https://doi.org/10.2352/ISSN.2470-1173.2018.15.COIMG-473
© 2018, Society for Imaging Science and Technology

Multiscale Matched Filter For Structured Light Decoding Using Sequential MAP Estimation

Hasib Siddiqui, Kalin Atanassov , and Magdi Mohamed; Qualcomm Technologies Inc.; San Diego, CA 92121

Abstract

Structured light depth sensors work by projecting a codeword pattern, usually made up of NIR light, on a scene and measuring distortions in the light received on an NIR camera to get estimates of the camera-projector disparities. A well-known challenge associated with using structured light technology for depth estimation is its sensitivity to NIR components in the ambient illumination spectrum. While various methodologies are employed to increase the codeword-to-ambient-light ratio – for instance, using narrow-band NIR filters and selecting a spectral band for the NIR laser where the interference from ambient light is expected to be low – structured light setups usually do not work well outdoors under direct sunlight. The standard deviation of shot noise increases as the square root of the ambient-light intensity, reducing the SNR of the received codeword pattern and making the decoding process challenging.

One way to improve the SNR of the received structured light pattern is to use codewords of larger spatial support for depth sensing. While large codewords do improve the SNR of the received pattern, the disadvantage is decreased spatial resolution of the estimated disparity field. In this paper, we use a multiscale random field (MSRF) to model the codeword labels and use a Bayesian framework, known as sequential MAP (SMAP) estimation, developed originally for image segmentation, for developing a novel multiscale matched filter for structured light decoding. The proposed algorithm decodes codewords at different scales and merges coarse-to-fine disparity estimates using the SMAP framework. We present experimental results demonstrating that our multiscale filter provides noise-robust decoding of the codeword patterns, while preserving spatial resolution of the decoded disparity maps.

Introduction

Structured light is a triangulation based active range finding technique which solves the stereo correspondence problem by transmitting a known pattern and comparing the received image to the recorded template in order to calculate the local disparity field [1].

The structured light transmission system usually comprises a near infra red (NIR) laser transmitter and a diffractive optical element (DOE). The laser light passes through the DOE, projecting a pattern of unique codewords on the scene. For our setup, the transmitted codewords are 2D binary sequences. The codewords reflected off the scene, and the unsolicited background image due to ambient light, are received by an NIR camera. Due to channel distortion, the received codewords on the sensor are no longer binary, but form a grayscale texture comprising *dots* and *holes*, with each dot or hole occupying a region of $P \times P$ pixels on the sensor. The schematic of a typical structured light setup and the received codeword patterns is shown in Figure 1.

Decoding of the received light pattern enables immediate recovery of the camera-projector disparities without having to explicitly solve the stereo correspondence problem. The codeword pattern acts as texture and hence disparities for texture-less objects in the scene can also be sensed. Once the disparity field is estimated, the depth of the scene is determined through knowledge of the NIR camera resolution, system baseline, and the receiver optics.

Since the projected pattern is made up of NIR light, the performance of structured light depth sensing deteriorates if the ambient illumination produces a strong interference signal in the NIR spectral band. Several methodologies are employed to increase the gamut of ambient illuminations where structured light depth sensing can work. For example, Kinect$^{\text{ToF}}$ and Kinect$^{\text{SL}}$ both make use of narrow band NIR filters to suppress the background light out of the range of the laser illumination [2, 3]. For outdoor depth sensing, it is a common practice to select a narrow wavelength band around 940 nm for the laser illumination, the spectral band where the radiant flux due to solar illumination is known to be small [4].

The challenge in structured light decoding is to correctly identify and label the received codeword pattern in the presence of channel distortion and strong ambient interference, particularly under direct sunlight. The standard deviation of shot noise increases as the square root of the ambient-light intensity, reducing the SNR of the received codeword pattern and making the decoding process difficult [5]. In this paper, our focus is to improve structured light decoding through algorithmic advancements.

We propose a novel multiscale matched filtering strategy that decodes structured codewords at different scales and merges coarse-to-fine disparity estimates using a Bayesian framework, known as sequential MAP estimation [6], originally proposed for segmenting multispectral images. We extend the SMAP theoretical framework for decoding structured-light patterns. We present experimental results demonstrating that our proposed multiscale filtering framework provides noise-robust decoding of the codeword pattern, while preserving spatial resolution of the decoded depth maps.

The rest of the paper is arranged as follows. In Section , we develop a single-scale matched filter that uses features and codeword dictionary elements of a fixed spatial support for structured light decoding. We shall look into limitations of such a filter and proceed to developing a multiscale matched filter, based on SMAP estimation, in Section . The experimental results are presented in Section , followed by the conclusions in Section .

Matched Filter for Structured Light Decoding

Structured light depth sensing, as shown in Figure 1, works by projecting a known codeword pattern on a scene and comparing an $L \times L$ local patch in the received image to recorded templates in a dictionary of M codeword elements that are known *apriori*. The objective of the decoding algorithm is to identify the codeword label, x, for the given patch. Knowledge of the codeword label enables immediate recovery of the camera-projector disparity through a deterministic transformation [1].

Signal Model

The image captured on the NIR sensor, $\mathscr{Y}$ is modeled as a linear combination of three components:

- *Codeword image*, $\mathscr{G}$, formed by the NIR structured light reflected off the scene;
- *Interference image*, $\mathscr{B}$ formed by the ambient light reflected off the scene; and
- *Noise image*, $\mathscr{W}$.

Let y_s, g_s, b_s, and w_s denote pixel values in $L \times L$ patches of images $\mathscr{Y}$, $\mathscr{G}$, $\mathscr{B}$, and $\mathscr{W}$, respectively, with their top-left corners at pixel s. We shall assume the pixel values in the local patches are arranged as L^2 dimensional column vectors. The observed feature y_s is given by

$$y_s = g_s + b_s + w_s.$$

The laser light passing through the diffractive optical element projects a binary dot pattern on the scene. Thus, the $L \times L$ patch, g_s, received at pixel s on the NIR sensor is a channel-distorted version of an $\frac{L}{P} \times \frac{L}{P}$ transmitted binary sequence, represented by the label

$$x \in \chi \triangleq \{0, 1, \ldots, M-1\},$$

where $P \times P$ is the spatial support of a *dot* or *hole* received on the NIR sensor (Figure 1) and M is the total number of unique $\frac{L}{P} \times \frac{L}{P}$ binary codes projected by the transmitter.

The distortion function is difficult to model accurately, as it depends on a number of factors including surface reflectance properties of objects in the scene, the point spread function (PSF) of the NIR camera lens, and the PSF of the NIR sensor. We shall assume that the cumulative effect of the above distortions is simply:

- a spatially-dependent attenuation of the transmitted signal and
- a spatially-invariant blurring operation, considered fixed regardless of the scene content and camera capture conditions.

Thus, denoting by $f(x) : \chi \to \mathbb{R}^{L^2}$ the blurred $L \times L$ codeword dictionary elements corresponding to labels $x \in \chi$, the received codeword g_s on the NIR sensor can be expressed as:

$$g_s \triangleq a_s f(x_s),$$

where x_s denotes the codeword label and $a_s > 0$ denotes the *codeword attenuation* at pixel s on the NIR sensor. Since the blurring kernel is considered fixed, the dictionary elements $f(x_s)$ can be computed *apriori*.

In our modeling, we shall further assume for simplicity that for each $x \in \chi$, the codeword pattern $f(x)$ is de-meaned and standard-deviation normalized, i.e.

$$\sum_i f_i(k) = 0 \text{ and } \frac{1}{L^2}\sum_i f_i^2(k) = 1 \ \forall k \in \chi.$$

For structured light decoding, it is reasonable to assume that the high-pass energy in the observed patch y_s is primarily due to noise or the received codeword. The interference image is treated as locally constant, contributing only a dc offset, denoted by the positive scalar b_s, to the local patch. Thus, the L^2 dimensional vector b_s can be simplified as:

$$b_s \triangleq b_s \mathbf{1}_{L^2},$$

where $\mathbf{1}_{\mathbf{L^2}}$ represents an L^2 dimensional column vector with all 1ś.

With these assumptions, the observation vector y_s can be modeled as:

$$y_s = a_s f(x_s) + b_s \mathbf{1}_{L^2} + w_s. \tag{1}$$

Given the observation patches, y_s for $s \in S$, where S denotes the set of all pixels in the image, the objective in structured light decoding is to estimate the codeword labels, x_s. Knowing the codeword labels, x_s, allows us to uniquely determine the disparity values, d_s, through a deterministic transformation:

$$d_s = h(s, x_s).$$

We shall assume for the rest of our discussion that the function $h(.,.)$ is known.

Maximum Likelihood (ML) Estimation

Suppose that $w_s \in \mathbf{R}^{L^2}$ in (1) is additive white Gaussian noise with a probability distribution $\mathscr{N}\left(0, \sigma^2 I_{L^2}\right)$, where I_{L^2} is an $L^2 \times L^2$ identity matrix. The conditional distribution of y_s given x_s, a_s, b_s, can then be written as:

$$p(y_s | x_s, a_s, b_s) = \frac{1}{\left(2\pi\sigma^2\right)^{L^2/2}} \exp\left\{-\frac{1}{2\sigma^2}\|y_s - a_s f(x_s) - b_s \mathbf{1}_{L^2}\|_2^2\right\}. \tag{2}$$

Assuming that the observation vectors y_s are conditionally independent of the neighboring pixels given x_s, a_s, and b_s, the log likelihood of the observed data can be written as a sum of independent terms:

$$\begin{aligned} -\log p(y|x,a,b) &= \sum_{s \in S} l(y_s|x_s) & (3)\\ &= \sum_{s \in S} \frac{1}{2\sigma^2}\|y_s - a_s f(x_s) - b_s \mathbf{1}_{L^2}\|_2^2 + \\ &\quad \frac{L^2}{2}\log(2\pi\ \sigma^2). & (4) \end{aligned}$$

The ML estimates of x_s, a_s, b_s are computed through minimization of the negative log likelihood function (3). A straight forward computation yields:

$$\hat{x}_s = \underset{k\in\chi}{\operatorname{argmax}}\, y_s^T f(k), \tag{5}$$

$$\hat{a}_s = \frac{1}{L^2} y_s^T f(\hat{x}_s), \text{ and} \tag{6}$$

$$\hat{b}_s = \frac{1}{L^2} y_s^T \mathbf{1}_{L^2}. \tag{7}$$

Thus, the ML estimate of the codeword label x_s is simply the index $k \in \chi$ for which the codeword dictionary element $f(k)$ is maximally correlated with the observation vector y_s. The per-pixel computation involved in determining this estimate is the multiplication of an $M \times L^2$ dimensional matrix, $F \triangleq [f(0),\ldots,f(M-1)]^T$, with an L^2-dimensional vector y_s. These computations can be efficiently performed in a GPU, or multi-threaded in a CPU, for real-time applications.

Outlier Rejection

To establish our confidence in the codeword labeling, $\hat{x}_s$, computed using ML estimation, we compute the posterior probability that $X_s = \hat{x}_s$ given the observation vector y_s, the codeword reflectance estimate $\hat{a}_s$, and the interference signal estimate $\hat{b}_s$, i.e.

$$p(\hat{x}_s|y_s,\hat{a}_s,\hat{b}_s) = \frac{p(y_s|\hat{x}_s,\hat{a}_s,\hat{b}_s)}{\sum_{k\in\chi} p(y_s|k,\hat{a}_s,\hat{b}_s)}. \tag{8}$$

The above expression for the posterior probability is based on the following two assumptions:

- All codewords are equally likely to occur in the coded image, i.e. $p(X_s = k) = \frac{1}{M}\ \forall k \in \chi$, and
- Channel distortion and background signal are independent of the transmitted codeword pattern, i.e. $p(a_s,b_s|x_s) = p(a_s,b_s)$.

The codeword estimate $\hat{x}_s$ is retained only if the posterior probability is above a pre-selected threshold, i.e.

$$p(\hat{x}_s|y_s,\hat{a}_s,\hat{b}_s) > T.$$

Otherwise, the estimated codeword label $\hat{x}_s$ is marked as invalid.

Limitations of ML Estimation

A major drawback of using the data model of (3) for codeword labeling is that it disregards spatial interaction between class labels of neighboring pixels. Thus, a small perturbation of features y_s, either due to noise or background, easily perturbs the codeword labeling, leading to inconsistent disparity estimates in a local neighborhood.

This is demonstrated in Figure 2. Figure 2 (a) shows an example NIR structured light image. Figure 2 (b) shows the disparity map estimated using 16×16 ($L = 16$) features, y_s, and codeword dictionary elements, $f(k)$, after outlier rejection. With 16×16 features, the estimated disparity map turns out to be noisy, with neighboring pixels in same-depth regions being assigned inconsistent disparity estimates. The outlier rejection algorithm removes the noisy disparity estimates and labels them as invalid, indicated by dark pixels in the figure.

Using overlapping features with larger spatial support is one way to allow neighboring pixels to share more information with each other and, hence, determine more consistent disparity estimates in a local neighborhood. The disparity map estimated using features of larger spatial support (28×28), after outlier rejection, is shown in Figure 2 (c). The disparity map has few regions with invalid depth, however, the disadvantage of using larger features is decreased spatial resolution of the estimated disparity map.

Multiscale Matched Filtering Based on Sequential MAP Estimation

In the previous section, we saw that the spatial resolution of disparity maps can be traded for reducing inconsistencies in the local disparity estimates by using larger codewords with more energy. Bayesian estimation provides an elegant framework for providing the best compromise between these two opposing objectives. Selecting a prior model, $p(x) \triangleq P(X = x)$, for describing the spatial interaction between codeword labels of neighboring pixels, Bayesian estimators attempt to minimize the average cost for an erroneous labeling.

In the following subsections, we shall use a multiscale random field (MSRF) [6] to model the codeword labels and use the sequential MAP [6] estimation framework, developed originally for image segmentation, for developing a novel multiscale matched filter for structured light decoding.

Multiscale Signal Model

Suppose $y_s^{(n)}$ denotes pixel values in an $L^{(n)} \times L^{(n)}$ local window at scale n and pixel s, arranged as an $L^{(n)2}$ dimensional column vector. As before, we shall assume that s denotes the top-left corner of the local window.

Let $n = 0$ denote the finest scale and $n = N$ denote the coarsest scale. The feature vector and codeword sizes for the $(N+1)$ scales are ordered as:

$$L^{(0)} < L^{(1)} < \cdots < L^{(N)}.$$

Let $x_s^{(n)}$ denote the codeword label at pixel s and scale n. The set of possible values for the codeword labels at scale n is denoted by $\chi^{(n)} \triangleq \left\{0,1,\ldots,M^{(n)}-1\right\}$. Notice that at each scale n, the total number of unique $L^{(n)} \times L^{(n)}$ codeword dictionary elements is different: $M^{(n)} \neq M^{(j)}$ for $n \neq j$. Let $f^{(n)}(x) : \chi^{(n)} \rightarrow \mathbf{R}^{L^{(n)2}}$ denote the de-meaned and standard-deviation normalized codeword pattern corresponding to label x at scale n.

Finally, assuming that a_s, b_s, and w_s, respectively, represent the codeword attenuation, signal component due to ambient light, and $L^{(n)2}$ dimensional noise vector at pixel s. The observation vector $y_s^{(n)}$ can be represented by the model:

$$y_s^{(n)} = a_s f^{(n)}(x_s^{(n)}) + b_s \mathbf{1}_{L^{(n)2}} + w_s^{(n)}. \tag{9}$$

Knowledge of the codeword label $x_s^{(n)}$ at scale n and pixel s, enables estimation of the disparity value through the following deterministic transformation $h^{(n)}(.,.)$:

$$d_s = h^{(n)}\left(s, x_s^{(n)}\right).$$

We shall assume $h^{(n)}(.,.)$ are known beforehand for all scales.

Statistical Modeling for Bayesian Estimation

We shall model the codeword labels using a multiscale random field $X^{(n)}$ that has a Markov structure in scale. Specifically, the distribution of $X^{(n)}$ given all other coarser fields is assumed to be depend only on $X^{(n+1)}$.

We shall denote by $Y^{(n)}$ the random field of observed features at scale n. The behavior of the observed vectors $Y^{(n)}$ is assumed to depend exclusively on the unobserved codeword labels $X^{(n)}$ through the relationship in (9).

The model parameters $a = \{a_s\}_{s\in S}$ and $b = \{b_s\}_{s\in S}$ are not of direct interest to us, but must be determined to solve the inversion. We shall make no assumptions about the values of these parameters and treat them as unknown deterministic, rather than random, quantities.

Forward Model

Let Y and X, respectively, denote the collection of observation vectors $Y_s^{(n)}$ and codeword labels $X_s^{(n)}$ $\forall s \in S$ and $\forall n \in \{0,1,\ldots,N\}$. Assuming $Y_s^{(n)}$ are conditionally independent given the class labels $X_s^{(n)}$ and the model parameters, the conditional density of the observed feature vectors is given by

$$p(y|x,a,b) = \prod_{n=0}^{N} p(y^{(n)}|x^{(n)},a,b) \tag{10}$$

$$= \prod_{s\in S}\prod_{n=0}^{N} p(y_s^{(n)}|x_s^{(n)},a_s,b_s), \tag{11}$$

where $p(y_s^{(n)}|x_s^{(n)},a_s,b_s)$ is a multivariate Gaussian distribution defined by an expression similar to that in (2).

Prior Model

As mentioned above, the random field $X^{(n)}$ is assumed dependent only on the previous coarser scale field $X^{(n+1)}$, giving $X^{(n)}$ a Markov chain structure in the variable n. This structure captures complex spatial dependencies in the codeword labels, while allowing for efficient computational processing.

Using this Markovian structure, the probability mass function of the random field X can be written as

$$\begin{aligned} p(x) &\triangleq P\left(X^{(n)} = x^{(n)}\ n \geq 0\right) \\ &= \prod_{n=0}^{N} P\left(X^{(n)} = x^{(n)} | X^{(l)} = x^{(l)}\ l > n\right) \\ &= \prod_{n=0}^{N} P\left(X^{(n)} = x^{(n)} | X^{(n+1)} = x^{(n+1)}\right) \\ &= \prod_{n=0}^{N} p\left(x^{(n)}|x^{(n+1)}\right), \end{aligned} \tag{12}$$

where we assume $p\left(x^{(N)}|x^{(N+1)}\right)$ equals $p\left(x^{(N)}\right)$.

We shall further assume that the class label of a pixel s at scale n, $X_s^{(n)}$, depends only on the class labels of pixels ∂s at the next coarser scale $(n+1)$, $X_{\partial s}^{(n+1)}$, where ∂s denotes the set of neighbors of pixel s. With this assumption, the prior distribution of X can be written as:

$$p(x) = \prod_{s\in S}\prod_{n=0}^{N} p\left(x_s^{(n)}|x_{\partial s}^{(n+1)}\right).$$

Joint Distribution

Using (10) and (12), the joint distribution function of Y and X can be written as

$$p(y,x|a,b) = \prod_{s\in S}\prod_{n=0}^{N} p\left(y_s^{(n)}|x_s^{(n)},a_s,b_s\right) p\left(x_s^{(n)}|x_{\partial s}^{(n+1)}\right). \tag{13}$$

Sequential MAP Estimate

For disparity estimation, we need to determine the codeword labels X given the observation features Y. Bayesian MAP estimation achieves this by finding x that maximizes the posterior distribution $p(x|y,a,b)$. However, there are two limitations of MAP estimation:

1. It assigns equal cost to erroneous labeling in both fine and coarse scales, which is not suitable for multiscale class labeling. Course scale errors are usually considered more serious as they result in mislabeling of a larger group of pixels [6].
2. MAP estimation requires computationally expensive iterative methods that are not suitable for real-time implementation [6].

The sequential MAP cost functional [6], proposed by Bouman and Shapiro, overcomes the limitations of MAP estimation by progressively assigning increasing cost to mislabeling at coarser scales. Specifically, the SMAP cost of estimating a true labeling X with the approximate labeling x is given by

$$C_{SMAP}(X,x) = \sum_{n=0}^{N} \alpha^n C_n(X,x), \tag{14}$$

where we assume $\alpha > 1$ and

$$C_n(X,x) = 1 - \prod_{i=n}^{N} \delta\left(X^{(i)} - x^{(i)}\right). \tag{15}$$

Thus, if K is the coarsest scale where the first mislabeling occurs then $C_{SMAP}(X,x) = \sum_{n=0}^{K} \alpha_n$.

The SMAP estimate of X is computed by performing the following minimization:

$$\hat{x} = \underset{x}{\operatorname{argmin}}\, E\left[C_{SMAP}(X,x)\,|y,a,b\right]. \tag{16}$$

Using a proof similar to the one given in previous work [6], we can show that for our data model (13), performing the minimization in (16) results in the following coarse-to-fine recursive equations for computing the SMAP estimate at each pixel:

$$\hat{x}_s^{(n)} = \underset{k^{(n)}\in\chi^{(n)}}{\operatorname{argmax}} \left\{\log p\left(y_s^{(n)}|k^{(n)},a_s,b_s\right) + \log p\left(k^{(n)}|\hat{x}_{\partial s}^{(n+1)}\right)\right\}. \tag{17}$$

Computation of SMAP Estimate

In this section, we will first present the specific form of the data term in (17) for the structured light decoding problem. We will then closely examine the structure of the codewords in our multiscale framework and develop a mathematical expression for the transitional probabilities in (17). Finally, we will present our multiscale matched filtering algorithm for recursively optimizing (17) to determine the SMAP estimates of the codeword labels.

Data Term for the SMAP Estimate

The data term in (17) is the log of the conditional probability of the observed feature $Y_s^{(n)}$ given the class label $X_s^{(n)} = k^{(n)}$ and the model parameters. Using the signal model in (9), the data term is given by

$$\begin{aligned} -\log p(y_s^{(n)}|k^{(n)},a_s,b_s) &\triangleq l_s^{(n)}(y_s^{(n)}|k^{(n)},a_s,b_s) && (18)\\ &= \frac{1}{2\sigma^2}\|y_s^{(n)} - a_s f^{(n)}(k^{(n)}) - b_s \mathbf{1}_{L^{(n)2}}\|_2^2 \\ &+ \frac{L^{(n)2}}{2}\log(2\pi\,\sigma^2), && (19)\end{aligned}$$

where $k^{(n)} \in \chi^{(n)}$ denotes a codeword label at scale n.

Prior Term for the SMAP Estimate

The prior model term in (17) requires knowledge of the conditional probability that $X_s^{(n)} = k^{(n)}$, where $k^{(n)} \in \chi^{(n)}$, given estimates of the codeword labels $\hat{X}_{\partial s}^{(n+1)} \in \chi^{(n+1)}$ in the neighborhood of s at the next coarser scale.

In order to determine these conditional probabilities, we refer to Figure 3, that shows an example $L^{(n+1)} \times L^{(n+1)}$ codeword with label $\hat{X}_s^{(n+1)} = m^{(n+1)}$, estimated at scale $(n+1)$. The schematic shows each codeword *dot* ("1") or *hole* ("0") occupying a single pixel on the sensor; however, in general, each of these dots or holes has a spatial support of $P \times P$ pixels. We see that the coarse scale codeword yields four $L^{(n)} \times L^{(n)}$ child codewords at pixel locations s, $s + P\cdot(0,1)$, $s + P\cdot(1,0)$, and $s + P\cdot(1,1)$, with $L^{(n)} = L^{(n+1)} - P$, at the next finer scale n. The labels of these child codewords shall be denoted by $c_i\left(m^{(n+1)}\right)$, where $i \in \{0,1,2,3\}$. The spatial locations of the four child codewords relative to pixel s are also shown in Figure 3.

Given the estimates of the coarse scale codeword labels $\hat{X}_{\partial s}^{(n+1)}$, it is possible to improve the prediction of $X_s^{(n)}$. Specifically, we shall assume a neighborhood ∂s that comprises the following four pixels: $s_0 = s$, $s_1 = s + P\cdot(0,-1)$, $s_2 = s + P\cdot(-1,0)$, and $s_3 = s + P\cdot(-1,-1)$. Also, let us assume the coarse scale label estimates at the four pixel locations are:

$$\hat{X}_{s_i}^{(n+1)} = m_i^{(n+1)} \text{ for } i \in \{0,1,2,3\},$$

where $m_i^{(n+1)} \in \chi^{(n+1)}$. Vectorially, the above relationship can be written as

$$\begin{aligned} \hat{X}_{\partial s}^{(n+1)} &\triangleq \left[\hat{X}_{s_0}^{(n+1)},\cdots,\hat{X}_{s_3}^{(n+1)}\right]^T \\ &= [m_0^{(n+1)},\cdots,m_3^{(n+1)}]^T \\ &= m^{(n+1)}. \end{aligned}$$

We observe from Figure 4 that each of the four coarse scale codewords, $m_i^{(n+1)}$, has an $L^{(n)} \times L^{(n)}$ child with label $c_i\left(m_i^{(n+1)}\right)$ that is more likely to be observed at the next finer scale n than the rest of the codewords in the set $\chi^{(n)}$. Thus, if θ_0 and θ_1 denote two positive numbers with $\theta_1 > \theta_0$, we compute the conditional probability that $X_s^{(n)} = k^{(n)}$ given the coarse scale label estimates $\hat{X}_{\partial s}^{(n+1)} = m^{(n+1)}$ as follows:

$$\begin{aligned} p\left(k^{(n)}|m^{(n+1)}\right) &\triangleq P\left(X_s^{(n)} = k^{(n)}|\hat{X}_{\partial s}^{(n+1)} = m^{(n+1)}\right) \\ &= \gamma\theta_1\,\omega\left(k^{(n)}|m^{(n+1)}\right) \\ &+ \gamma\theta_0\left(1 - \omega\left(k^{(n)}|m^{(n+1)}\right)\right), && (20)\end{aligned}$$

where

$$\omega\left(k^{(n)}|m^{(n+1)}\right) \triangleq 1 - \prod_{i=0}^{3}\delta\left(k^{(n)} \neq c_i\left(m_i^{(n+1)}\right)\right). \quad (21)$$

Thus, $p\left(k^{(n)}|m^{(n+1)}\right) = \gamma\theta_1$ if $k^{(n)} \in \left\{c_i\left(m_i^{(n+1)}\right)\right\}$ and $\gamma\theta_0$ otherwise. Expressed in words, the codewords at scale n have an increased probability of detection if they are the children of codewords estimated at the courser scale $(n+1)$ in the neighborhood of a given pixel.

The variable γ is a normalization factor that ensures $\sum_{k\in\chi^{(n)}} p(k|m^{(n+1)}) = 1$ and is given by:

$$\gamma \triangleq \frac{1}{\theta_0 M^{(n)} + (\theta_1 - \theta_0)\sum_{k\in\chi^{(n)}}\omega\left(k, m^{(n+1)}\right)}. \quad (22)$$

Multiscale Matched Filtering Algorithm

Having determined the specific forms of the data (18) and prior model (20) terms in (17), the SMAP estimate of the codeword labels can be computed efficiently using a sequence of coarse-to-fine scale optimization steps. The pseudo code for the multiscale matched filtering algorithm that performs this optimization is shown in Algorithm 1.

We begin at the coarsest scale, $n = N$. For this scale, we assume that the prior term (20) is simply a uniform distribution, i.e.

$$p(k^{(N)}|m^{(N+1)}) = p(k^{(N)}) = \frac{1}{M^{(N)}}\ \forall k^{(N)} \in \chi^{(N)}.$$

With this assumption, the codeword labels, $X_s^{(N)}$, and the model parameters, $a_s^{(N)}$ and $b_s^{(N)}$, can be estimated using a set of equations similar to the one used for ML estimation in Section . The ML estimation of parameters at the coarsest scale is shown in lines (1)-(5) of the pseudo code.

After the estimates for the coarsest level have been determined, the SMAP estimates for the rest of the $(N-1)$ scales are computed recursively using lines (6) to (17) of the pseudo code.

Our final estimates for the codeword labels and the model parameters are the estimates computed at the finest scale, i.e., $\hat{x}_s^{(0)}$, $\hat{a}_s^{(0)}$, and $\hat{b}_s^{(0)}$. Finally, as in Section , to establish our confidence in the codeword labeling, $\hat{x}_s^{(0)}$, we compute the posterior probability that $X_s = \hat{x}_s^{(0)}$ given the observation vector $y_s^{(0)}$ and the model parameter estimates at the finest scale:

$$p(\hat{x}_s^{(0)}|y_s^{(0)},\hat{a}_s^{(0)},\hat{b}_s^{(0)}) = \frac{p(y_s^{(0)}|\hat{x}_s^{(0)},\hat{a}_s^{(0)},\hat{b}_s^{(0)})}{\sum_{k\in\chi^{(0)}} p(y_s^{(0)}|k,\hat{a}_s^{(0)},\hat{b}_s^{(0)})}. \quad (23)$$

The codeword estimate $\hat{x}_s^{(0)}$ is retained only if the posterior probability is above a pre-selected threshold, i.e.

$$p(\hat{x}_s^{(0)}|y_s^{(0)},\hat{a}_s^{(0)},\hat{b}_s^{(0)}) > T.$$

Algorithm 1 Multiscale Matched Filter Based on SMAP Estimation

$n \leftarrow N$
$c \leftarrow \frac{1}{L^{(n)2}}$
$\hat{x}_s^{(n)} \leftarrow \underset{k \in \chi^{(n)}}{\text{argmax}}\ y_s^{(n)T} f^{(n)}(k)$
$\hat{a}_s^{(n)} \leftarrow c y_s^{(n)T} f^{(n)}(\hat{x}_s^{(n)})$
$\hat{b}_s^{(n)} \leftarrow c y_s^{(n)T} \mathbf{1}_{L^{(n)2}}$
for $n = (N-1)$ **to** 0 **do**
    $c \leftarrow \frac{1}{L^{(n)2}}$
    for $s \in S$ **do**
        for $k \in \chi^{(n)}$ **do**
            $a(k) = c y_s^{(n)T} f^{(n)}(k)$
        end for
        $m \leftarrow \hat{x}_{\partial s}^{(n+1)}$
        $\hat{x}_s^{(n)} = \underset{k \in \chi^{(n)}}{\text{argmax}}\ a(k)^2 + 2c\sigma^2 \log p(k|m)$
        $\hat{a}_s^{(n)} \leftarrow a\left(\hat{x}_s^{(n)}\right)$
        $\hat{b}_s^{(n)} \leftarrow c y_s^{(n)T} \mathbf{1}_{L^{(n)2}}$
    end for
end for

Otherwise, the estimated codeword label $\hat{x}_s^{(0)}$ is marked as invalid.

EXPERIMENTAL RESULTS

The multiscale matched filtering algorithm used for generating results for this paper has a total of four scales $N = 3$. The finest scale is denoted by $n = 0$ and the coarsest scale is denoted by $n = N$.

Each codeword *dot* ("1") or *hole* ("0") in our setup has a spatial support of 4×4 pixels on the NIR sensor; thus, $P = 4$.

The feature vector sizes for fine to coarse scales are selected as: $L^{(0)} = 16$, $L^{(1)} = 20$, $L^{(2)} = 24$, and $L^{(3)} = 28$. Since each dot or hole in our codewords occupies a region of $P \times P$ pixels, we notice that the underlying 2D binary sequence corresponding to an $L^{(n)} \times L^{(n)}$ codeword pattern is $\frac{L^{(n)}}{P} \times \frac{L^{(n)}}{P}$.

The number of unique $L^{(n)} \times L^{(n)}$ codeword dictionary elements, $f^{(n)}(.)$, at the various scales, n, are: $M^{(0)} = 405$, $M^{(1)} = 420$, $M^{(2)} = 424$, and $M^{(3)} = 428$.

The parameters for the data and prior model terms for the SMAP estimation are selected as: $\sigma = 2.5$, $\theta_0 = 1$, and $\theta_1 = 5$.

Finally, the parameter T for thresholding the posterior probability for outlier rejection is selected as $T = 0.1$.

The performance gains with our proposed multiscale matched filtering algorithm are shown in Figures 5 and 6. Figures 5 (a) and 6 (a) show examples of two structured light NIR images. The corresponding disparity maps for the two example NIR images estimated using single scale matched filtering algorithm of Section with $L = 16$ are shown in Figures 5 (b) and 6 (b). The estimated disparity maps estimated using the 16×16 feature elements are very noisy. Removal of the noisy regions using the outlier rejection scheme of Section results in invalidation of large areas of the estimated disparity. Figures 5 (c) and 6 (c) show the disparity maps estimated using the single scale matched filter with features of larger, 28×28 ($L = 28$), spatial support. With larger feature vectors, the depth of the low-frequency background is correctly estimated, but the high-spatial-frequency regions of the foreground objects cannot be resolved. Finally, Figures 5 (d) and 6 (d) show the disparity maps estimated using the multiscale matched filter algorithm based on SMAP estimation developed in Section . As evident from the results, the multiscale algorithm provides a good compromise between noise suppression and preservation of spatial resolution for disparity estimation from structured light images.

CONCLUSIONS

We developed a multiscale matched filtering algorithm based on Bayesian sequential MAP estimation framework, originally proposed for segmenting multispectral images. The proposed algorithm can easily be parallelized for efficient implementation in software. The experimental results convincingly demonstrate that the proposed multiscale algorithm outperforms single-scale matched filtering strategies for structured light depth sensing.

References

[1] Scharstein, Daniel, and Richard Szeliski. "High-accuracy stereo depth maps using structured light." Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on. Vol. 1. IEEE, 2003.

[2] Sarbolandi, Hamed, Damien Lefloch, and Andreas Kolb. "Kinect range sensing: Structured-light versus time-of-flight kinect." Computer vision and image understanding 139 (2015): 1-20.

[3] Butler, D. Alex, et al. "Shake'n'sense: reducing interference for overlapping structured light depth cameras." Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM, 2012.

[4] Park, Byeonghoon, et al. "Outdoor Operation of Structured Light in Mobile Phone." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017.

[5] Healey, Glenn E., and Raghava Kondepudy. "Radiometric CCD camera calibration and noise estimation." IEEE Transactions on Pattern Analysis and Machine Intelligence 16.3 (1994): 267-276.

[6] Bouman, Charles A., and Michael Shapiro. "A multiscale random field model for Bayesian image segmentation." IEEE Transactions on image processing 3.2 (1994): 162-177.

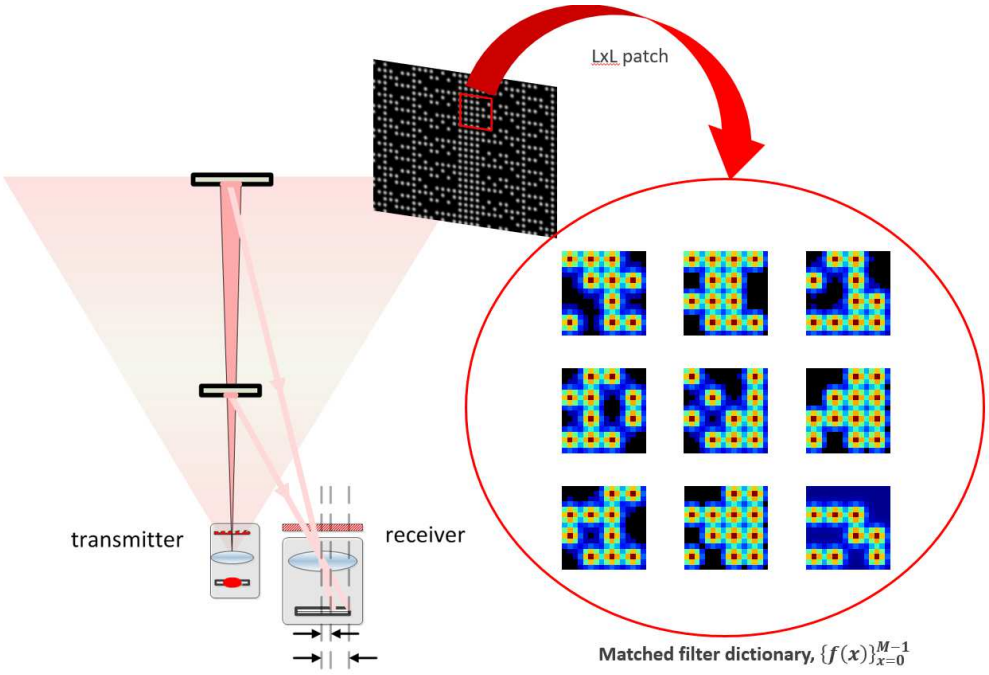
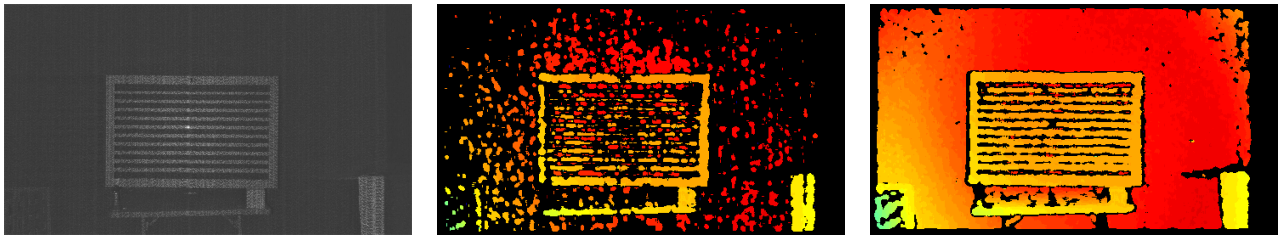


Figure 1. *Structured light setup for depth sensing. The NIR transmitter projects a pattern of 2D binary codewords on the scene. The codeword patterns are reflected off the scene and received by an NIR camera. The received patterns after channel distortion are no longer binary. The decoding algorithm compares a local $L \times L$ patch in the received image to recorded templates in a dictionary of codeword elements and returns the label $x \in \{0, \ldots, M-1\}$ of the closest matching template.*



(a) (b) (c)

Figure 2. *(a) Example structured light image captured on the NIR sensor. (b) Disparity map estimated using 16×16 ($L = 16$) features and codeword dictionary elements. The dark areas in the disparity map are high-noise regions, marked as invalid by the outlier rejection algorithm. (c) Disparity map estimated using 28×28 ($L = 28$) features and codeword dictionary elements. Using features of larger spatial support reduces noise (i.e., few areas in the background are marked as invalid), however, the disadvantage is decreased spatial resolution of high-frequency regions in the scene.*

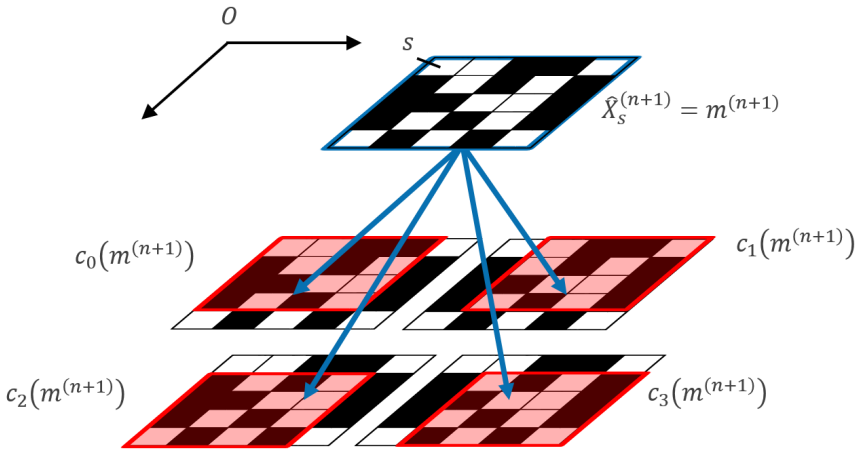


Figure 3. *An example coarse-scale codeword with label $\hat{X}_s^{(n+1)} = m^{(n+1)}$ at scale $(n+1)$ and its four children $c_i\left(m^{(n+1)}\right)$, where $i \in \{0,1,2,3\}$, at the next finer scale n. The parent codeword has a spatial support of $L^{(n+1)} \times L^{(n+1)}$ pixels and its top-left corner is located at pixel s. The children codewords have spatial supports of $L^{(n)} \times L^{(n)}$ and are located at s, $s+P\cdot(0,1)$, $s+P\cdot(1,0)$, and $s+P\cdot(1,1)$. For the schematic shown, $P=1$, and $L^{(n)} = L^{(n+1)} - P$, where P denotes the spatial support of each codeword* dot *("1") or* hole *("0").*

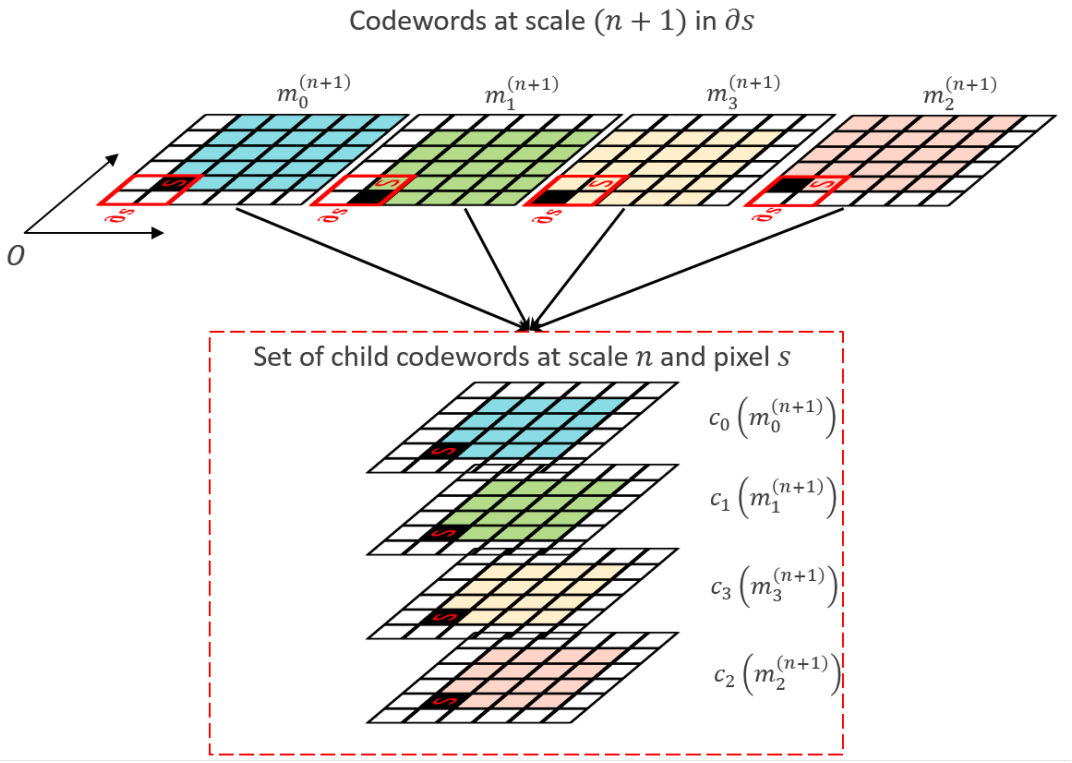
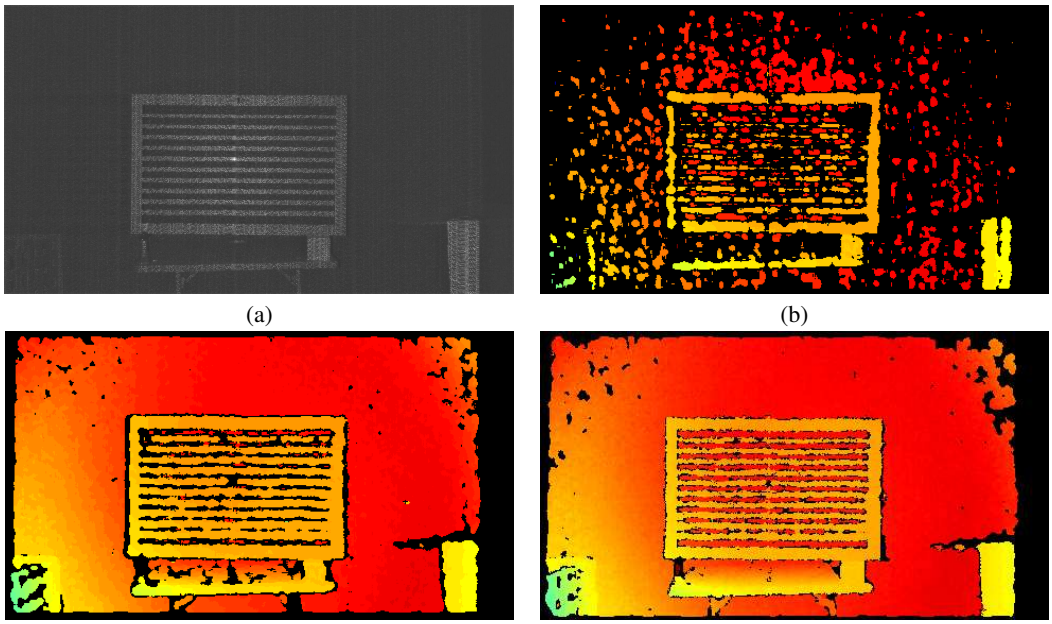
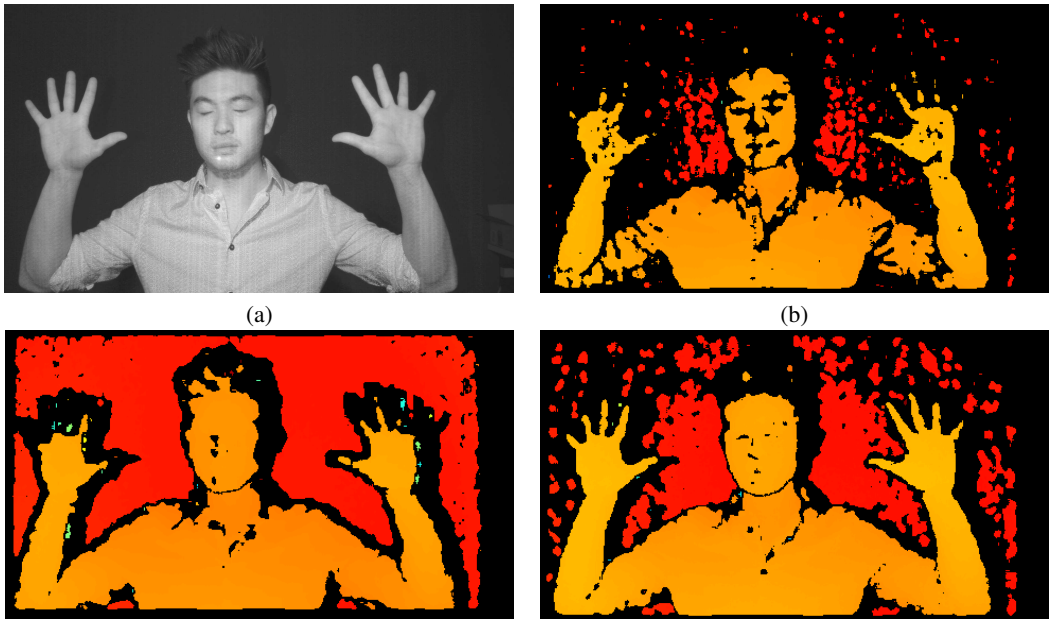


Figure 4. *The top row shows example codewords, $m_i^{(n+1)}$, at scale $(n+1)$ in the neighborhood of pixel s. The neighborhood ∂s comprises four pixels: $s_0 = s$, $s_1 = s+P\cdot(0,-1)$, $s_2 = s+P\cdot(-1,0)$, and $s_3 = s+P\cdot(-1,-1)$, where $P=1$ for the example schematic. Each $L^{(n+1)} \times L^{(n+1)}$ codeword at the coarse scale has a smaller $L^{(n)} \times L^{(n)}$ subset or child, denoted by $c_i\left(m_i^{(n+1)}\right)$, that shapes our prior knowledge about codeword labels at the next finer scale n at pixel location s.*



(a) (b) (c) (d)

Figure 5. *(a) Example structured light image captured on the NIR sensor. (b) Disparity map estimated using single scale matched filter with* $L = 16$*. (c) Disparity map estimated using single scale matched filter with* $L = 28$*. (d) Depth map estimated using proposed multiscale matched filter.*



(a) (b) (c) (d)

Figure 6. *(a) Example structured light image captured on the NIR sensor. (b) Disparity map estimated using single scale matched filter with* $L = 16$*. (c) Disparity map estimated using single scale matched filter with* $L = 28$*. (d) Depth map estimated using proposed multiscale matched filter.*