Abstract — Time-of-flight range sensors and passive stereo have complimentary characteristics in nature. To fuse them to get high accuracy depth maps varying over time, we extend traditional spatial MRFs to dynamic MRFs with temporal coherence. This new model allows both the spatial and the temporal relationship to be propagated in local neighbors. By efficiently finding a maximum of the posterior probability using Loopy Belief Propagation, we show that our approach leads to improved accuracy and robustness of depth estimates for dynamic scenes.

Index Terms — Stereo, MRFs, Time-of-Flight Sensor, Data Fusion, Global Optimization.

I. INTRODUCTION

Depth is an important visual cue for understanding a scene. For decades, recovering 3D depth from stereo geometry has been an active topic in computer vision [1], artificial intelligence [2] and machine learning [3]. Although significant progress has been made during the last few years [4], the fundamental problems in stereo matching, such as textureless regions, repetitive patterns and occlusion, remain to be solved.

Recently, Time-of-Flight (TOF) range sensors are available from companies such as Swissranger [5], Canesta [6], PMD [7] and 3DV Systems [8]. TOF sensors use an active technique to obtain a full depth frame simultaneously at real-time. The basic principle of the TOF sensor is based on estimating the round-trip of a pulse of light. It is different from traditional laser-based Light Detection and Ranging (LIDAR) systems in that a TOF sensor can provide a full-frame depth map in one shot, instead of one single measurement. However, depth maps produced by TOF sensors are often in low resolution, noisy, and poorly calibrated which make their applications limited.

In this paper, we fuse a TOF sensor and passive stereo to get better depth maps. Different than existing approaches, we extend the fusion technology to the temporal domain, which generates improved depth map for dynamic scenes, and therefore strengthens one of the significant advantages of using TOF sensors over other active depth sensing techniques.

We develop our fusion algorithm based on the theory of Markov Random Field (MRF) [9] [10]. According to the Middlebury benchmark [11], most of MRFs-based stereo methods can generate high-quality depth maps by enforcing global constraints such as visibility [12].

However, most of algorithms are designed for one stereo image pair, e.g., one MRF per depth map in which the local relationship is propagated around its spatial neighborhood. We call such MRFs Spatial MRF (SMRF). In this paper, we extend SMRF to the temporal domain, which we refer to as a Spatial-Temporal MRF (STMRF), taking the temporal coherence into account. The idea behind STMRF is that a layered structure of SMRF is developed to infer depth from both spatial and temporal neighbors. The later is calculated by an dense optical flow method. Instead of treating the flow vector as a hard constraint, we formulate it as a data term in the MRF framework, increasing the robustness over erroneous flow vectors. We demonstrate the effectiveness of our approach by providing depth estimation results from a number of dynamic scenes.

Our algorithm can be also viewed as a temporal denoising and smooth scheme for dynamic depth maps. We found this particular important for the TOF sensor since its depth accuracy degrades as a function of object motion: the faster the object moves, the faster “shutter” time it requires, leading to the less accurate depth measurement polluted by sensor noise (e.g., lower signal to noise ratio).

The contributions of this paper are two folds: a new spatial temporal MRF model for robust depth inference and an efficient algorithm to fuse stereo and TOF sensor for dynamic scenes.

II. RELATED WORKS

In this section, we survey existing works on fusion of cameras with TOF sensors. We also review recent works on dynamic MRFs.

Most of previous fusion work on cameras and TOF sensors aim to (1) improving depth accuracy by stereo cameras and one TOF sensor and (2) upsampling the resolution of depth maps by one color/high resolution camera with one TOF sensor.

In [13], a stereo rig is fused with a PMD sensor [7] to improve the depth quality. Both sensors are adjusted to a common optical axis. The depth from PMD is simply replicated and merged with stereo straightforwardly. Instead of a straight forward approach, Guðmundsson [14] employed a non-direct method using a probability density function. By weighting the measurements differently (such as regions with rich texture or textureless regions), this method achieves better quality. However, this local method lacks power to estimate depth discontinuity correctly, which causes a visual artifact. Instead, our previous work [15] provides a global method to optimize depth from stereo matching and that from the TOF sensor. The results show decent depth maps for complex static scenes.

To upsample the low resolution depth map directly reported from a TOF sensor, Diebel et al. [16] provides a MRF model with four types of nodes. The cost function they designed is based on the fact that depth discontinuity tends to co-align with color difference. In addition, Yang et al. [17] provides a simple but effective upsampling method by a Bilateral Filter with a sub-pixel refinement post process.

However, previous fusion methods are constrained on a per-frame basis where local [13] [14] [17] and global fusion methods [16] [15] are limited in spatial space. And only results on static scenes are demonstrated. In this paper, we follow up our previous work and introduce a novel fusion approach by incorporating temporal coherence to infer depth from dynamic scenes.
Here, we also review recent works on MRFs applied to video as an extension to spatial MRFs. Most of these algorithms have been proposed using Graph Cuts or Belief Propagation for inference, where discrete labeling tasks can be well expressed.

Kohli and Torr [28] introduced a st-mincut approach to speed up the inference process using flows in MRFs with slightly different energy terms in video. By reusing the solution of the previous frame, they apply it to video segmentation by soft constrain.

Instead of recycling flows, Juan and Boykov [29] recycled cuts from previous frames. Their results show that the efficiency of st-mincut is proportional to the amount of motion between two consecutive frames. Alahari et al. [33] introduced an efficient approach for multi-label energy minimization by also reusing the labeling of the previous frame. This provides a good initialization for the subsequent instance which can substantially reduce the computing power.

Isard and Macormick [30] provided an approach that can inference depth, occlusion and motion simultaneously by Belief Propagation. However, the algorithm is limited for low resolution video because of the large labeling space by numbers from disparity × motion.

In addition, Yin and Collins [31] employed a spatial-temporal MRFs with a 6-connected nodes for moving object detection. And Chen and Tang [32] proposed a new spatial temporal MRFs for video denoising.

Most of these models, however, either take the flow as a deterministic temporal correspondence or label the current frame using previous solution. Unlike previous approaches, we incorporate a N-layer MRFs to do forward and backward belief propagation which can substantially increase the accuracy and robustness.

III. SPATIAL-TEMPORAL MRF MODEL FOR DEPTH INFERENCE

In this section, we first introduce our approach by formulating a binary MRF model that incorporates both spatial and temporal relationship. And then we present our labeling scheme which is benefited from the MRF model with different truncational potential functions. We also give a short explanation of Loopy Belief Propagation which is used to do approximate inference on our MRF model.

A. Energy Minimization Formulation

Given evidence \( z \) (such as an image intensity map) from stereo, estimation of the depth map \( d \) can be viewed as a conditional probability problem. Its posterior probability can be written as:

\[
P(d|z) = \frac{P(d,z)}{P(z)} \propto P(d,z)
\]

To resolve the joint distribution \( P(d,z) \), we model it as a MRF, where nodes represent depth values and observations, edges represent dependencies and each clique is associated with a potential function. This MRF model is a binary MRF, because the size of its maximal clique is two. According to the Hammersley-Clifford theorem, the joint distribution of \( P(d,z) \) can be written as:

\[
P(d,z) \propto \prod_i \phi(d_i, z_i) \prod_{j \in N(i)} \varphi(d_i, d_j)
\]

where \( N(i) \) represents the neighborhood of the node \( i \). Function \( \phi(d_i, z_i) \) is called the potential function and encodes the local evidence for the node \( i \) based on the initial pixel-wise matching cost. Function \( \varphi(d_i, d_j) \) is called the compatible function. It is a symmetric function measuring the smoothness assumption on the node \( i \).

Using negative logarithm on both sides of equation (2), we get

\[
-\log(P(d, z)) = -\log(\prod_i \phi(d_i, z_i) \prod_{j \in N(i)} \varphi(d_i, d_j))
\]

\[
= \sum_i -\log(\phi(d_i, z_i)) + \sum_{j \in N(i)} -\log(\varphi(d_i, d_j))
\]

The maximum probability assignment given the evidence can be minimized by:

\[
\max(p(d_i, z_i)) = \min(\sum_i -\log(\phi(d_i, z_i)) + \sum_{j \in N(i)} -\log(\varphi(d_i, d_j)))
\]

The first sum in above equation represents a data term which is calculated from the observed data (evidence). The second sum is usually called a smoothness term, which encodes a prior knowledge of confidence on which depth should be selected, given by its neighbors’ depth. We use function \( D(i) \) to represent the data term and function \( V(i, j) \) to represent the smoothness term. \( j \) is neighbors of the pixel \( i \). An energy minimization function in our problem can be formulated as:

\[
E = \sum_i D(i) + \sum_{i \in N(i)} V(i, j)
\]

B. Extension to the TOF sensor

Depth from the TOF sensor can be incorporated as another evidence \( y \) besides that from stereo \( z \). Assuming \( z \) and \( y \) are independent, our posterior probability function becomes:

\[
P(d|z, y) = \frac{P(d, z, y)}{P(z, y)} = \frac{P(d, z, y)}{P(z)P(y)} \propto P(d, z, y)
\]

We derive a similar energy function by introducing two weighting factors to allow more flexibility for the data term to fuse the depth from stereo matching and that from the TOF sensor. That is

\[
D(i) = w_s \cdot f_s(d_i, z_i) + w_r \cdot f_r(d_i, y_i),
\]

where \( f_s \) and \( f_r \) are functions of the data cost for stereo and the TOF sensor, respectively. We will present detailed method of calculating these two cost functions in the later section. \( w_s \) and \( w_r \) are corresponding weighting factors for each method, respectively.

C. Extension to temporal coherence

To incorporate the temporal coherence, we extend SMRF to STMRF using a layered structure. In our model, each layer is a binary MRF itself and edges among layers represent the temporal relationship. Figure 1 illustrates a 3-layer example.

In this model, the unknowns (depth values represented by white circles) are connected by undirected edges which means they are statistically dependent in their neighborhood. There are no edges among observation nodes which means evidences are statistical independent.

By extending SMRF to temporal coherence, we incorporate the third evidence \( t \) besides \( z \) and \( y \), and similarly by assuming \( z, y \)
and $t$ are independent, our posterior probability becomes:

$$P(d|z,y,t) = \frac{P(d,z,y,t)}{P(z,y,t)} = \frac{P(d,z,y,t)}{P(z)P(y)P(t)} \propto P(d,z,y,t) \quad (8)$$

Instead of incorporating $y$ inside data term, here we incorporate $t$ inside smoothness term. By introducing another weighted smoothness function $V_t(\cdot)$ to constrain temporal smoothness in equation (5), the final energy function minimized in our model can be written as:

$$E = \sum_i (w_s f_s(i) + w_r f_r(i)) + \sum_{i \in N_s(i)} V_s(i,j) + w_t \sum_{j \in N_t(i)} V_t(i,j) \quad (9)$$

where $N_s(i)$ and $N_t(i)$ are spatial and temporal relationship for pixel $i$, respectively. $V_s(\cdot)$ and $V_t(\cdot)$ are spatial and temporal smoothness functions, respectively. $w_t$ is the weight coefficient for temporal smoothness. In the later section, we will introduce how to calculate all the weighting factors adaptively.

D. Calculate Different Terms

In this section, we present details on cost function formulation and weight coefficient calculation in equation (9).

a) Data terms: Labeling on data terms is based on the observed data. In our implementation, we design cost functions based on geometric distances between current interested pixel (a pixel in left view) and its candidates (a vector of pixels in right view as potentially matched).

Stereo Matching

The cost function of stereo matching $f_r$ is designed as pixel dissimilarity between the left and right views (the two views are rectified in which matched pixels can be searched on a scan line) with an aggregation process. To weight on both smooth and discontinuous regions, an appropriate window should be selected during the cost aggregation. In a sense, the window should be large enough to cover sufficient area in textureless regions, while small enough to avoid crossing regions with depth discontinuities. In our implementation, we incorporate a color weighted aggregation to obtain this reliable correlation volume, which is based on adaptive weight aggregation strategy [34].

We compute the weights using both color and spatial proximity to the central pixel of the support window. The color difference in this support window (in the same view) is expressed in RGB color space as:

$$\Delta C(x,y) = \sum_{c=R,G,B} |I_c(x) - I_c(y)| \quad (10)$$

where $I_c$ is the intensity of the color channel $c$. The weight of pixel $x$ in the support window of $y$ (or vice versa) is then determined using both its color and spatial difference as:

$$w_{xy} = e^{-\frac{\Delta C(x,y)}{\gamma_C} + \frac{\Delta G_{xy}}{\gamma_G}} \quad (11)$$

where $\Delta G_{xy}$ is the geometric distance from pixel $x$ to $y$ in the 2D image grid. $\gamma_C$ and $\gamma_G$ are constant parameters controlling the shape of the weighting function, which are determined empirically.

The data term (cost in the left and right views) is then an aggregation with the soft windows defined by the weights as:

$$f_s(x_l, x_r) = \frac{\sum_{y_l, y_r \in W(x_l) \times W(x_r)} w_{xy} d(y_l, y_r)}{\sum_{y_l, y_r \in W(x_l) \times W(x_r)} w_{xy}} \quad (12)$$

where $W(x)$ is the support window around $x$; $d(y_l, y_r)$ represents the pixel dissimilarity using Birchfield and Tomasi’s approach [42]; $x_l$ and $y_l$ are pixels in the left view; $x_r$ and $y_r$ are pixels in the right view. Results show that $f_s$ can provide both moderate smoothness and preserve boundary sharpness on depth.

The TOF sensor

The cost function of the TOF sensor $f_r$ is designed as the depth difference reported from the TOF sensor $d_r$ and a vector of depth candidates from stereo triangulation $d_s$. The transformation between them is estimated by a pre-calibration step similar to the geometric calibration step in [15] where the three cameras (in calibration, the TOF sensor is regarded as a regular camera because it can return a gray scale image besides a depth map.) are unified into one coordinate system. We explain the method in the following.

For each pixel $p$ in the left view, given a vector of disparity candidates $d_c$, we get a vector of 3D points as candidates of $p$ in 3D space using stereo triangulation. By assuming that the real 3D point is one of the member in this vector, we assign the depth costs of pixel $p$ by their difference to the point reported by the TOF sensor. This process is further illustrated in Figure 2.

To maintain large depth variations, we incorporate a linear truncation model:

$$f_r(i) = \exp \frac{-s_1 \times \min\{|d_s(i) - d_r(i)|, T_1\}}{\gamma_D} \quad (13)$$

where $\gamma_D$ controls the shape of the weighting function. $T_1$ is the truncation value. In this paper, we set its value by 0.3 meters while the working volume is around 2 meters. To normalize the cost to [0, 1], we set $s_1$ as the inverse of $T_1$.

b) Smoothness term: Smoothness term is typically used to specify the smoothness assumptions on the labeling. In our problem, this term encodes the depth consistency which is based on the assumption that depth should be piecewise constant. In the sense, the smoothness cost should be decreased at depth edges, while it should be increased on regions that is locally smooth.

Spatial Smoothness By a further assumption that depth and color tend to co-align, we design our smoothness cost function using a quadratic truncational model where small differences cause smaller penalties and large differences cause larger penalties. This
encourages a few places where nearby pixels that change their costs significantly as:

$$V_s(i, j) = s_2 \times \min\{(I_i - I_j)^2, T_2\}, j \in N_s(i)$$  \hspace{1cm} (14)$$

where $I_i$ and $I_j$ are the intensity values of selected pixel $i$ and its neighbors $j$. In this paper, we set $T_2$ to 30 while the maximum intensity value is 255. $s_2$ is also set to the inverse of $T_2$ to normalize costs between $[0, 1]$.

**Temporal Smoothness**

During our implementation, we record the previous frame’s final costs $c^{t-1}$. We also compute the next frame’s local cost $c^{t+1}$, $c^{t-1}$ and $c^{t+1}$ are used as evidence in the temporal domain. Considering the noise in optical flow estimation, we define a local neighborhood $N^{t-1}_s$ on the previous frame and compute the mean cost within $N^{t-1}_s$ as the final temporal evidence. The cost from the next frame is similarly computed. As a result, we derive our cost function of temporal smoothness as:

$$V_t(i^t, i^{t-1}, i^{t+1}) = \sum_{k \in N_s(i^{t-1})} \frac{c_k^{t-1}}{|N_s(i^{t-1})|} + \sum_{k \in N_s(i^{t+1})} \frac{c_k^{t+1}}{|N_s(i^{t+1})|}$$  \hspace{1cm} (15)$$

where $i^t, i^{t-1}, i^{t+1}$ are matched pixels in three sequential frames. $|N_s(i^{t-1})|$ and $|N_s(i^{t+1})|$ is the number of pixels in the temporal neighborhood of pixel $i$ at frame $t-1$ and $t+1$, respectively.

All the terms are defined, a 6-node grid will be used to do message updating. Figure 3 shows such an example.

We choose Loopy Belief Propagation (LBP) [26] to do depth inference. Although, LBP is not guaranteed to converge or be correct, it has been applied with experimental success [35]. Besides, Graph Cuts (GC) is another common algorithm for approximating inference to energy minimization problems like ours. The comparison between LBP and GC can be found from an excellent work from Tappen [27].

Without any simplification, LBP requires enormous time to converge. We employ hierarchical belief propagation [38] to speed up the convergence. The difference between the hierarchical BP and general BP is that the former works in a coarse-to-fine manner, first performing BP at the coarsest scale, then using the beliefs from the coarser scale to initialize the input for the next scale.

**IV. EXPERIMENT RESULTS**

In this section, we first introduce our setup, and then we analyze three key problems in the fusion approach: calibration, synchronization and cost balance. By investigating answers, we provide an experimental system architecture. We evaluate results from different methods qualitatively and quantitatively.

A. Multi-sensor Setup

We fuse a TOF sensor with a pair of stereo cameras (as shown in Figure 4). The cameras we are using is the DragonFly2 IEEE-1394 CCD camera which can capture 200 frames per second at most. The TOF sensor we have is a SwissRanger SR3000 [5], which can continuously produce a depth map of $176 \times 144$ resolution with an operational range up to 7.5 meters. In our current setup, two cameras have a baseline about 12cm and they are verged towards each other around 10 degree from the parallel setup.

B. Practical Issues

There are three main challenges of fusing stereo with a TOF sensor: (1) Calibration. TOF sensors are usually poorly calibrated, which makes it hardly usable for computer vision algorithms. (2) Synchronization. The sampling rate of the camera and the TOF sensor are different. The TOF sensor has no means of providing or accepting an external sync signal. (3) Cost balance. To maximize the depth accuracy, the weights of the data terms $w_s$, $w_r$ and $w_t$ have to be carefully selected. This, however, is hard in practice since different scenes may require different weights.

To resolve the first problem, we follow up Zhu [15]'s work to calibrate the TOF sensor with stereo cameras in both geometrical and photometrical. The basic idea of geometrical calibration is that the TOF sensor we have (Swissranger SR-3000) can also
return a gray scale image besides a depth map. This allows us to regard the TOF sensor as a regular camera, which makes traditional camera calibration techniques usable to calibrate these three cameras into one unified coordinate system. The idea behind photometric calibration is based on the assumption that sensitiveness of the TOF sensor to different scene reflectance is a linear function. This allows us to calibrate the depth bias using two planar planes in white and black color. For other scene reflectance in different colors, we simply do an linear interpolation. Other works of calibration a TOF sensor using CCD cameras can also be found in Schiller [49] and Linder [50].

ToF calibration is an active topic and is important to our fusion results. In general, existing calibration of Time-of-Flight sensors can be categorized into two major classes: photometric calibration and geometric calibration. Photometric calibration usually identifies the systematic error at various exposure times and stores it for later compensation. Geometric calibration measures the geometric relationship between TOF sensors with other sensors. The result can be used for fusing other information from multi-sensors.

In photometric calibration, approaches are provided according to different noise sources that may reduce the accuracy of depth measurement. Anisotropic LED lights and lens distortions are regarded as internal errors which can be reduced by calibration using a look-up-table [44] or B-Splines [46]. Ambient light is usually regarded as the main external uncertainty and can be removed by calibration using multi-shutter [47]. Other external uncertainties can also be calibrated by providing additional equipments. One example is to calibrate the environmental temperature using a high precise temperature sensor [48].

In geometric calibration, the TOF sensor is usually regarded as a regular camera because it can return an intensity image besides a depth map. Traditional camera calibration techniques therefore can be used to compute the relative pose between the TOF sensor and cameras [49], [50]. Besides, calibration can also be achieved by estimating the center of a ball in multi-sensors from a video sequence in [51].

We calibrate the TOF sensor with stereo cameras both geometrically and photometrically. The basic idea of geometrical calibration is that the TOF sensor we have (Swissranger SR-3000) can also return a gray scale image besides a depth map. This allow us to regard the TOF sensor as a regular camera, which makes traditional camera calibration techniques usable to calibrate these three cameras into one unified coordinate system. The idea behind photometric calibration is based on the assumption that sensitiveness of the TOF sensor to different scene reflectance is a linear function. This allows us to calibrate the depth bias using two planar planes in white and black color. For other scene reflectance in different colors, we simply do an linear interpolation.

To resolve the second problem, we synchronize the two video cameras using the MultiSyn toolkit which is provided by Pointgrey company [39]. To synchronize the stereo cameras with the TOF sensor, we incorporate a soft method by recording their time stamps and finding the closest in the temporal domain. Because video cameras’ capturing rate (we set it to 100 FPS) is much higher than the TOF sensor (around 20 FPS), this nearest-neighbor approximation yields satisfactory result.

To resolve the third problem, we adaptively adjust the weight coefficient in equation (13). Given a vector of depth candidates, we know that the best candidate should have a low cost while others are obviously larger. We therefore intuitively define \( w_s, w_r, w_t \) as how distinctive its best cost \( c_1^{st} \) and its second best cost \( c_2^{nd} \) is:

\[
w_{k \in \{s,r,t\}} = \begin{cases} 
1 - \frac{c_2^{nd}}{c_2^{max}} & \text{if } C_{k}^{2nd} > T_c \\
0 & \text{otherwise}
\end{cases}
\]

\( T_c \) is a small value to avoid \( c_2^{nd} \) equals zeros. Results show that our adaptive weighting can accurately describe the complimentary nature of stereo matching, depth from the TOF sensor and smoothness from temporal neighbors.

C. Experimental Architecture

In this section, we introduce our experimental architecture in Figure 5. The architecture works like this: given the first synchronized stereo-TOF pair at \( t-1 \), it calculates all nodes’ data cost by equation (7) and smooth cost by equation (14). And then it iteratively optimizes costs globally from local neighbors using
LBP. This cost is restored and summed with the cost from stereo at $t + 1$ by equation (15). This final cost is treated as the temporal evidence. The temporal coherence between each two sequential images are calculated using a dense optical flow approach [40]. Since the capture frequency of stereo is high (the displacement between two sequenced frames are small), it is safe to assume that the dense temporal correspondence is available.

Given the temporal matching, we have two sources of input for unsynchronized image pairs (evidence from stereo matching and evidence from temporal neighbors). However, we have an extra input (evidence from the TOF sensor) for synchronized stereo-TOF pair. Results show that this architecture helps to maintain evidence from previous/next frame and propagate both spatial and temporal relationship in local neighborhoods globally.

One significant benefit of our architecture is that even without evidence from the TOF sensor, STMRF works as well on regular cameras. And it can be easily extended to other sensors.

### D. Evaluation

Before comparing results from SMRF and STMRF, we did a simple test to show that the quality of depth maps from SMRF method is influenced by the Signal to Noise Ratio (SNR) of the TOF sensor. In one of our previous work in fusing stereo with TOF sensors [15], we focused on static scenes and took several depth maps and compute the mean. The result shows that it is less susceptible to image noise. The mean depth map was used as input. In Figure 6, we visualize 3D points and depth maps using one shot (low SNR) and the averaged of 10 shots (high SNR).

We can clearly see the depth map from multiple shots is better than that from the single shot. However, simple average over multiple shots can be applied only to static scenes. Our STMRF method uses temporal correspondences to reduce noises and improve depth estimate in dynamic scenes. In this way, it can also be viewed as a temporal denoising scheme when applying to the TOF sensor alone.

To verify the effectiveness of enforcing temporal smoothness $V_{t}$, we generate a set of 5-frame ground truth data of a moving scene. The object is manually rotated and acquired in stop motion. The ground truth depth is obtained using structured light techniques [36]. As shown in Table I, both mattes and depth are improved. One frame of qualitative comparison of matting with/without $V_{t}$ from video sequence can be found in Figure 7.

We evaluate our STMRF method on a number of real scenes. For each scene, we compare depth maps and their geometrical representations (3D point clouds). All of them are using the left camera as the reference view. To make a fair comparison, we set the message truncation value $T_{m}$ = 0.3, and we set the TOF sensor’s capturing frequency around 20 FPS and the stereo rig around 100 FPS. All compared frames are using the TOF sensor’s time stamp as the reference.

In the first test, we choose a simple scene includes only two boxes. One with some patterns on it, the other is almost textureless. We make the boxes do an uniform rotation. We compare the depth maps of 14th and 92th frames in Figure 8.

We can see SMRF method [15] can substantially optimize the depth by making use of the complimentary characteristics, and the overall depth map is correct. However SMRF can not remove the incorrect depth discontinuity and still performs poorly on local parts. Our STMRF method incorporates evidence on the temporal axis additionally, and thus the depth map is much more acceptable both in overall and local details.

Besides qualitative comparison, we plot the $5^{th}$ scanline depth and record one pixel’s depth at $(80, 66)$ in 100 frames. We compare results from SMRF and STMRF in Figure 9.

The left bottom figure shows the data along a single scan line as scaled disparity values. We can see there are many false depth jumps from SMRF, while STMRF can effectively remove these local peaks by incorporating additional evidence from temporal domain.

The right figure helps us to investigate the reason that our method (STMRF) performs well on global consistency by comparing one pixel’s depth in a 100 frames sequence. The line from SMRF obviously encodes the ridging artifacts. Our analysis indicates that these artifacts are due primarily to the fact that SMRF does not taking into account constraints among temporal neighbors. On the contrary, STMRF can substantially maintain globally consistency by a probability inheriting temporal smoothness, which presumably leads to much better depth estimates.

We also evaluate results with two other scenes. Such scenes are challenging because of some textureless regions, occlusions and part of non rigid motion. These scenes are important because they are much more likely to be captured in our daily life. We select two frames for each scene and compare their depth maps and 3D points rendered with colors in Figure 10 and Figure 11, respectively.

Finally, we test our algorithm on a textureless scene. We show results in Figure 13. We can see although there are many false depth discontinuities on the background (this is due to the bad matching from stereo algorigh), our STMRF approach can still achieve satisfied results.

We can see that our STMRF is superior than SMRF in overall depth estimates, especially in local smoothness which is an important cue for visual appearance. Part of sequence of the results presented here are included in our supplementary video.

### E. Discussion

Results show that our STMRF performs well on dynamic, complex and even little textured scenes, however, it does poorly when high dynamic movements are presented in the scene. This is mainly caused by two reasons: motion blur and synchronization. The TOF sensor reports scene depth either by multiple shots [8] or integration over time [5]. Rapid movement causes motion blur on depth boundaries which reduces our trust to the term $f_{c}$ from the TOF sensor. We show two blurred example in Figure

The second reason is that although the frame rate of video cameras can reach 100, the TOF sensor has only a capacity of 20 FPS (25 at most from the hardware specification of a Swissranger SR-3000) in practice. Therefore, the synchronized frames are actually down-sampled to 20 FPS using our soft synchronization.
Fig. 6. Comparison of depth maps and 3D points from one shot and averaged 10 shots. The noisy data from one shot leads to poor depth map, while averaging 10 shots for the same scene reduces the 3D noise and improves the depth estimates.

Fig. 7. An example (frame 4) of depth map with ground truth.

Fig. 8. Comparison of depth maps and 3D points of 14th and 92th frame. From the depth maps, we can see STMRF performs well on local details than SMRF. This can be also observed from 3D point clouds, where the geometry from STMRF is well preserved. A complete comparison of depth maps in the video sequence is included in our supplementary video.

We can see fast movement in a scene is actually a challenge for our approach. An interesting extension is to interpolate the depth reported from the TOF sensor using the optical flow estimated from color images. We envision this anticipated extension will increase the FPS of the TOF sensor 2 to 4 times in temporal space, which can finally generate decent fast motion results.

V. SUMMARY

We have presented a spatial and temporal MRF to infer high quality dynamic depth by fusing of stereo and a TOF sensor. We regard both sensor’s measurement as a probability density function, and model the measurements with a binary MRF. By minimizing the energy function composed of fused data term and extended smoothness term with temporal coherence, our method
outperforms traditional spatial only MRFs and enhances temporal smoothness. Beyond stereo matching, we believe our strategy of creating temporal constrained MRFs and aligning/merging different cost functions can also be extended to other problems such as data segmentation or data completion over a video sequence.

However, our STMRF approach can not generate satisfied results when high dynamic motions presented in the scene. Recently, by assuming that the color and depth are co-aligned, segmentation based approach can improve the depth estimation [41]. Although we have not incorporated this technique, we envision that color segmentation can provide cues for (1) improving the accuracy of temporal correspondence (2) detecting and recovering the depth on the motion boundaries for the TOF sensor which are not addressed in this paper.

REFERENCES

Fig. 11. Comparison of 3D point clouds from depth maps using SMRF and STMRF. Artifacts of jumping in depth map and holes in 3D points from SMRF are correctly removed by STMRF. Besides, 3D noises are efficiently suppressed by STMRF.

Fig. 12. Comparison of depth maps from different methods in a textureless scene. The whole sequence of comparison is included in the supplemented video.

Fig. 13. Two examples of motion blur in depth reported by a Swissrange sensor. We can see blur occurs near boundaries on the moving hand and the box.


