

3D integral imaging depth estimation of partially occluded objects using mutual information and Bayesian optimization

PRANAV WANI AND BAHRAM JAVIDI* 

Electrical and Computer Engineering Department, University of Connecticut, 371 Fairfield Road, Storrs, Connecticut 06269, USA

*Bahram.javidi@uconn.edu

Abstract: Integral imaging (InIm) is useful for passive ranging and 3D visualization of partially-occluded objects. We consider 3D object localization within a scene and in occlusions. 2D localization can be achieved using machine learning and non-machine learning-based techniques. These techniques aim to provide a 2D bounding box around each one of the objects of interest. A recent study uses InIm for the 3D reconstruction of the scene with occlusions and utilizes mutual information (MI) between the bounding box in this 3D reconstructed scene and the corresponding bounding box in the central elemental image to achieve passive depth estimation of partially occluded objects. Here, we improve upon this InIm method by using Bayesian optimization to minimize the number of required 3D scene reconstructions. We evaluate the performance of the proposed approach by analyzing different kernel functions, acquisition functions, and parameter estimation algorithms for Bayesian optimization-based inference for simultaneous depth estimation of objects and occlusion. In our optical experiments, mutual-information-based depth estimation with Bayesian optimization achieves depth estimation with a handful of 3D reconstructions. To the best of our knowledge, this is the first report to use Bayesian optimization for mutual information-based InIm depth estimation.

© 2023 Optica Publishing Group under the terms of the [Optica Open Access Publishing Agreement](#)

1. Introduction

Object localization is the task of locating an instance of a particular object category in an image. It typically aims to specify a tightly cropped bounding box centered on the object [1]. Two-dimensional (2D) object localization is a slightly simpler problem than object detection. A target object is known to exist in the query image, and the aim is to locate this object and give its 2D bounding box. The task of object localization is required in many-safety critical systems. Under this problem statement, traditional methods use object templates from different viewpoints to find the one which shares the most similarity (in terms of features) with the target image [2–4]. Recent methods use feedforward deep networks that evaluate a fixed number of object proposals in a target image [5–7]. There have also been some attempts to use neural networks for direct three-dimensional (3D) localization instead of 2D in monocular images [8,9]. In this paper, we use 3D integral imaging (InIm) for the depth estimation of the object present within the 2D bounding box obtained via 2D localization. Henceforth, we assume the availability of this bounding box, and as 2D localization is a simpler problem than detection, we also extend this assumption in partially occluded environments.

3D InIm is a prominent technique that works by capturing angular information about the object scene [10–14]. 3D InIm records multiple 2D elemental images of the scene from diverse perspectives [15–25]. This is achieved using either a camera array, a single imaging sensor with a lenslet array, or a single camera mounted on a moving platform. 3D scenes are reconstructed by integrating the 2D elemental images optically or computationally. Computational reconstruction is achieved by back-propagating the 2D elemental images through a virtual pinhole array. 3D

reconstruction can be obtained at any depth within the limits of the depth of field of captured 2D elemental images. Passive depth estimation is one of several applications of InIm that has been extensively studied in the literature [26–30]. InIm has also been used in several occlusion-aware depth estimation problems [31–35]. InIm has several advantages over other commonly used imaging modalities for imaging in adverse environmental conditions [36,37]. A recent study [38], inspired by [39], uses mutual information as a metric to evaluate the fidelity of the 3D InIm reconstruction with and without the presence of partial occlusions. This allows for passive depth estimation of both object and occlusion. This method differs from others in the sense that it performs computation on the entire 2D bounding box as opposed to patch-based analysis. The computation of mutual information curve can be broken down into three parts: InIm 3D reconstruction, mutual information computation, and mutual information curve generation. Assuming a patch of size $n \times m$, the combined time complexity of these three steps is $O(l(nm + b_1^2 b_2^2))$ [38]. Here, b_1 , and b_2 are the bin sizes for possible pixel values, and possible clique potentials (see Sec. 2.3) and l is the number of 3D reconstructions used to generate the average MI curve. In this paper, we advance this method by using Bayesian optimization for sample efficient inference of the objective function. The scope of this manuscript is limited to the application of Bayesian optimization for minimizing the number of required 3D reconstructions. We use laboratory experiments to demonstrate the efficacy of this approach. A rigorous study of its various applications or performance is not considered here. We postulate that our analysis can be extended to shape-to-focus methods available in the literature [40].

Bayesian optimization is a global optimization scheme for sample efficient optimization of expensive-to-evaluate objective functions [41–43]. Bayesian optimization iteratively builds a statistical model of the objective function according to all past observations and selects the next evaluation by maximizing some acquisition criteria. It has been successful in several domains including material design [44–46], tuning/calibration [47–49], simulations [50,51], machine learning [52–54], and reinforcement learning [55]. Several other methods exist for multi-modal optimization. Some of the commonly used are the gradient descent methods, the quasi-Newton methods [56,57], and the simplex methods [58]. These require an analytical form of the objective function and may get trapped in a local optimum. Evolutionary optimization can be used in domains with no available analytical representation. Some commonly used evolutionary optimization methods are genetic algorithms [59–61], clonal selection algorithms [62], and artificial immune networks [63,64]. However, these methods rely on heuristics and require excessive function evaluations. This prevents their reliable deployment in many applications.

In this paper, the objective functions were obtained by computing the mutual information of 3D InIm reconstructions. These objective functions have some common underlying structures which depend on InIm system parameters. This lends them as good candidates for Bayesian optimization. We assume these functions to be statistically stationary and their acquisition process to be corrupted by homoscedastic noise. This paper discusses parameter estimation for the Gaussian process with a small number of samples. We study different kernel functions, acquisition functions, and parameter estimation algorithms for Bayesian optimization-based inference. We evaluate their performance based on errors between the predicted and the true depth of objects as a function of the number of 3D reconstructions. We also compare a few methods for finding local optima instead of the global optimum. This allows us to estimate the depth of both the object and occlusion. For our experiments, we can achieve these objectives with reasonable accuracy in approximately ten 3D reconstructions, which are approximately half of that required for mutual information-based depth estimation using spline interpolation. We also discuss parallelization to further speed up computations.

2. Integral imaging-based depth estimation

2.1. Integral imaging (InIm)

InIm is a passive 3D imaging approach that integrates the diverse perspectives of the 2D elemental images to gain information about the light field. This can be accomplished by a single camera mounted on translating stage or a camera array [15–25]. InIm can also use a single imaging sensor and a lenslet array to capture the light field. 3D reconstruction can be achieved by backpropagating the rays through a virtual pinhole. The reconstruction can be anywhere within the depth of fields of the elemental images. InIm records elemental images using multiple parallaxes. This helps mitigate the effects of partial occlusion. 3D InIm is optimal in the maximum likelihood sense for read-noise dominant images [11–14]. This enables the 3D reconstructed scenes to have a better signal-to-noise ratio. An excellent overview of integral imaging research can be found in [65,66].

In our experiments, we use a synthetic aperture integral imaging setup which uses a camera array or a single camera mounted on a translation stage [67]. The pickup stage of the synthetic aperture InIm is shown in Fig. 1(a). Once the 2D elemental images are captured, the 3D reconstruction of the scene can be achieved computationally as shown in Fig. 1(b). Figure 1(c) and (d) show the InIm camera pickup and reconstruction process using one image sensor and a lenslet array.

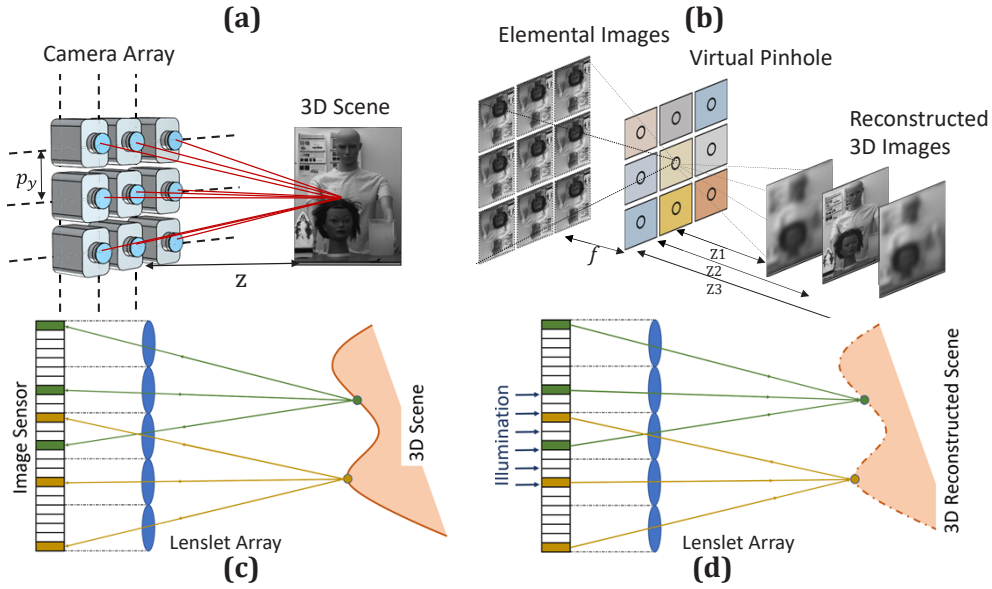


Fig. 1. (a) Synthetic aperture integral imaging (InIm) setup using camera array for image pickup process. (b) The reconstruction process of the integral imaging setup of (a). (c) Integral imaging setup using a lenslet array and a single imaging sensor. (d) The reconstruction process of integral imaging setup of (c).

3D reconstruction is achieved by backpropagating the captured 2D elemental images through a virtual pinhole. Reconstructed 3D scene intensity $I_z(x, y)$ is computed as [38]:

$$I_z = \frac{1}{O(x, y)} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \left[I_{mn} \left(x - \frac{m \times L_x \times p_x}{c_x \times \frac{z}{f}}, y - \frac{n \times L_y \times p_y}{c_y \times \frac{z}{f}} \right) + \varepsilon \right] \quad (1)$$

where (x, y) is the pixel indices, $O(x, y)$ is the number of overlapping pixels in (x, y) . I_{mn} is a 2D elemental image, with (m, n) representing its indices, and (M, N) representing the total

number of elemental images. (c_x, c_y) , (p_x, p_y) , and (L_x, L_y) represent the sensor size, the pitch size between cameras, and the resolution of the camera sensor, respectively. f is the focal length of the camera lens, and z is the reconstruction distance of the 3D object from the camera array. ε is the additive camera noise. Assuming that the incoming rays originate from an object voxel with approximately similar intensities, 3D reconstruction at the true depth will minimize the variation of these rays [68].

2.2. Experimental setup

Our experimental synthetic aperture InIm setup uses nine cameras in a 3×3 configuration as shown in Fig. 1. Pitch size of the cameras is 4 cm in both the x and y directions. The objects are placed at roughly 350 cm from the camera sensor plane and the occlusion is present at roughly 250 cm. We modify these distances to some degree for each scene. The field of view of cameras at the object plane is roughly $400 \text{ cm} \times 400 \text{ cm}$. Images are recorded using a visible sCMOS sensor (Hamamatsu C11440-42U). The focal length of each camera lens is 5 cm and the diameter is 4 cm giving an F-number of 1.25. The sensor size is 2048×2048 pixels with each pixel of 6.5×6.5 micrometers. Our experimental dataset consists of 20 different scenes with each scene containing three distinct objects. Out of these scenes, 12 scenes contain partial occlusion and the remaining do not. Thus, in total, 36 partially occluded objects and 24 occlusion-free objects are used to generate experimental results. This dataset will be used to study kernel functions, acquisition functions, and parameter estimation algorithms (Sec. 3). Similarly, this data set will also be used to compare the Bayesian optimization inference with the spline-based inference (Sec. 4). Sample 2D elemental image is shown in Fig. 2(a). Its corresponding 3D reconstructions at the occlusion plane and the object plane are shown in Fig. 2(b) and (c). A few more sample scenes are shown in Fig. 2(d)-(f). Objects of interest are circumscribed with bounding boxes shown in red boundaries. As mentioned previously, we assume the availability of these 2D bounding boxes using any of the contemporary object localization techniques.

2.3. Mutual information

We denote X and Y as two random variables corresponding to input and output variables. In our case, these variables correspond to two images. The mutual information (MI) between X and Y can now be defined in terms of the probability density function of the pixel values [69]:

$$MI(X; Y) = \sum_{g_1 \in I} \sum_{g_2 \in I} f_{xy}(g_1, g_2) \log \frac{f_{xy}(g_1, g_2)}{f_x(g_1)f_y(g_2)} \quad (2)$$

Here I is the set of pixel intensity values available in the image. However, pixel-to-pixel correspondence fails to capture the spatial information that exists in an image. Earlier studies on image registration using mutual information found that a lack of spatial information results in poor robustness to experimental factors like noise and misalignment errors [70]. Some authors tried to mitigate this by incorporating additional information such as gradients with mutual information [71] or by using higher-order mutual information [72]. However, these techniques lead to an exponential rise in computational costs and data requirements. In [70] the authors handle this with dimensionality reduction using either independent component analysis (ICA) or principal component analysis (PCA). A graphical approach for incorporating spatial information is discussed in [73] that considers the Ising model. It uses the Gibbs random field formulation which states that the conditional probabilities of a site's gray level corresponding to its neighborhood are proportional to the exponential sum of the potentials of its associated cliques. Thus, different neighborhood configurations that produce the same potential $U(x)$ can be

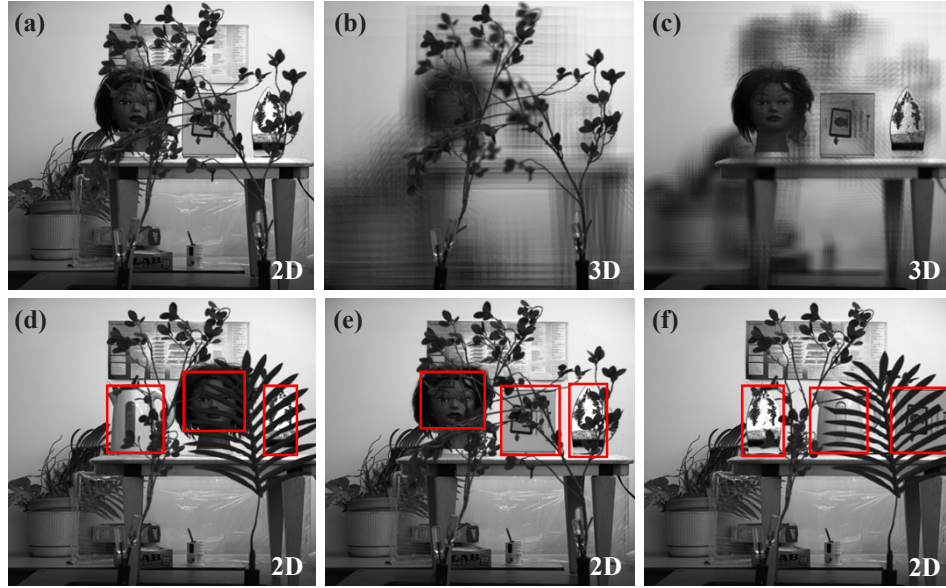


Fig. 2. InIm experiments. (a) Sample 2D central elemental image. (b) Corresponding 3D reconstruction at the plane of occlusion. (c) Corresponding 3D reconstruction at the plane of the objects. (d)-(f) Central 2D elemental images corresponding to sample scenes used in experiments. Objects of interest are circumscribed with bounding boxes shown in red boundaries.

grouped as a single state α . Mutual information between two images is then given as [73]:

$$MI(X; Y) = \sum_{g_x \in I} \sum_{g_y \in I} \sum_{\alpha_x} \sum_{\alpha_y} f_{xy}(g_x, \alpha_x, g_y, \alpha_y) \log \frac{f_{xy}(g_x, \alpha_x, g_y, \alpha_y) f_x(\alpha_x) f_y(\alpha_y)}{f_{xy}(\alpha_x, \alpha_y) f_x(\alpha_x, g_x) f_y(\alpha_y, g_y)} \quad (3)$$

Here I is the set of pixel intensities. g_x and g_y are the intensity values of pixels. α_x and α_y are the unique states corresponding to different neighborhood configurations that produce the same potential. This approach was adopted by [38] for 3-bit images with one adjacency neighborhood used for spatial information. This gives $I = \{0, 1, 2, 3, 4, 5, 6, 7\}$ and the number of α equals nine. Thus the total combination of the pairs (α, g) is 72. This formulation of mutual information has been used henceforth.

Two types of normalized mutual information curves are discussed in [38]. First, normalized mutual information is computed between the bounding box of 3D reconstructed image and the corresponding bounding box of 2D central elemental image. Second, normalized mutual information is computed between bounding boxes of two adjacent axially 3D reconstructed images. Here, we follow the first method to obtain mutual information. Figure 3 presents samples of mutual information curves obtained by our experimental setup, with and without occlusion.

The peaks in Fig. 3 indicate the presence of either an object or occlusion in the scene. We use Bayesian optimization to find these peaks while keeping the number of required 3D reconstructions as low as possible.

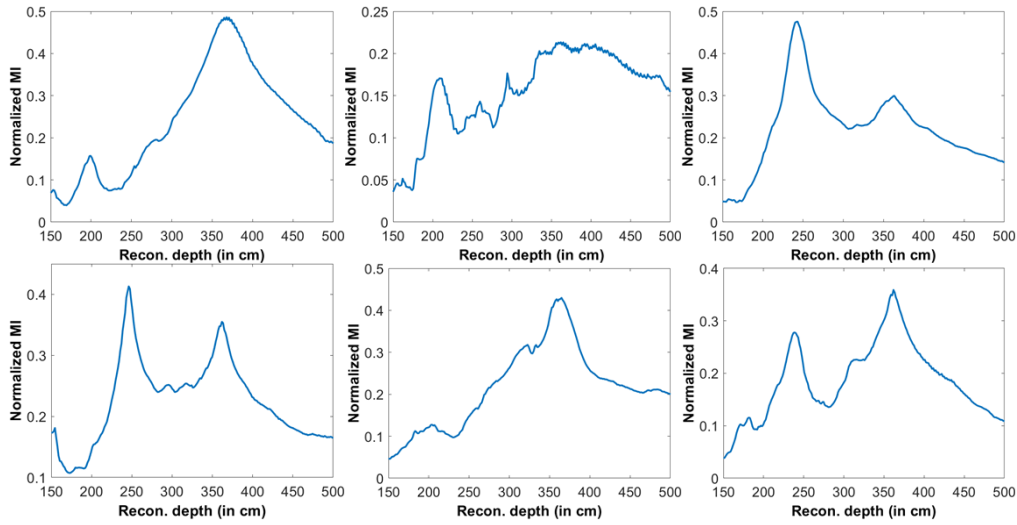


Fig. 3. Passive range estimation using InIm using normalized mutual information (MI). Sample MI curves vs object reconstruction depths for the scenes in Fig. 2. The object depth plots are obtained by computing mutual information between bounding boxes in 3D reconstructed scenes and their corresponding boxes in 2D central elemental images. For details see Fig. 2 for sample scenes and bounding boxes of objects of interest. Peaks indicate presence of either an object or occlusion in scenes. Their relative height is determined by degree or severity of occlusion.

3. Bayesian optimization

3.1. Background

Bayesian optimization emerged from the works of Kushner [74], Zilinskas [75,76], and Mockus [77,78]. It gained popularity with the development of the Efficient Global Optimization algorithm [79]. This work was further advanced to study multi-fidelity optimization [80,81], multi-objective optimization [82–84], and convergence rates [85–88]. Bayesian optimization performs sample-efficient optimization of expensive non-convex objective functions [43,89]. It is particularly useful when, as in our case, the objective functions have no closed-form representation and only a noisy point-based evaluation is possible. Each noisy point-based sample evaluation (or function evaluation) involves the 3D computational reconstruction of the scene contained within a bounding box, and the computation of mutual information between the bounding box in this 3D reconstructed scene and the corresponding bounding box in the 2D central view image.

The optimization problem is to find $z^* = \arg \max_{z \in Z} mi(z)$, where $Z \subset \mathbb{R}^d$ is a compact set and $mi(z)$ is the objective mutual information curve which is a function of z . It typically requires that the domain of z i.e. $Z \in \mathbb{R}^d$ has dimensions $d < 20$ and the objective function mi is continuous. This allows for the use of Gaussian process regression [90], which is the most commonly used surrogate model for Bayesian optimization. For our problem, we do not have access to derivatives of the objective function. However, if available, they could be incorporated into Bayesian optimization [91]. Bayesian optimization performs a sequential search, and at every iteration k selects a new location z_{k+1} to evaluate mi and observe its value. Gaussian process regression, the most commonly used surrogate model for Bayesian optimization, provides the posterior distribution according to previous observations. The sequential selection in Bayesian optimization is achieved through an acquisition function $a : Z \rightarrow \mathbb{R}$, defined over the posterior of the Gaussian process.

3.2. Gaussian process regression

Gaussian process [92,93] is a collection of random variables $\{MI_{z1}, MI_{z2}, \dots\}$ for which any finite subset has a joint multivariate Gaussian distribution. Thus, for any finite-length vector $\mathbf{z} = [z^1, z^2, \dots, z^n]^T$ its corresponding observation values $\mathbf{MI}_{\mathbf{z}} = [MI_{z1}, MI_{z2}, \dots, MI_{zn}]$ are jointly normally distributed:

$$\mathbf{MI}_{\mathbf{z}} \sim N\{\mu_0(\mathbf{z}), k(\mathbf{z}, \mathbf{z})\} \quad (4)$$

Here elements of $\mu_0(\mathbf{z})$ are given by a prior mean function $\mu_0(z_i)$, and k is the kernel (or covariance) function. For k to be a valid kernel $k(\mathbf{z}, \mathbf{z})$ needs to be a square, positive-semidefinite matrix for any \mathbf{z} [94]. We obtain the values $\mathbf{MI}_{\mathbf{z}}$ by noisily observing the function $mi(z)$ at indices \mathbf{z} , i.e. $MI_z = mi(z) + \varepsilon$, where $\varepsilon \sim N(0, \sigma_\varepsilon^2)$ is independent and identically distributed (i.i.d.). The Gaussian process regression infers the posterior of mi given the observations $\mathbf{MI}_{\mathbf{z}}$. The posterior distribution at some new point $z \in X$ is Gaussian with mean and variance [92,93]:

$$\mu(MI_z | \mathbf{MI}_{\mathbf{z}} = \mathbf{mi}) = \mu_0(z) + k(z, \mathbf{z})(k(\mathbf{z}, \mathbf{z}) + \sigma_n^2 I)^{-1}(\mathbf{mi} - \mu_0(\mathbf{z})) \quad (5a)$$

$$\sigma^2(MI_z | \mathbf{MI}_{\mathbf{z}} = \mathbf{mi}) = k(z, z) - k(z, \mathbf{z})(k(\mathbf{z}, \mathbf{z}) + \sigma_n^2 I)^{-1}k(\mathbf{z}, z) \quad (5b)$$

The kernel (or covariance) matrix $k(\mathbf{z}, \mathbf{z}) + \sigma_n^2 I$ depends only on the observed values and is Cholesky factored instead of inverted. The posterior mean is a linear combination of n kernel functions, each one centered at an observed data point. In absence of observation noise σ_n , a small number should be added to the diagonal of $k(\mathbf{z}, \mathbf{z})$ to prevent eigenvalues from being too close to zero.

3.3. Kernel functions

Gaussian process regression is a non-parametric generalization of the normal linear regression model $MI = \mathbf{Z}^T w + \varepsilon$, where $\varepsilon \sim N(0, \sigma^2)$ is i.i.d. random variable and the likelihood $p(MI | \mathbf{Z}, w, \sigma^2)$ is normal [91]. Non-parametric formalization is achieved by marginalizing the weights of the likelihood by placing a zero-mean Gaussian prior on the regression coefficients $w \sim N(0, V_0)$, where V_0 is some prior covariance matrix. A Gaussian prior, being a conjugate to the Gaussian likelihood, preserves the Gaussianity of the posterior yielding $p(MI | \mathbf{Z}, \sigma^2) \sim N(0, \mathbf{Z}V_0\mathbf{Z}^T + \sigma^2 I)$. The generalization of this model comes with the use of non-linear basis functions $\Phi : Z \rightarrow \mathbb{R}$ [41]. This allows us to model the objective function mi with a linear combination $mi(\mathbf{z}) = \Phi(\mathbf{z})^T w$. Under this basis transformation, a slightly different likelihood is obtained $p(MI | \mathbf{Z}, \sigma^2) \sim N(MI | 0, \Phi(\mathbf{Z})V_0\Phi(\mathbf{Z})^T + \sigma^2 I)$. The kernel (or covariance) function k and the kernel (or covariance) matrix K can now be defined using the positive semi-definite matrix $\Phi(\mathbf{Z})V_0\Phi(\mathbf{Z})^T \in \mathbb{R}^{n \times n}$ as:

$$K_{i,j} = k(z_i, z_j) = \Phi(z_i)V_0\Phi(z_j)^T = \langle \Phi(z_i), \Phi(z_j) \rangle_{V_0} \quad (6)$$

The kernel function k dictates the structure of the response functions that we can fit. For example, a periodic kernel is good for a periodic response function. Since we assume statistical stationarity, we only consider stationary shift-invariant kernel functions. We consider four commonly used kernel functions: squared exponential, Matern 3/2, Matern 5/2, and exponential.

$$k_{\text{squared_exponential}}(z_i, z_j | \theta) = \sigma_f^2 \exp\left(-\frac{1}{2} \frac{(z_i - z_j)^T (z_i - z_j)}{\sigma_l^2}\right) \quad (7a)$$

$$k_{\text{exponential}}(z_i, z_j | \theta) = \sigma_f^2 \exp\left(-\frac{r}{\sigma_l}\right) \quad (7b)$$

$$k_{\text{matern } 3/2}(z_i, z_j | \theta) = \sigma_f^2 \left(1 + \frac{\sqrt{3}r}{\sigma_l}\right) \exp\left(-\frac{\sqrt{3}r}{\sigma_l}\right) \quad (7c)$$

$$k_{\text{matern } 5/2}(z_i, z_j | \theta) = \sigma_f^2 \left(1 + \frac{\sqrt{5}r}{\sigma_l} + \frac{5r^2}{3\sigma_l^2} \right) \exp \left(-\frac{\sqrt{5}r}{\sigma_l} \right) \quad (7d)$$

Here, $r = \sqrt{(z_i - z_j)^T (z_i - z_j)}$, σ_f is the signal standard deviation, and σ_l is the characteristic length scale. σ_f and σ_l together form the hyperparameter vector of the kernel function, denoted by θ . Squared exponential kernels give rise to Gaussian process whose samples are infinitely differentiable, while those with Matern 3/2, Matern 5/2, and exponential kernels are ones, twice, and nowhere differentiable respectively. All the discussed kernel functions are differentiable with their hyperparameters θ . The marginal likelihood of the data can thus be optimized to obtain a type II maximum likelihood (type II ML) estimate of kernel parameters.

We evaluate the performance of the kernel functions for our data using two error metrics. First, we define the squared error (SE) as the total squared error between the predicted posterior mean $y_{\text{posterior}}$ and the ground truth y_{ground} evaluated over its entire domain. Second, we define max point error (MPE) as the Euclidian distance between the position of the global maxima predicted by the posterior mean and the true global maxima.

$$\text{Squared error (SE)} = \int |m_{\text{posterior}}(z) - m_{\text{ground}}(z)|^2 dz \propto \sum_i |m_{i,\text{posterior}} - m_{i,\text{ground}}|^2 \quad (8a)$$

$$\text{Max point error (MPE)} = (\arg \max_z m_{\text{posterior}}(z) - \arg \max_z m_{\text{ground}}(z))_+ \quad (8b)$$

We compute these errors for Gaussian process regression on the experimental normalized mutual information curves. We plot the values of average errors with varying numbers of support points, that is, number of observations made on the objective function. For each curve and each number of support points, we iterate 50 times, each time randomly selecting the positions of support points from the entire domain. Figure 4 shows average errors computed by averaging over all the curves and all the iterations. Here we assume complete knowledge of the hyperparameters associated with each curve and each kernel function (hyper parameter estimation is discussed in Sec. 3.6).

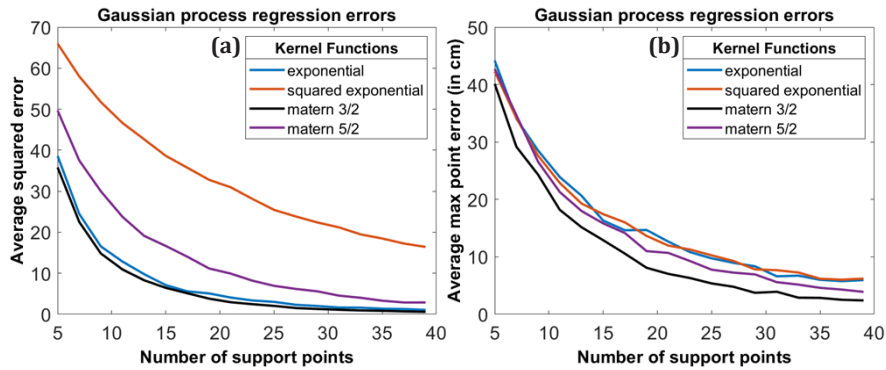


Fig. 4. Gaussian process regression errors of normalized mutual information [see Eq. (8)] for different kernel functions [see Eq. (7)]. (a) Average squared error (SE) with a varying number of support points. (b) Average max point error (MPE) (in cm) with a varying number of support points.

In our data, the Matern 3/2 kernel gives the lowest average squared error (SE) and max point error (MPE) for Gaussian process regression. As such, Matern 3/2 kernel function will be used henceforth unless mentioned otherwise.

3.4. Acquisition functions

Bayesian optimization uses acquisition functions to sequentially probe the objective function to get point estimates. These functions have to balance the tradeoff between the exploration of search space and the exploitation of current promising areas to be useful. In our case, the optima of these objective functions correspond to the 3D reconstruction depth that generates the highest mutual information and indicates the true depth of the objects.

The probability of improvement (PI / max-PI / MPI) criteria, as developed by Kushner [74], selects the next acquisition point according to its probability of improving upon the best current observation $z^{n+1} = \arg \max_z P[MI_z \geq \max_i(mi_{z_i}) + \xi(n)]$. Since the posterior distribution is Gaussian, this can be written analytically as:

$$PI(z, \xi) = P[MI_z \geq \mu_{\max} + \xi] = 1 - \Phi\left(\frac{\mu(z) - (\mu_{\max} + \xi)}{\sigma(z)}\right) \equiv \frac{\mu(z) - (\mu_{\max} + \xi)}{\sigma(z)} \quad (9)$$

Here Φ is the standard normal cumulative density function. \equiv indicates that both representations give the same maxima/minima as Φ is a monotonic function. The expected improvement (EI / max-EI / MEI) criteria, as developed by Mockus [76], chooses the point which gives the lowest expected loss between the predicted and the true maximum. This means choosing a point that upon observation would produce the largest maximum posterior mean (also referred to as the “one-step” method). Its statement and an analytical expression for Gaussian posterior are given as:

$$z^{n+1} = \arg \max_{z^{n+1}} E[\max_{z^{n+2}} E[MI_{z^{n+2}} | mi_{z^{1:n+1}}] | mi_{z^{1:n}}] \quad (10a)$$

$$EI(z, \xi) = E[(MI_z - (\mu_{\max} + \xi))_+] = (\mu(z) - (\mu_{\max} + \xi)) \Phi\left(\frac{\mu(z) - (\mu_{\max} + \xi)}{\sigma(z)}\right) + \sigma(z) \Theta\left(\frac{\mu(z) - (\mu_{\max} + \xi)}{\sigma(z)}\right) \quad (10b)$$

Here Θ is the standard normal probability density function and Φ is the normal cumulative density function. Both of these criteria prefer points with higher posterior mean and variance although with different tradeoffs. ξ controls the tradeoff in both these criteria. The probability of improvement (PI) becomes “greedy” when ξ is 0. The same cannot be said about expected improvement (EI). These strategies have been extensively studied in [55,79,95] and recently convergence rates have been proven for expected improvement [96]. Lia and Robbins [97] use the upper confidence bound (UCB) to control the exploration and exploitation tradeoff $UCB(z) = \mu(z) + \beta\sigma(z)$. This acquisition function works on the principle of selecting an optimistic point under uncertainty. For every query point z , UCB uses a fixed probability best-case scenario according to the underlying probabilistic model. The parameter β controls the exploration-exploitation tradeoff. A high value of β enables more exploration while a value of 0 leads to full exploitation. This criterion often has a provable cumulative regret bound [41], as shown recently by Srinivas et.al. [98] by formulating Bayesian optimization as an infinite-armed bandit problem with correlated arms.

A distinct class of acquisition functions makes entropy-based inferences. The entropy search (ES) criterion uses information about the location of global maxima [99]. It chooses a point that maximizes the decrease in differential entropy. The differential entropy of a continuous distribution $p(z)$ is defined as $\int p(z) \log(p(z)) dx$. A smaller differential entropy indicates less uncertainty. The predictive entropy search (PES) criterion seeks the same point but formulates the entropy reduction objective in terms of mutual information [100].

$$ES(z) = H(P(z^*)) - E_{mi(z)}[H(P(z^* | mi(z)))] \quad (11a)$$

$$PES(z) = ES(z) = H(P(mi(z))) - E_{z^*}[H(P(mi(z) | z^*))] \quad (11b)$$

Here z^* is the location of the global maxima, $H(P(z^*))$ denotes the entropy of the uncertainty about the position of the global maxima, and $H(P(mi(z)))$ denotes the entropy of the uncertainty

about the values of the function at z . Theoretically, both (entropy search (ES) and predictive entropy search (PES)) should yield the same point. However, practical computational techniques used for approximations yield different results. Equation (11(b)) (predictive entropy search) is slightly easier to evaluate than Eq. (11(a)) (entropy search). The first term in Eq. (11(b)) (predictive entropy search) can be computed in closed form. The second term requires approximations, such as the expectation propagation algorithm [101]. Even then, predictive entropy search (PES) is expensive to evaluate and thus unsuitable for our problem (and problems involving real-time execution). Max-value entropy search (MES) [102] tries to mitigate the computational complexity issue of entropy search (ES) and predictive entropy search (PES) by proposing a modified criterion. It uses information about the maximum value $MI^* = mi(z^*)$ instead of information about the location of the maximum value z^* .

$$MES(z) = H(P(mi(z))) - E_{MI^*}[H(P(mi(z)|MI^*))] \quad (12)$$

This is an easier problem to solve. The computation time of sampling the global minimizer can be further reduced by approximating the black-box function with parabolic form [103].

The marginal distribution of $mi(z)$ for any z is Gaussian, and hence the distribution of MI^* can be viewed as the maximum of an infinite collection of dependent Gaussian random variables. [102] uses a “mean-field” approximation and considers these infinite Gaussian random variables as independent and identically distributed. It allows the CDF of MI^* to be represented by Gumbel distribution. This parametric formulation allows the computation of the probability distribution of MI^* with fewer numbers of MI^* samples. Henceforth, the max-point entropy search method will be used with Gumbel approximation for the CDF of MI^* (MES-G).

Many of the discussed methods can be classified as “one-stage” methods, as they seek to obtain a point whose observation could minimize a certain quantity. Similarly, two-stage or multi-stage methods can be formulated using the principles of dynamic programming. However, such formulations do not significantly improve the results [89]. A detailed analysis of exploration and exploitation tradeoffs in these methods is presented in [104].

In this section, we probe the empirical performance of acquisition functions in Bayesian optimization using the max point error (MPE). From non-entropy-based functions, we select probability of improvement (PI), expected improvement (EI), and upper confidence bound (UCB) for evaluation. From entropy-based functions, we select the max-value entropy search with Gumbell sampling (MES-G) [102] for evaluation. We use Matern 3/2 kernel function and assume complete knowledge of its hyperparameters for each curve. Figure 5(a)-(c) shows the effect of the acquisition function hyper-parameter on the average max point error. Figure 5(d) compares the different acquisition functions using max point error. The best hyper-parameters selected from Fig. 5(a)-(c) are used in Fig. 5(d). We start the Bayesian optimization with 5 initial function evaluations and then plot max point error (MPE) for the 10th, 15th, and 20th function evaluation (or the 5th, 10th, and 15th BO iteration).

Max-value entropy search with Gumbell sampling (MES-G) has a better initial convergence rate than other considered acquisition functions. However, it tends to get stuck in local maxima. One example of such behavior is shown in Fig. 6(a). Figure 6(b) uses the same objective function as Fig. 6(a) to compute the percent of misidentification of global maxima by changing the initial sample locations 1000 times. MES-G’s runtime is comparable to the fastest method expected improvement (EI) or probability of improvement (PI). MES-G is also much more robust to the number of y^* sampled from the GP posterior to estimate the acquisition function than PES is to the number of x^* sampled [102]. Figure 6(c) shows that MES-G performs competitively even with as low as 10 samples.

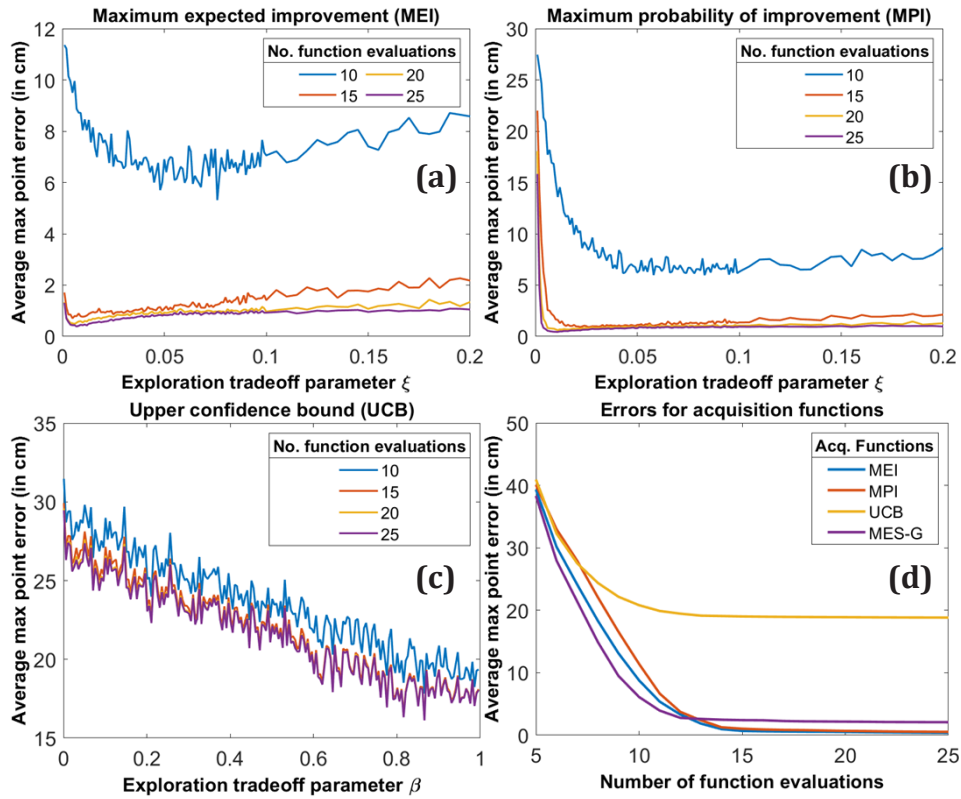


Fig. 5. Bayesian optimization errors for different acquisition functions (Acq. Functions). (a)-(c) Average max point error (MPE) curves [see Eq. (8b)] for maximum expected improvement (MEI) [see Eq. (10b)], maximum probability of improvement (MPI) [see Eq. (9)], and upper confidence bound (UCB), respectively with varying exploration tradeoff parameters. (d) Average max point error curves comparing MEI, MPI, UCB, and max-value entropy search with Gumbell sampling (MES-G) [see Eq. (12)]. For all these error curves Bayesian optimization was initialized with 5 initial function evaluations.

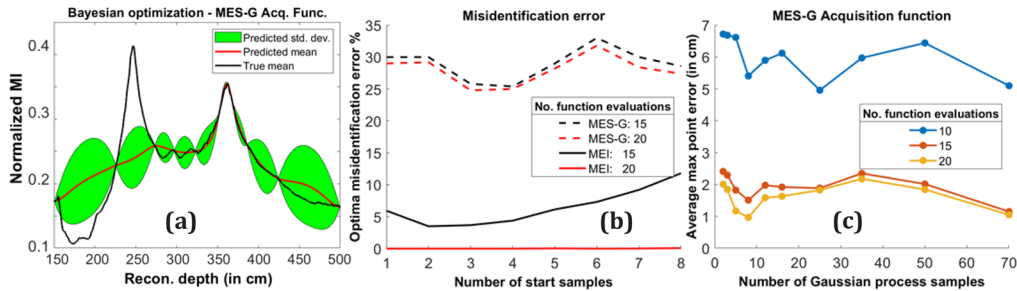


Fig. 6. (a) Sample objective function with the output of the Bayesian optimization at the 10th function evaluation using max-point entropy search acquisition function (MES-G) [see Eq. (12)]. (b) Misidentification error rate of global maxima on the same objective function as (a) for max-value entropy search with Gumbell sampling (MES-G) and maximum expected improvement (MEI) [see Eq. (10b)] acquisition functions at the 15th and 20th function evaluation. (c) Average max point error [see Eq. (8b)] for Bayesian optimization using MES-G acquisition function vs. the number of y^* sampled from the Gaussian process posterior. *MI*: Mutual information, *Recon. Depth*: Reconstruction depth, *Std. Dev.*: Standard deviation.

3.5. Mean function

We use a constant prior mean for the Gaussian processes. The choice of mean functions affects the overall fit of the posterior model with the ground truth and the rate of convergence. We evaluate the goodness of fit of the Gaussian process posterior for four different types of constant mean functions: minimum, maximum, maximum likelihood, and user-defined. The minimum constant mean function uses the minimum observed value as the prior mean; the maximum constant mean function uses the maximum observed value. The maximum likelihood (ML) constant mean function optimizes the likelihood function to obtain the estimate. The likelihood function for a Gaussian process with prior mean function $\mu(z)$ and kernel k is:

$$\log p(MI_z = \mathbf{mi}) = -\frac{1}{2}(\mathbf{mi} - \mu)^T K^{-1}(\mathbf{mi} - \mu) - \frac{1}{2} \log |K| - \frac{N}{2} \log 2\pi \quad (13)$$

Here $K = k(\mathbf{z}, \mathbf{z}) + \sigma_n^2 I$. In the above equation, \mathbf{z} , \mathbf{mi} , and N are completely determined by the data, whereas σ_n^2 , μ , σ_l , and σ_f are free model parameters. σ_l and σ_f parametrize the kernel function k . The ML estimate for the prior mean, assuming a constant mean function $\mu(z) = \mu$ is:

$$\mu^* = \left(\frac{\mathbf{1}^T K^{-1} \mathbf{mi}}{\mathbf{1}^T K^{-1} \mathbf{1}} \right) \mathbf{1} \quad (14)$$

Figure 7 shows the average squared error (SE) for the Gaussian process regression with a varying number of support points for different mean functions. Figure 7(a) compares the minimum, maximum, and maximum likelihood (ML) constant mean functions and Fig. 7(b) compares several user-defined constant mean functions. We assume complete knowledge of the kernel (Matern 3/2) hyperparameters.

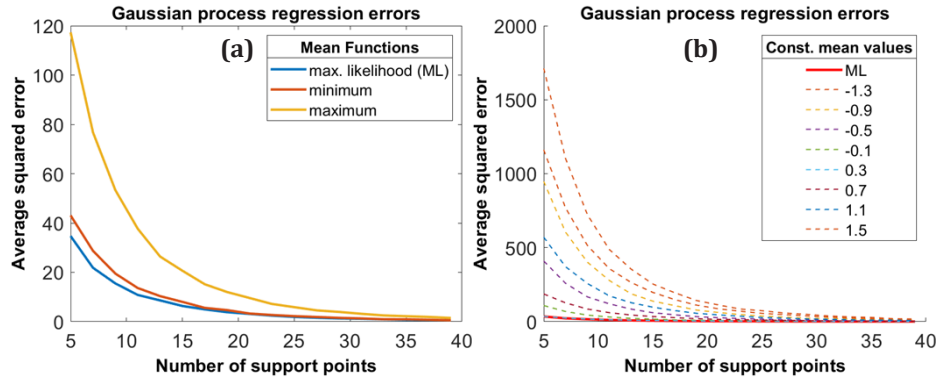


Fig. 7. Gaussian process regression errors for different constant mean functions. (a) Gaussian process regression average squared error (SE) [see Eq. (8a)] with a varying number of support points for minimum, maximum, and maximum likelihood (ML) constant mean functions. (b) Same curve as (a) for different user-defined values of constant mean functions.

Maximum likelihood constant mean function gives the lowest squared error compared to other constant mean functions. We had assumed prior knowledge of kernel hyperparameters. When kernel hyperparameters do not match, the difference in squared error for different mean functions grows starker. A detailed analysis of mean functions is given in [105]. They show that the minimum mean function makes the acquisition function more exploratory while the maximum mean function makes it more exploitative.

3.6. Parameter estimation

We consider three approaches to estimate kernel hyper-parameters. The first is to find the maximum likelihood estimate [52,90]. We substitute the Maximum likelihood (ML) estimated

mean function (Eq. (14)) in the likelihood equation (Eq. (13)). The likelihood function is then optimized for other free model parameters to get their corresponding maximum likelihood (ML) estimates. This method can easily fall into traps for a small number of samples [96]. The second approach imposes a prior on the hyperparameters. This yields the *maximum a posteriori* (MAP) estimate [106]. We use a Gaussian prior, although, log-normal or truncated Gaussian priors can also be used for positive parameters. Figure 8 shows the histogram of the model parameters for our experimental data set. The noise standard deviation σ_n depends significantly on the fineness of the sampling grid used to represent the objective function. The selection of σ_n prior thus becomes an engineering choice.

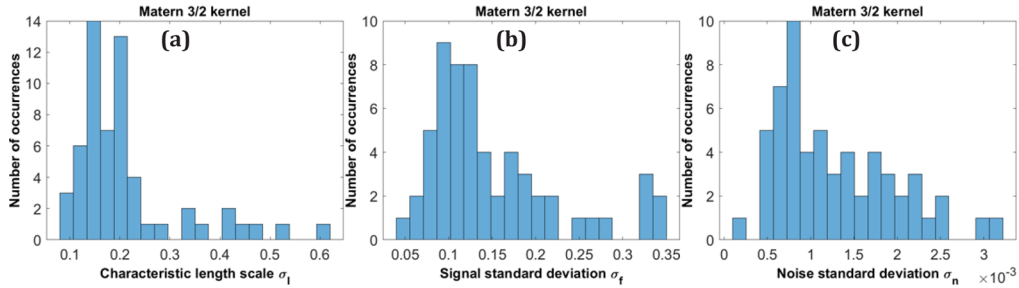


Fig. 8. Histograms of the free model parameters for the experimental data set with Matern 3/2 kernel [see Eq. (7)]. Gaussian fit on these parameters is used as a prior for the maximum a posteriori (MAP) estimate. The prior mean function is evaluated using the maximum likelihood (ML) estimate and does not require a prior.

The third approach is a fully Bayesian treatment. It integrates out the hyperparameters using quadrature or Monte Carlo methods [43,107,108]. Hyperparameters can be marginalized to obtain the posterior distribution or the acquisition function. The *maximum a posteriori* (MAP) approach can be considered an approximation to the Bayesian inference, and the maximum likelihood approach can be considered an approximation to the *maximum a posteriori*.

Figure 9(a) uses the max point error (MPE) to compare the maximum likelihood and *maximum a posteriori* approaches. We evaluate both these methods for informed and un-informed initial guesses. Informed guess is set as the average value of the parameters obtained experimentally (refer to Fig. 8). Un-informed guess is set $\exp(0)$ for the characteristic length σ_l and signal standard deviation σ_f , and $\exp(-1)$ for the noise standard deviation σ_n . Five starting samples (function evaluations) were used for the Bayesian optimization. Figure 9(b) uses the best approach from Fig. 9(a) to evaluate the effect of the number of starting samples.

Maximum a posteriori (MAP) approach with informed guess gives the lowest max point error (MPE) for each number of total function evaluations (refer to Fig. 9(a)). The average max point error difference for *maximum a posteriori* is small while using informed versus un-informed initial guesses, and it gets smaller with more function evaluations. Whereas, the average max point error difference for maximum likelihood is comparatively large while using informed versus un-informed initial guess. The number of starting samples has a considerable effect on the max point error for less number of total function evaluations (refer to Fig. 9(b)). This effect gets muted with more function evaluations. It is a result of the balancing act between the parameter estimation error and the Bayesian optimization process. Less number of starting samples leads to more errors in parameter estimation while accommodating more BO iterations for a constant number of function evaluations. Less number of start samples (two/three) also leads to frequent failure of the maximum likelihood approach, thereby increasing the performance advantage of *maximum a posteriori* even more. Henceforth, we use two starting samples, unless specified otherwise.

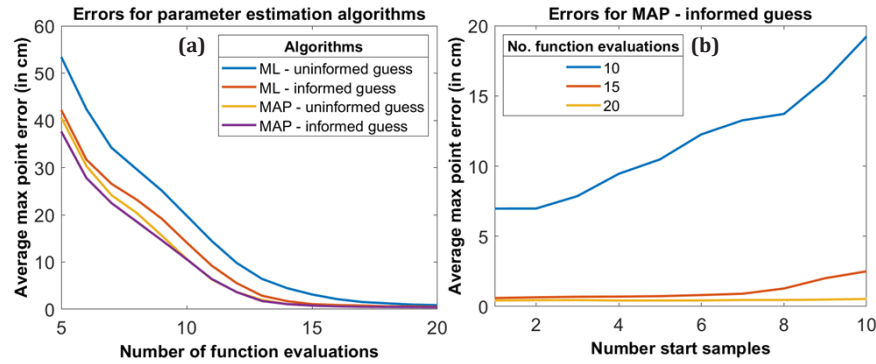


Fig. 9. Bayesian optimization errors for different parameter estimation algorithms. (a) Max point error (MPE) [see Eq. (8b)] for maximum likelihood (ML) and maximum a posteriori (MAP)-based approaches using informed and un-informed initial guess. The MAP with informed guess provides the lowest max point error (MPE) for each number of total function evaluations. (b) Max point error for MAP with an informed guess with a varying number of starting samples.

We incorporate an approximation to the full Bayesian treatment by marginalizing out the hyperparameters using Laplace approximation. Here we consider only the characteristic length scale σ_l and the signal standard deviation σ_n for marginalization. We approximate the posterior likelihood by a multi-variate (bivariate for our case) Gaussian distribution. This is achieved by centering the multi-variate Gaussian on the maximum of the posterior likelihood and setting its covariance matrix equal to the Hessian at that point. Parameters are now sampled from this Gaussian distribution – 40 samples for our case. We consider two different approaches to marginalization. First, we marginalize the hyperparameters to get the posterior distribution. The acquisition function is then used on this posterior distribution to compute the next point for probing (*Laplace – posterior*). Second, we marginalize the hyperparameters to directly get the acquisition function (*Laplace – acquisition*). In Fig. 10(a) we compare the two Laplace methods with the best approach from Fig. 9(a). The *Laplace – acquisition* method gives the lowest max point error, albeit with only a limited improvement. In Fig. 10(b) we plot the max point error (MPE) for Laplace – acquisition with a varying number of parameters sampled from the multi-variate Gaussian. It takes about 20 samples to get the full benefit of the marginalization. This benefit is also restricted to a smaller number of total function evaluations.

Laplace approximation relies on the goodness of the posterior likelihood and the ability to get its maximum point. This exposes it to the same concerns as that for the maximum likelihood (ML) and *maximum a posteriori* (MAP) methods. However, the failure in Laplace is more severe than that in ML or MAP. A failure of the ML method is a failure to find the maximum likelihood point. However, this problem gets compounded in Laplace approximation as the multi-variate Gaussian fitted on a point that is not the maximum is very ill-conditioned. This results in the sample parameter values straying far away from their true value. We define a failure if the Hessian (which is equated to the covariance matrix) is not positive-definite, or if the multi-variate Gaussian is very ill-conditioned. Figure 10(c) shows the failure rate of the Laplace method as a function of the number of support points (function evaluations). The same four methods as shown in Fig. 9(a) are used for the posterior likelihood computation. The *maximum a posteriori* (MAP) method gives a zero percent failure rate with both informed and un-informed initial guesses. The failure percentage with the maximum likelihood (ML) method depends on whether the initial guess is informed or un-informed. For all the methods, the failure decreases with the number of support points.

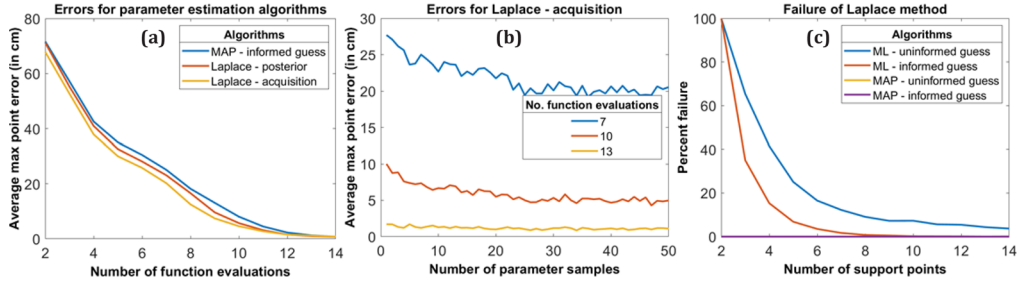


Fig. 10. Bayesian optimization errors for different parameter estimation algorithms. (a) Max point error (MPE) [see Eq. (8b)] for the two Laplace-based approaches: *Laplace – posterior* and *Laplace – acquisition*. These methods are compared with the best approach from Fig. 9(a), i.e. *maximum a posteriori* (MAP) with an informed initial guess. (b) Max point error for the *Laplace – acquisition* method with a varying number of parameter samples acquired from the multi-variate Gaussian. (c) Failure of the Laplace-based approaches with a varying number of support points. Both MAP with informed guess and MAP with un-informed initial guess give a 0 percent failure for each number of support points considered. *ML*: *Maximum likelihood*.

For an added computational cost the marginalization method provides only a limited gain in the max point error (MPE) compared to the *maximum a posteriori* (MAP) method with an informed initial guess. As such, we do not use marginalization in its current form for our problem.

3.7. Local maxima

As can be seen in Fig. 3, partially occluded objects give more than one peak, each corresponding to either the object or occlusion. In many instances, the peak corresponding to the object is not the global maxima. It is thus imperative to find all local maxima. For our problem, we define a local maximum as the peak corresponding to either the object or occlusion.

The posterior of the Gaussian process model, representing the objective function at the next sampling point $z \in Z$, is Gaussian:

$$p_{mi} \triangleq mi(z)|\mathbf{mi} \sim N(\mu_z, \Sigma_{z,z}) \quad (15a)$$

Due to the linear property of the derivative operator, the derivative of the posterior mi' is also subject to the Gaussian process model. Thus, at a different sampling point $z_1 \in Z$, another posterior can be written as:

$$p_{mi'} \triangleq mi'(z_1)|\mathbf{mi} \sim N(\mu_{z_1}, \Sigma_{z_1,z_1}) \quad (15b)$$

The two posteriors $mi(z)|\mathbf{mi}$ and $mi'(z_1)|\mathbf{mi}$ are not independent and identically distributed (i.i.d.) as they relate to the same objective function mi . These two posteriors are jointly normally distributed. Their combined mean and covariance matrix is given as [109]:

$$\mu \triangleq \begin{bmatrix} \mu_z & \mu_{z_1} \end{bmatrix}^T = \mathbf{A}k(\mathbf{z}, \mathbf{z})^{-1}(\mathbf{mi} - \mu_0(\mathbf{z})) \quad (16a)$$

$$\sigma^2 \triangleq \begin{bmatrix} \Sigma_{z,z} & \Sigma_{z,z_1}; & \Sigma_{z_1,z} & \Sigma_{z_1,z_1} \end{bmatrix} = \begin{bmatrix} k(z, z) & k'(z, z_1); & k'(z_1, z) & k''(z_1, z_1) \end{bmatrix} - \mathbf{A}k(\mathbf{z}, \mathbf{z})^{-1}\mathbf{A}^T \quad (16b)$$

$$\mathbf{A} = \begin{bmatrix} k(z, \mathbf{z}) & k'(z_1, \mathbf{z}) \end{bmatrix}^T, \quad k'(\cdot) \triangleq \frac{d}{dz_1}(\cdot), \quad \text{and} \quad k''(\cdot) \triangleq \frac{d^2}{dz_1^2}(\cdot), \quad (16c)$$

All other symbols have the same meaning as discussed in sec. 3.2. Equation (16) can be used to obtain the conditional posterior $p_{mi}|p_{mi'}$, characterized by the conditional mean $\bar{\mu}$ and variance $\bar{\sigma}^2$ as:

$$\bar{\mu} = \mu_z + \Sigma_{z,z_1} \Sigma_{z_1,z_1}^{-1} (m_i'(z_1) - \mu(z_1)) \quad (17a)$$

$$\bar{\sigma}^2 = \Sigma_{z,z} - \Sigma_{z,z_1} \Sigma_{z_1,z_1}^{-1} \Sigma_{z_1,z} \quad (17b)$$

In [110], the authors set two conditions to guarantee the discovery of a local optimum solution: the value of the objective function at the test point should be larger than elsewhere nearby, and the first-order derivative of the objective function should be close to zero. These conditions can be represented as $p_{mi}|p_{mi'} \geq \xi$ and $|p_{mi'}| \leq \varepsilon$. The authors incorporate these conditions to define a modified probability of improvement (PI) based acquisition function by first letting $z \equiv z_1$ and then defining the acquisition function as [110]:

$$PI(z, \xi) \approx \int_{\xi}^{\infty} (p_{mi}|p_{mi'}) d\xi \cdot \int_{-\varepsilon}^{\varepsilon} p_{mi'} d\varepsilon = Q\left(\frac{\xi - \bar{\mu}(z)}{\bar{\sigma}(z)}\right) \left[Q\left(\frac{-\varepsilon - \bar{\mu}(z)}{\Sigma_{z_1,z_1}^{1/2}}\right) - Q\left(\frac{\varepsilon - \bar{\mu}(z)}{\Sigma_{z_1,z_1}^{1/2}}\right) \right] \quad (18)$$

The Q-function represents the tail distribution function of the standard normal distribution.

We compare three approaches for finding local maxima. The first uses the modified probability of improvement (PI)-based acquisition function [110] as described in Eq. (18). This approach requires the first and second derivatives of the Gaussian process samples. Till now we had used the Matern 3/2 kernel function which gives rise to Gaussian process whose samples are once differentiable. To mitigate this, we switch to the squared exponential kernel for all three approaches used for finding local maxima. In the second approach, we partition the domain space into equal-length compartments (three for our case). A new probing point is computed for each of these partitions by maximizing the acquisition function inside its domain. This method generates multiple probing points in each Bayesian optimization iteration. Thus, for a fixed number of function evaluations, this approach requires less number of Bayesian optimization iterations. The third approach is sometimes referred to as synchronous parallel optimization in literature. Similar to the second method, this allows obtaining multiple function evaluations in parallel in the time that would ordinarily be required to obtain just one with sequential evaluation. Within every Bayesian optimization iteration, we choose the next probing point $z^i, i = 1, 2, \dots, n$ ($n = 3$ for our case) sequentially assuming that points $z^j, j < i$ have already been observed, and have values equal to a constant (expected value $m_i(z^j)$ of the posterior in our case).

In Fig. 11 we compare the three approaches for finding local maxima. We call these methods *local maxima PI*, *partition*, and *parallel* respectively. We use three partitions for the second method and three parallel acquisitions for the third method. We also compare the maximum expected improvement (MEI) method for finding global maxima as discussed in Sec. 3.4. In all these methods we use the squared exponential kernel function and the parameters are estimated using the *maximum a posteriori* (MAP) approach with informed initial guess. Four function evaluations are used for starting the Bayesian optimization. We evaluate the performance of these methods by computing the mean of local max point errors (MPes) corresponding to each local maxima (Mean-local MPE / Mean-LMPE). For example, if the ground truth objective function has two known local maxima of interest, then we consider the two local optima of posterior with the highest value and compute the mean of their position distances from those in the ground truth.

$$\text{Mean local max point error (Mean - LMPE)} = \frac{\sum_{i=1:N} MPE_i}{N} \quad (19)$$

The *partition* and *parallel* methods have almost similar average mean local max point error (LMPE). They perform similarly to *local maxima PI* in the initial few iterations of Bayesian

optimization. However, their performance drops slightly as the Bayesian optimization progresses. These three methods give a significantly smaller average mean local max point errors (LMPE) compared to the maximum expected improvement (MEI) method used for finding the global maxima.

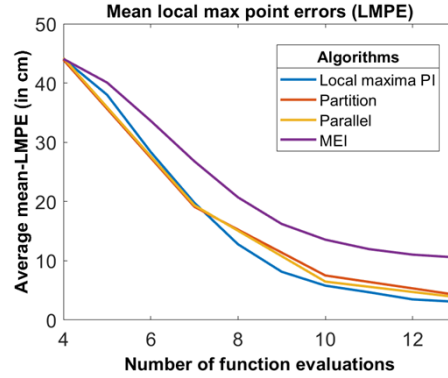


Fig. 11. Bayesian optimization errors for different local maxima acquisition functions. Average mean of local max point errors (Mean-LMPE) [see Eq. (19)] corresponding to each local maxima is plotted as a function of the number of function evaluations. The curves are plotted using the squared exponential kernel function along with maximum a posteriori (MAP) for parameter estimation. *PI*: Probability of improvement, *MEI*: Maximum expected improvement.

4. Results

We compare the discussed Bayesian optimization inference with traditionally used cubic spline inference. We compare these methods on two different tasks. First, the task of finding global maxima is evaluated using max point error (MPE). Second, the task of finding local maxima is evaluated using mean local max point errors (LMPE). For both these tasks we compare the average and median performances. For the first task with Bayesian optimization, we use Matern 3/2 kernel, *maximum a posteriori*-based parameter estimation using informed initial guess, and the maximum expected improvement (MEI) acquisition function. For the second task with Bayesian optimization, we use squared exponential kernel, *maximum a posteriori*-based parameter estimation using informed initial guess, and the *local maxima PI* acquisition function. Figure 12 shows these comparisons using the average and median values. Figure 12(a) shows the average max point error (MPE) for the first task, and Fig. 12(b) and (c) show the median max point error. Figure 12(d) shows the average mean local max point errors (LMPE) for the second task, and Fig. 12(e) and (f) show the median mean local max point errors (LMPE).

Figure 12 shows that the average performance of Bayesian optimization is considerably better than spline-based inference, and the disparity in the median performance is even higher. Bayesian optimization achieves an acceptable degree of object depth localization (within 5-10 cm for objects placed at roughly 350 cm from the camera array) with ten 3D reconstructions or less (Fig. 12(a) and (d)). Compared to spline-based inference, Bayesian optimization achieves an acceptable max point error (Fig. 12(b) and (c)) and mean local max point error (Fig. 12(e) and (f)) in approximately half the number of function evaluations. The standard deviation with Bayesian optimization is also significantly smaller than that with spline-based inference.

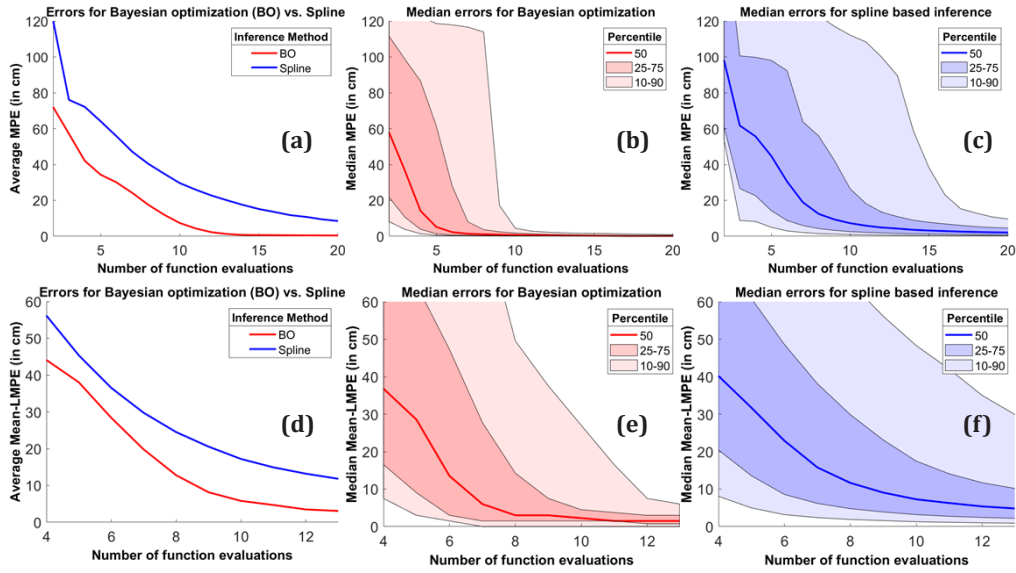


Fig. 12. Bayesian optimization and spline based inference errors for mutual information based depth estimation. (a) Average max point error (MPE) for Bayesian optimization vs. cubic spline-based inference. (b) Median MPE for Bayesian optimization. (c) Median MPE for cubic spline-based inference. (d) Average mean local max point error (mean-LMPE) for Bayesian optimization vs. cubic spline-based inference. (e) Median mean-LMPE for Bayesian optimization. (f) Median mean-LMPE for cubic spline-based inference.

5. Conclusions

We have considered object depth estimation under partial occlusion using integral imaging. A recent study accomplishes this by computing mutual information between the bounding boxes of the 3D reconstructed scenes and the 2D central elemental image. Mutual information analysis as a function of reconstruction depth shows that there are prominent peaks at the true depth of the object and occlusion. To improve upon this method, we have investigated the application of Bayesian optimization to achieve a reasonable depth estimation of both object and occlusion with only a few 3D reconstructions. We have studied different kernel functions, acquisition functions, and parameter estimation algorithms for Bayesian optimization-based inference. We evaluated their performance based on errors between the predicted and the true depth of objects as a function of the number of 3D reconstructions. Our preliminary results show that a handful of 3D reconstructions may be sufficient to obtain a reasonably accurate depth estimation of objects. This number is approximately half of that required for traditionally used spline-based inference.

This manuscript discussed certain aspects of Bayesian optimization like kernel functions, acquisition functions, and parameter estimation. We also compared a few frameworks for finding local maxima as opposed to global maximums. However, a rigorous study of its various applications and performance was not considered here as it is outside the scope of this work. In the future, we plan to study the performance of this framework in real-world scenarios with additional environmental degradations such as fog, brownout, and low illumination conditions. We also plan to study in detail the role of integral imaging system parameters to further enhance the performance. We postulate that parameters such as the number of cameras and pitch size can be optimized for different degradations. Lastly, we also plan to extend this methodology to dynamic object localization and tracking.

Appendix A

We list all the abbreviations defined in the paper in Table 1.

Table 1. Definitions of commonly used abbreviations in this paper

Abbreviations	Definitions	Abbreviations	Definitions
BO	Bayesian optimization	MES-G	Maximum entropy search with Gumbell sampling
ES	Entropy search	ML	Maximum likelihood
GP	Gaussian process	MPE	Max point error
InIm	Integral imaging	Max-PI / MPI	Maximum probability of improvement
LMPE	Local max point error	PES	Predictive entropy search
MAP	Maximum a posteriori	SAII	Synthetic aperture integral imaging
Max-EI / MEI	Maximum expected improvement	SE	Squared error
MES	Maximum entropy search	UCB	Upper confidence bound

Funding. National Science Foundation (2141473); Office of Naval Research (N000142012690, N000142212349, N000142212375); Air Force Office of Scientific Research (FA9550-21-1-0333).

Disclosures. The authors declare no conflict of interest.

Data availability. Data underlying the results are not publicly available at this time but may be obtained from the authors upon reasonable request.

References

1. A. D. Rhodes, M. H. Quinn, and M. Mitchell, "Fast on-line kernel density estimation for active object localization," *arXiv*, arXiv: 1611.05369v1 (2016).
2. L. Liu, H. Li, and Y. Dai, "Efficient global 2D-3D matching for camera localization in large-scale 3D map," in *2017 IEEE Int. Conf. on Computer Vis.* (2017), pp. 2372–2381.
3. T. Sattler, B. Leibe, and L. Kobbelt, "Efficient and effective prioritized matching for large-scale image-based localization," *IEEE Trans. Pattern Anal. Mach. Intell.* **39**(9), 1744–1756 (2017).
4. C. Toft, E. Stenborg, L. Hammarstrand, L. Brynte, M. Pollefeys, T. Sattler, and F. Kahl, "Semantic match consistency for long-term visual localization," in *European Conf. on Com. Vis. (ECCV)* (2018), pp. 383–399.
5. K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conf. on Comp. Vis. and Pat. Recog. (CVPR)* (2016), pp. 770–778.
6. R. Girshick, "Fast R-CNN," in *Int. Conf. on Comp. Vis. (ICCV)* (2015), pp. 1440–1448.
7. J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *IEEE Conf. on Comp. Vis. and Pat. Recog. (CVPR)* (2016), pp. 779–788.
8. D. Z. Chen, A. X. Chang, and M. Niebner, "ScanRefer: 3D object localization in RGB-D scans using natural language," in *European Conf. on Com. Vis. (ECCV)* (2020), pp. 202–221.
9. X. Zhou, Y. Peng, C. Long, F. Ren, and C. Shi, "MoNet3D: towards accurate monocular 3D object localization in real time," *Proc. ICML'20 1066*, 11503–11512 (2020).
10. S. Komatsu, A. Markman, A. Mahalanobis, K. Chen, and B. Javidi, "Three-dimensional integral imaging and object detection using long-wave infrared imaging," *Appl. Opt.* **56**(9), D120–D126 (2017).
11. A. Markman and B. Javidi, "Learning in the dark: 3D integral imaging object recognition in very low illumination conditions using convolutional neural networks," *OSA Continuum* **1**(2), 373–383 (2018).
12. D. Aloni, A. Stern, and B. Javidi, "Three-dimensional photon counting integral imaging reconstruction using penalized maximum likelihood expectation maximization," *Opt. Express* **19**(20), 19681–19687 (2011).
13. X. Shen, A. Carnicer, and B. Javidi, "Three-dimensional polarimetric integral imaging under low illumination conditions," *Opt. Lett.* **44**(13), 3230–3233 (2019).
14. B. Tavakoli, B. Javidi, and E. Watson, "Three dimensional visualization by photon counting computational integral imaging," *Opt. Express* **16**(7), 4426–4436 (2008).

15. G. Lippmann, "Epreuves reversibles donnant la sensation du relief," *J. Phys.* **7**, 821–825 (1908).
16. N. Davies, M. McCormick, and L. Yang, "Three-dimensional imaging systems: a new development," *Appl. Opt.* **27**(21), 4520–4528 (1988).
17. H. Arimoto and B. Javidi, "Integral Three-dimensional Imaging with digital reconstruction," *Opt. Lett.* **26**(3), 157–159 (2001).
18. F. Okano, H. Hoshino, J. Arai, and I. Yuyama, "Real-time pickup method for a three-dimensional image based on integral photography," *Appl. Opt.* **36**(7), 1598–1603 (1997).
19. M. Martinez-Corral, A. Dorado, J. C. Barreiro, G. Saavedra, and B. Javidi, "Recent advances in the capture and display of macroscopic and microscopic 3D scenes by integral imaging," *Proc. IEEE* **105**(5), 825–836 (2017).
20. A. Stern and B. Javidi, "Three-dimensional image sensing and reconstruction with time-division multiplexed computational integral imaging," *Appl. Opt.* **42**(35), 7036–7042 (2003).
21. E. H. Adelson and J. R. Bergen, "The plenoptic function and the elements of early vision," *Computational Models of Visual Processing* **1**, 3–20 (1991).
22. J. Liu, D. Claus, T. Xu, T. Keßner, A. Herkommer, and W. Osten, "Light field endoscopy and its parametric description," *Opt. Lett.* **42**(9), 1804–1807 (2017).
23. G. Scrofanì, J. Sola-Pikabea, A. Llavador, E. Sanchez-Ortiga, J. C. Barreiro, G. Saavedra, J. Garcia-Sucerquia, and M. Martinez-Corral, "FIMic: design for ultimate 3D-integral microscopy of in-vivo biological samples," *Biomed. Opt. Express* **9**(1), 335–346 (2018).
24. J. Arai, E. Nakasu, T. Yamashita, H. Hiura, M. Miura, T. Nakamura, and R. Funatsu, "Progress overview of capturing method for integral 3-D imaging displays," *Proc. IEEE* **105**(5), 837–849 (2017).
25. M. Yamaguchi, "Full-parallax holographic light-field 3-D displays and interactive 3-D touch," *Proc. IEEE* **105**(5), 947–959 (2017).
26. X. Xiao, B. Javidi, M. M. Corral, and A. Stern, "Advances in three-dimensional integral imaging: sensing, display, and applications," *Appl. Opt.* **52**(4), 546–560 (2013).
27. A. M. Uso, P. L. Carmona, J. M. Sotoca, F. Pla, and B. Javidi, "Depth estimation in integral imaging based on a maximum voting strategy," *J. Disp. Technol.* **12**(12), 1715–1723 (2016).
28. W. A. Roberts and B. S. Thurow, "Correlation-based depth estimation with a plenoptic camera," *AIAA J.* **55**(2), 435–445 (2017).
29. Y. Wang, L. Wang, G. Wu, J. Yang, W. An, J. Yu, and Y. Guo, "Disentangling light fields for super-resolution and disparity estimation," *arXiv*, arXiv: 2202.10603v2 (2022).
30. L. Shi, C. Liu, D. He, X. Zhao, and J. Qiu, "Matching entropy based disparity estimation from light field data," *Opt. Express* **31**(4), 6111–6131 (2023).
31. W. Williem and I. K. Park, "Robust light field depth estimation for noisy scene with occlusion," in *IEEE Conf. on Comp. Vis. and Pat. Recog. (CVPR)* (2016), pp. 4396–4404.
32. T. C. Wang, A. A. Efros, and R. Ramamoorthi, "Depth estimation with occlusion modeling using light-field cameras," *IEEE Trans. Pattern Anal. Mach. Intell.* **38**(11), 2170–2181 (2016).
33. H. Sheng, P. Zhao, S. Zhang, J. Zhang, and D. Yang, "Occlusion-aware depth estimation for light field using multi-orientation EPIs," *Pattern Recognition* **74**, 587–599 (2018).
34. X. Jiang, M. L. Pendu, and C. Guillemot, "Depth estimation with occlusion handling from a sparse set of light field views," in *25th IEEE Int. Conf. on Img. Processing (ICIP)* (2018), pp. 634–638.
35. X. Long, L. Liu, C. Theobalt, and W. Wang, "Occlusion-aware depth estimation with adaptive normal constraints," in *European Conf. on Comp. Vis. (ECCV)* (2020), pp. 640–657.
36. P. Wani, K. Usmani, G. Krishnan, T. O'Connor, and B. Javidi, "Lowlight object recognition by deep learning with passive three-dimensional integral imaging in visible and longwave infrared wavelengths," *Opt. Express* **30**(2), 1205–1218 (2022).
37. K. Usmani, T. O'Connor, P. Wani, and B. Javidi, "3D object detection through fog and occlusion: passive integral imaging vs active (LiDAR) sensing," *Opt. Express* **31**(1), 479–491 (2023).
38. P. Wani, G. Krishnan, T. O. Connor, and B. Javidi, "Information theoretic performace evaluation of 3D integral imaging," *Opt. Express* **30**(24), 43157–43171 (2022).
39. S. R. Narravula, M. M. Hayat, and B. Javidi, "Information theoretic approach for accessing image fidelity in photon-counting arrays," *Opt. Express* **18**(3), 2449–2466 (2010).
40. S. Pertuz, D. Puig, and M. A. Garcia, "Analysis of focus measure operators for shape-from-focus," *Pattern Recognition* **46**(5), 1415–1432 (2013).
41. B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. de Freitas, "Taking the human out of the loop: A review of Bayesian optimization," *Proc. IEEE* **104**(1), 148–175 (2016).
42. J. Wu, M. Poloczek, A. G. Wilson, and P. I. Frazier, "Bayesian optimization with gradients," *arXiv*, arXiv: 1703.04389v3 (2017).
43. J. Snoek, H. Larochelle, and R. P. Adams, "Practical Bayesian optimization of machine learning algorithms," *Proc. NIPS'12*, 2951–2959 (2012).
44. P. I. Frazier and J. Wang, "Bayesian optimization for materials design," *Information science for materials discovery and design* (Springer, 2016), Vol. 225.
45. Y. Zhang, D. W. Apley, and W. Chen, "Bayesian optimization for materials design with mixed quantitative and qualitative variables," *Sci. Rep.* **10**(1), 4924 (2020).

46. R.-R. Griffiths and J. M. Hernandez-Lobato, "Constrained Bayesian optimization for automatic chemical design using variational autoencoders," *Chem. Sci.* **11**(2), 577–586 (2020).
47. V. Dalibard, M. Schaarschmidt, and E. Yoneki, "BOAT: Building auto-tuners with structured Bayesian optimization," *Proc. WWW'17*, 479–488 (2017).
48. M. Kim, Y. Ding, P. Malcolm, J. Speckaert, C. J. Siviyy, C. J. Walsh, and S. Kuindersma, "Bayesian optimization of wearable device parameters," *PLoS One* **12**(9), e0184054 (2017).
49. R. A. Vargas-Hernandez, "Bayesian optimization for calibrating and selecting hybrid-density functional models," *J. Phy. Chem. A* **124**(20), 4053–4061 (2020).
50. A. Marco, F. Berkenkamp, P. Hennig, A. P. Schoellig, A. Krause, S. Schaal, and S. Trimpe, "Virtual vs. real: Trading off simulations and physical experiments in reinforcement learning with Bayesian optimization," *Proc. ICRA*, 1557–1563 (2017).
51. L. Acerbi and W. J. Ma, "Practical Bayesian optimization for model fitting with Bayesian adaptive direct search," *Proc. NIPS'17*, 1834–1844 (2017).
52. J. Bergstra, R. Bardenet, Y. Bengio, and B. Kegl, "Algorithms for hyper-parameter optimization," *Proc. NIPS'11*, 2546–2554 (2011).
53. K. Swersky, J. Snoek, and R. P. Adams, "Multi-task Bayesian optimization," *Proc. NIPS'13*, 2004–2012 (2013).
54. C. Thornton, F. Hutter, H. H. Hoos, and K. Leyton-Brown, "Auto-WEKA: combined selection and hyperparameter optimization of classification algorithms," *Proc. KDD'13*, 847–855 (2013).
55. E. Brochu, V. M. Cora, and N. de Freitas, "A tutorial on Bayesian optimization of expensive cost functions, with applications to active user modeling and hierarchical reinforcement learning," *arXiv*, arXiv: 1012.2599v1 (2010).
56. H. Chen, H. C. Wu, S. C. Chan, and W. H. Lam, "A stochastic quasi-newton method for large-scale nonconvex optimization with applications," *IEEE Trans. Neural Netw. Learning Syst.* **31**(11), 4776–4790 (2020).
57. A. S. Lewis and M. L. Overton, "Nonsmooth optimization via quasi-newton methods," *Math. Program.* **141**(1–2), 135–163 (2013).
58. K. Butt, R. A. Rahman, N. Sepehri, and S. Filizadeh, "Globalized and bounded Nelder-Mead algorithm with deterministic restarts for tuning controller parameters: Method and application," *Optimal Control Applications and Methods* **38**(6), 1042–1055 (2017).
59. C. Stoean, M. Preuss, R. Stoean, and D. Dumitrescu, "Multimodal optimization by means of a topological species conversion algorithm," *IEEE Trans. on Evolutionary Computation* **14**(6), 842–864 (2010).
60. J. P. Li, "Truss topology optimization using an improved species-conversion genetic algorithm," *Engineering Optimization* **47**(1), 107–128 (2015).
61. Y. Liang and K. S. Leung, "Genetic algorithm with adaptive elitist-population strategies for multimodal function optimization," *Applied Soft Computing* **11**(2), 2017–2034 (2011).
62. L. De Castro and F. J. V. Zuben, "The clonal selection algorithm with engineering application," *Proc. GECCO 2000*, 36–39 (2001).
63. L. N. De Castro and J. Timmis, "An artificial immune network for multimodal function optimization," *Proc. CEC'02*, 699–704 (2002).
64. L. N. De Castro and F. J. V. Zuben, "aiNet: an artificial immune network for data analysis," *Data Mining: A Heuristic Approach* (IGI Global, 2002), pp. 231–260.
65. M. M. Corral and B. Javidi, "Fundamentals of 3D imaging and displays: a tutorial on integral imaging, light-field, and plenoptic systems," *Adv. Opt. Photonics* **10**(3), 512–566 (2018).
66. B. Javidi, A. Carnicer, J. Arai, T. Fujii, H. Hua, H. Liao, M. M. Corral, F. Pla, A. Stern, L. Waller, Q. H. Wang, G. Wetzstein, M. Yamaguchi, and H. Yamamoto, "Roadmap on 3D integral imaging: sensing, processing, and display," *Opt. Express* **28**(22), 32266–32293 (2020).
67. J. S. Jang and B. Javidi, "Three-dimensional synthetic aperture integral imaging," *Opt. Lett.* **27**(13), 1144–1146 (2002).
68. M. Daneshpanah and B. Javidi, "Profilometry and optical slicing by passive three-dimensional imaging," *Opt. Lett.* **34**(7), 1105–1107 (2009).
69. T. M. Cover and J. A. Thomas, *Elements of information theory*, (John, Wiley & Sons, 1991).
70. D. B. Russakoff, C. Tomasi, T. Rohlfing, and C. R. Maurer, "Image similarity using mutual information of regions," in *European Conf. on Comp. Vis. (ECCV)* (2004), pp. 596–607.
71. J. P. Pluim, J. B. Maintz, and M. A. Viergever, "Image registration by maximization of combined mutual information and gradient information," *IEEE Trans. Med. Img.* **19**(8), 809–814 (2000).
72. D. Rueckert, M. J. Clarkson, D. L. G. Hill, and D. J. Hawkes, "Non-rigid registration using higher-order mutual information," *Proc. SPIE* **3979**, 438–447 (2000).
73. E. Volden, G. Giraudon, and M. Berthod, "Information in Markov random fields and image redundancy," *Selected papers from the 4th Canadian workshop on information theory and applications II*, 250–268 (1996).
74. H. J. Kushner, "A new method of locating the maximum point of an arbitrary multippeak curve in the presence of noise," *Journal of Basic Engineering* **86**(1), 97–106 (1964).
75. A. G. Zilinskas, "Single-step Bayesian search method for an extremum of functions of a single variable," *Cybernetics and System Analysis* **11**(1), 160–166 (1975).
76. J. Mockus, V. Tiesis, and A. Zilinskas, "The application of Bayesian methods for seeking the extremum," *Towards Global Optimization 2* (Springer, 1979), pp. 117–12.

77. J. Mockus, "On Bayesian methods for seeking the extremum," in *Optimization Techniques IFIP Technical Conference* (1974), pp. 400–404.
78. J. Mockus, *Bayesian Approach to Global Optimization: Theory and Applications*, (Springer, 1989).
79. D. R. Jones, M. Schonlau, and W. J. Welch, "Efficient global optimization of expensive black-box functions," *Journal of Global Optimization* **13**(4), 455–492 (1998).
80. D. Huang, T. T. Allen, W. I. Notz, and R. A. Miller, "Sequential kriging optimization using multiple-fidelity evaluations," *Structural and Multidisciplinary Optimization* **32**(5), 369–382 (2006).
81. A. Sobester, S. J. Leary, and A. J. Keane, "A parallel updating scheme for approximating and optimizing high fidelity computer simulations," *Structural and Multidisciplinary Optimization* **27**(5), 371–383 (2004).
82. A. J. Keane, "Statistical improvement criteria for use in multiobjective design optimization," *AIAA J.* **44**(4), 879–891 (2006).
83. J. Knowles, "ParEGO: a hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems," *IEEE Transactions on Evolutionary Computation* **10**(1), 50–66 (2006).
84. J. B. Mockus and L. J. Mockus, "Bayesian approach to global optimization and application to multiobjective and constrained problems," *Journal of Optimization Theory and Applications* **70**(1), 157–172 (1991).
85. J. M. Calvin, "Average performance of a class of adaptive algorithms for global optimization," *The Annals of Applied Probability* **7**(3), 711–730 (1997).
86. J. M. Calvin and A. Zilinskas, "One-dimensional global optimization for observations with noise," *Computers and Mathematics with Applications* **50**(1-2), 157–169 (2005).
87. J. Calvin and A. Zilinskas, "On the convergence of the P-algorithm for one-dimensional global optimization of smooth functions," *Journal of Optimization Theory and Applications* **102**(3), 479–495 (1999).
88. J. Calvin and A. Zilinskas, "One-dimensional P-algorithm with convergence rate $O(n^{-3+\delta})$ for smooth functions," *Journal of Optimization Theory and Applications* **106**(2), 297–307 (2000).
89. P. I. Frazier, "A tutorial on Bayesian optimization," *arXiv*, arXiv: 1807.02811v1 (2018).
90. C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*, (MIT Press, 2005).
91. D. J. Lizotte, "Practical Bayesian optimization," Ph.D. dissertation (University of Alberta, 2008).
92. C. K. I. Williams, "Prediction with Gaussian processes: From linear regression to linear prediction and beyond," *Learning in Graphical Models* (Springer, 1998), pp. 599–621.
93. C. E. Rasmussen, "Gaussian processes in machine learning," *Advanced Lectures in Machine Learning: ML Summer School 2003* (Springer, 2003), pp. 63–71.
94. J. S. Taylor, *Kernel Methods for Pattern Analysis*, (Cambridge, 2004).
95. D. R. Jones, "A taxonomy of global optimization methods based on response surfaces," *Journal of Global Optimization* **21**(4), 345–383 (2001).
96. A. D. Bull, "Convergence rates of efficient global optimization algorithms," *Journal of Machine Learning Research* **12**(88), 2879–2904 (2011).
97. T. L. Lai and H. Robbins, "Asymptotically efficient adaptive allocation rules," *Advances in Applied Mathematics* **6**(1), 4–22 (1985).
98. N. Srinivas, A. Krause, S. Kakade, and M. Seeger, "Gaussian process optimization in the bandit setting: no regret and experimental design," in *Proc. of the 27th Int. Conf. on Machine Learning (ICML'10)* (2010), pp. 1015–1022.
99. P. Hennig and C. J. Schuler, "Entropy search for information-efficient global optimization," *The Journal of Machine Learning Research* **13**, 1809–1837 (2012).
100. J. M. Hernandez-Lobato, M. W. Hoffman, and Z. Ghahramani, "Predictive entropy search for efficient global optimization of black-box functions," *Proc. NIPS'14*, 918–926 (2014).
101. T. P. Minka, "A family of algorithms for approximate Bayesian inference," Ph.D. dissertation (Massachusetts Institute of Technology, 2001).
102. Z. Wang and S. Jegelka, "Max-value entropy search for efficient global optimization," *Proc. ICML'17*, 3627–3635 (2017).
103. B. Ru, M. McLeod, D. Granzio, and M. A. Osborne, "Fast information-theoretic Bayesian optimization," *Proc. ICML'18*, (2018).
104. G. D. Ath, R. M. Everson, A. A. M. Rahat, and J. E. Fieldsend, "Greed is good: Exploration and exploitation trade-offs in Bayesian optimization," *ACM Trans. on Evol. Learning and Opt.* **1**(1), 1–22 (2021).
105. G. D. Ath, J. E. Fieldsend, and R. M. Everson, "What do you mean?: the role of mean function in Bayesian optimization," *Proc. GECCO'20*, 1623–1631 (2020).
106. A. Gelman, J. B. Carlin, H. S. Stern, D. B. Dunson, A. Vehtari, and D. B. Rubin, *Bayesian Data Analysis*, 2nd ed. (CRC Press, 2014).
107. E. Brochu, T. Brochu, and N. D. Freitas, "A Bayesian interactive optimization approach to procedural animation design," *Proc. SCA'10*, 103–112 (2010).
108. M. A. Osborne, R. Garnett, and S. J. Roberts, "Gaussian processes for global optimization," in *LION* (2009).
109. E. Solak, R. M. Smith, W. E. Leithead, D. J. Leith, and C. E. Rasmussen, "Derivative observations in Gaussian process models of dynamic systems," *Proc. NIPS'02*, 1057–1064 (2002).
110. Y. Mei, T. Lan, M. Imani, and S. Subramaniam, "A Bayesian optimization framework for finding local optima in expensive multi-modal functions," *arXiv*, arXiv: 2210.06635v1 (2022).