{N}\mu(o_j)\prod_{j=1}^{N}\mu(a_j)\right]}_{(\triangle)}\end{aligned} \tag{8}$$

We can simplify (□) similar to the way (†) was simplified. Since $o_2 = 1$, the ray potential reduces to $\nu(a_2)$ and thus can be taken out of the summations/integrals expect the integral over a_2. Again, assuming the incoming messages are normalized such that they integrate/sum to 1, all other terms vanish as shown below:

$$(\square) = \left[\int_{a_2} \nu(a_2)\mu(a_2)\,\mathrm{d}a_2\right] \underbrace{\sum_{o_3}\cdots\sum_{o_{i-1}}\sum_{o_{i+1}}\cdots\sum_{o_N}\int_{a_1}\int_{a_3}\cdots\int_{a_N}\prod_{\substack{j=3\\j\neq i}}^{N}\mu(o_j)\prod_{\substack{j=1\\j\neq 2}}^{N}\mu(a_j)}_{\text{this term evaluates to 1.}} \tag{9}$$

At this point, it is worth re-writing Eq. 6.

$$\begin{aligned}\mu_{\psi\to o_i}(o_i=1) =& \mu(o_1=1)\left[\int_{a_1}\nu(a_1)\mu(a_1)\,\mathrm{d}a_1\right] + \mu(o_i=0)\mu(o_2=1)\left[\int_{a_2}\nu(a_2)\mu(a_2)\,\mathrm{d}a_2\right]\\ &+ \mu(o_i=0)\mu(o_2=0)(\triangle)\end{aligned} \tag{10}$$

We see that the equation is incrementally simplified as the summations over occupancy variables are expanded. We continue expanding the summations until (including) o_{i-1}. It can be verified that, each such expansion brings in a term of the form $\mu(o_j = 1)\prod_{k=1}^{j-1}\mu(o_k = 0)\,\rho_j$. Once all summations are expanded, the resulting expression is:

$$\mu_{\psi\to o_i}(o_i=1)=\sum_{j=1}^{i-1}\mu(o_j=1)\prod_{k=1}^{j-1}\mu(o_k=0)\left[\int_{a_j}\nu(a_j)\mu(a_j)\,\mathrm{d}a_j\right]+ \tag{11}$$

$$\prod_{k=1}^{i-1}\mu(o_k=0)\underbrace{\left[\sum_{o_{i+1}}\cdots\sum_{o_N}\int_{a_1}\cdots\int_{a_N}\psi(o_1=0,..,o_{i-1}=0,o_i=1,..,o_N,\mathbf{a})\prod_{j=i+1}^{N}\mu(o_j)\prod_{j=1}^{N}\mu(a_j)\right]}_{(\Diamond)} \tag{12}$$

Finally, $(\Diamond)$ can be simplified. The ray potential in $(\Diamond)$ evaluates as follows $\psi(o_1=0....,o_{i-1}=0,o_i=1,...,o_N,a_1,...,a_N)=\nu(a_i)$. Plugging this into $(\Diamond)$, the expression becomes:

$$(\Diamond)=\left[\int_{a_i}\nu(a_i)\mu(a_i)\,\mathrm{d}a_i\right]\underbrace{\sum_{o_{i+1}}\cdots\sum_{o_N}\int_{a_1}\cdots\int_{a_N}\prod_{j=i+1}^{N}\mu(o_j)\prod_{\substack{j=1\\j\neq i}}^{N}\mu(a_j)}_{\text{evaluates to 1.}} \tag{13}$$

Substituting $(\Diamond)$ back into Eq. 12, we get the final form of the message:

$$\mu_{\psi\to o_i}(o_i=1)=\sum_{j=1}^{i-1}\mu(o_j=1)\prod_{k=1}^{j-1}\mu(o_k=0)\,\rho_j+\prod_{k=1}^{i-1}\mu(o_k=0)\,\rho_i \tag{14}$$

where we use the shorthand notation as in Eq. 12 in the submission:

$$\rho_j=\int_{a_j}\nu(a_j)\mu(a_j)\,\mathrm{d}a_j \tag{15}$$

Note that while the general form of the message equation (Eq. 5) requires summing over 2^{N-1} states of the occupancy variables, the simplified form (Eq. 14) can be computed in *linear* time. Although Eq. 14 looks quadratic at a first glance, the visibility terms $\prod_{k=1}^{i-1}\mu(o_k=0)$ can be accumulated during a linear pass from $i=1$ to N. This allows computing all messages $\mu_{\psi\to o_i}(o_i=1)$ in a single linear pass.

Negative case: The derivation for the negative case, $o_i=0$, is very similar to the positive case. The general form of the message is:

$$\mu_{\psi\to o_i}(o_i=0)=\sum_{o_1}\cdots\sum_{o_{i-1}}\sum_{o_{i+1}}\cdots\sum_{o_N}\int_{a_1}\cdots\int_{a_N}\psi(\mathbf{o},\mathbf{a})\prod_{\substack{j=1\\j\neq i}}^{N}\mu(o_j)\prod_{j=1}^{N}\mu(a_j) \tag{16}$$

We follow the same strategy of expanding the summations one by one. The major difference is the terms involving voxels after i. For the positive case $o_i=1$, the summations/integrals for all variables after i vanished (see Eq. 13). For the negative case ($o_i=0$), however, these terms do not vanish. The resulting equations are as follows:

$$\mu_{\psi\to o_i}(o_i=0)=\sum_{j=1}^{i-1}\mu(o_j=1)\prod_{k=1}^{j-1}\mu(o_k=0)\,\rho_j+\frac{1}{\mu(o_i=0)}\sum_{j=i+1}^{N}\mu(o_j=1)\prod_{k=1}^{j-1}\mu(o_k=0)\,\rho_j \tag{17}$$

1.2 Factor to appearance variable messages

For the appearance variables, the general message equation (Eq. 1) is,

$$\mu_{\psi\to a_i}(a_i) = \sum_{o_1} ... \sum_{o_N} \int_{a_1} ... \int_{a_N} \psi(\mathbf{o},\mathbf{a}) \, . \prod_{j=1}^{N} \mu(o_j) \prod_{\substack{j=1 \\ j\neq i}}^{N} \mu(a_j) \tag{18}$$

We follow the same strategy as in the derivation of the occupancy messages and expand all the summations over the occupancy variables one by one. We begin by expanding the summation over o_1.

$$\begin{aligned}\mu_{\psi\to a_i}(a_i) = &\mu(o_1=1)\Bigg[\overbrace{\sum_{o_2}\cdots\sum_{o_N}\int_{a_1}\cdots\int_{a_{i-1}}\int_{a_{i+1}}\cdots\int_{a_N}\psi(o_1=1,o_2,...,o_N,\mathbf{a})\prod_{j=2}^{N}\mu(o_j)\prod_{\substack{j=1\\ j\neq i}}^{N}\mu(a_j)}^{(\dagger)}\Bigg]+\\ &\mu(o_1=0)\Bigg[\overbrace{\sum_{o_2}\cdots\sum_{o_N}\int_{a_1}\cdots\int_{a_{i-1}}\int_{a_{i+1}}\cdots\int_{a_N}\psi(o_1=0,o_2,...,o_N,\mathbf{a})\prod_{j=2}^{N}\mu(o_j)\prod_{\substack{j=1\\ j\neq i}}^{N}\mu(a_j)}^{(\ddagger)}\Bigg]\end{aligned} \tag{19}$$

For (†), the ray potential evaluates to, $\psi(o_1 = 1, o_2, ..., o_N, \mathbf{a}) = \nu(a_1)$ and thus, can be taken out of the integrals/summations that do not involve a_1:

$$(\dagger) = \underbrace{\left[\int_{a_1}\nu(a_1)\mu(a_1)\,\mathrm{d}a_1\right]}_{\rho_1}\underbrace{\sum_{o_2}\cdots\sum_{o_N}\int_{a_1}\cdots\int_{a_{i-1}}\int_{a_{i+1}}\cdots\int_{a_N}\prod_{j=2}^{N}\mu(o_j)\prod_{\substack{j=2\\ j\neq i}}^{N}\mu(a_j)}_{\text{this term evaluates to 1.}} \tag{20}$$

where we use the shorthand notation ρ in Eq. 15. The terms inside (‡) can be expanded similarly in a recursive manner. It can be verified that each such expansion brings in a term of the form $\mu(o_j = 1) \prod_{k<j} \mu(o_k = 0)\,\rho_j$. However, the expansion for the ith variable yields an exception since there's no appearance integral for a_i nor an incoming message $\mu(a_i)$ for the ith variable. It can be verified that this expansion brings in the term $\mu(o_i = 1) \prod_{k<i} \mu(o_k = 0)\,\nu_i$. Thus, the final form of the message is:

$$\mu_{\psi\to a_i}(a_i) = \sum_{j\neq i}\mu(o_j=1)\prod_{k<j}\mu(o_k=0)\,\rho_j + \mu(o_i=1)\prod_{k<i}\mu(o_k=0)\,\nu(a_i) \tag{21}$$

2 Bayes Optimal Depth Estimate Derivation

In this section, we provide details of our depth estimation procedure. As described in Section 4.2 of the submission, the Bayes optimal depth estimate along a ray can be computed as,

$$D^* = \arg\min_{D} \mathbb{E}_{p(D')}[\Delta(D, D')]. \tag{22}$$

For the ℓ_1-loss, $\Delta(D, D') = |D - D'|$, considered in this paper, the minimizer to Eq. 22 is given by D^* where $p(D < D^*) = p(D \geq D^*) = 0.5$; i.e., the median of $p(D)$ [3].

This estimation requires computing $p(D)$. However, our probabilistic model specifies occupancy variables o and appearance variables a, but not depth. We first express depth along a ray in terms of the occupancy variables in our model and then show the depth distribution $p(D)$ can be computed similarly to our inference procedure which estimates occupancy and appearance marginals. Once we have an estimate of $p(D)$, the Bayes optimal decision D^* is computed simply as the value such that $p(D < D^*) = p(D \geq D^*) = 0.5$.

We begin by defining the depth variable D in terms of occupancy variables o in our model. Similar to the image formation process, the depth forward process can be specified as

$$D(o_1, o_2, ..., o_N) = \sum_{i=1}^{N} o_i \prod_{j<i} (1 - o_j)\, d_i \tag{23}$$

where d_i is the distance of voxel i from the camera along the ray. The observed depth D is the depth of the first occupied voxel along the ray. We expose the special structure of the depth forward process as follows:

$$D(o_1, o_2, ..., o_N) = \begin{cases} d_1 & \text{if } o_1 = 1 \\ d_2 & \text{if } o_1 = 0, o_2 = 1 \\ d_3 & \text{if } o_1 = 0, o_2 = 0, o_3 = 1 \\ \vdots & \\ d_N & \text{if} o_1 = 0, ..., o_{N-1} = 0, o_N = 1 \end{cases} \tag{24}$$

Note that we do not consider here the possibility that the depth might be have been generated by a voxel outside the bounding volume, i.e., there exists at least one occupied voxel along the ray. Based on Eq. 24, $p(D = d_i)$ can be expressed in terms of the occupancy variables as follows,

$$p(D = d_i) = p(o_1 = 0, ...o_{i-1} = 0, o_i = 1) \tag{25}$$

Hence, the depth distribution requires computing the *joint* probability of occupancies along the ray. While our inference procedure estimates *marginal* distributions of occupancy (and appearance), it can be modified easily to compute the joint probabilities as well.

Namely, the sum-product belief propagation equations can be used to express the joint probability of occupancy and appearance variables along a ray as follows:

$$p(o_1, ..., o_N, a_1, ..., a_N) \propto \psi(\mathbf{o}, \mathbf{a}) \prod_{i=1}^{N} \mu(a_i) \prod_{i=1}^{N} \mu(o_i) \tag{26}$$

where μ denote the incoming messages and ψ is the ray potential. Since we're only interested in the joint probability of occupancies, the appearance variables are integrated out:

$$p(o_1, ..., o_N) \propto \int_{a_1} \dots \int_{a_N} \psi(\mathbf{o}, \mathbf{a}) \prod_{i=1}^{N} \mu(a_i) \prod_{i=1}^{N} \mu(o_i) \tag{27}$$

First, note that Eq. 25 does not require computing the joint distribution $p(o_1, ..., o_N)$ *for all* combinations of the occupancy variables. We only need to evaluate the distribution at certain configurations: $p(o_1 = 0, ...o_{i-1} = 0, o_i = 1)$ for all $i = 1, ..., N$.

We can compute $p(o_1 = 0, ...o_{i-1} = 0, o_i = 1)$ simply by summing out all other occupancy variables as follows:

$$p(o_1 = 0, ...o_{i-1} = 0, o_i = 1) = \sum_{o_{i+1}} ... \sum_{o_N} p(o_1 = 0, ...o_{i-1} = 0, o_i = 1, o_{i+1}, ..., o_N) \tag{28}$$

$$\propto \sum_{o_{i+1}} ... \sum_{o_N} \int_{a_1} \dots \int_{a_N} \psi(o_1 = 0, ...o_{i-1} = 0, o_i = 1, o_{i+1}, ..., o_N, \mathbf{a})$$

$$\times \prod_{j=1}^{N} \mu(a_j) \prod_{o=i+1}^{N} \mu(o_j) \times \left[\mu(o_i = 1) \prod_{j=1}^{i-1} \mu(o_j = 0)\right] \tag{29}$$

Note that the expression inside the square brackets does not depend on the variables summed/integrated and therefore can be taken out. Moreover, the ray potential ψ evaluates to $\nu(a_i)$, see Eq. 4. The equation simplifies as follows:

$$p(o_1 = 0, ...o_{i-1} = 0, o_i = 1) \propto$$

$$\left[\mu(o_i = 1) \prod_{j=1}^{i-1} \mu(o_j = 0)\right] \times \sum_{o_{i+1}} ... \sum_{o_N} \int_{a_1} \dots \int_{a_N} \nu(a_i) \prod_{j=1}^{N} \mu(a_j) \prod_{o=i+1}^{N} \mu(o_j) \tag{30}$$

The term $\nu(a_i)$ can also be taken out of the summations/integrals except the integral involving a_i as follows:

$$p(o_1 = 0, ...o_{i-1} = 0, o_i = 1) \propto$$

$$\left[\mu(o_i = 1) \prod_{j=1}^{i-1} \mu(o_j = 0)\right] \underbrace{\left[\int_{a_i} \nu(a_i)\mu(a_i)\, \mathrm{d}a_i\right]}_{\rho_i} \times \underbrace{\sum_{o_{i+1}} ... \sum_{o_N} \int_{a_1} \dots \int_{a_{i-1}} \int_{a_{i+1}} \dots \int_{a_N} \prod_{\substack{j=1 \\ j \neq i}}^{N} \mu(a_j) \prod_{o=i+1}^{N} \mu(o_j)}_{\substack{\text{assuming messages } \mu \text{ are normalized such that} \\ \text{they sum/integrate to 1, this term evaluates to 1.}}} \tag{31}$$

Thus, the joint probability simplifies to

$$p(o_1 = 0, ...o_{i-1} = 0, o_i = 1) \propto \mu(o_i = 1) \prod_{j=1}^{i-1} \mu(o_j = 0)\, \rho_i \tag{32}$$

Note the similarity of this equation to the ray-factor to variable message equations (Eq. 21, Eq. 17, Eq. 14). It can be easily computed as a by-product of the inference scheme.

Now that we have an expression for the joint probability, we can express the depth distribution using Eq. 25 as follows,

$$p(D = d_i) = \frac{1}{Z}\,\mu(o_i = 1)\prod_{j=1}^{i-1}\mu(o_j = 0)\,\rho_i \tag{33}$$

where Z is the normalization constant such that the depth distribution sums to 1.

3 Computing the Appearance Integrals

In this section, we present our method to compute integrals that arise in the sum-product message computations, Eq. 15. These integrals are of the form $\int_a \nu(a)\mu(a)\,\mathrm{d}a$, where $\mu(a)$ denote the incoming appearance messages and the $\nu(a)$ is a Gaussian distribution centered at the observed intensity I. We know the incoming appearance message up to a normalizing constant:

$$\mu_{a_i\to\psi}(a_i) = \prod_{g\in\mathcal{F}_{a_i}\setminus\psi}\mu_{g\to a_i}(a_i) = \frac{1}{Z}.\frac{p(a_i)}{\mu_{\psi\to a_i}(a_i)} \tag{34}$$

where $p(a)$ is the current belief (approximated with a mixture of gaussian distribution). Unfortunately these integrals do not admit to a closed form solution. We propose a sampling method and perform Monte Carlo integration. We write,

$$I = \int_a \nu(a)\mu(a)\,\mathrm{d}a = \frac{1}{Z}\int_a \nu(a)\frac{p(a_i)}{\mu_{\psi\to a_i}(a_i)}\,\mathrm{d}a \approx \frac{1}{Z}\frac{1}{N}\sum_{s=1}^{N}\nu(a^{(s)})\frac{p(a^{(s)})}{\mu(a^{(s)})} \tag{35}$$

where samples $a^{(\cdot)}$ are drawn from $p(a)$. We further approximate the normalizing constant Z using importance sampling as follows:

$$Z = \int_a \frac{p(a)}{\mu(a)}\,\mathrm{d}a \approx \frac{1}{N}\sum_{s=1}^{N}\frac{1}{\mu(a^{(s)})} \tag{36}$$

using the same set of samples drawn from $p(a)$.

4 Octree Construction

We use an octree data structure that is amenable to GPGPU processing [2]. The data structure consists of a uniform grid of shallow octrees (max. 4 levels). The boundaries of the working volume are automatically extracted from the Structure-from-Motion point cloud that is provided with the dataset [5]. Each octree is initialized as a single root cell. We alternate inference and octree refinement steps. In the inference step, we carry out the message update equations for each image once. We then traverse each octree cell and subdivide cells with substantial occupancy probability (larger than 0.3) into eight children cells. The messages for the children cells are initialized to a uniform distribution. We repeat this alternation procedure until a maximum number of iterations are reached or the beliefs do not change much.

Algorithm	Run time per image(seconds)
PM	1.02
Our algorithm	6.20

Table 1: Run times per image. The timings were averaged over all images in the downtown dataset.

5 Run Times

We present run times for the Pollard and Mundy (PM) algorithm [4] and our algorithm. For a fair comparison, we evaluate the run times on the same octree (roughly 18 million voxels). Since our implementations of both algorithms process one image at a time, we evaluate the run times per image. Table 1 presents the average run times per image in the downtown dataset.

References

[1] F. Kschischang. Factor Graphs and the Sum-Product Algorithm. *Information Theory, IEEE Transactions on*, 47(2):498–519, 2001. 1

[2] A. Miller, V. Jain, and J. L. Mundy. Real-time Rendering and Dynamic Updating of 3-d Volumetric Data. In *Proceedings of the Fourth Workshop on General Purpose Processing on Graphics Processing Units*, 2011. 8

[3] K. P. Murphy. *Machine Learning: A Probabilistic Perspective*. The MIT Press, 2012. 6

[4] T. Pollard and J. L. Mundy. Change Detection in a 3-d World. In *CVPR*, 2007. 9

[5] M. I. Restrepo, A. O. Ulusoy, and J. L. Mundy. Evaluation of Feature-based 3-d Registration of Probabilistic Volumetric Scenes. *ISPRS Journal of Photogrammetry and Remote Sensing*, 98(0):1–18, 2014. 8